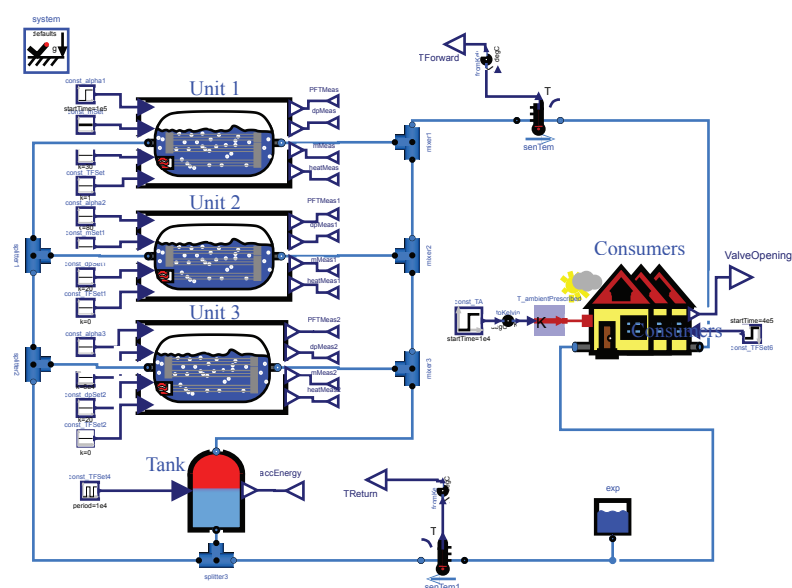

Economic Production Balancing of District Heating Portfolio



A Master's Thesis Submitted to the Department
of Electronic Systems - Control and Automation
Aalborg University, by

Kim Hylling Sørensen

June 2015

In collaboration with:

Added[®]
Values



AALBORG UNIVERSITY
STUDENT REPORT

FACULTY OF Engineering and Science
SCHOOL OF Information &
Communication Technology
DEPARTMENT OF Electronic Systems
SECTION FOR Control & Automation
June 2015

Aalborg University
The Faculty of Engineering and Science
School of Information and
Communication Technology
Department of Electronic Systems
Section for Control and Automation

Frederik Bajers Vej 7
DK – 9220 Aalborg Ø
Phone.: +45 9940 9730
Telefax: +45 9940 9725
<http://www.es.aau.dk>

Title: Economic Production Balancing
of District Heating Portfolio

Theme: Control of Complex Systems

Project period: Sep. 1st – Juni 3rd 2015.

Project group: 14gr935 (3rd – 4th semester)

Participants:

Kim Hylling Sørensen

Supervisor: Jan Dimon Bendtsen

Printed copies: 4

Pages: 93

Appendices: A - E

Attachment: 1 CD

Author



Kim Hylling Sørensen

khso10@student.aau.dk, kim@hylling.com

BSc in Electronics & IT, Aalborg University, Denmark, 2013

BSc thesis: Current Control of Gas Metal Arc Welding

MSc student in Control & Automation, Aalborg University, Denmark



Abstract:

AALBORG UNIVERSITY
STUDENT REPORT

Denmark is one of the leading countries in the world on the use district heating. It is therefore important to have a well functioning and optimal controlled system. This thesis is based upon the wish to see if it is possible to design balancing controller which generates economic optimal operation conditions for portfolio of energy production units, using model predictive control. This is to support the integration of green energy plants and to improve the flexibility in the energy market. This must be done while meeting all consumer demands. The MPC optimization problem is designed in two parts, one which maintains the pressure in the distribution network and one that minimizes the production costs. The optimization must be done while respecting all physical constraints. Producer and consumer district heating models are developed with the modelling language Modelica. It is implemented with and tested in Simulink Matlab. It is possible to design the controller which is able to maintain the required operation conditions while minimizing the fuels costs. The linear prediction model had a difficult time predicting the very nonlinear behaviour in the district heating models. A few method is taken to accommodate for this problem. This thesis is made possible because of a collaboration, combined with an internship period, together with the Danish company Added Values™.

The content of this report is freely available, but publication (with source reference) may only take place by agreement with the authors.

Preface

This report documents the development process of an online economic optimization controller which job is to generate set points for heat production units. The set point must ensure economic optimal operation while meeting the required district heating demands. The project is done in cooperation with the Danish company Added Values™ located in Denmark, Vejle 7100. This research is done during a 3rd and 4th semester in the Masters programme Control and Automation at Aalborg University during the autumn of 2014 and spring of 2015.

There are two primary focuses of the thesis. The first is to develop a district heating model capturing the key dynamics of the production units, consumers and the district heating network as a whole. The second is to develop a controller which minimizes production costs.

Reading Guide

The report is built from three parts.

- Part I analyses of the energy trading markets and the controller concepts together with development of the adequate models.
- Part III developing and implementing the proposed controller with the developed models and measurement data.
- Part IV discuss the obtained results

All matrices are written with bold face, e.g. **A** and vectors with italic bold face e.g. ***Y***. All bold face arrows in diagrams represent vectors values. Blue lines represent flow of water and green lines represent flow of power.

References to program variables are done with a typewriter font e.g. `THouseSet`.

Reading the thesis in a chronologically order will give the reader the best insight into the project.

All sources are given in square brackets indicating the author and the year of publication, i.e. [Jens Jensen, 2008]. Figures, equations and tables are numbered according to chapter and location, e.g. third figure in chapter five is numbered 5.3. Units are given in square brackets in symbolic equations and otherwise without brackets.

Below the table of content symbols, acronyms and a glossary are listed in the nomenclature. Appendices are referred to by capital letters, and are placed after the Bibliography. The attached CD contains simulation data, copies of references found on the Internet, MATLAB scripts, Modelica code, and a digital version of this report.

Acknowledgement

A thanks to Added Values™ for allowing the project becoming reality. Also a special big thanks to Engineer Rene Just Nielsen at Added Values™ for helping with getting into Modelica, Dymola, for his large insight in modelling and control of thermal production units and for sharing his thoughts on various subjects during the thesis period. Also a thanks to Kasper Vinther from Aalborg University for aiding with various tasks.

Contents

Preface	I
Reading Guide	I
Acknowledgement	II
Nomenclature	VIII
Glossary	VIII
Acronyms	VIII
Symbols	X
1 Introduction	1
Initial Aim of Thesis	1
I Analysis on the Background of the Control Problem	3
2 Analysis of the Control Problem	4
2.1 District Heating in Denmark	4
2.2 The Danish Energy Market	5
2.3 Case Study - Sønderborg District Heating	5
2.4 Economic Production Balancing	9
2.5 Production Costs	11
3 Scope of Thesis	14
3.1 Problem Statement	14
3.2 Thesis Contribution	15
3.3 Previous Work	15
3.4 Thesis Outline	17
4 Software Structure and Platform	18
4.1 CVX: Matlab Software for Disciplined Convex Programming	18
4.2 Modelica and Dymola	19
4.3 FMU-Models and FMI-Toolbox	19
4.4 Matlab and Simulink	20

4.5	Overview of Software Structure	20
II	District Heating Modelling	21
5	Non-Linear Plant Modelling	22
5.1	Overview of the District Heating Model	22
5.2	Consumer Model	23
5.3	Producer Model	35
5.4	Accumulator Tank Model	39
5.5	Component Models	43
5.6	State Selection	51
5.7	Combining the Models	52
5.8	Model Linearisation	55
III	Control Solution to the Cost Optimization Problem	62
6	Introduction to the Control Solution	63
6.1	Requirement Specification	64
6.2	Choice of Balance Controller	64
7	The Balance Control Problem	68
7.1	The Cost Function	68
7.2	Model Predictive Control	71
7.3	Model Constraints	71
8	Implementation and Results	74
8.1	General Simulation Implementation	74
8.2	MPC Tests	78
8.3	Test Case 1 - 3 Producers and Scaled Consumer	79
8.4	Test Case 2 - With an Accumulator Tank	84
IV	Evaluating the Performance of the Balance Controller	89
9	Acceptance Test	90
10	Conclusion	91
11	Discussion	93
	Literature	94
	Appendix A Introduction to Modelica and Dymola	98

Appendix B	Linearization of System Models	101
Appendix C	Convex Sets and Functions	103
Appendix D	Stall of Mosek Solver	105
Appendix E	Matlab MPC Function Source Code	106
Appendix F	Attached CD	112

Nomenclature

Glossary

Dymola

Dymola is a commercial Dynamic Modeling Laboratory based on Modelica.

Elspot

Day-ahead energy market. Bids on production volumes for each hour the following day are bid to Elspot at noon every day and auctioned by the TSO. The Elspot is a sub-market of the Nord Pool Spot.

Modelica

Modelica is an open object-oriented modelling programming language and is property of the non-profit organisation, the Modelica Association.

up-regulation

The state where the grid is a deficit and the TSO needs more power to up-regulate.

wind power penetration

The ratio between the produced amount of wind power in a given hour and the consumed amount of power in that same hour. Large wind power penetration corresponds to wind power covering a large amount of the energy consumed.

Acronyms

AD	Average Deviation
AHP	Absorption Heat Pump
CHP	Combined Heat and Power
DAE	Differential Algebraic Equation

DH	District Heating
DHW	District Heating Water
FDW	Forward District Water
FMI	Functional Mock-up Interface
FMU	Functional Mock-up Unit
FMU-CO	Functional Mock-up Unit Co-Simulation
FMU-ME	Functional Mock-up Unit Model-Exchange
I/O	Input/Output
LP	Linear Programming
MIMO	Multiple Inputs and Multiple Outputs
MPC	Model Predictive Control
O-O	Object-Oriented
OP	Operating Point
PBP	Production Balancing Problem
PI-Controller	Proportional and Integral Controller
PSO	Particle swarm optimization
RDW	Return District Water
RHP	Right Half Plane
RMSD	Root Mean Square Deviation
RPM	Revolutions Per Minute
SDH	Sønderborg District Heating
SISO	Single Input and Single Output
TPD	Terminal Pressure Difference
TSO	Transmission Service Operator. Energinet.dk is the Danish TSO.
UCP	Unit Commitment Problem
UDP	Unit Dispatch Problem
UML	Unified Modeling Language
VAT	Value-Added Tax

Symbols

R_{Roof}	The total thermal resistance of the wall model.	$[m^2 K/W]$
$C_{p,w}$	The specific heat for water	$[J/(kg K)]$
ρ_p	The density of the medium inside the pipe.	$[kg/m^3]$
N	time horizon of the optimization problem, $N = 48$ is used throughout Part III	[h]
h_{bw}	The thickness of the brick walls.	[m]
$h_{p,a}$	The specific enthalpy at inlet port a of the pipe component.	[J/kg]
$u_{p,a}$	The medium specific internal energy in the pipe.	[J/kg]
$U_{p,a}$	The internal energy of the medium in the pipe.	[J]
q_f	The heat flow from the floor into the air.	[J/s]
k_{ai}	The thermal conductivity of air.	$[W/m K]$
k_{bw}	The thermal conductivity of the air foil insulation between walls.	$[W/m K]$
k_{mw}	The thermal conductivity of wall mineral wool insulation.	$[W/m K]$
k_{wp}	The thermal conductivity of wood plating.	$[W/m K]$
M_p	The mass the medium inside the pipe.	[kg]
\wp	The linear momentum of the fluid inside the pipe.	[kg m/s]
A_{Roof}	The area of the modelled consumer roof.	$[m^2]$
A_{Wall}	The area of a consumer wall.	$[m^2]$
d_p	The diameter of the pipe .	[m]
h_{ai}	The height of modelled of staggered air.	[m]
h_{wp}	The height of the mineral wool used in wall insulation.	[m]
h_{wp}	The height of the wall wood plating.	[m]
l_p	The length of the pipe.	[m]
$m_{cv,nom}$	The nominal mass flow through consumer valve.	$[kg/s]$
$m_{cv,scaled}$	The scaled nominal mass flow through consumer valve.	$[kg/s]$
$m_{p,a}$	The mass flow at the inlet port a of pipe component.	$[kg/s]$
V_p	The volume of the pipe component.	$[m^3]$
$\Delta_{p,fric}$	The is pressure loss duo to friction in the pipe.	[Pa]
ΔP_{Pump}	The is pressure drop across the pumps two ports of the pump.	[Pa]
ΔP	Producer pressure difference across each of the terminals port_a and port_b.	[Pa]
$P_{p,a}$	The pressure at inlet port a of the pipe component.	[Pa]

$Q_{P,1h}$	The heat production over one hour.	[W]
R_{ai}	The total thermal resistance of staggered air.	[$m^2 K/W$]
R_a	The thermal resistance of the air foil insulation between walls.	[$m^2 K/W$]
R_{bw}	The thermal resistance of a brick wall.	[$m^2 K/W$]
R_{mw}	The total thermal resistance of mineral wool.	[$m^2 K/W$]
R_{wp}	The total thermal resistance of wall wood plating.	[$m^2 K/W$]
N_{nom}	The nominal rotational speed for flow characteristics of the pump.	[rpm]
N_{rs}	The rotational speed for flow characteristics of the pump.	[rpm]
$T_{p,a}$	The is the temperature at inlet port a of the pipe component.	[$^{\circ}C$]
$Q_{D,1h}$	The consumer heat demand over one hour.	[W]
q_p	The Heat energy flow to or from the pipe to an adjacent component	[W]

Introduction

Energy producers are challenged everyday by the increasing demand for cheaper energy and renewable energy sources. As the penetration of renewable energy is steadily increasing, and in particular for wind power penetration in Denmark, the overall power production fluctuation is increasing as well. This is due to the nature of renewable energy sources which are governed by stochastic behaviours. During 2014 wind turbines reached a new record of producing 39.1 % of the Danish consumption [Energinet.dk, 2015]. Denmark's aim by 2020 is to have 50 % of the energy consumption supplied by renewable energy sources [Energinet.dk, 2014]. As a consequence of these changes improvements regarding flexibility in existing deterministic power production sources are needed. This is to function better with stochastic power production i.e. to produce from deterministic production units when wind is not blowing and their by still meeting the supply requirements.[EURELECTRIC, 2011] [Energytransition.de, 2013].

Energy companies are trying to adapt to this ongoing shift in power production, from mainly deterministic production into part deterministic and part stochastic production. In order for a deterministic power producer e.g. "conventional" power plants, to better work together with a dominated stochastic energy market, deterministic production units must strive towards being more flexible and adaptable [Energinet.dk, 2013]. This strategy gives a more comprehensive toolbox maintaining the supply requirements, and utilize the potential of the stochastic renewable energy sources, from a deterministic producer point of view e.g. by ramping up power production when stochastic production is low.

Power producers portfolio of production units can include several different stochastic and deterministic production facilities. In order to utilize the portfolio most efficient all production units must be balanced in an economic optimal way.

An example of what is being done right now is the government financial supported project Green Flex. The project is a collaboration between Added Values, Aalborg University, Sønderborg Fjernvarme AMBA, and EMD INTERNATIONAL A/S. The project concerns with developing models and software to evaluate the effects of making production units more flexible and how make develop control to ensure economic optimal operation of energy production units in existing producers production portfolio. [energiforskning.dk, 2014].

This thesis is done in collaboration with Added Values and the goal is to try and answer some of the questions above. The focus areas of this work is described below as an initial problem statement.

Initial Aim of Thesis

The initial aim of the thesis is to investigate how to economically operate a producers portfolio of production units. This must be done while taking into account the requirements of meeting consumer demands,

flexibility demands, production unit dynamics and production unit costs. All of this must then be done with the goal of maximizing revenue for the producer. In order to achieve this a controller together with an adequate testing simulation environment must be developed. All modelling in the Greenflex project is done with the non-causal modeling language Modelica in combination with the development platform Dymola. It is there chosen to use these tools for all modelling during the thesis. Both tools are described in Appendix A. The Controller is to be implemented and simulated in Matlab and Simulink.

From the knowledge of Chapter 1 the following initial thesis statement is outlined.

Initial Problem Statement

How is a controller be designed which economically balances production for a portfolio of production units in order to maximize revenue.

Initial Thesis Objectives

To solve and accomplish the initial thesis statement a set of preliminary objectives are defined.

- *Understanding the systems:* In order to achieve an insight into energy production systems a brief analysis of the following subjects should be made, the different energy markets, the economics of heat and power production, a case study of a heat and power producer, and the aim of the controller.
- *Develop Simulation Models:* This entails modelling of producers production units together with the District Heating (DH) system network. More specific this should include designing models of heat and power production units, heat consumer units, and heat storage units. All must be connected together in order to represent a true DH system.
- *Develop Controller:* The development of the controller can be split up into two objectives.
 - *Optimization problem:* This entails understanding and developing of a optimization problem which have the ability to maximize revenue or to minimize the cost of meeting production demands with a portfolio of production units.
 - *Controller:* Designing a controller which online generates optimal operating set points for each production unit.
- *Controller Tests:* The controller must be tested to generate results to be evaluated upon.
- *Evaluate Controller:* The developed controllers performance must be concluded upon and compared to existing methods.

In Part I of the thesis an analysis is made in order to better understand the different concepts related to the stated initial thesis statement and from this develop the starting concept of the economically balances controller. The analysis involves understanding how production of district heating and power in Denmark is done, how the Danish energy market works, looking into how a true existing producer works.

PART

I

**ANALYSIS ON THE BACKGROUND OF THE
CONTROL PROBLEM**

Analysis of the Control Problem

This first chapter introduces the concept of district heating in Denmark, the Danish energy market and economic, an example of a power and district heating producer, and an introduction to the initial controller design thoughts.

2.1 District Heating in Denmark

In the context of this report DH is defined as the following:

DH: the supply of heat or hot water from one source to a district or a group of buildings.

During the past 100 years Denmark has been working on decentralising heat and hot water production which today is better known as DH by Combined Heat and Power (CHP) plants. DH was invented to get rid of waste in an efficient way. Today Denmark makes one of the world leading countries within developing innovative technologies and applying for district heating [dbdh.dk, 2015a]. Figure 2.1 shows the overall concept of district heating. DH systems consist of three parts, one or more production units, a District Heating Water (DHW) distribution network, and heat consumers. In this manner the objective is to make the DH supply meet DH consumption demands. All DH production is covered by 617 DH plants distributed all over Denmark. The DH distribution pipe networks are operated at rather low temperatures and pressures compared to other countries in Europe. The maximum specification is designed to be 120 °C at 25 bars whereas most operation supply is done at 80-100 degrees at 6.5-10 bars [dbdh.dk, 2015b]. The consumption is handled by 63 % (1.6 million) Danish households which all are supplied by DH [dbdh.dk, 2015a] [energitisynet.dk, 2014].

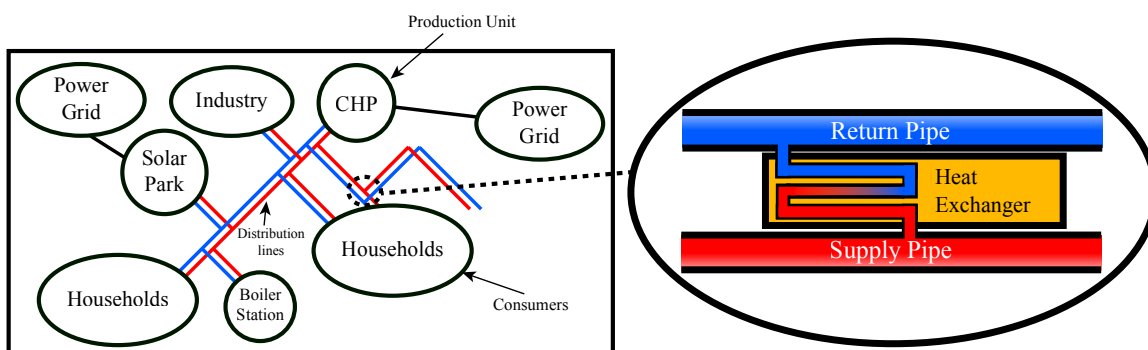


Figure 2.1: Illustration of a district heating network with producer and consumers.

The DH producers are paid a price for each MWh necessary to make the DH production meet the con-

sumption. This is done in accordance to a deal between the producer and the consumers. The consumer price is approximately in a national average 596 kr/MWh including Value-Added Tax (VAT) energitilsynet.dk [2015]. Adding all costs from the production, distribution and consumption shown Figure 2.1 gives a approximately total cost of 29,4 billion kr. yearly. This number is only partly correlated with the consumer costs as the consumer only pays the distribution costs [energitilsynet.dk, 2014]. This number is quite large and what to take from it is the potential of developing control systems which potentially can reduce/convert some costs to income.

An example of the the three core parts of district heating is seen in Figure 2.1. Producers maintain the production and consumers heat exchange with the supply pipes.

2.2 The Danish Energy Market

In order to understand how power is traded in Denmark a very brief introduction is given in this section. The Nordic and Baltic countries share a common online marketplace for trading power. Geographically all countries are split up into bidding areas where Denmark consist of DK1 for Jutland and DK2 for Funen and Zealand. Within the overall Danish energy market there exists 3 markets where trading occurs. These markets are very briefly described below.

Elspot Day-ahead Market

In the Elspot market is where almost all trades occur. It is an online auction house where producers places offers and buyers places bids for power contracts each hour for the following day. Each day at 12:00 trading closes for that period and prices are calculated.

Regulating Market

This market is a tool for the transmission service operator (TSO) to maintain consumer demands by activating additional power production or consumption for a given hour. A set of committed producers/consumers ensures this option.

Balancing Market

The balancing market is where prices are set for the trading imbalances relative to the regulating market.

2.3 Case Study - Sønderborg District Heating

Sønderborg is located in the southern part of Denmark and Sønderborg District Heating (SDH) is the main supplier of district heating with over 10.000 connected consumers [Sønderborg Fjernvarme]. SDH have a maximum heat production capacity of 154 MW [Sønderborg Fjernvarme]. Power and heat production is primary from a CHP plant located in Vestermark delivering 161.000 MWh heat and 27.000

MWh power. This is produced from 74.000 tons of waste and 8.000 tons of fossil fuels. In 2014 the total costs for running the CHP was 52.34 millions kr. paid by SDH and Sønderborg Kommune [Sønderborg Fjernvarme]. A pumping system is located in Spang delivering hot geothermal subsoil water to an absorption heat pump plant next to the CHP [geotermi.dk, 2013]. Peak and reserve loads are maintained by having 5 central boilers distributed in various places in Sønderborg. Furthermore a 4500 m^2 sun park is located in Vollerup. The location of the different production units are shown on the map in Figure 2.2. The operation conditions for Sønderborg DH distribution network is 75 - 82°C at a pressure difference across the consumer at 0.3 – 0.6Bar [Johansen, 2014]. The total district heating grid and all production units are in a shared ownership between Sønderborg District Heating and Sønderborg Kommune [Sønderborg Fjernvarme, 2014].



Figure 2.2: Sønderborg area together with the locations of Sønderborg District Heating productions units.

2.3.1 Production Portfolio

The production Portfolio of Sønderborg District Heating contains the previously mentioned production units and is listed below. Each unit produces either heat, power or both. All units does this in a deterministic manner except for the solar park.

- **Sønderborg CHP I/S:** Deterministic heat and power production with a production share of 54%
- **Absorption heat pumps with geothermal energy:** Deterministic heat production with a production share of 32%
- **Central Boilers:** Deterministic heat production with a production share of 8%
- **Solar park:** Stochastic power and heat production with a production share of 6%

2.3.2 Present Control Method

All production units can regulate either mass flow relative to a pressure difference or just to a mass flow. The enumerate below shows how the current control method at SDH works, in order to deliver the required DH [Johansen, 2014].

1. Enable an initial heat production unit and set it to maintain 0.6 bar of terminal pressure difference by adjusting the mass flow.
2. If the pressure differences reference can not be maintained enable the next unit in the portfolio to maintain the pressure.
3. Reconfigure unit 1 to control mass flow and set it to deliver a constant mass flow lower than maximum capacity.
4. Repeat

This control scheme is executed by manually enabling and disabling production units given a historical data and forecasts generated load plans.

2.3.3 District Heating Economics

During 2014 SDH produced 296.000 MWh of district heating and domestic water. The MWh heat consumer prices in Sønderborg is currently 401 kr/MWh including VAT excluding VAT maintaining round number this gives 321 kr/MWh [energitilsynet.dk, 2015]. The income from the district heating production Sønderborg by SDH is.

$$296.000 \text{ MWh} \cdot 321 \text{ kr/MWh} = 95.02 \text{ million kr.} \quad (2.1)$$

From this number 224.000 MWh went to the consumers and 72.000 MWh went to transmission losses. This makes (2.1) a potential income for the SDH but in reality the income is stated by (2.2) subtracting transmission losses.

$$224.000 \text{ MWh} \cdot 321 \text{ kr/MWh} = 71.90 \text{ million kr.} \quad (2.2)$$

This is still a quite large number though it is import to notice that production and maintaining costs are still not included. This illustates that there is quite some potential for increasing reducing loss of income if optimal operation conditions are applied to the production units. This could be done by running with the optimal supply temperatures at all times to ensure the least transmission losses while still maintaining the consumer demands. Intelligent production unit scheduling will also has an direct impact on this income number because the costs of input heat most likely reduces the income greatly.

2.3.4 Extreme Production Scenarios

From a business point of view producers have two methods of creating revenue. One by selling district heating to households and another by trading power in the energy market. In general weather conditions dominates the quantities of heat and power production and therefore to some degree how much revenue is generated. In particular the ambient temperature and the amount of wind on a given day is important. Table 2.1 outlines four somewhat extreme cases of production conditions with focus on outdoor temperatures and wind penetration levels.

Case	Temperature	Wind
1	Low	Low
2	High	Low
3	High	High
4	Low	High

Table 2.1: Four extreme production cases.

Case 1: The producer has the potential to make a good profit on delivering district heating and selling the power generated on the energy market to a relatively high price. Cold weather implies large needs for DH and no wind translates to more need for power production because wind turbines are not producing any power. This especially is true in Denmark where wind turbines are becoming increasingly dominant as discussed in Chapter 1. When the power grid is in up-regulation, i.e. needs more power, the price of power is high.

Case 2: The need for DH is low but the need for power is high. This yields the option of producing and trading power, given it yields a profit. In this situation heat can be considered as a bi-product of producing power because power generates the most revenue. The producer will produce as much power as long it is possible to get rid of the produced heat e.g. either by storing it in accumulator tanks or delivering it as DH.

Case 3: This case is the opposite of case 1. The demand for heat is low which means less profit from district heating. Also the power produced is traded for a low price due to high production from wind turbines. In these kind of situations producers will run with low production or go on weekend stop. Weekend stop is where the production is stopped and DH is delivered by hot water in accumulator tanks, this is usually done during weekends.

Case 4: There is a large demand for heating and less demand for power. Power will here be treated as the bi-product and traded on the market for additional revenue to a possible low price. The heating on the other hand is very necessary to be produced and will generate a lot of revenue. In some special cases the power prices becomes negative which gives the nice option of producing power from grid power to increase revenue.

2.4 Economic Production Balancing

Economic operation of DH systems has been investigated for a long time and is by no mean trivial due to the non-linear and non-stationary nature of these systems. Most DH producer operates the production units by manually setting set points as described for SDH in Subsection 2.3.2. Operating conditions and production facilities are also very diverse going from one producer to another. All of this adds up and increases the complexity of the economic production balancing problem and generating a versatile and general solution it.

The overall goal is to match DH demands while generating most revenue. This problem can be split into two challenges:

- Deciding when to enable or disable production units. This is also known in research literature as the Unit Commitment Problem (UCP).
- Defining economic optimal production conditions for each enabled production units. In literature this is also referred to as Unit Dispatch Problem (UDP) or to as production balancing.

The focus of this thesis is not on the unit commitment problem and it is assumed a set of enabled production units is given in advance. This set consists of an arbitrary number of production units with various dynamics and different types input fuel. This set will be different relative to a producers production portfolio e.g. one producer might have a gas-turbine available while another producer does not and so on.

In order to ensure economic daily operation of production units the following objective must be in focus:

- Maximizing the difference between revenue and cost or minimizing production costs while meeting DH demands.

An overview diagram of the system structure and information flow is seen in Figure 2.3. The block Economic Production Planning constructs a hourly based production plan dictating heat and power production each hour. The production plan is generated from information about sold power and DH, weather forecasts, and district heating forecast, and from prior production knowledge. In Figure 2.3 a set of production units are available similar to SDHs production portfolio, three boiler stations, one CHP, Absorption Heat Pumps (AHPs), and a solar park. Economic production set points must be generated to the deterministic production units. This generation must take into account the stochastic production from the solar park, fuel costs, dynamics of each unit, and production requirements. In order to further explain the control objective Example 2.1 is made to illustrate the concept of economic production planning relative to e.g. input fuel costs.

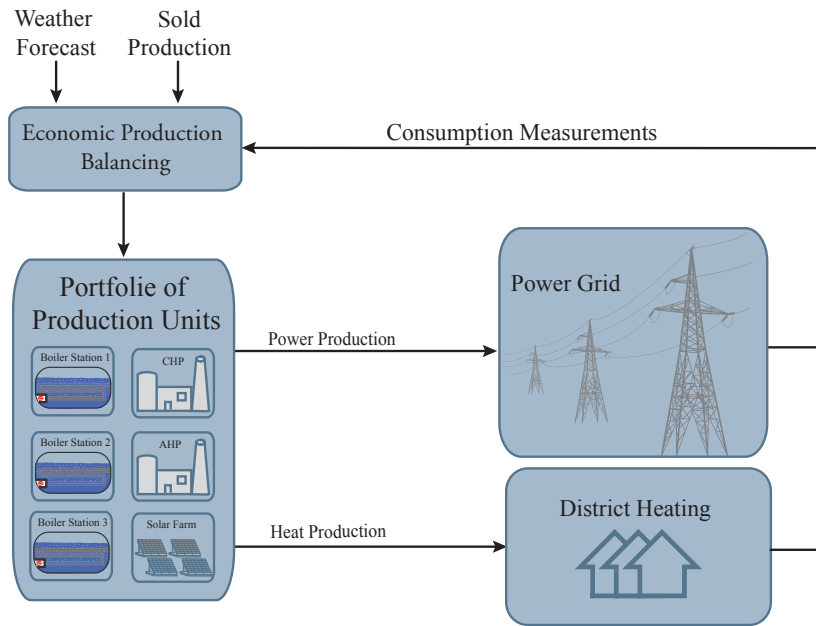


Figure 2.3: Overview of the control structure, information, and heat and power flow.

Example 2.1 (Economic Production Planning)

In this example an arbitrary DH load plan is given in Figure 2.4. The objective is to deliver the required heat in an economic optimal way i.e. produce the heat with the least cost.

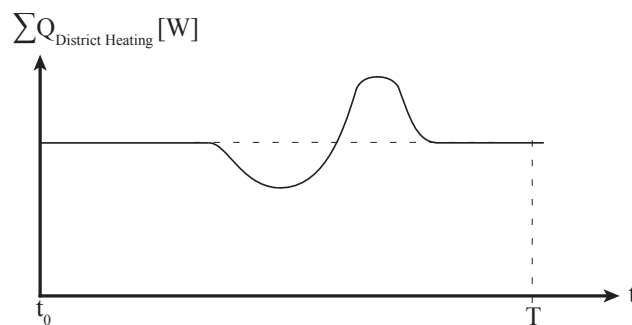


Figure 2.4: This is an example of a DH load function describing the necessary heat production in the interval $[0, T]$. This interval could be a 24 hour period.

In order to produce the heat a set of production units are enabled. In this case a waste burning plant, a heat pump cycle, and an oil fired boiler are enabled. The three units shares the load and delivers the required heat. The waste burning plant ensures the base load in the whole period $[t_0, T]$, the heat pumps supplies base load peaks in $[t_0, t_1; t_2, T]$, and to meet the high peak demands the boiler is enabled in the interval $[t_3, t_4]$ as demonstrated in the 3 graphs in Figure 2.5.

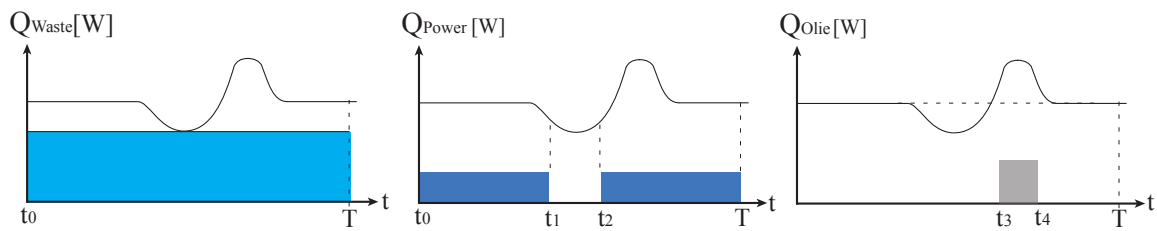


Figure 2.5: This is an example of how DH consumption is met by enabling three production units and producing heat in different periods to supply the heat most economic.

The joint production from the three enabled production units can be seen in Figure 2.6.

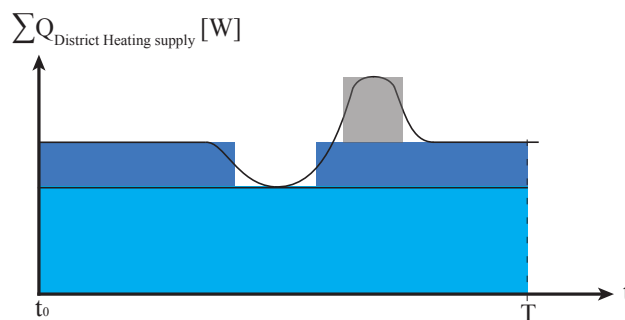


Figure 2.6: All the production from the units are the jointed into a total production and meeting the load plan.

The controller achieves the objective by enabling low cost production units for longer periods of time compared high cost ones. Waste burning plants are cheap production units, and usually the companies are paid to burn waste yielding negative input fuel costs. This is quite good from a cost minimization point of view, though there is an associated cost for starting up and maintaining the plant which is relatively high for a large CHPs. Oil is an expensive fuel input and oil burners are relative costly production units. This is why the oil burner only is enabled in short high peak periods to meet the demand and ramped down in non peak periods.

2.5 Production Costs

For DH systems and production units there exists different kind of costs and some a listed and described here.

- **Production costs:** The cost of e.g. producing heat from a boiler station including the cost of input energy, pumping costs, and people operation costs. The cost of pumping is much less than the energy costs and therefore it can be neglected.
- **Maintenance costs:** The cost fixing and repairing production units. This cost is very dependent on how production units and DH network is utilized. Changing set points a lot for a production unit

will generate more stress on mechanical components. As a consequence of this lifespan of parts is reduced and maintenance costs increases.

- Start-up and shut-down costs: Every time a production unit is shut down it will cost an additional cost of starting up the production unit again. This is because during start-up phases normal operation conditions are not present and it might cost more input energy during a warm up phase of for a gas turbine.
- Transmission heat loss: Heat losses are a function of the forward supply temperature and decreasing forward temperatures also lowers heat losses.

2.5.1 Production Balance Controller Interface

This section is made to give an overview the production balance controller interfaces. The controller is split into two controllers in order to differ between heat and power interfaces. One controller which optimizes within district heating and the market and one for energy production and the energy market. Interface diagram in Figure 2.7 illustrates this. Both controllers interfaces with the production portfolio but have different relevant inputs and outputs for heat and power.

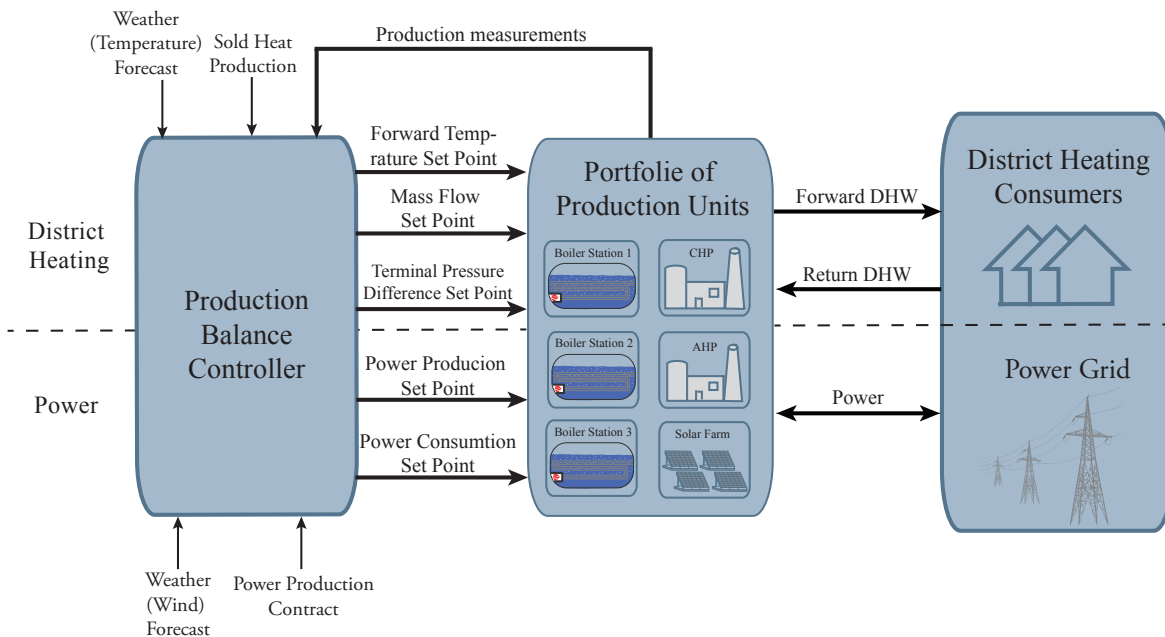


Figure 2.7: Controller, production portfolio, and consumer/market interface diagram.

Each heat and power producer unit is given 5 set point values, forward temperature, mass flow, pressure difference, power production, and power consumption. Heat production units are only given the three first inputs as indicated by the horizontal dashed line. The production output is measured and fed back to the controller. The two controller gets information about load plans in form of sold heat production and traded power production. Also information about the weather conditions is given to the controllers.

Now an analysis on the background of the control problem has been conducted. The chapter started out with understanding how the DH works in general and in Denmark. A case study of Sønderborg District Heating as a DH producer was presented and analysed with respect to production units, production capacity, economics, operating conditions, and present controller method. After this the concept of the balance controller was presented together with an example. In the end a brief discussed on production costs was made together with an introduction to the controller interface diagram.

Scope of Thesis

The initial thesis statement from Chapter 1 defined the focus areas of the thesis. The purpose of this chapter is to narrow down the initial aims and outline a scoped set of objectives necessary to accomplish the goals. This chapter also outlines the contributions of the thesis research, a review of previously work done for the production balancing problem and together with information of the overall thesis structure.

The overall objective is to develop a controller which economically balances production for a energy producer. This must be done while meeting production demands for both power and DH. This must also be done while minimizing production costs and thereby maximizing revenue for the producer. In order to narrow down the focus it is now chosen only to balance DH production and to meet demands only within the DH market. This choice confines the balancing problem only towards DH and all aspects towards power production is neglected.

With this outlined the following final problem statement is formulated in Section 3.1.

3.1 Problem Statement

How is a controller designed which can economically minimize the cost of meeting district heating consumer demands.

The initial objectives are now re-evaluated to fit the new thesis statement:

Final Thesis Objectives

The preliminary objectives was defined in Chapter 1 and until now the first objective is achieved and therefore given a check mark. The focus of the thesis is now scoped down to only optimize with respect to DH and therefore the objectives are reformulated.

- ✓ Understanding the systems
- Develop Simulation Models: This entails modelling of the DH system network. More specific this now includes designing models of heat production units, heat consumer units, and heat storage units. All must be connected together in order to represent a true DH system.
- Develop Controller The development of the controller can be split up into two objectives.
 - Optimization Problem Design: This entails the understanding and the development of a minimization problem which have the abilities to minimize the production costs while meeting DH consumer demands, given a portfolio of heat production units.
 - Controller Design: Designing a controller which online generates economic optimal operating set points for each heat production unit.

- *Controller Tests*: The controller must be tested to generate results to be evaluated upon.
- *Evaluate Controller*: The developed controller must be evaluated on its performance to fulfill the requirements specified in Section 6.1.

3.2 Thesis Contribution

This thesis is done in close cooperation with the Added Values and Aalborg University. Added Values is the primary consulting company of the Green Flex Project for Sønderborg District Heating. The contributions of this thesis is to both industry and academia. The contributions to Added Values are mainly how to model a district heating network using Modelica and Dymola, but also how to develop an adequate controller which can balance DH production. The contribution to Aalborg university is also how the balancing problem is solved but with weight on which software tools are utilized in order to do this.

3.3 Previous Work

This section presents the previous research done within economic production balancing of power and district heating production units. This research subject is in literature also referred to as the Unit Dispatch Problem (UDP). This review is conducted to discover what other researchers have chosen to do with respect to modelling and balance controller. In this section is the control problem referred to as the Unit Dispatch Problem (UDP) and the Production Balancing Problem (PBP) interchangeably.

3.3.1 The Production Balancing Problem / Unit Dispatch Problem

During the past 3 decades effort have been put into researching the UDP. Often the UDP is solved for a power production units as in [Ashfaq et al., 2014], DH, or combined plants. Each of these are Multiple Inputs and Multiple Outputs (MIMO) systems and they are ideal to be controlled using the UDP in order to do economic optimization.

In the article by [Ashfaq et al., 2014] the economic dispatch problem for a power system is solved using Linear Programming (LP), then compared to the firefly algorithm and the lambda iteration method. The firefly algorithm is a nature inspired optimization algorithm and is used to find all optimums of a highly nonlinear optimization problems. The algorithm works by all "fireflies" attaches each other proportional to brightness and distance, and they both decrease as their distance increases [Yang and He, 2013]. The lambda iteration method is where an Lagrangian optimization problem is designed and iteratively solved by interpolating a higher and lower, than the optimum, lambda values until a designed tolerance value is reached and the optimum is found [Dogra et al., 2014].

The motivation for their research is solving the UDP will result in a significant amount of fuel savings and cost reductions for power systems. This work argues that fuel and cost curves of modern power generation units are highly nonlinear and therefore methods which are able to handle this non-linearity

is applied by other researchers. The researchers state that these solutions often results in stochastic behaviours which gives local minima and maxima but the goal is to get a definite optimal point of operation. They suggest converting the non-linear behaviours into a linear function with objectives and constraints and solving it with the LP method. They conclude to have developed an algorithm which solves the UDP for a three and six unit systems and that it is simpler than other heuristic algorithms.

In the article by [Stéphane Velut, Per-Ola Larsson, Johan Windahl, 2014] a short term thermal economic production balancing problem is solved together with physical modelling of Vattenfall AB district heating plant using Modelica. The solution is done in two steps first a binary UCP using standard mixed integer linear models is solved and then a economic UDP is solved by non-linear optimization techniques from the JModelica.org platform. The outputs of the UDP is economic optimal set points for supply-temperatures, flow rates, turbine by-pass valves. This research discovered that this method improves short-term production planning and sets the supply temperatures significantly lower than normal operation, a more extensive usage of external coolers for higher efficiency and electricity production.

An alternative approach to solving the UDP is outlined in [Leandro dos Santos Coelho, 2008]. In this article the UDP is solved for a set of committed power generation units so that the required load demands at minimum operating costs are met. Optimization approaches inspired by swarm intelligence and evolutionary computation approaches are used as a potential alternative to solve difficult UDPs. A Particle swarm optimization (PSO) technique using Gaussian probability distinction functions is used to solve the economic UDP. Their PSO algorithm is tested on 15 and 20 power production units and prove to be a feasible method and approaches for an efficient method of solving UDPs with constraints.

In [Edlund et al., 2008] a Model Predictive Control (MPC) control scheme is used to solve the economic UDP for power generation units meeting consumption demands. A committed set of production units stated as a portfolio is given and the objective is to reference track and reject disturbances. The objective function of the optimization is a combination of l_1 norms and linear weights. Using simulation it showed a significant improvement of the performance of the MPC compared to the distributed Proportional and Integral Controller (PI-Controller) structure, both in terms of minimizing cost and minimizing deviations.

Summarize of the essential points from the work presented. In [Ashfaq et al., 2014] it is argued that production balancing saves input fuel i.e. reduced the production cost and increases revenue. They also supports a linear approach rather than a non-linear one to ensure one optimum. In [Stéphane Velut, Per-Ola Larsson, Johan Windahl, 2014] modelling of a district heating plant is done with modelica and production is economic balanced with a non-linear scheme. In [Leandro dos Santos Coelho, 2008] a swarm algorithm is used to coordinate a large set of production units. In [Edlund et al., 2008] a set of committed power production units is given and a optimization problem of norms showed improvement compared to a distributed controller structure.

Quite some work have been done trying to solve the production balancing problem. Several approaches have been outlined in research and there is not a supreme control and modelling approach to it. All researchers agree that there are significant saving to be made. The modelling is quite often kept relative simple. The control methods are often some algorithms and optimization with norms. With this knowledge the thesis work is continued

3.4 Thesis Outline

This thesis is divided into 4 part described the over all thesis areas of the report. The first part is an analysis on the background of the control problem, the second part is the modelling of the district heating system, the third part designing and testing of the picked control solution, and the fourth and last part is of the evaluate of the controller performance and conclusion.

Chapter 2: outlines the concept of district heating in Denmark, the Danish energy market, a brief case study of Sønderborg District Heating as an energy producer, and the initiating concept of the production balance controller.

Chapter 3: defines the scope of the thesis and outlines a problem statement. In here the contribution of the thesis is defined together with a research in to existing literature of the economic production balancing control problem.

Chapter 4: is an introduction to the not trivial software structure utilized during the thesis work. This includes informing of the Modelica and Dymola created district heating models. The use of the optimization solver extension to matlab CVX and how all the software interacts.

Chapter 5: in this chapter where the modelling of the whole district heating concept is made. This entail modelling of producers and consumers units connected to the distribution pipes.

Chapter 6: is where the balance controller type is selected together with an introduction of the MPCs controllers workings. In this chapter the controller requirements and assumptions are also defined. The requirements are evaluated upon in Chapter 9.

Chapter 7: is where the optimization problem is designed. This entails design of a cost function which is to be minimized and the constraints. The MPC control structure is also defined and presented in the end of this chapter.

Chapter 8: is where the controller and models are implemented into Simulink and tested for different simulation configurations.

Chapter 9: is where the controller requirement specification is evaluated to see what requirements the controller can satisfy and which it can not.

Chapter 10-11: in these two chapters the conclusion and discussion of the thesis work is presented. The conclusion entails concluding on the problem statement.

Software Structure and Platform

In this chapter an introduction to the software structure and platform is given. Firstly the chosen optimization platform is presented, then the used modelling language and environment, and then Matlab and Simulink. After this a total overview of the software structure is provided together with a description of the software goals in form of Unified Modeling Language (UML) diagrams.

4.1 CVX: Matlab Software for Disciplined Convex Programming

All optimization done with the thesis is defined and solved within the CVX structure. CVX is a Matlab-based convex modeling framework which turns standard matlab syntax into a optimization modeling language. CVX allows to specify objectives, constraints and solving disciplined convex programs [CVX Research, 2015]. The definition of disciplined convex programming is used for optimization problems which are constructed using CVXs composition rule set¹. Two free solvers, SeDuMi 1.34 and SDPT3 4.0, and two commercial solvers, Gurubi 6.0 and Mosek, are available within the CVX environment. The commercial solvers are free using a professional academia license from AAU. Mosek is chosen as solver duo to its greater ability at solving large scale sparse problems and in general it seemed faster than Gurubi, SeDuMi, and SDPT3 for the thesis objectives.

Example 4.1 (CVX Optimization Problem)

An example of how a CVX optimization problem looks like with matlab syntax. x and y are defined as variables. The goal is now to find the values of x and y which minimizes the 1-norm of the expression $Ax-y$ without violating the constraint $y \geq 0$.

```
cvx_begin
    variables x y;
    minimize(norm(Ax - y));
    subject to
        y >= 0;
cvx_end
```

As default this problem is solved with the SDPT3 4.0 solver.

¹To read this rule set go to www.web.cvxr.com/cvx/doc/dcp.html#dcp

4.2 Modelica and Dymola

Modelica is an object-orientated modelling language for modelling of multi-domain complex physical systems. The Modelica language is open source and is developed by the non-profit Modelica Association. The main goal with modelica is to make it easy to exchange models and model libraries. This is done by object-oriented constructs that facilitate reuse of already developed models. Modelica works as a non-causal modelling language with true ordinary differential and algebraic equations. [Elmqvist and Mattsson, 1997].

A complete DH model library containing producer and consumer models is developed in Modelica code. Figure 4.1(a) shows the developed DH library when imported into Dymola. The entries with white square symbols represents packets and they can be expanded as seen in Figure 4.1(b). In the producer, consumer, and AccumulatorsTanks packets exists the developed model components which can be dragged into different simulations. Also in each packet there is a unit test made to show and verify different functionalities of the unit. The tests are always located in the bottom.

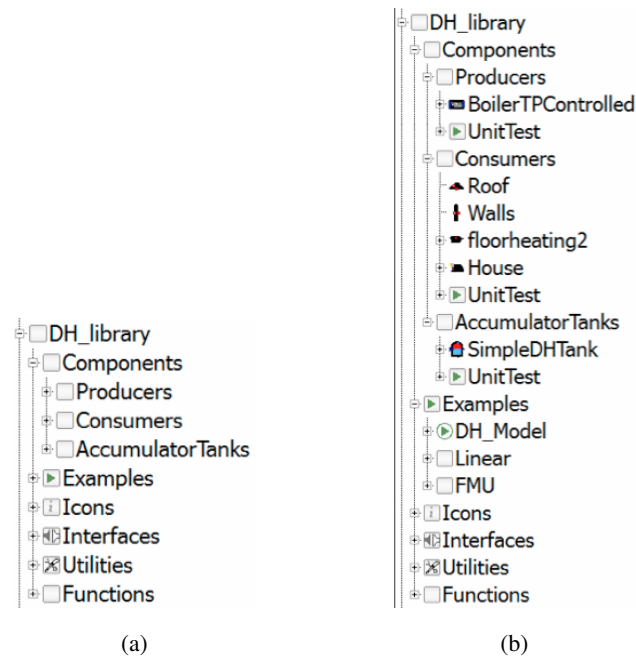


Figure 4.1: (a): The developed Modelica library folded together. (b): The developed Modelica library opened.

The overall idea of the library is that anyone can load in the DH library and use the developed and tested models to similar modelling projects.

4.3 FMU-Models and FMI-Toolbox

In order to interface the developed Modelica/Dymola physical models with Matlab/Simulink Functional Mock-up Unit (FMU) models are exported from Dymola. A Matlab toolbox called Functional Mock-

up Interface (FMI) gives the option of importing FMU models into Simulink. There are two exporting options of FMUs available namely Functional Mock-up Unit Co-Simulation (FMU-CO) and Functional Mock-up Unit Model-Exchange (FMU-ME). FMU-COs models contain both the model and an internal solver. In here the simulation which the FMU model has been imported into act as a master to the solver to the FMU-CO. The FMU-MEs only contains the model and uses the solver om the imported simulation environment [Blochwitz et al.].

4.4 Matlab and Simulink

The overall control and simulation framework is implemented in Matlab[®] release 2014a. Matlab is a high-level scripting language for technical computing. It provides visualization and simulation in an easy-to-use environment. Simulink[®] is a causal graphical programming language toolbox for modelling, simulation, and analysis of dynamical systems within Matlab. Both programs are developed by MathWorks[®].

4.5 Overview of Software Structure

An overview of the software structure is seen in Figure 4.2. A non-linear FMU model is developed in Dymola and imported into Simulink using an FMI Toolbox. An embedded Matlab function is run within Simulink working as the controller. The MPC controller measures outputs from the FMU model, solves the optimization problem by calling CVX, Mosek returns optimal values, and then MPC specifies input set points for the FMU model.

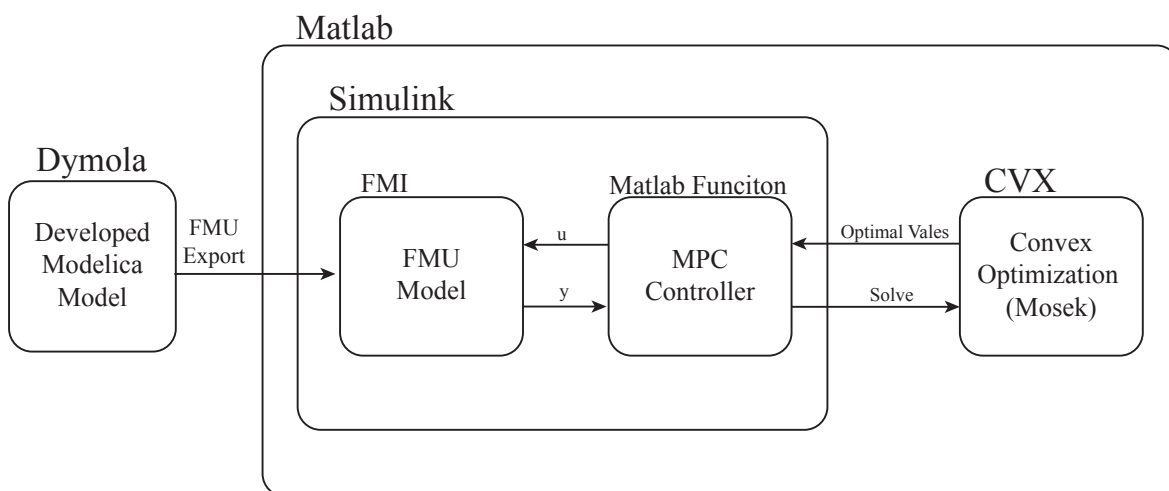


Figure 4.2: The overall software structure using Modelica/Dymola, Matlab, Simulink, and matlab toolboxes CVX and FMI.

This concludes the introduction to chosen software platform and how the different software programs are used and structured. In the next chapter, Chapter 5, modelling of the DH system is made.

PART

II

DISTRICT HEATING MODELLING

Non-Linear Plant Modelling

This chapter is about creating a model of the district heating network. This requires modelling of heat production units, storage units, consumer units, and the hydraulic connection between them. Each model is to be designed, implemented, parameterized and verified. All of this is done with the modelling language Modelica together with the commercial development environment Dymola. These tools are daily used at Added Values. To get an initial understanding of the Modelica language and Dymola tool see Appendix A.

5.1 Overview of the District Heating Model

An Object-Oriented (O-O) DH model is made by combining producers and consumers in parallel as indicated by Figure 5.1. In this configuration n parallel producers, p parallel consumers and a single accumulator tank are connected to the DH network. The producers output heat is distributed to households radiators and floor heating pipes through the DH pipe network. Local distributed control systems adjust the individual consumption of each consumer to ensure a wanted house temperature. Control systems at producer makes sure to maintain a terminal pressure difference across all the consumers. In this fashion producers ensure that the consumer heat demands are always satisfied. The producers use the feeding pumps in Figure 5.1 to maintain the pressure and consumers use the valves to achieve their indoor temperatures. The pressure difference will always be sufficient close to the producers but if the feeding pumps capacity is reached distant consumers in the DH network might go in a heat deficit. The forward temperatures are adjusted relative to heat demands and seasonal transport losses.

In Figure 5.1 each production unit is given set points in order to meet the momentary DH consumption.

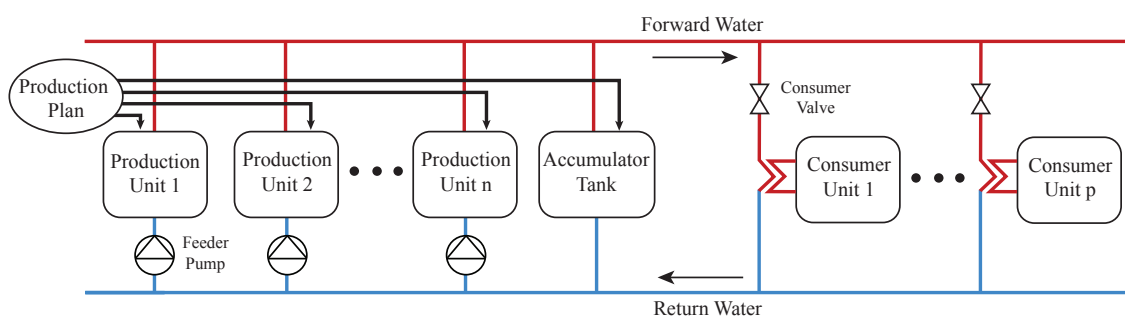


Figure 5.1: This figure represent how the DH system is modelled. n Parallel producer, p parallel consumer and a accumulator tank is included in the DH system model. The bottom blue lines represent the return water, the top red lines is the hot ward forward water, and the black lines are information flow.

5.2 Consumer Model

This section contains modelling of a consumer connected to the district heating supply line and working as a DH load. The goal is to create a nominal consumer model representing a single household. The model must be complex enough to capture DH dynamics without going into unnecessary details. The consumer model must also be scalable in order to represent a cluster of houses or to approximate consumption of a town.

Design Thoughts

- House dimensions: The household must be a standard sized house with an average heat consumption. Modelling the consumer as a standard house will neglect the increases and decreases in consumption from factories and small student apartments. The goal is represent house parameters as normal distributed and this is assumed to be sufficient because the consumer will be scaled to represent a large number of consumer.
- Thermal interface: It must be possible to heat exchange between the consumer and the DH supply lines.
- House temperature control: Consumers must be able to demand more or less heating e.g. by adjusting the temperature inside the house.
- Heat losses: Heat losses to the ambient environment must be that of an average sized house. Heat losses must also be seasonal dependent i.e. greater losses in winter. For model simplicity the heat losses to domestic water is neglected. This should of course be included in further model development because the heat going to the domestic water is significant when modelling a true consumer model. The reason it can be neglected is because it does not contribute features in order to test a possible balancing controller algorithm.
- Consumer thermal dynamics: The thermal dynamics must match that of a real average sized house. This must be done because it is necessary to have similar dynamics to test the controller against a model representing consumption as true as possible.
- Consumer scalability: It is assumed to be.

In a normal DH supplied household hot DHW flows through floor heating pipes and radiators to heat up the house. The heat travels through the floors and radiators in a conductive manner. Natural convection ensures that the heat is distributed from floors and radiators into the air and other materials inside the house. The households are not perfectly insulated and the heat continues to travel to the ambient environment. This occurs through walls, windows, opening of windows, roof, doors, ventilation systems, opening of doors, ect. A simple illustration of this is shown in Figure 5.2. Heat losses are indicated by full blue arrows and heat gains by full red arrows. In houses the temperature can usually be adjusted using thermostats and this is represented as the temperature control box in the bottom left corner in Figure 5.2. This controller should make the indoor temperature reach the set point $T_{HouseSet}$ by adjusting the intake of DHW using the consumer valve. The two large blue and red arrows indicate the convective behaviour occurring.

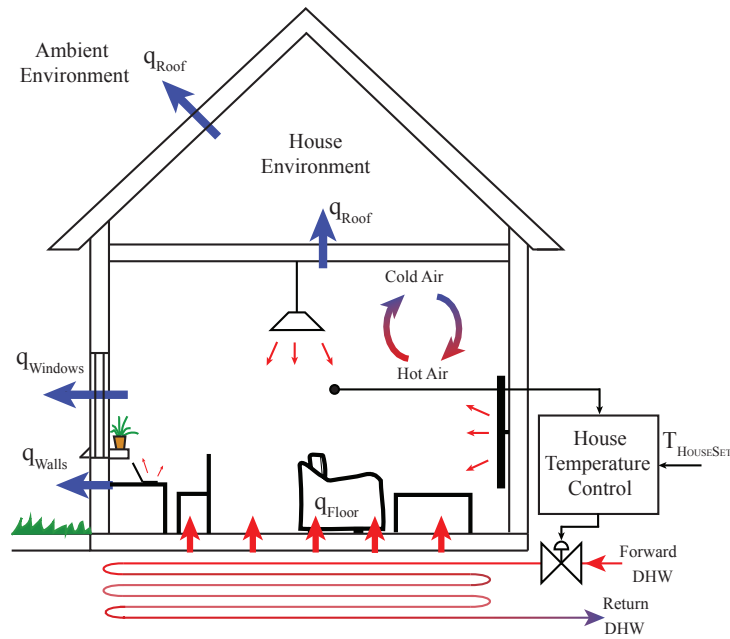


Figure 5.2: A simple model concept of a consumer supplied by district heating using floor-heating.

From the above consumer description the following assumptions and simplifications are chosen for the consumer model.

Assumptions and Simplifications

- **Domestic water:** Heating of domestic water by district heating is not included in the model. These heat losses are stochastic step like disturbances from when people take showers ect. They are less dominant compared to the ambient heat losses. Including this in the model will unnecessary increase the model complexity and the heat consumption from heating of domestic water is including in the dimensioning of the ambient heat loss model. Doing this simplification excludes this step disturbance though it can be compensated for e.g. by stepping on the indoor temperature and in this way generate a similar disturbance response.
- **Heat sources:** Radiators are neglected and only floor heating is modelled. The heat contribution of possible radiators are included in the floor heating. The objective is to deliver heat and it is of less importance to model how this is achieved i.e. if it is through floor heating or radiators. Modelling of a radiator also seems complex compared to a floor heating where heat is evenly disturbed in a relative simple geometry. The thermal dynamics and heat losses must of course fit that of a real house.
- **House dimensions:** The consumer house is modelled as a rectangle with an equal wall length and heigh. The roof is 150 % area of the floor area assumption that it is angled. This simplifies dimensioning and it is assumed that a rectangle makes a somewhat good approximation of general houses.
- **Heat losses:** The most dominant heat losses are to the ambient environment and therefore are heat losses to walls and roof modelled. All other less dominant heat losses are neglected. In

reality there are heat losses through everything and even heat gains from stoves, fireplaces, people, lighting, and consumer electronics. The idea is as long the total heat losses are represented it is of less importance how. Heat loss to the ground is also neglected.

- *DH distribution grid*: It is chosen not to modelled the DH distribution grid in order to simplify the modelling. Modelling of the distribution grid is not necessary in order to test the functionalities of a production balancing controller. This choice also fits well with scaling of the consumer e.g. to fit the consumption of a real area.

Design and Implementation

The consumer model is designed and implemented in Dymola as two layer component. The diagram layer and an icon layer as seen in Figure 5.3 and Figure 5.7 respectively.

The diagram layer includes the consumer temperature control, wall heat loss model, roof heat loss model, thermal house dynamics, floor-air convection model, and floor heating model. Each model is construed from thermal and fluid component models. All component models are explained in Section 5.5.

The temperature control is implemented as a PI-Controller by measuring from `Temperature sensor` as seen in Figure 5.3. The sensor is placed inside the house and the PI-Controller actuates on mass flow by adjusting opening levels of `Consumer valve`. `port_a` is connected to the DH supply pipes and hot water flows in and heat exchanges with the floor and leaves through the return pipes at `port_b`. Heat from the floor is transferred to the air using a `convection model`. A thermal capacitor, `Thermal capacitor house`, is included in order to capture temperature dynamics of the house. It is dimensioned to be twice the thermal capacitance of the volume of air inside the house. It is doubled in order to accommodate for thermal capacitances in furnitures and other materials.

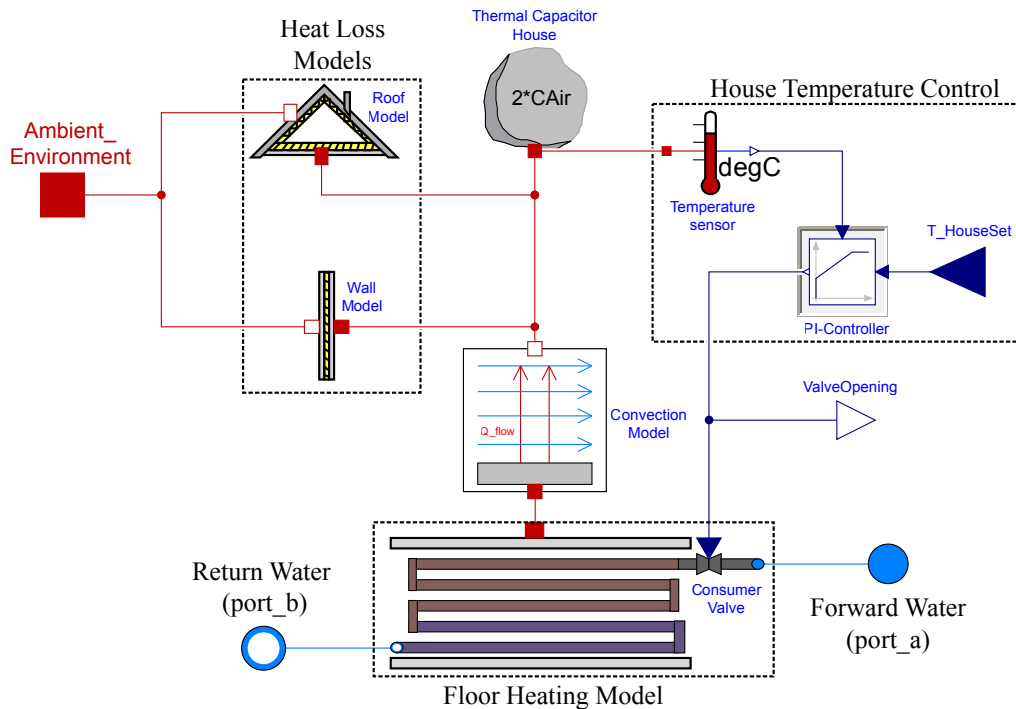


Figure 5.3: Diagram layer for the implemented consumer model with temperature control and heat losses to the outside environment.

Heat losses to the ambient environment are modelled with a wall and a roof model. The thermal connector `Ambient_Environment` represents the ambient temperature and the total heat loss either of which can be fixed outside the consumer model. If the ambient temperature is fixed the heat flow rate Q will be calculated and vice versa. The thermal connection to the outside can also be seen in the icon layer in Figure 5.7. In the next sections are the wall, roof, and floor heating model explained.

Floor Heating Model

The floor heating model has the purpose of describing how heat flows through the floor into the convection model. A `Conduction Floor Model` is used to model the thermal resistance of the floor material. The floor material is chosen as concrete. The heat capacitance, `Thermal Capacitor Floor`, is included to capture the temperature dynamics of the concrete floor. The concrete thermal capacitance is important in order to capture the overall thermal dynamics of the consumer because it is one of the dominating dynamics. The `consumer valve` is included to adjust flow of water. The pipe model component represents the floor heating pipes. The pipes are modelled to be 100 m long circular pipes with a diameter of 0.015 m. Further more the pipe is split into 1 discrete segments along the flow path. For the segment one total mass balance and energy equation is defined. This discrete segment could be increased to improve accuracy of the model though it is not wanted here as it is the overall dynamics which are of interest. An ideal heat transfer model is chosen for the thermal heat transfer from the water to the concrete floor. This choice is because the thermal conductivity of the concrete is much more dominant than the water and pipe walls and is therefore neglectable. For more information about chosen heat transfer and also flow

models see Subsection 5.5.1. The icon seen in Figure 5.3 is shown again for each model in the top right corner.

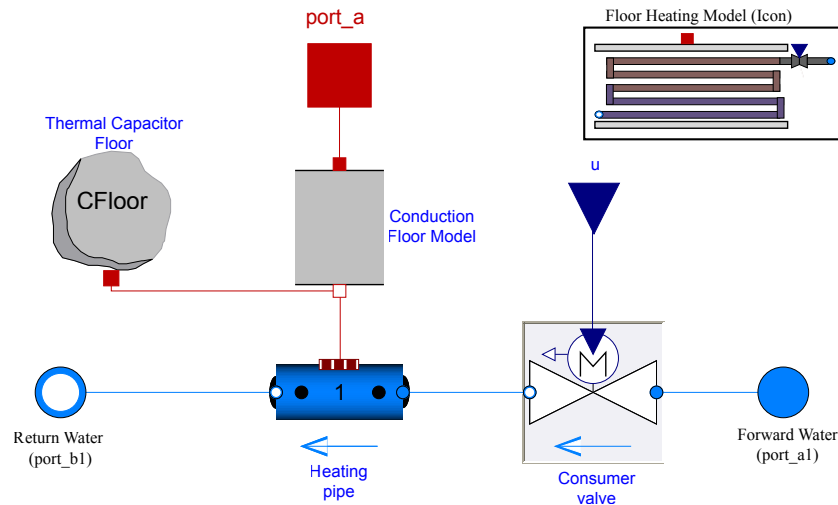


Figure 5.4: Floor heating Modelice model.

Roof Model

The roof is modelled as three conduction models in series. An air foil is assumed to be existing in the roof which helps with insulation. The roof consist of a wood plating together with mineral wool insulation. Dimensioning values for the model parameters, thickness, thermal conductivity constants, and thermal resistances are seen in Table 5.1.

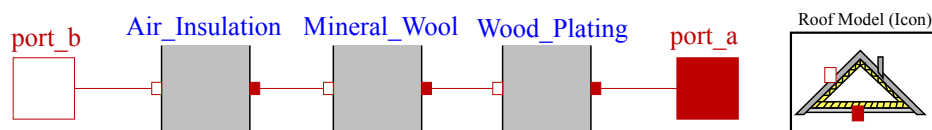


Figure 5.5: Roof conductive model with 3 thermal resistances in series.

The total thermal resistance for the roof can be calculated as the sum of resistances of each conductive model [Serway and John W. Jewett, 2010]. The calculation results are given in Table 5.1.

$$R_{Roof} = R_{wp} + R_{mw} + R_{ai} = \frac{h_{wp}}{k_{wp}} + \frac{h_{mw}}{k_{mw}} + \frac{h_{ai}}{k_{ai}} \quad (5.1)$$

Some values have been adjusted in order to account for the heat losses not model with the roof and walls models in order to have a average heat loss for the entire model.

	where	Values	Unit	Source
R_{wp}	is thermal resistance of wood plating.	0.33	$[m^2 K/W]$	Calculation
R_{mw}	is thermal resistance of mineral wool insulation.	5.0	$[m^2 K/W]$	Calculation
R_{ai}	is thermal resistance of air foil insulation.	1.95	$[m^2 K/W]$	Calculation
k_{wp}	is thermal conductivity of wood plating.	0.15	$[W/mK]$	[Haynes, 2014]
k_{mw}	is thermal conductivity of mineral wool insulation.	0.04	$[W/mK]$	[Haynes, 2014]
k_{ai}	is thermal conductivity of air foil insulation.	0.0257	$[W/mK]$	[Haynes, 2014]
h_{wp}	is thickness of wood plating.	0.05	$[m]$	Design
h_{mw}	is thickness of mineral wool insulation.	0.3	$[m]$	Design
h_{ai}	is thickness of air foil insulation.	0.1	$[m]$	Design
A_{Roof}	is area of the roof.	150	$[m^2]$	Design

Table 5.1: Parameter table for consumer roof model.

Wall Model

The wall model is very similar to the roof model. It has three conduction models in series. An outer and inner brick wall with insulation between. The thickness, thermal conductivity, thermal resistance, and area for each material is listed in Table 5.2. The thermal conductivity of the air is the same as for the roof model.

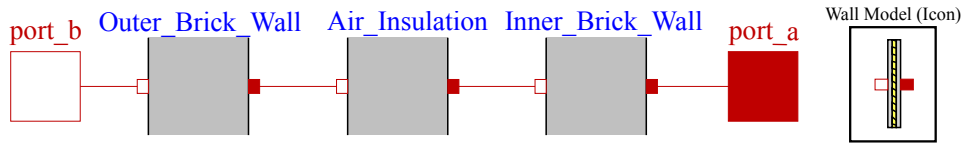


Figure 5.6: Conductive model of the consumers walls.

The thermal resistance for each conductive model is therefore given as.

$$R_{Wall} = R_{bw} + R_{ai} = \frac{h_{bw}}{k_{bw}} + \frac{h_{ai}}{k_{ai}} \tag{5.2}$$

The consumer model also has an icon layer. The interface connections to this layer are written in Table 5.3. The icon layer simply represents a few parallel consumer households together with input and outputs. R_{Roof}

	where	Values	Unit	Source
R_{bw}	is thermal resistance of brick wall	0.13	$[m^2 K/W]$	Calculation
R_a	is thermal resistance of the air foil insulation	4.12	$[m^2 K/W]$	Calculation
k_{bw}	is thermal conductivity of brick wall	0.8	$[W/mK]$	[engineeringtoolbox.com, 2015]
h_{bw}	is thickness of brick wall	0.1	$[m]$	Design
A_{Wall}	is area of a wall.	100	$[m]$	Design

Table 5.2: Parameter table for consumer roof model.



Figure 5.7: Icon layer for the consumer model.

Consumer Model Interfaces (Connectors)

Type	Name	Description
Real Input	T_HouseSet	The consumers desired temperature inside the house.
Thermal connector	Ambient_Environment	The heat loss to the output side environment.
Real output	ValveOpening	Measured opening level of consumer valve.
Fluid connector	port_a	Forward DHW.
Fluid connector	port_b	Return DHW.

Table 5.3: Interface table for consumer component model

House Geometry

The average household floor area of detached houses, terraced houses, and flats in Denmark is 101 m² [Petersen and Gram-Hanssen, 2005]. The consumer household floor area is therefore modelled to be 100 m² or as a square house of 10 by 10 m. The roof is modelled to be 150 % of floor area which gives a roof area of 150 m² due to a roof pitch of roughly 45 degrees. The height of the house is chosen to be 2.5 meters. From these quantities necessary masses, areas, and volumes are calculated. Densities, specific heat capacities, and thermal conductivities are found from look-up tables in Modelica.

Heat Losses

The average annual heat consumption of detached houses, terraced houses, and flats in Denmark is 12524 kWh/year [Petersen and Gram-Hanssen, 2005]. The hourly consumption or heat loss is therefore given as:

$$Q_{D,1h} = \frac{12524 \text{ kWh/year}}{365 \text{ d} \cdot 24 \text{ h}} = 1.42968 \text{ kWh/h} \sim 1429.68 \text{ W} \quad (5.3)$$

where

$Q_{D,1h}$ is consumer heat demand over one hour. [W]

The consumer household is therefore dimensioned to heat loss of approximately 1400 W at an ambient temperature of -12 °C. The standard for dimensioning heat losses is done at an ambient temperature

of -12 °C and an indoor living area temperature of 20 °C [Winther et al., 2012] [DANSK INGENIØRFORENING, 1968]. The heat loss of 1400 W should be represented from the roof and walls heat loss models. If it was modelled exactly the consumer heat losses should all be governed by different heat loss models but since there is only derived models for roof and walls the total heat loss must be fitted onto these two.

Temperature Control

A PI-Controller works as temperature control to achieve the desired indoor temperature of 20 °C. The controller have a minimum and maximum saturation ranging from 0 to 1 where 0 represent a fully closed valve and 1 a fully open valve. If limits are reached anti-integrator windup compensation is applied to drive the integrator state to zero. The temperature sensor feeds back a measurement of the indoor house temperature. The PI-Controller is tuned by using the normal gain k and integrator time constant and T_i . It is import that this controller is tuned to given an realistic thermal response e.g. if there is a step in the indoor temperature. For a 1 °C step in the indoor temperature it is should approximately an hour.

Consumer Scalability

The consumer model is made scalable in heat consumption by having an integer parameter called n_scale which denotes the amount of n houses in parallel connected to the supply line. This parameter scales up all areas, nominal flows, thermal capacities and the number of parallel floor heating pipes. Increasing the areas of the walls, roof and floor increases the heat flow through them and thereby scales up the total heat loss. The thermal capacitances of the house and concrete floor are also scaled up in order to maintain equal time constants of 1 or n parallel consumers. The nominal mass flow through the consumer valve is scaled up to take in more hot DH water as the numer of house increases. This is also done to have one valve adjusting the total consumer heat demand. The nominal mass flow through the valve is dimensioned for one consumer to have the capacity to deliver the required 1400 W as seen in (5.4). It is assumed that the return water temperature is approximately 40 °C. As the model is scaled up the the nominal mass flow together with number of pipes is scaled up as well as seen in (5.5).

$$m_{cv,nom} = \frac{Q_{P,1h}}{C_{p,w} \Delta T} = \frac{1400 \text{ W}}{4186 \cdot 40} = 0.00836 \text{ kg/s} \quad (5.4)$$

$$m_{cv,scaled} = n_scale \cdot m_{cv,nom} \quad (5.5)$$

where

$Q_{P,1h}$	is the heat production over one hour.	[W]
$C_{p,w}$	is the specific heat for water.	[J/(kg K)]
$m_{cv,nom}$	is the nominal mass flow through consumer valve.	[kg/s]
$m_{cv,scaled}$	is the scaled nominal mass flow through consumer valve.	[kg/s]

Sønderborg Outdoor Temperature Data

The ambient temperature data is acquired from SDH and it is measurements taken from the Sønderborg area. An example sequence of this data is seen in Figure 5.8 ranging from 18-03-2014 to 27-03-2014. The temperature time series is applied to the thermal connector `Ambient_Environment` in order to calculate the corresponding heat flow. Data for several years are available from SDH.

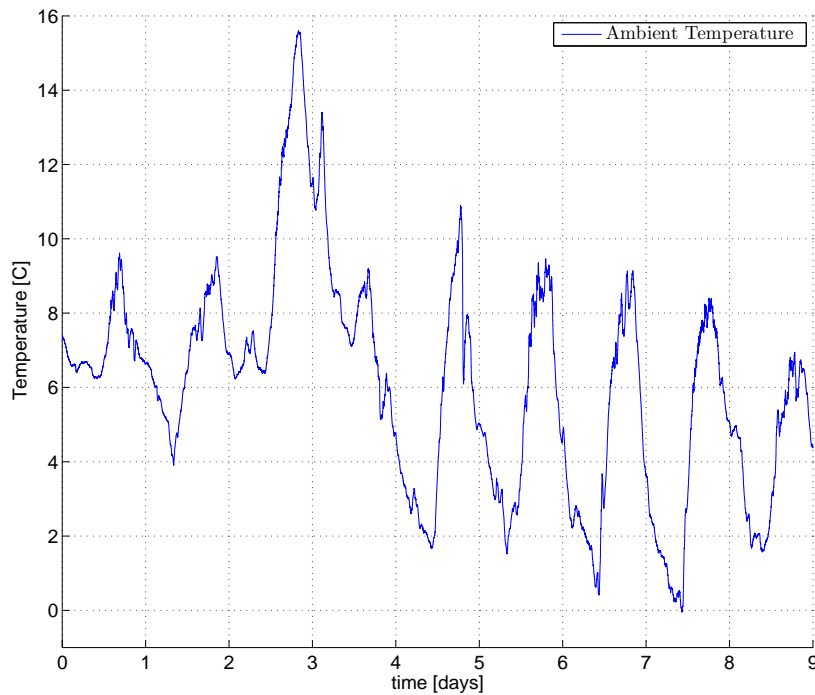


Figure 5.8: Example of Sønderborg ambient temperature data from 18-03-2014 to 27-03-2014.

5.2.1 Model Verification

In the following section three verification tests are made. One test for scalability, a step response test to verify the dynamics and a test where the ambient temperature data is applied to the model. The tests are done with the Dymola test configuration seen in Figure 5.9. The consumer model is placed between the source and the sink. The source resembles the forward DHW conditions with an outlet temperature of 80 °C at 6.6 bars and the sink represent the return conditions of 40 °C at 6 bars. The outdoor temperature `T_Ambient` is set to -12 °C and the indoor temperature `T_HouserSet` to 20 °C.

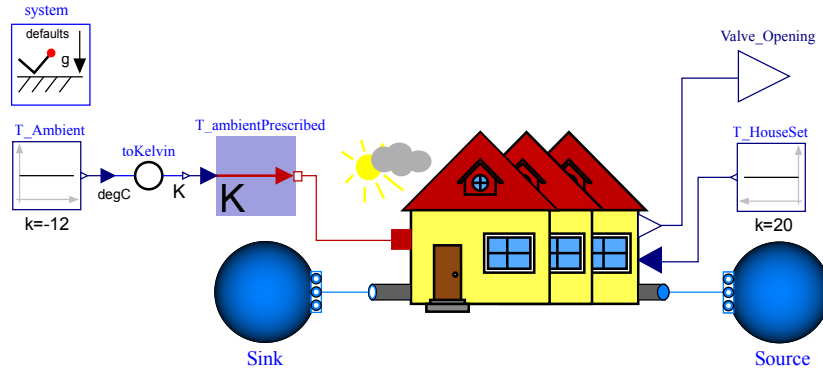


Figure 5.9: Consumer model verification Dymola test setup.

Verification Tests

1. Scalability test: In Figure 5.10(a) n_scale is set to 1 and the nominal heat losses are distributed between wall and roof models. The total heat loss seen from the black curve in reaches very close to 1400.1 W which is a bit lower than the previously mean consumption of 1429.68 W but sufficiently close enough. This is done with very little calibration of the heat losses. The heat losses should be somewhat less than the true heat loss since the models does not include heat losses to the ground, doors, openings ect. The heat loss through the wall model is 738 W and 654 W for the roof model. All heat losses are in Figure 5.10 are negative due to the sign convention which Modelica uses. In Modelica positive flows are defined as flows into a component and negative flows as flows out of a component.

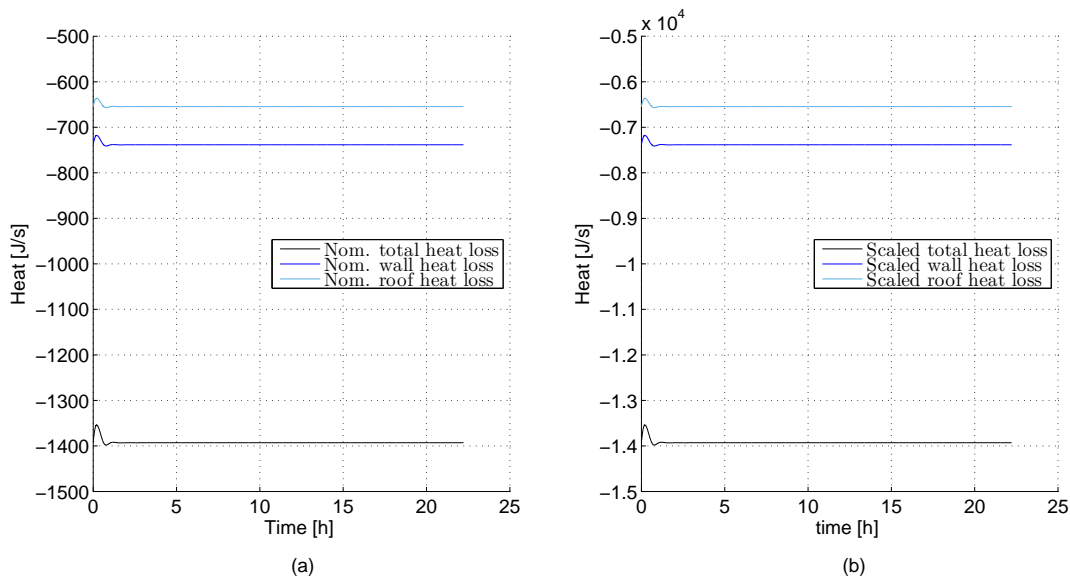


Figure 5.10: (a): Total heat loss together with heat losses from roof and wall model. Scaled by a factor 1, $n_scale = 1$. (b): Scaled by a factor 10, $n_scale = 10$.

In Figure 5.10(b) the heat loss is scaled up by a factor of 10 and n_scale is set to 10. The total heat loss is now scaled up to 13930 W resembling 10 households in parallel. It takes a bit of time

before the dynamics reach steady state due to some initial values are a bit off.

2. *Step response test:* In this second test a step in the indoor house temperature set point is made in order to verify the nominal and scaled dynamics. It seems reasonable that it should take a bit around an hour to change the house temperature from 20 °C to 21 °C. The temperature step occurs at the 2.7 hour and reaches 21 °C an hour later as seen in Figure 5.11(a). In Figure 5.11(d) the system is scaled up and the dynamics are maintained. The valve also acts the same as seen from Figure 5.11(b) and (e). The mass flow in Figure 5.11(f) is scaled with a factor 10 because n_scale is 10.

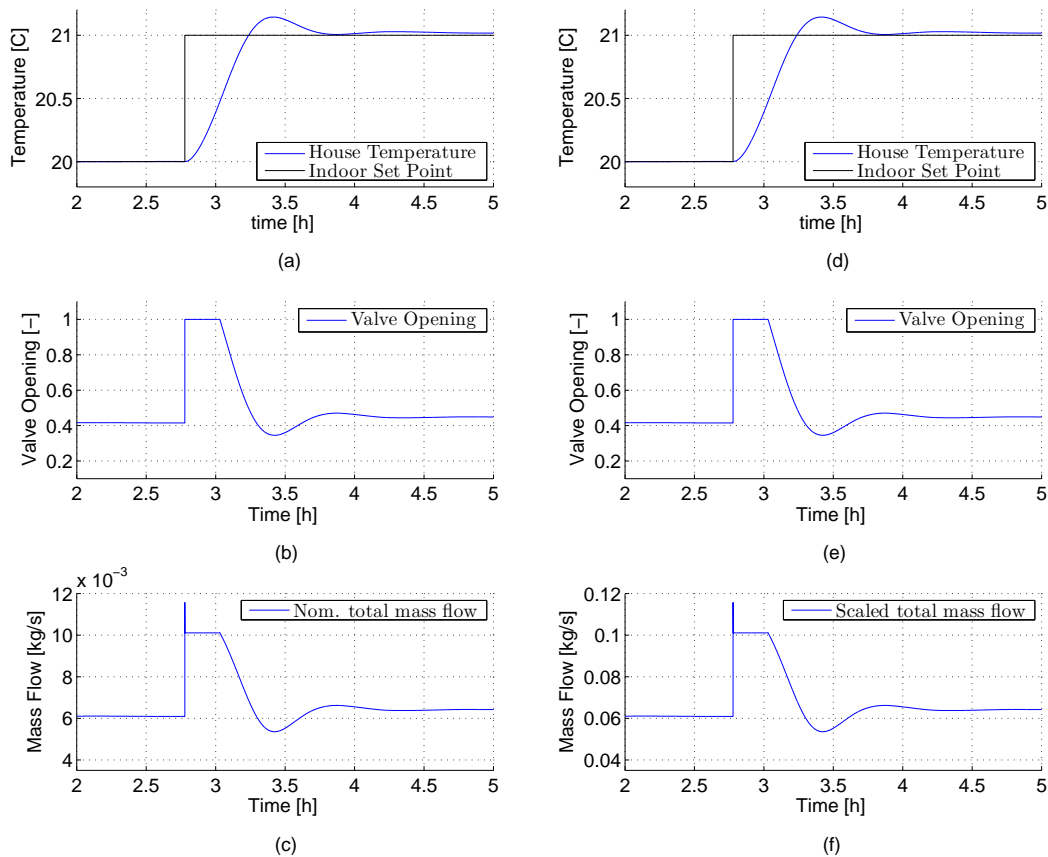


Figure 5.11: Measured consumer house temperature. An indoor temperature step occurs at the 0.7 hour. a,b,c are scaled by a factor 1 and d,e,f are scaled by a factor 10.

From the step responses in Figure 5.11 can high-order response be seen. This is due to the thermal inertias in the water pipes, concrete floor, house air, and controller dynamics. The dominating dynamics are the concrete floor and air. The lesser dominant are the water and pipe walls and controller dynamics. This is because the steel in pipe wall have a relative low mass with a high thermal conductivity versus the large mass of the concrete floor with a low conductivity. The PI-Controller also adds additional non-linear dynamics by using a saturated valve to generate the response. The valve opening degree in Figure 5.11 (b) and (e) shows how the controller tries generate a faster response but reaches the physical limitations of the valve. After the quick opening

in the valve to raise the temperature the opening valve converges to a value 0.45, a bit above from before the step. The mass flow follows the valve opening which makes sense.

3. *Applied ambient temperature data:* The last test verifies how consumers heat consumption is decided from T_{Ambient} using the temperature data from Sønderborg. n_{scale} is set to 1. The weather data is 10 days in may 2014, also plotted back in Figure 5.8 and is shown in the plot (a) of Figure 5.12. In plot (b) it can be seen how consumption decrease and increases relative to the ambient temperatures. During high ambient temperatures low heat losses occurs. The house temperature controller continuously accommodate the changes in heat losses by adjusting the consumer valve continuously in order to maintain the 20 degrees inside the house and this is plotted in (c).

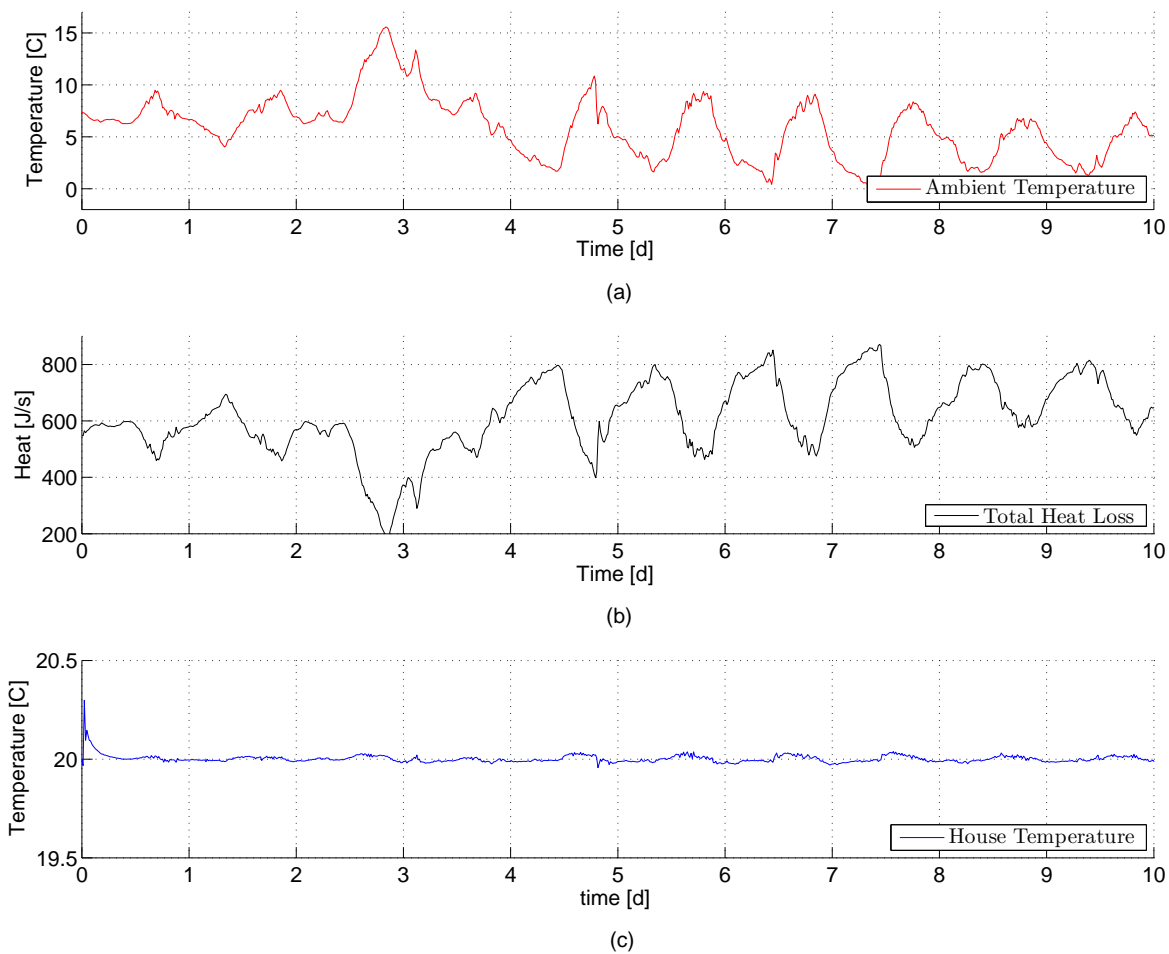


Figure 5.12: Consumer model with ambient temperature, heat consumption, and house temperature

This disturbance test concludes the modelling and verification for the consumer. In the section this is done for the producer unit is a similar approach.

5.3 Producer Model

This section contains the modelling and development of a deterministic DH producer unit. The producer unit design is based upon one of SDHs boiler stations and the design is similar to the hot water boiler tank illustrated in Figure 5.13. The producer unit must obey two fundamental requirements.

- *Forward supply temperature:* The forward supply temperature must be in between 75-82°C. This gives the option of decreasing or increase relative to transport losses and seasonal temperatures. This limited interval also ensures lesser thermal stress on pipes because the temperature can not vary a lot and thereby reducing maintenance costs in the DH pipe network.
- *Pressure difference:* The producer must maintain terminal pressure difference between 0.3-0.6 bar. This requirement ensure that the heat demands is satisfied and pressure are kept within safe operating bounds.

The hot water boiler is assumed to be a smoke tube boiler, i.e. a closed tank with an inlet, outlet and heating flue gas pipes inside the tank. The flue gas is generated in the furnace which can be fired e.g. by nature-gas or oil. A pump feeds Return District Water (RDW) into the tank to ensure flow and to create terminal pressure difference as seen in Figure 5.13.

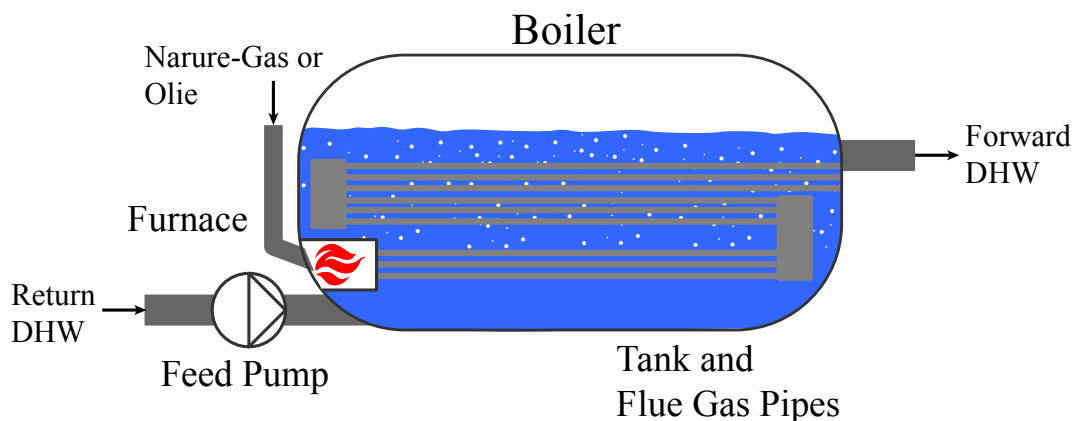


Figure 5.13: Illustration of a hot water boiler with flue gas pipes heating the water inside a tank. It could also be reversed with a tanked filled with flue gas and heating water inside pipes.

Two controllers are included to ensure the forward temperatures and terminal pressure difference are within the required bounds. In real distribution systems there always is a production which controls the pressure connected to the network. The forward temperature is measured at the outlet of the tank and the return water is reheated to the forward supply temperature set point, $T_{ForwardSet}$. The required heat is transferred to the water relative to the mass flow. If there is a large mass flow more heat is required to heat the water to reach the specified temperature in time before the water is at the outlet. The terminal pressure difference is also measured and the mass flow is adjusted to accommodate for increases or decreases in DH demands.

Design and Implementation

A Modelica model of the discussed design thoughts is developed in this section. The boiler Modelica model is designed and implemented with an icon and diagram layer. These layers are shown in Figure 5.14 and Figure 5.15.

The icon layer: contains external interfaces and picture of the model. The icon layer has a set of both signal and fluid interfaces as seen from the triangles in Figure 5.14. A description of all interfaces are seen in Table 5.4. The external interfaces are blue triangles denoting causal inputs and white triangles causal outputs. The full blue and the blue/white circles are acausal fluid connectors. A small introduction to connectors is outlined in Appendix A Section A.

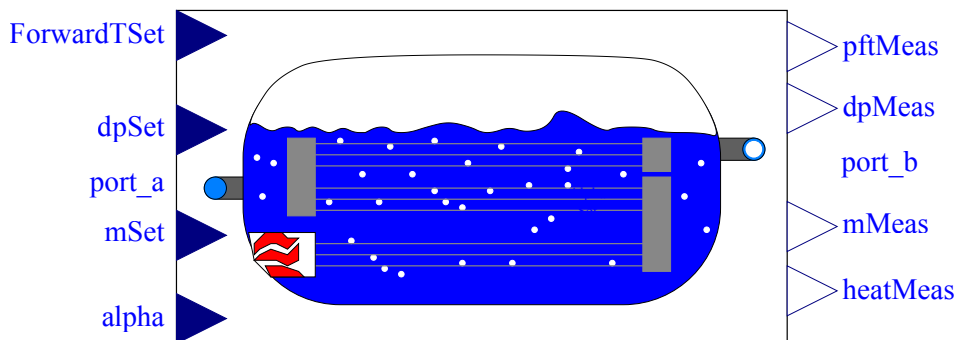


Figure 5.14: Layer 1: Icon and external interface layer.

Producer Interfaces (Connectors)

Type	Name	Description
Real Input	ForwardTSet	Producer forward temperature.
Real input	dpSet	Producer ΔP between terminals port_a and port_b
Real input	mSet	Set point for mass flow controller.
Real input	alpha	Alpha is a value in the interval [0;1] denoting the choice of controller.
Real output	ptfMeas	Measurement of the producer forward water temperature.
Real output	mMeas	Measurement of the mass flow thought the producer.
Real output	dpMeas	Measurement of the ΔP over the producer terminals.
Real output	heatMeas	Measurement of the heat going into the forward heating water.
Fluid connector	port_a	Return DHW.
Fluid connector	port_b	Forward DHW.

Table 5.4: Interface table for the producer unit.

The diagram layer: consist of temperature control, terminal pressure difference control, a water pipe storage component, and a set of Input/Outputs (I/Os). The forward temperature control indicated by the top blue dashed square in Figure 5.15 is a PI-Controller. The PI-Controller uses the measured forward temperature at `senTem` to generate the control signal passed on to the `prescribedHeatFlow` component. The control signal is a real value given to `prescribedHeatFlow`, which as the name indicates outputs a wanted heat flow. The pipe component has both a fluid interface and a heat transfer interface which makes it possible to heat fluid inside the pipe. The heat transfer between the `prescribedHeatFlow` output and the pipe heat port is modelled as a laminar and turbulent because it is suited for forced convection and

if both types of flows are occurring. The heat transfer coefficient is set sufficiently high because the pipe walls are thin. The pipe component is one of the base components which originates from the fluid library within Dymola and model is seen in Subsection 5.5.1. The pipe component in Figure 5.15 are the heating pipes inside the closed tank. The pipes are modelled to be 300 in parallel with a length of 5 m and a diameter of 0.015 m.

A description of all interfaces is seen in Table 5.4.

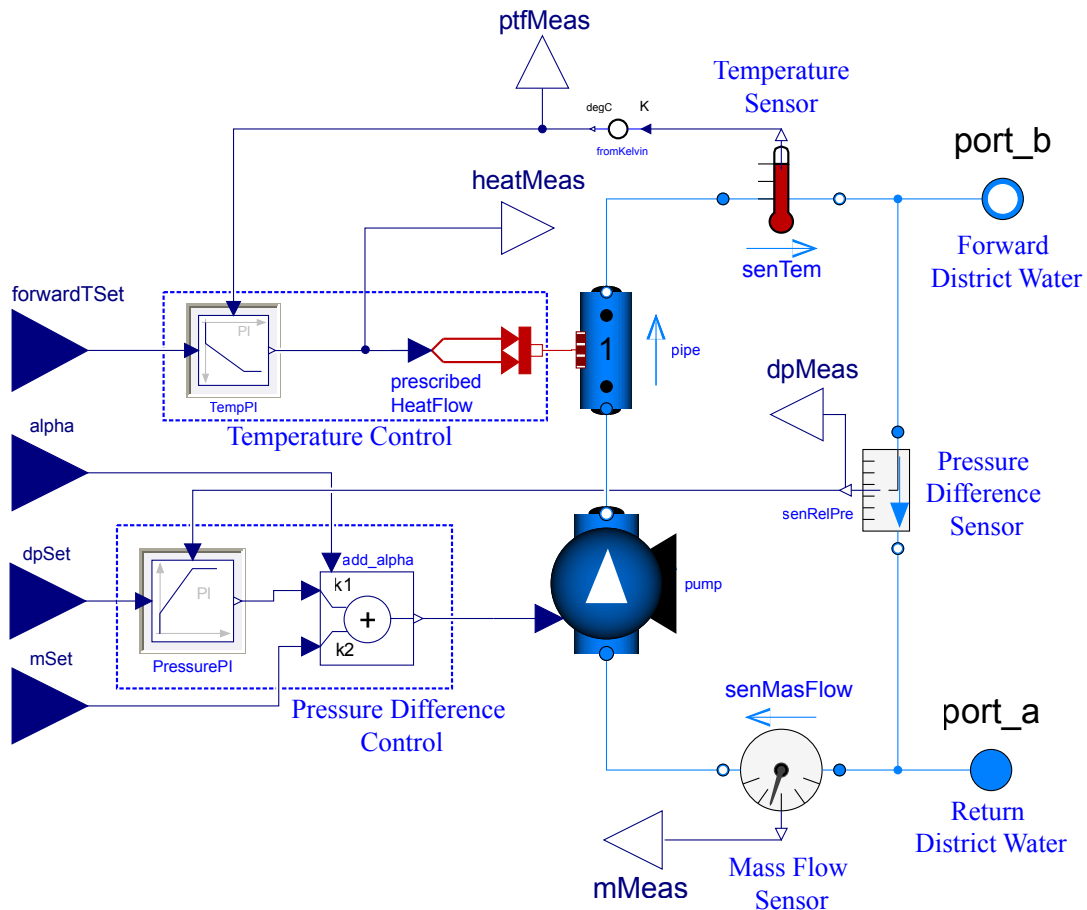


Figure 5.15: Layer 2: Producer unit component with temperature control.

The pressure control, in the bottom blue dashed square, can either control pressure difference or mass flow. The two controllers use the pump as actuator adjusting the mass flow. The pressure difference control uses the PI-Controller while the mass flow control uses an internal controller inside the pump. Switching between controllers is done by setting the input α to 1 or 0. If α is 1 the producer's mass flow is controlled from $dpSet$ if α is 0 then $mSet$. Choosing a value in between will generate a combined control signal from both of the references which should be avoided. Measurements of producer's terminal pressure difference is taken by $senRelPre$ and fed back to the PI-Controller. The mass flow measured through the pump, at $senMasFlow$, is made for analysis purposes. RDW flows in from $port_a$ in Figure 5.15 and out as Forward District Water (FDW) at $port_b$.

Model Verification

In this section the producer model is tested and verified. The tests are done with the Dymola test configuration seen in Figure 5.16. One producer unit is connected to a consumer valve and both of them are connected to a boundary model, *Sink*. In this test the producer must maintain the pressure difference set point, $dpSet$, and the forward temperature set point, $ForwardTSet$. The consumer valve works as a disturbance and describes heat consumption for a group of consumers. Alpha is set 1 which makes the producer control for pressure. The boundary sink specifies a pressure of 0.6 bar and a return water of 40 °C.

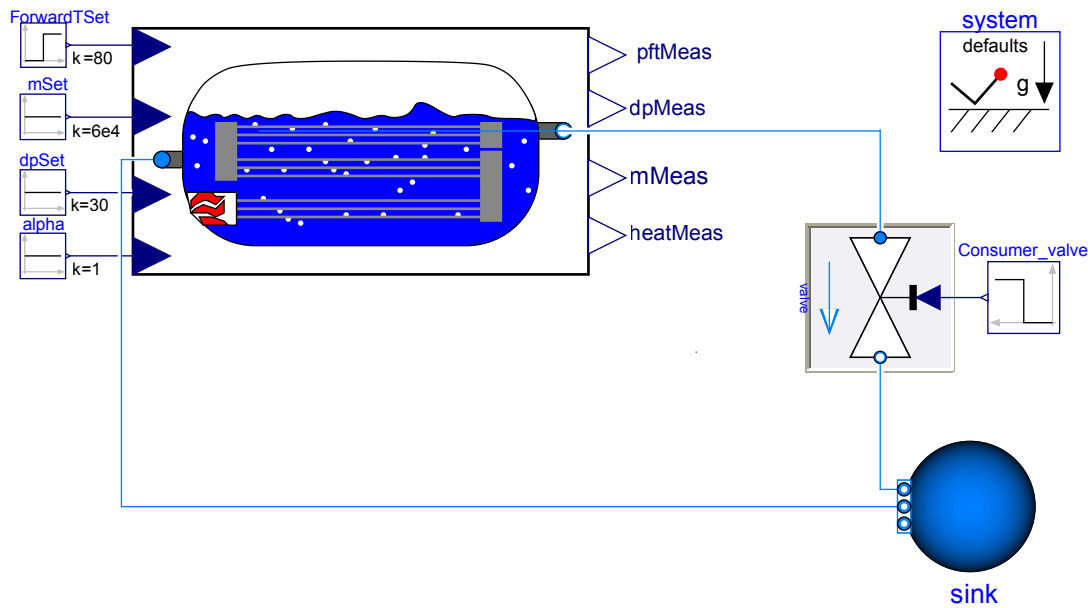


Figure 5.16: Producer model verification Dymola test setup.

Verification Test

The validation test consist of a step in forward temperature and a step in consumer consumption at 25%, 50%, and 100% consumer loads.

Step in forward temperature 80 to 81 °C: In order to verify the temperature control a step from 80 to 81 °C is made at 5.55 hour for three different loads as seen in Figure 5.17(a). At a 25% load it takes 1.5 minutes to reach the 81 °C. At 50 % load it takes around 4 minutes and at 100 % it takes close to 7 minutes. Ideally all step times should be around 1.5 minutes because it a realistic step time for a real boiler. In order to achieve this different tuned controllers should be applied at different loads. For simplicity it is chosen not to do this and only one controller is used. In Figure 5.17(e) is the measured heat plotted and at 5.55 hour the heat is increased a bit in order to raise the temperature. At all loads the mass flow controller uses approximately 1.5 minutes to step which is quite fine and realistic time period. A little faster at 25 % load and a bit slower at 100 %. Roughly the same times apply for the mass flow controlled boiler.

Step in consumption/valve opening: At 13.8 hour a step in the consumer valve opening is made. For the 25 % load the valve steps from 0.25 to 0.35, for the 50 % load the valve steps from 0.5 to 0.6, and For the 100 % load the valve steps from 0.9 to 1. As a consequence of the increased valve opening the pressure drops as seen in Figure 5.17(c) which is then compensated for by increasing the mass flow as seen in Figure 5.17(d). The increase in mass flow descresses the forward temperature for a short while until the temperature controllers catches up as shown in Figure 5.17(b). The required increase in temperature is done by increasing the input heat at hour 14 as seen in Figure 5.17(f).

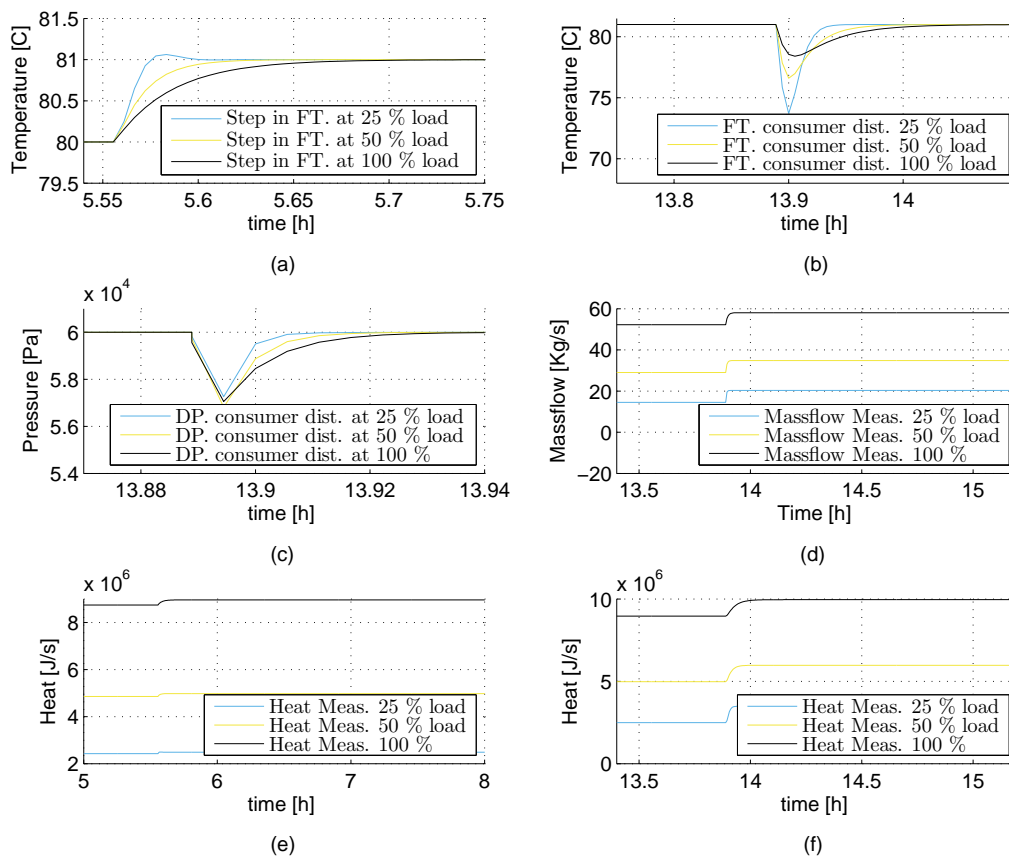


Figure 5.17: Pressure controller producer validation test results.

5.4 Accumulator Tank Model

In this section a hot water storage tank is modelled and implemented in Modelica code. The storage tank is connected to the return and supply line in parallel with the producers. The tank can be used to contribute with additional hot water in periods with high heat demands or when it is economic beneficial. The tank can also be used to take in or to put out additional return water in order to decreasing the return heat and to increase efficiency. When the tank is charged up it is slowly filled up with hot water forcing

cold return water out and decreasing the volume of cold water. The hot and cold water is separated by a relative thin thermocline layer where the temperature gradient is large and the temperature rapidly changes. These kind of tanks are referred to as stratified tanks. The tank can be charged and discharged with thermal energy as wanted given the limitations of the tank. The tank can also be charged fully with hot water before a weekend to give the option of weekend stop and supply from the tank during the weekend. An illustration of the tank is seen in Figure 5.18. The storage tank will have a heat loss to the air and this will be greater in high charged periods due to the increased hot water area.

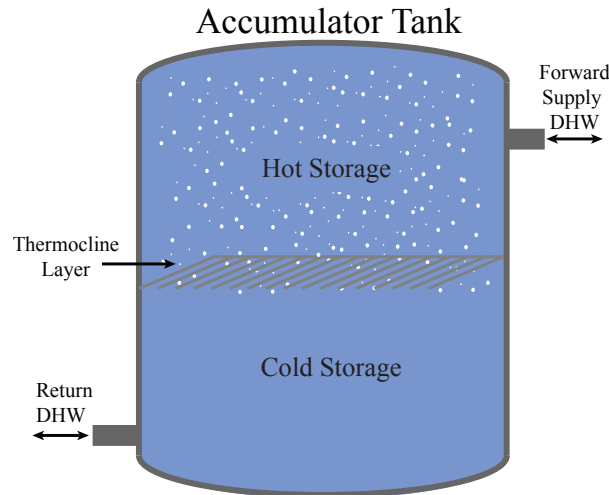


Figure 5.18: Accumulator tank for buffering hot water or cold water.

The accumulator tank unit must obey these two requirements.

- Contribution of hot and cold water: In reality the tank is equipped with an advanced set of charge and discharge pumps and valves to ensure a proper outlet temperature and pressure. In this strongly simplified model a fixed outlet temperature for charging and discharging are specified to be 80 to 90 °C and 30 - 40 °C respectively. This can be specified by the designer.
- Physical limitations: The tank must have a maximum and minimum storage capacity.

Design and Implementation

As for the other units the tank is also implemented in Modelica. The icon used for the tank model component is seen in Figure 5.19. The tank consist of the standard fluid connection to the return and supply lines. The have a input called `charge` which denote how much heat is charged e.g. setting `charge` to 1 for 10 seconds will change up 10 Joule and `AccEnergy` outputs this momentary amount of stored Joule in the tank.

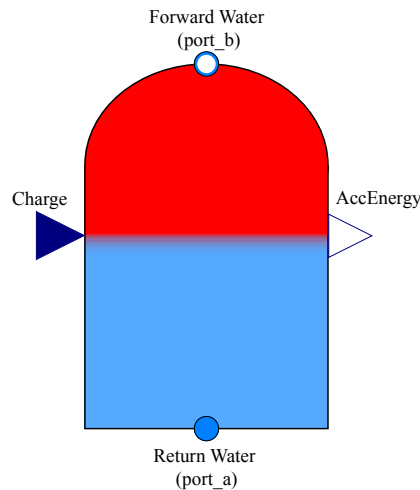


Figure 5.19: Accumulator tank for buffering hot water or cold water.

Producer Interfaces (Connectors)

Type	Name	Description
Real Input	Charge	Decides how much heat should be charged into the tank over a second.
Real input	AccEnergy	Outputs the current value of stored heat in the tank.
Fluid connector	port_a	Forward DHW.
Fluid connector	port_b	Return DHW.

Table 5.5: Interface table for the tank unit.

The tank implementation is purely implemented in modelica code rather than predefined components as seen in Code snip E.1.

```

1 model SimpleDHTank "Simple model of a district heating tank"
2
3 "Defines the choice of medium"
4 replaceable package Medium = Modelica.Media.Water.StandardWater constrainedby Modelica.Media.Interfaces.PartialMedium annotation (
   choicesAllMatching=true);
5
6 "Discharge exit temperature"
7 parameter SI.Temperature T_discharge=SI.Conversions.from_degC(90)
8
9 "Charge exit temperature (cold end)"
10 parameter SI.Temperature T_charge=SI.Conversions.from_degC(30)
11
12 parameter SI.Energy energy_start=10e6 "Initial energy contents";
13 SI.Energy energy(start=energy_start,min=0,max=100e6) "DH tank Energy";
14
15 "Define fluid interfaces"
16 Modelica.Fluid.Interfaces.FluidPort_a port_a(redeclare package Medium = Medium); "Connected to the cold return line"
17 Modelica.Fluid.Interfaces.FluidPort_b port_b(redeclare package Medium = Medium); "Connected to the hot supply line"
18
19 "Define real value interfaces"
20 Modelica.Blocks.Interfaces.RealInput charge;
21 Modelica.Blocks.Interfaces.RealOutput AccEnergy;
22
23 equation

```

```
24 "Mass balance"
25 0 = port_a.m_flow + port_b.m_flow;
26
27 "Energy balance"
28 0 = port_b.m_flow*actualStream(port_b.h_outflow) + port_a.m_flow*actualStream(port_a.h_outflow) - charge;
29 port_b.h_outflow = Medium.specificEnthalpy(Medium.setState_pTX(port_b.p,T_discharge));
30 port_a.h_outflow = Medium.specificEnthalpy(Medium.setState_pTX(port_a.p,T_charge));
31
32 der(energy) = charge;
33 energy = AccEnergy;
34 end SimpleDHTank;
```

Code snip 5.1: Source code for the Modelica accumulator tank model

In line 25 in Code snip E.1 a static mass balance is defined for the tank because an equal amount of water goes out when something goes in and that the tank is always filled up with incompressible water. A static energy balance is defined in line 28 with a bi-directional flow. The flow must be modelled with an operator which can handle bi-directional flows because the flow direction changes when charging and discharging the tank. The two enthalpies are found by look-up table from the defined charge and discharge temperature in line 10 and 7 together with the momentary pressure at each port. The `actualStream()` operator evaluates if the mass flow is out going (negative) or in going (positive) and finds the correct enthalpy a potential chain of previous enthalpies from other components. Setting a positive charge value corresponds to a positive mass flow into `port_b` and a negative mass flow into `port_a` i.e. the water is leaving the tank through `port_a`. The temperature/enthalpy of the water entering the tank through `port_b` is given by the other components connected to the tank. The enthalpy of the water leaving the tank is given by the fixed discharge temperature (90 °C) and the pressure on `port_a` given by the other components connected to the tank. The `AccEnergy` is given a maximum value of 100 MJ and a start value of 10 MJ.

Model Verification

In this section the tanks functionalities are verified. There are made two simple tests, one to test if the tank can charge and another to test discharge. The sink and source boundary models are used again. A value of $\pm 1e5$ is given to the `charge` input. The simulation is over a 100s this yields a changing and discharging scenario of $1e5$ Joule each second over the 100s.

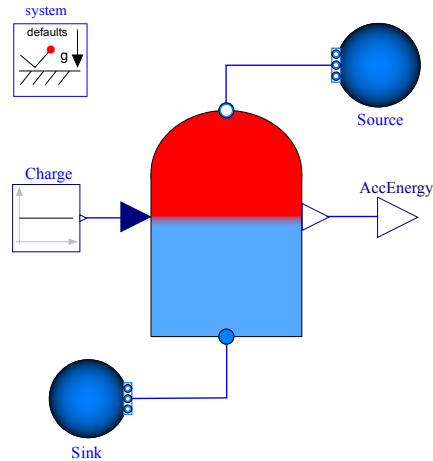


Figure 5.20: Accumulator tank for buffering hot water or cold water.

The step in charge occurs at $t = 10$ s as seen in Figure 5.21(a). The accumulated energy during the period is seen in (b) with a starting value of 10 MJ. The corresponding mass flows for port_b are seen in (c). port_b is positive when charging and negative in discharging periods as expected. The the cases are not done at the same time just plotted on the same graphs.

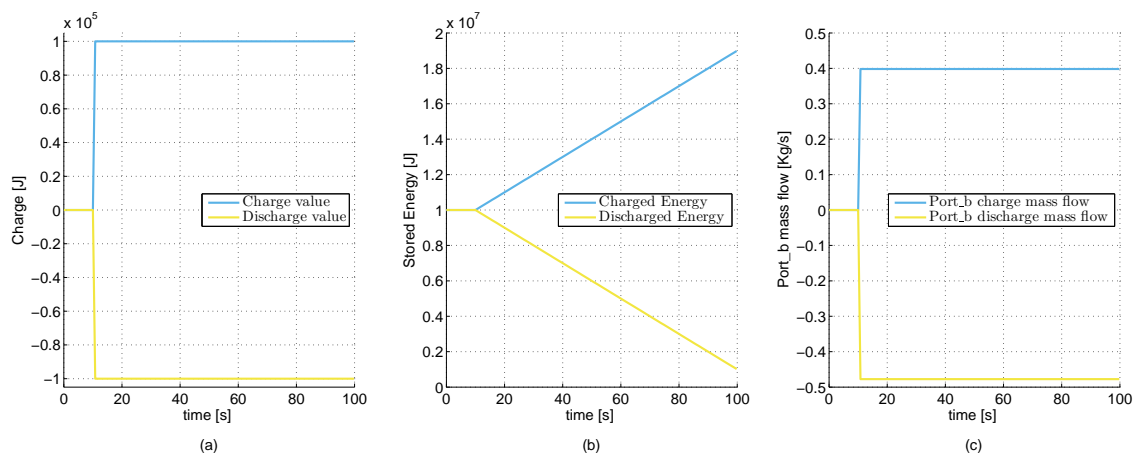


Figure 5.21: Accumulator tank verify test results.

5.5 Component Models

This section contains the modelling of the essential components from which the producer and consumer model is constructed from. In Table 5.6 an overview of the different components used in the previously section. For each of these components a simple model is derived.

Models	Pipe	Pump	Valve	Thermal Capacitance
Producer	X	X	-	-
Consumer	X	-	X	X

Table 5.6: This table contains an overview of the different component models which is necessary to represent the different system configuration.

All component models derived in this section is general models because the pipe component model is used in more than one system model.

5.5.1 Dynamic Pipe Model

The pipe model is required by almost all of the system models and the following models are included:

- Convective heat transfer model
- Dynamic mass balance
- Dynamic energy balance
- Static momentum balance

Convective Heat Transfer Model

Convection heat transfer occurs both in the producers and consumer models. There are two different types of convection models, namely forced and natural convection. Two forced convections appears, one in the heat transfer between input heat and RDW pipes in the producer and in the heat transfer between heating pipes and floor in the consumer. Forced convection refers to a mechanical generated mass flow of a fluid e.g. by use of a pump [Andersen and Pedersen, 2006]. In the consumer natural convection also appears between the floor and air. Natural convection is when density differences generates mass flows due to temperature variations.

The heat transfer for the power $q_p(t)$ is given by (5.6) [Andersen and Pedersen, 2006]

$$q_p(t) = UA [T_1(t) - T_2(t)] \tag{5.6}$$

where

- q_p is heat energy flow to or from the pipe to an adjacent component [J/s]
- A is the area of the heat transfer surface [m^2]
- U is heat transfer coefficient [$J/(s^\circ C m^2)$]
- $T_1(t)$ is the temperature of the pipe surface [$^\circ C$]
- $T_2(t)$ is the temperature of the pipes adjacent environment e.g. air [$^\circ C$]

The heat transfer coefficient U is a function of of fluid properties and pipe geometry but in many cases it

can be regarded as a constant [Andersen and Pedersen, 2006]. The Dymola DynamicPipe model provides the user with a selection of different heat transfer models:

- **IdealHeatTransfer:** Ideal heat transfer without thermal resistance.
- **ConstantHeatTransfer:** Constant heat transfer coefficient.
- **LocalPipeFlowHeatTransfer:** Laminar and turbulent forced convection in pipes.

The laminar and turbulent model captures makes the heat transfer coefficient a function of time relative to flow which is a more realistic model than the two others.

For the sake of simplicity terms like $q_p(t)$ is denoted without function of time as q_p for the rest of this chapter.

A figure of a Dymola dynamic pipe model with variables and parameters is seen in Figure 5.22. The upstream variables are defined with subscript a i.e. m_a for mass flow and the downstream with subscript b.

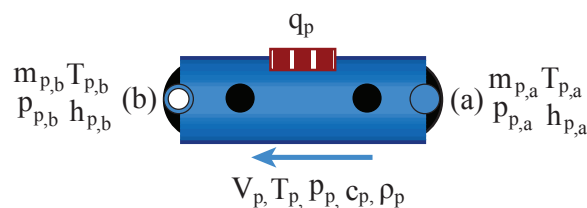


Figure 5.22: Dymola dynamic pipe model including internal, upstream and downstream variables for both fluid and heat interfaces.

where

port (a)	is a fluid connector where a positive design flow direction is from port a to port b	[-]
port (b)	is a fluid connector where a negative design flow direction is from port b to port a	[-]
$m_{p,a}$, $m_{p,b}$	is mass flow at the inlet port a and outlet port b	[kg/s]
$T_{p,a}$, $T_{p,b}$	is the temperature at inlet port a and outlet port b	[°C]
$P_{p,a}$, $P_{p,b}$	is pressure at inlet port a and outlet port b	[Pa]
$h_{p,a}$, $h_{p,b}$	is the specific enthalpy at inlet port a and outlet port b	[J/kg]
V_p	is the volume of the pipe	[m ³]
P_p	is the pressure inside the pipe	[Pa]
ρ_p	is density of the medium inside the pipe	[kg/m ³]

In Modelica connectors are used.

Dynamic Mass Balance

The model for the mass balance is chosen to be static because there is conservation of mass in the pipe. This done by choosing the dynamics of the pipe to be (massDynamics=SteadyState) in the Dymola

DynamicPipe model and solving the system for static mass balance. If this is not done the mass balance is given as the time derivative of the mass, $\frac{d(M_p)}{dt}$ with $m = v\rho$.

$$\frac{d(M_p)}{dt} = m_{p,a} + m_{p,b} = 0 \quad (5.7)$$

where

M_p is the mass the medium inside the pipe [kg]

Dynamic Energy Balance

The objective of the pipe is to transfer heat. In the consumer the heat is transferred into to the floor. For the production models the water inside the pipes is heated.

$$\frac{d(U_p)}{dt} = h_{p,a}m_{p,a} + h_{p,b}m_{p,b} + q_p \quad (5.8a)$$

$$U_p = M_p u_{p,b} \quad (5.8b)$$

$$u_p = cT_p \quad (5.8c)$$

where

$U_{p,a}$, $U_{p,b}$ is the internal energy of the medium in the pipe [J]

$u_{p,a}$, $u_{p,b}$ is the medium specific internal energy in the pipe [J/kg]

q_p is the added heat power to or from the pipe [J/s]

c is the specific heat capacity [J/(kg °C)]

T_p is the temperature inside the pipe [°C]

Static Momentum Balance

The momentum balance equation describes the relationship between pressure and flow. The equation is given as (5.9a). The subscript a denotes entering e.g. $h_{p,a}$ is the enthalpy entering the pipe and subscript

b denotes leaving.

$$\frac{d\phi}{dt} = \frac{d(M_p v_p)}{dt} = p_{p,a} A_p + p_{p,b} A_p - \Delta F_{fric} \Rightarrow \quad (5.9a)$$

$$V_p = A_p l_p \quad (5.9b)$$

$$v_p = \frac{m_p}{\rho_p A_p} \quad (5.9c)$$

$$M_p = \rho_p V_p \quad (5.9d)$$

$$\frac{1}{A_p} \frac{d\left(\rho_p A_p l_p \frac{m_p}{\rho_p A_p}\right)}{dt} = \frac{l_p}{A_p} \frac{d(m_p)}{dt} = p_{p,a} + p_{p,b} - \Delta_{p,fric} \quad (5.9e)$$

The flow friction model can be chosen to as:

- **NominalLinearFlow:** Linear pressure loss for nominal values.
- **NominalTurbulentPipeFlow:** Quadratic turbulent pressure loss for nominal values.
- **TurbulentPipeFlow:** Pipe wall friction in the quadratic turbulent regime (using mu to regularize laminar region).
- **DetailedPipeFlow:** Pipe wall friction in the laminar and turbulent regime (detailed characteristics).

The friction force $\Delta_{p,fric}$ is a function of material properties and geometry. The DetailedPipeFlow model is chosen in order to have a full model for friction which allows for both turbulent and laminar flows. The other models are more specific i.e. only for laminar or turbulent flow.

where

ϕ	is the linear momentum of the fluid	[kg m/s]
V_p	is the velocity of the medium inside the pipe	[m/s]
l_p	is the length of the pipe	[m]
A_p	is cross-sectional area of the pipe	[m ²]
$\Delta_{p,fric}$	is pressure loss duo to friction	[Pa]
d_p	is the diameter of the pipe	[m]
ρ	is the density of the medium inside the pipe	[kg/m ³]
V_p^2	is the volume of the medium inside the pipe	[m ³]
m_p	is the mass flow the medium inside the pipe	[kg/s]

In order to make the simulated system less stiff the dynamic part of the momentum balance is put equal to 0. This can be done because the time constant of this is very small compared to the energy balance and relative to what is of interest which is the overall system dynamics.

$$0 = p_{p,a} + p_{p,b} - \Delta_{p,fric} \quad (5.10)$$

5.5.2 Valve Model

The valve model is chosen to be a linear model seen in (5.11a) and the valve component is seen in Figure 5.23.

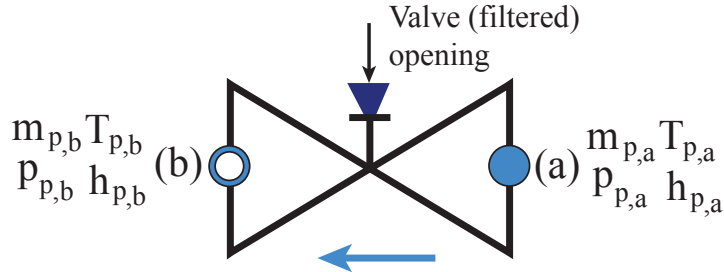


Figure 5.23: Representation of the valve component used in Dymola with two fluid connectors and an opening input.

There is linear behaviour between the pressure drop across the valve and the flow through it. The opening value reaching from $[0;1]$ can also be specified to be filtered meaning rather than a step response when opening the valve it opens as a first order response.

$$m_v = \alpha K_v (P_b - P_a) \quad \alpha \in [0; 1] \quad (5.11a)$$

$$m_v = \alpha K_v \Delta P_{b,a} \quad (5.11b)$$

$$K_v = \frac{m_{v,nom}}{\Delta p_{nom}} \quad (5.11c)$$

where

$P_b - P_a$, $\Delta P_{b,a}$	is the pressure drop across the valve	[Pa]
m_v	is the flow through the valve	[kg/s]
K_v	is the hydraulic conductance of the valve at full opening	[-]
α	denotes the position of the valve.	[-]
$m_{v,nom}$	is the nominal mass flow through the valve at full opening	[kg/s]
Δp_{nom}	is the nominal pressure drop across the valve at full opening	[Pa]

The valve model includes an isenthalpic state transformation i.e. no storage or loss of energy.

$$0 = h_{v,a} + h_{v,b} \quad (5.12)$$

and also a static mass balance.

$$0 = m_{v,a} + m_{v,b} \quad (5.13)$$

5.5.3 Pump Model

The pump is modelled as a device which can deliver a pressure difference and the Dymola component is seen in Figure 5.24.

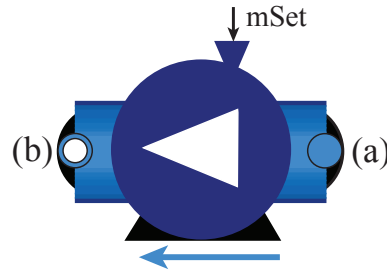


Figure 5.24: Mass flow controlled pump from Dymola with fluid connectors port a and port b.

The pumps head-flow characteristic at variable Revolutions Per Minute (RPM) is modelled as.

$$\Delta H = \frac{\Delta P_{pump}}{\Delta P_{nom}} = \left(\frac{N_{rs}}{N_{nom}} \right)^2 - \left(\frac{m}{m_{nom}} \right)^2 \quad (5.14a)$$

where

ΔP_{pump}	is the pressure drop across the pumps two ports	[Pa]
N_{rs}	is the rotational speed for flow characteristics of the pump	[RPM]
N_{nom}	is the nominal rotational speed for flow characteristics of the pump	[RPM]

The mass balance is given as for the pipe due to no loss or storage of water.

$$0 = m_{pump,a} + m_{pump,b} \quad (5.15)$$

The energy balance is given by.

$$\frac{d(U_{pump})}{dt} = h_{pump,a}m_{pump,a} + h_{pump,b}m_{pump,b} + W_{pump} \quad (5.16a)$$

$$(5.16b)$$

The pump adds work to the water due to friction.

5.5.4 Convection Model

Inside consumer model free convection or natural convection and the heat transfer is dependent on the flow of air molecules which occurs because of Buoyancy forces, or density differences due to variations of thermal temperature in the air. This makes the heat transfer coefficient U a function of temperature.

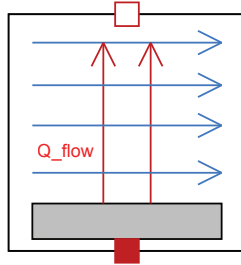


Figure 5.25: Dymola convection model with two heat ports.

The Buoyancy force heat transfer coefficient h for floors is given as.

$$h = k|\Delta T|^{\frac{1}{3}} \quad \text{with} \quad \begin{cases} k = 1.51 & \text{if } T_{Floor} > T_{Air} \\ k = 0.76 & : \text{ Otherwise} \end{cases} \quad (5.17)$$

The convective flux from the floor into the air is given as.

$$q_f = h\Delta T \quad (5.18)$$

where

q_f is the heat flow from the floor into the air [J/s]

ΔT is the floor temperature minus the air temperature [°C]

5.5.5 Thermal Capacitor Model

Heat capacity model. Geometry is only included for the total volume. The temperature is modelled as uniform in the volume. The heat capacity is also assumed independent of temperature.

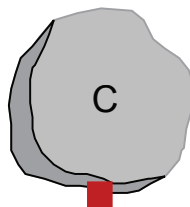


Figure 5.26: Dymola heat capacity model.

The thermal capacity is given as:

$$C = c_p M \quad (5.19)$$

where

C	is the heat capacity of the volume consisting of the modelled material.	[J/K]
C_p	is the specific heat capacity of the material.	[J/kg K]
M	is the mass.	[kg]

5.6 State Selection

Dymola automatically selects dynamic states dependent on the model. The most obvious choice is to select the variables that have time derivatives in a model, e.g. select x when the expression $\text{der}(x)$ appears in the model. In dynamic fluid systems the time derivatives of some of the fluid properties are often needed and Dymola is given a "hint" by the Modelica implementation of the fluid properties (the person who implemented them knows which variables are suitable). For models containing water/steam properties, pressure, p , and specific enthalpy, h , are "recommended" since these can be used to calculate all other properties see ???. The tedious and error prone work of expanding the dynamic mass and energy balances in thermo-hydraulic models is done automatically by Dymola. The default choice of states can be manually be changed in the model by using the selection command in Dymola [Dassault Systèmes AB, 2014] [Francesco Casella].

Example 5.1 (Manuel Selection of States)

If an ideal gas is used Dymola selects the recommended states by the fluid properties (p and T) the state expansion in (5.20a) and (5.20c) is done automatically:

Here p , T is chosen as states:

```
AbsolutePressure p(stateSelect = StateSelect)
```

```
Temperature T(stateSelect = StateSelect)
```

The mass and energy balance equations are defined and written out until the dynamic expression for pressure and temperature appears i.e. the state variables differentiated with respect to time.:

$$\frac{d(M)}{dt} = \frac{d(V\rho)}{dt} = V \left(\frac{\partial \rho}{\partial p} \frac{dp}{dt} + \frac{\partial \rho}{\partial T} \frac{dT}{dt} \right) \quad (5.20a)$$

$$\frac{d(U)}{dt} = \frac{d(Mu)}{dt} = \frac{d(V\rho c_p T)}{dt} = V \left(\rho c_p \frac{dT}{dt} + \rho T \frac{dc_p}{dt} + c_p T \frac{d\rho}{dt} \right) \quad (5.20b)$$

$$= V \left(\rho c_p \frac{dT}{dt} + \rho T \left(\underbrace{\frac{\partial c_p}{\partial T} \frac{dT}{dt} + \frac{\partial c_p}{\partial p} \frac{dp}{dt}}_{\frac{dc_p}{dt}} \right) + c_p T \left(\underbrace{\frac{\partial \rho}{\partial p} \frac{dp}{dt} + \frac{\partial \rho}{\partial T} \frac{dT}{dt}}_{\frac{d\rho}{dt}} \right) \right) \quad (5.20c)$$

The volume, V , is assumed to be constant and left out of the derivatives. The state selection command solves for the derivatives, indicated by red terms in (5.20a) and (5.20c), and isolates the dynamic state variables automatically.

Densities and specific enthalpies are found through look-up tables.

$$\rho = f_{\rho}(p, T), \quad c_p = f_{c_p}(p, T) \tag{5.21a}$$

In Dymola can these be accessed by calling functions e.g. by calling $c_p = \text{SpecificHeat_pT}(p, T)$ for the specific heat.

5.7 Combining the Models

In this section the producer and consumer model are combined in a closed fluid system in order model the return water and to have a final DH model. This will the how the return will vary in temperature relative to consumption and production demands mismatches. In Figure 5.27 is one producer, a accumulator tank, and the consumer model set work as a DH network.

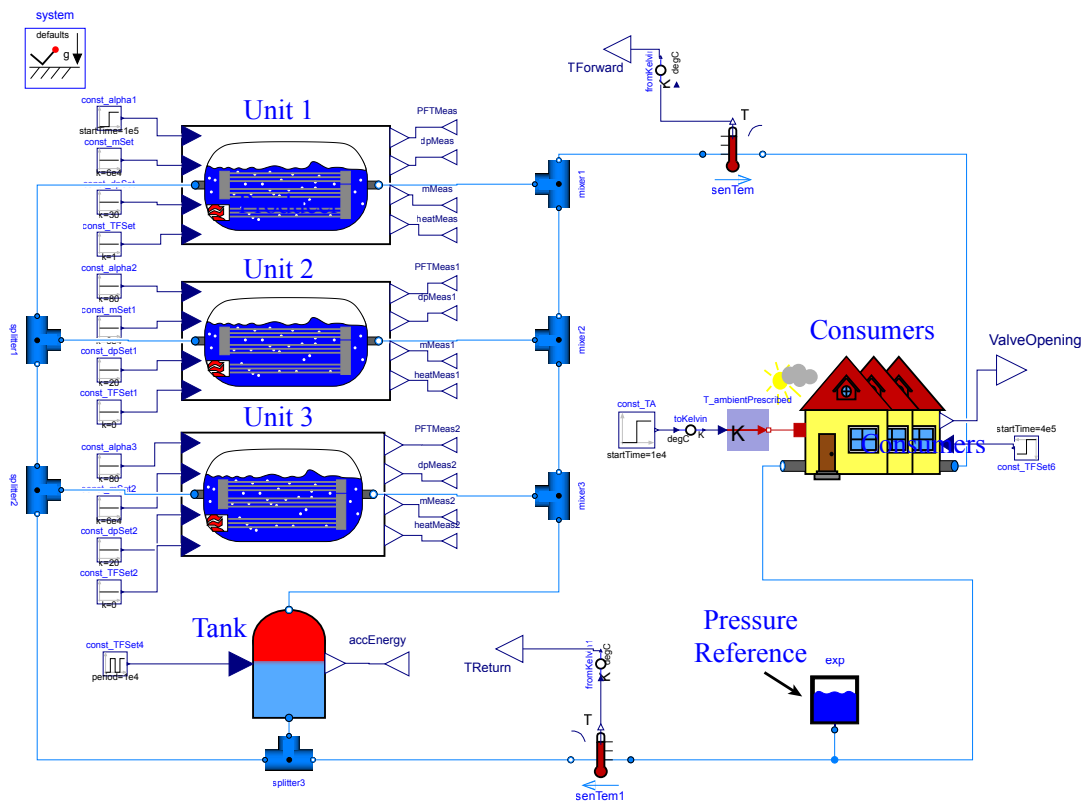


Figure 5.27: Dymola convection model with two heat ports.

In this combined model the pressure source exp is imposed. This is an expansion vessel which accounts for expansion of the fluid and adds a reference pressure to the model. This is similar to a ground point in circuit electronics. It is important for closed fluid systems that there only is one pressure reference because it leads to an over specified system. Without one it is not possible to calculate the pressure and the model will become structurally singular or in other words will fail in the initialization. Without the pressure reference the model only accounts for relative changes in pressure and not an absolute pressures.

Initializing the Model in Steady-State

It is possible to specify Dymola to make the model start in steady-state. This makes Dymola's quasi newton-rampson solver try to solve the system in steady-state. The quasi newton-rapson solver requires initial guess values. It can be quite hard give good starting valyes for complex thermodynamic fluid systems. If the initial values are good enough the solver converges and a steady-state solution is found. If not then the model is started up with dynamics. The dynamical system solver A RadauIIa - order 5 stiff dynamic is used to find a dynamic solution.

DH Model Verification

In order to test the functionalities of the combined model a series of input signals are applied to the model. This entails a step in the supply temperature, step in the consumption, and charging the accumulator tank.

Results

- At hour 28 a step in the supply temperature is made for unit 1. This is seen in Figure 5.28(a). The other two supply temperatures are maintained at 80 °C. This step makes the mass flow in Figure 5.28(b) drop a bit.
- A step in consumption is applied at hour 111. This is a step in the indoor temperature from 20 to 21 °C. In practise this is an increase in consumer valve opening and this makes the pressure drop as seen in Figure 5.28(c). The pressure is raised again by the increase in mass flow from unit 1 seen in Figure 5.28(b) short time after hour 111. As a consequence of the indoor house temperature increase so does the return water temperature increase as seen in Figure 5.28(d).
- At hour 277 the tank is charged with 990 MW.

5.7 Combining the Models

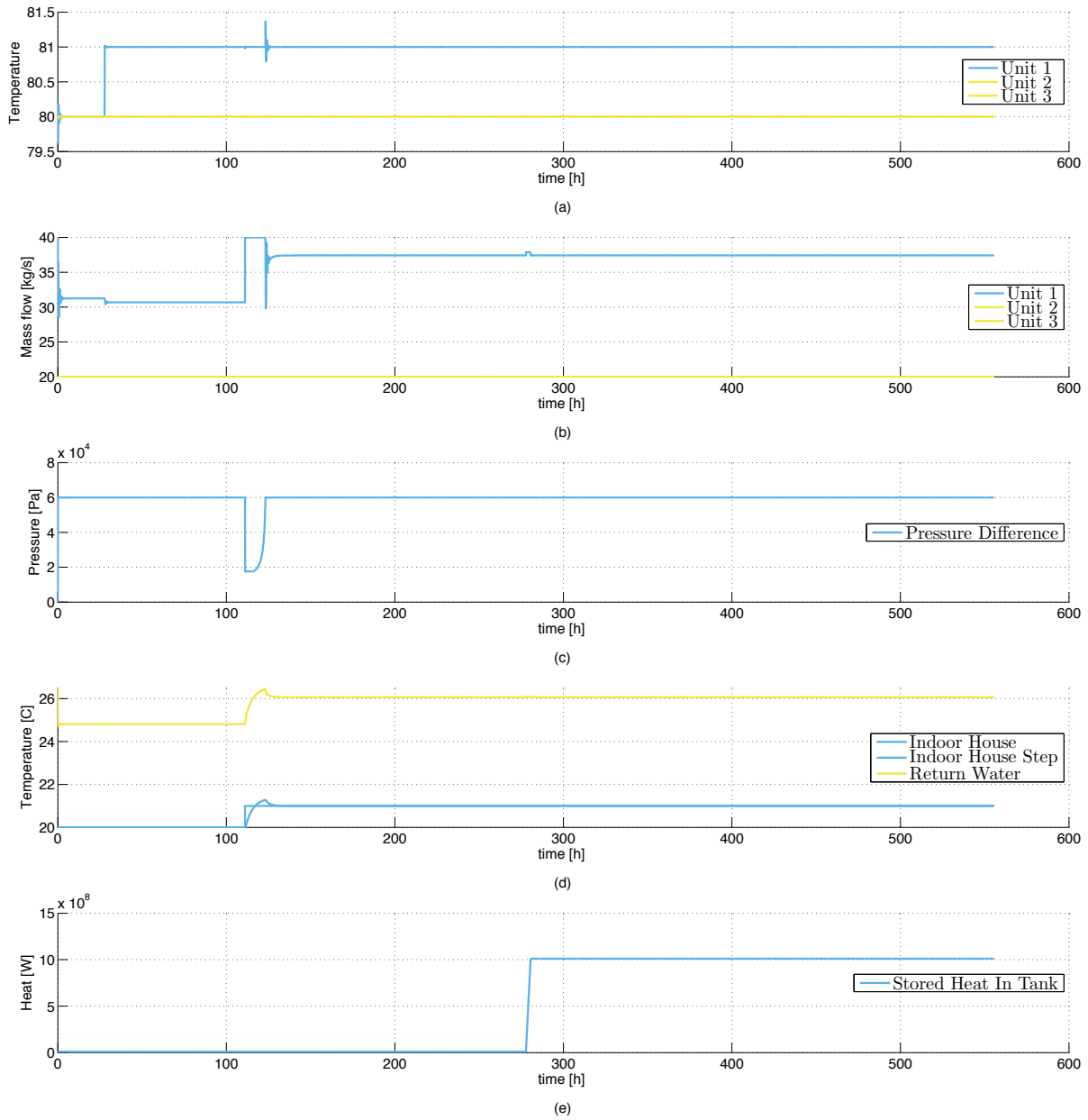


Figure 5.28: Combined Modelica Model Test Results.

5.8 Model Linearisation

In this section a linear model is created by linearising the nonlinear DH model in Figure 5.29 using Dymola. The linearisation is done by the following four steps listed below.

Linearise Procedure

1. Define all inputs and outputs in the model to be linearised using blue triangles for inputs and white triangles for outputs.
2. Add constant offsets to inputs in order to linearise in the Operating Points (OPs) of interest. The chosen OPs for this model are listed in Table 5.7.
3. Go to the Dymola simulation tab and write `importInitial()` in the console to load in steady-state values and then press the linearise button.
4. A `dslin.mat` file will be created containing a state space realization of the model. Using the following matlab function `[A, B, C, D, xName, yName] = loadin('dslin')` it is possible to load in the linear state space model, state names, and output names into matlab. Paths to the folder MFiles within Dymolas directory must be added before the function can be used.

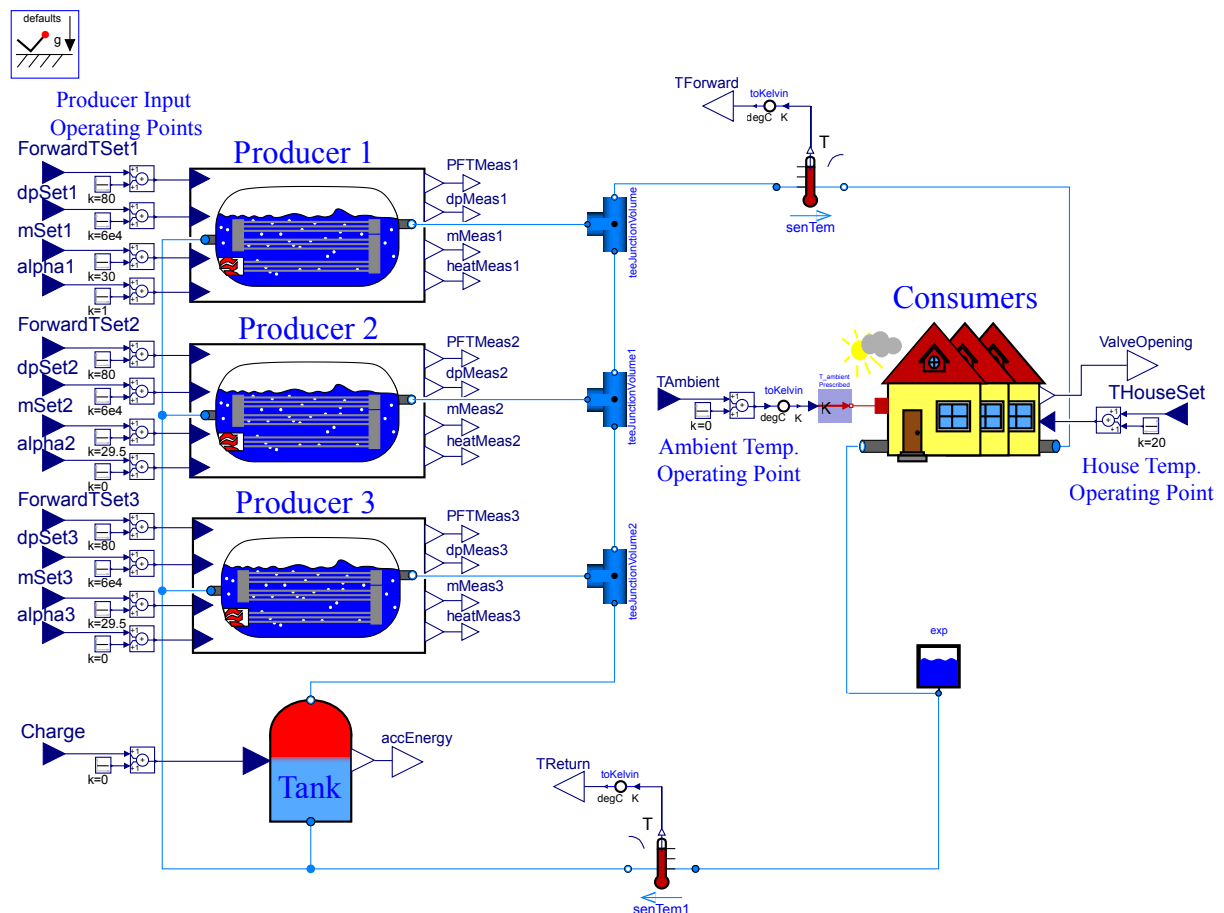


Figure 5.29: Full district heating modelica model.

Choice of Operating Points

In order to linearise the model it is necessary to specify OPs for each input. The supply temperatures are chosen to have the OPs of 80°C because it is the goal to have the supply temperature to be in small perturbations around 80°C. The same applies for the terminal pressure. For the mass flow controlled units the OPs are chosen to be at 50 % load which is at 20 kg/s. This choice gives a mass flow for the pressure controlled unit to be 28.3 kg/s. The alpha OPs are specified to have 1 pressure and two mass flow controlled units. For the tank the OP is 0 because it need to vary around charging and discharging. The OP for the ambient temperature is chosen to be 6°C because this is the mean temperature in Sønderborg for 2014. The indoor temperature must always be around 20°C.

Linear Model Input Operating Points		
Model Input	Operating Point, \bar{x}	Unit
ForwardTSet1, ForwardTSet2, ForwardTSet3	80	[°C]
dpSet1, dpSet2, dpSet3	6e4	[Pa]
mSet1	27.57	[kg/s]
mSet2, mSet3	20	[kg/s]
alpha1	1	[-]
alpha2, alpha3	0	[-]
charge	0	[-]
TAmbient	6	[°C]
THouseSet	20	[°C]

Table 5.7: Inputs and chosen operation points.

The Linear Model

The continuous-time model which is generated from the linearisation is a linear state space system, as in (5.22), of Figure 5.29 in the operating points \bar{x} of interest seen from Table 5.7. The matrix A,B,C,and D is calculated from the taking the partial derivatives, as seen in (B.7), of the nonlinear function. The overall linearization procedure with first-order Taylor approximation is described in Appendix B.

$$\text{Continuous-time Linear Model: } \hat{\dot{x}}(t) = A\hat{x}(t) + B\hat{u}(t) \quad (5.22a)$$

$$\hat{y}(t) = C\hat{x}(t) + D\hat{u}(t) \quad (5.22b)$$

where

$\hat{x} = x - \bar{x}$ is the small signal perturbation vector

\bar{x} is the steady-state operating point vector

x is the state vector

In this case $x(t)$ is a 35x1 state vector, $y(t)$ is a 16x1 output vector, and $u(t)$ is a 15x1 input vector.

Using the linear model in a MPC context it is important that the linear model approximates the non-linear model quite well because it is used to generate set points for the plant. The linear model is therefore compared to the non-linear model to verify this approximation in the next section.

Before the linear model is verified it is discretised in Matlab. This is done by zero order holding with a sampling time of 1 hour or 3600s using the `c2d` function in matlab. A sampling time of an hour is deemed adequate due to the slow nature of temperature changes, one hour sampling time also fits nicely with the energy market resolution, and a 24 hour day cycle. Hourly long sampling times also yields the option of simulating over extensive time periods because at every sample time set points are calculated and this takes time. The discrete linear model is formulated as (5.23).

$$\text{Discrete-time Linear Model: } \hat{x}(k+1) = A_d \hat{x}(k) + B_d \hat{u}(k) \quad (5.23a)$$

$$\hat{y}(k) = C_d \hat{x}(k) + D_d \hat{u}(k) \quad (5.23b)$$

Linear Model Analysis

In to see what to expect from the linear model a small analysis of the poles and zeros is made.

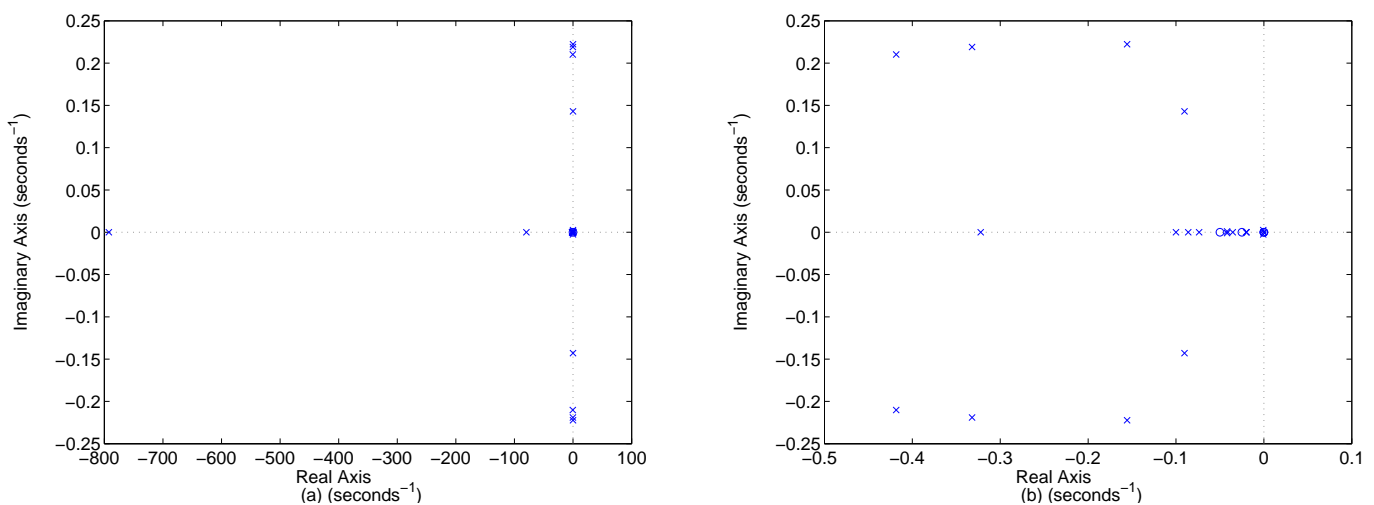


Figure 5.30: Pole-Zero map of the continuous linear state-space model. (a) zoomed out view and (b) is zoomed in.

From the two pole-zero maps it can clearly be seen that there are no poles in the Right Half Plane (RHP) meaning that the system is stable or that all exponential components of the system are decaying. The complex conjugate pole pairs introduce components with decaying sinusoid responses. There is a very fast component at the pole at least at -800 in Figure 5.30(a) and some more rapid dynamics of the complex pairs in Figure 5.30(b). Moving towards zero on the real axis in Figure 5.30(b) is seen a lot of poles with relatively more dominant dynamics than the complex pairs. There are four poles at zero which gives four integrators in the system. This makes sense because the models include integrators. There are also no

zeros in the RHP making it a minimum phase system.

Linear Model Verification

In order to verify the linearised model errors between states from the discrete linear model and non-linear model are looked at. This is done as seen in Figure 5.31. Each model is given the same input vector and the difference between the states is outputted.

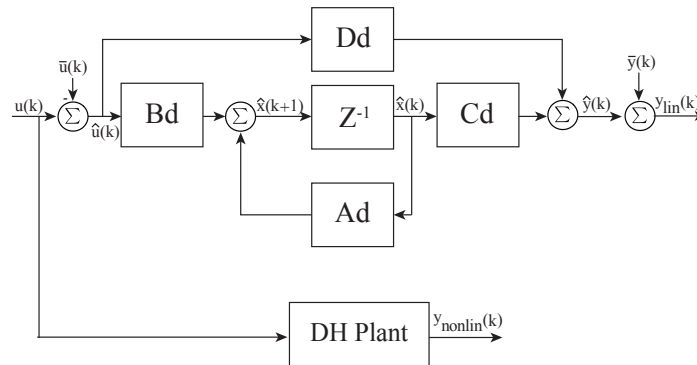


Figure 5.31: The linear and nonlinear plant model output comparison.

In Figure 5.32 are the supply temperature and mass flow for the linear and nonlinear models compared. The linear model outputs are dashed lines with markers and the nonlinear model are just lines. In (a) a two degree step in supply temperature is made at hour 6 for unit 1. The linear model approximates the nonlinear model quite fine here. A 5 degree step is applied at hour 25 for unit 2 and also here the linear model is doing good. Larger steps than 5 degrees in supply temperature is not necessary because the supply temperature must be within 75-82 degrees. In (b) a 2 kg/s step in mass flow for unit 2 is applied. The linear model also performs adequate here. In general the linear model fits nicely for the controlled variables. At hour 40 a large 20 kg/s step in mass flow is done for unit 3. The linear model for unit fits good though the mass flow for linear unit 1 starts to deviate from the nonlinear when moving too far away from the OP as expected.

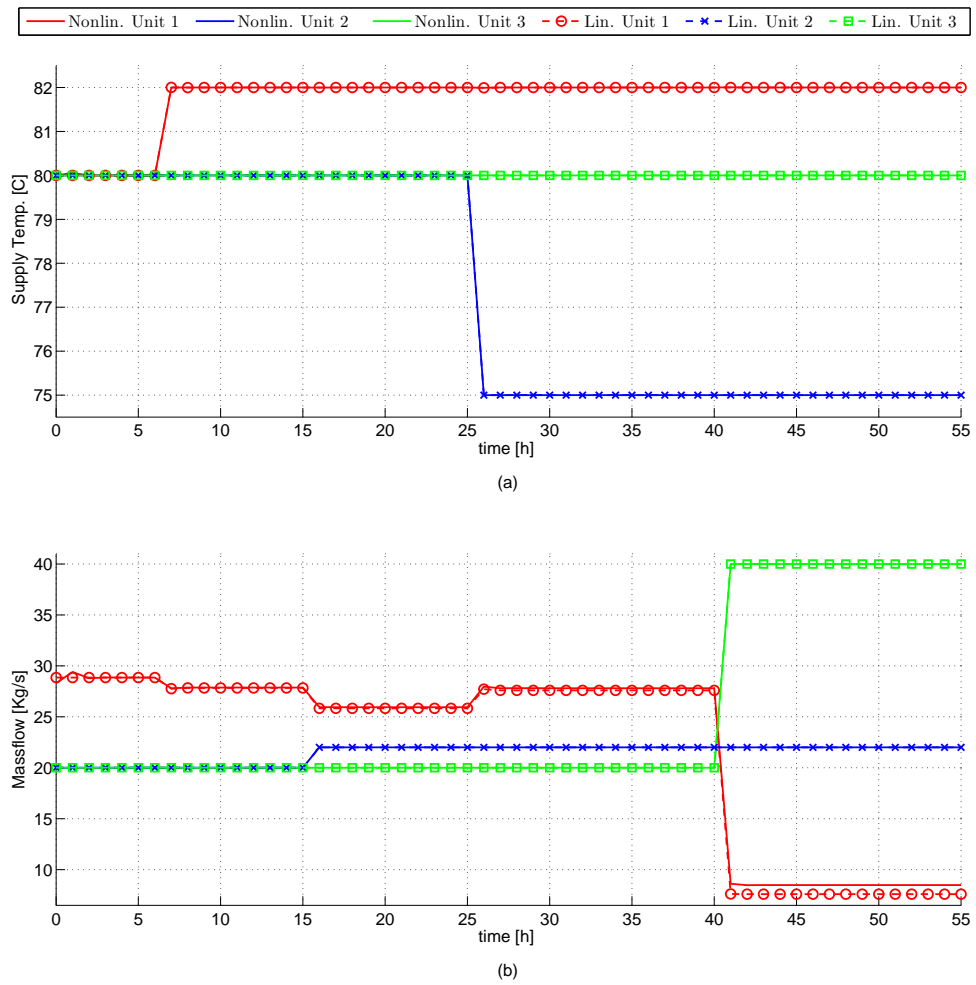


Figure 5.32: Nonlinear and linear model output comparison. The legend in the top applies for both (a) and (b).

In Figure 5.33 a step is made in the ambient temperature. The step is from 6 to 4 °C and occurs at hour 20. The pressured controlled unit raises the mass flow to compensate for the increased consumption. The nonlinear model is at full load at 40 kg/s and is there saturated. This creates the pressure drop in (b) and increased house temperature drop in (c). The linear model is not constrained by any saturation and therefore increases the mass flow above 40 kg/s for a short period. It is important that the linear model is constrained in order to represent the nonlinear model because the saturations are lost in the linearisation because these are not linear.

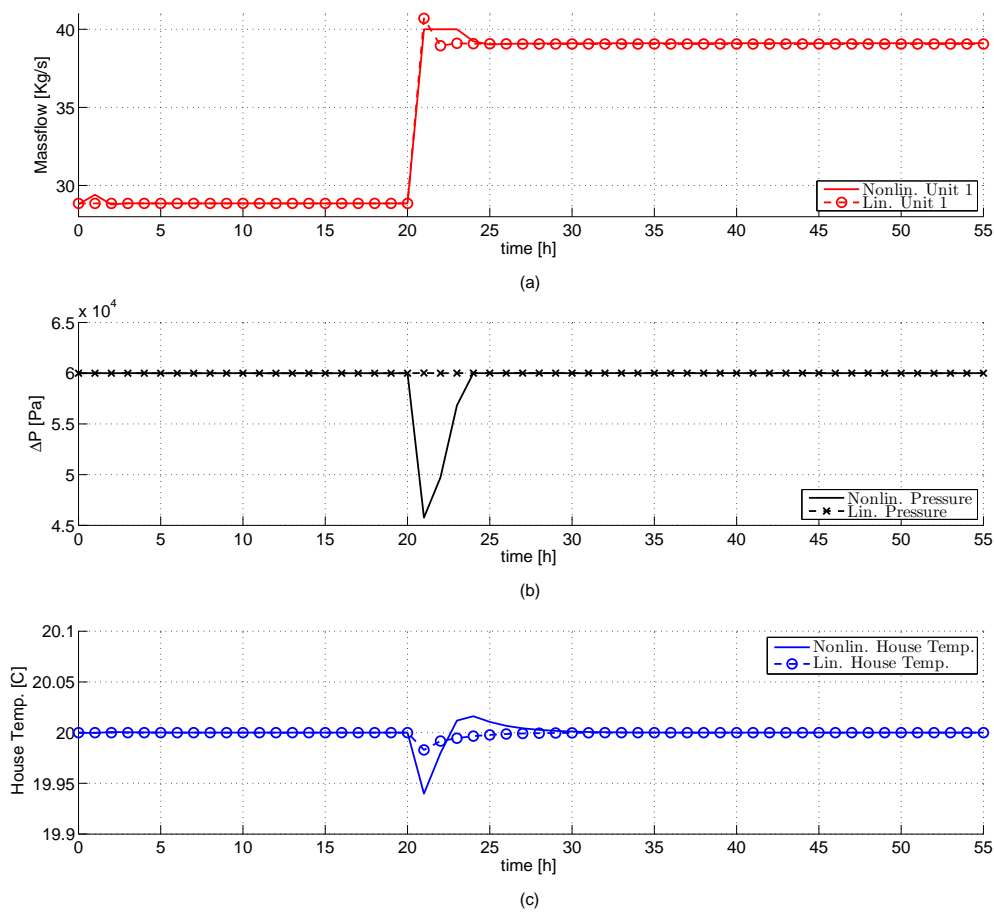


Figure 5.33: Nonlinear and linear model output comparison.

In general the linear model approximates the nonlinear model quite fine when not going too far away from the operating points.

Each model has now been designed and combining into the full DH model as specified in Chapter 5. The linear model have been verified and the next chapter starts the control and controller design part.

PART

III

**CONTROL SOLUTION TO THE COST
OPTIMIZATION PROBLEM**

Introduction to the Control Solution

The key subjects from the introduction and initial thesis statement in Chapter 1 have now been analysed in Chapter 2. This gave rise to the final scoping of the problem statement and objectives defined in Chapter 3. The narrowed objectives created the foundation for the modelling in Chapter 5. The model has now been completely developed and test. In this part and in the next chapters the optimization problem and balancing controller is to be developed. These tasks are initiated by defining the underlying assumptions, the limitations, and the requirements for the balancing controller. After this a controller type is selected. In the end the controller is designed, implemented and verified. For good measures are the objectives recapitulated here:

Status of Thesis Objectives

The preliminary objectives were defined in Chapter 1 and until this point the first two objectives have been achieved and therefore given a check mark.

- ✓ Understanding the system
- ✓ Develop Simulation Models
- Develop Controller: The development of the controller can be split up into two sub-objectives.
 - Optimization Problem Design: This entails the understanding and the development of a minimization problem which have the abilities to minimize the production costs while meeting DH consumer demands, given a portfolio of heat production units.
 - Controller Design: Designing a controller which online generates economic optimal operating set points for each heat production unit.
- Controller Tests:
- Evaluate Controller:

When developing the controller the assumptions seen in Table 6.1 are presumed.

Assumptions and Limitations

Assumption	Motive
Production minimization is only done with respect to heat production.	From the problem statement in Chapter 3 is has been chosen to focus on production minimization for district heating units.
Start-up/shut-down phases will not be considered as a part of the optimal operation.	In start-up/shut-down phases variables tend to go outside the normal operating bounds. These phases are therefore left out
The power consumption of electrical components is ignored in the minimization.	The cost of power is very small relative to the fuel costs.
A portfolio of committed production units are given a priori i.e. no costs is associated with start-ups or shut-downs are considered.	The focus is on the balancing of production set points and not on the commitment problem which arises when another producer unit must be enabled to meet production demands.

Table 6.1: General assumptions/limitations defined upon knowledge from previously chapters.

6.1 Requirement Specification

This section contains a set of controller requirements which must be satisfied in order to be a feasible solution which can compete with the existing control solutions by producers.

Controller Requirements

Requirement	Motive
The forward supply temperatures must be within 75 - 82 °C	Depending on annual seasons the losses in the distribution pipes are greater or less. In summer periods less hot water is needed. It is also important not to vary the water temperatures too much to avoid thermal stress on pipes.
The Terminal Pressure Difference (TPD) must be kept within 0.3-0.6 Bar.	Producer must supply a certain mass flow in order to deliver a specific heat power at consumers and to ensure a certain level of consumer comfort i.e. the water should not come out the tap with too much or too little pressure.
Economic optimal operation set points must be calculated at each sample time for all production units.	The consumption changes continuously and production set points must be recalculated every time step.
Minimization of fuel costs must be possible.	Fuel costs are by far the most costly part of production and it is here the most potential for economic savings exists.
Physical plant limitations and constraints must be included in the optimization.	Each production unit have different physical constraints which must be obeyed.
The consumer heat demand must always be met by the DH production units.	It is the primary job of the DH producer to maintain the heat production at the consumption level.
The consumer household temperature must be maintained within ± 1 °C of the set point.	It necessary to ensure that the consumers can specify the desired indoor temperature and achieve it.

Table 6.2: Requirement specification for the economic balance controller.

6.2 Choice of Balance Controller

From the requirement specification it can be deduced that the controller must be able to handle MIMO systems, system constraints, and must run in an online fashion. This immaterially rules out classic PID controllers which are designed for Single Input and Single Output (SISO) systems. Classic optimal control excluded as well because it is not good at handling of system constraints in the optimization problem. The choice of controller is therefore a variation of optimal control called MPC.

Now an introduction to MPC is given. Figure 6.1 illustrates the general idea of MPC and its workings. In the top graph of Figure 6.1 time is at the k sample and from here a sequence of predicted control inputs and outputs are calculated over a finite prediction horizon. Doing this makes the states optimally reach the reference trajectory relative to some cost/performance function. Only the first calculated control input of the whole sequence is applied to the plant for that time step. This is indicated by the dashed circle in Figure 6.1.

In the bottom graph time is at the next time step $k+1$ and the first control input has been applied to the plant. A new sequences of predicted control inputs and outputs are now calculated. The predict horizon is continuously shifted in time in a receding horizon manner. Since it is impossible to develop a prediction model which is equal to the plant an error between the predicted and measured trajectory will happen. This error is indicated by the small dashed pink line in the bottom graph. This is accommodated for by online recalculating the control inputs every time sample iteration and updating the initial conditions of the prediction model with most recent measurement data.

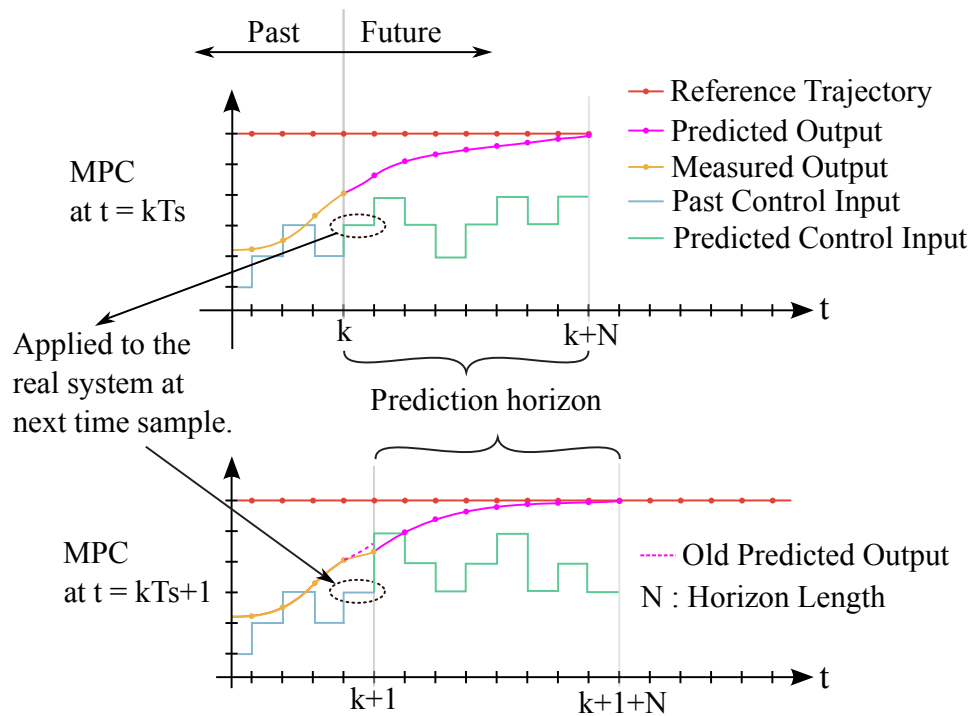


Figure 6.1: The workings of the Model Predictive Control Scheme.

The control input sequence is calculated relative to a given cost function and constraints. The costs of this function must be convex, see Appendix C for a further explanation of convexity. The prediction model must be linear. The size of prediction horizon, N , must be larger than the slowest dynamics of the system. This is for the system to be able to foresee and react to disturbances within the predict horizon knowing the dynamics of the system.

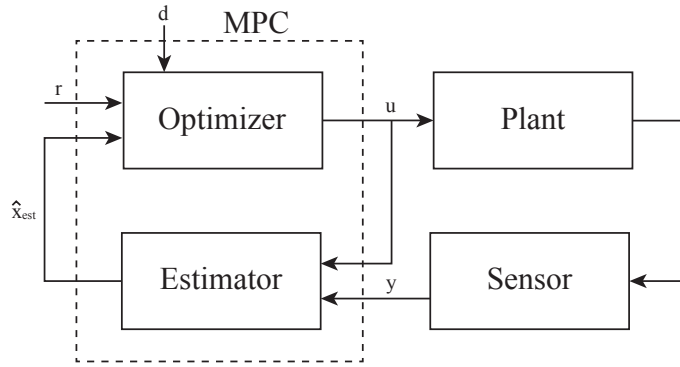


Figure 6.2: The workings of the Model Predictive Control Scheme.

Figure 6.2 shows the control concept of MPC. Unless all states are measurable which is quite uncommon MPC consist of a controller and an estimator. The controllers purpose is to solve the optimization problem as and output the control signal, u . The estimator does the model updating using the historical state information, \hat{x}_{est} . The controller is also given forecasts of future disturbances, d .

For the MPC control scheme the following pros and cons are considered.

MPC cons:

- Need a linear model of the system to be controlled.
- Optimization problems must be solved online.
- Converges problems can appear where solution times are too long.
- Infeasibility where no solution exist.
- Difficult to prove stability for online MPC.
- Requires the development of a cost function.

MPC Pros:

- Considers system constraints.
- Good at MIMO systems.
- Takes future events into considerations.
- Receding horizon/online control are great for non-stationary control problems.

In general MPC has usually only been applied on systems with slow dynamics because the optimization problem is required to be calculated every time step. It is possible to use MPC on fast system applications but additional work must be done as in [Wang and Boyd, 2010]. The increased calculation speed is archived by using calculating the entire control law offline and implementing it as look-up tables. This is possible but often standard MPC has been seen applied a lot within relative slow dynamic thermodynamic systems. In this thesis stability of MPC is not in focus and therefore touched upon.

The production balance control must be on a system which is relatively slow and therefore calculation time problems should not become a problem. The system have a lot of input and MPC is therefore chosen as the control method.

6.2.1 MPC Cost Function

In this section an example of a MPC cost function is presented. $V(k)$ is the cost function and it consists of two terms. The fist punishes deviations of the output measurements y from the reference r and the

second punishes changes in inputs, Δu .

$$V(k) = \underbrace{\sum_{i=N_w}^{N_p-1} (\mathbf{r}[k+i] - \mathbf{y}[k+i])^T \mathbf{Q}_e (\mathbf{r}[k+i] - \mathbf{y}[k+i])}_{\text{Punishment of trajectory errors}} + \underbrace{\sum_{i=0}^{N_u-1} \Delta \mathbf{u}[k+i]^T \mathbf{Q}_{\Delta u} \Delta \mathbf{u}[k+i]}_{\text{Punishment of control moves}} \quad (6.1a)$$

where

$r(k+i)$ is the reference signal depend on measurements up to time k .

$y(k+i)$ is the output measurement.

Δu is the changes in control/input values.

N_p is the predict horizon.

N_w is the predict horizon delay window.

N_u is the control horizon.

The delay window N_w gives the option of delaying with penalizing the error and it is assumed that $N_p \geq 0$. It also assumed that the control horizon is $N_u \leq N_p$.

As a part of the optimization problem there is often connected the dynamics of the model and a set of equality and inequality constraints.

MPC designers often also uses a linear and quadratic weights in order to punish the performance deviations. These weights have different design features as shown in the Figure 6.3.

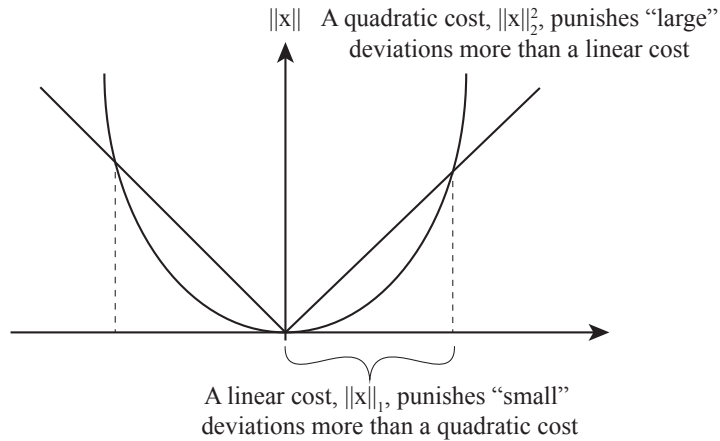


Figure 6.3: Linear and quadratic weighting of variables.

In Figure 6.3 the linear and quadratic cost is defined as the equalant norm-1 and (norm-2)²

The Balance Control Problem

The aim for this section is to outline the optimization problem by developing a cost function. After this the MPC controller is designed using the knowledge from the just developed optimization problem. These things must be done to develop a controller in order to satisfy the control requirements as specified from the requirement specification in Chapter 6.

7.1 The Cost Function

The structure of the optimization problem is given in advance as (7.2). The development is explained in details during the following sections. The cost function is inspired from the work done in the paper [Edlund et al., 2008, Introducing Model Predictive Control for Improving Power Plant Portfolio Performance].

$$\min_{U \in \mathbb{R}^m} J(k, u) = J_e(k, u) + J_y(k, u) + J_{\Delta u}(k, u) \quad (7.1)$$

$$= \mathbf{e}[k]^T \mathbf{Q}_e \mathbf{e}[k] + \mathbf{q}_y^T \mathbf{y}[k] + \mathbf{q}_{\Delta u}^T \Delta \mathbf{u}[k] \quad (7.2)$$

The goal of (7.2) is to minimize the cost function J by choosing a vector $\mathbf{U} = [u_1[k], u_2[k], \dots, u_m[k]]^T$ of m inputs which minimizes the error to the reference trajectory, measured input energy, and changes in input signals. The optimal inputs \mathbf{u} defined as (7.3) is the \mathbf{u} argument that minimizes the cost.

$$\mathbf{u}^*(k) = \arg \min_{U \in \mathbb{R}^m} J(k, u) \quad (7.3)$$

The input power to electrical components is neglected from the cost function as specified in Table 6.2, because the electrical power is much cheaper compared to fuel costs. In this way the cost of fuel makes for a much more dominant cost term.

In order to solve the optimization problem, the cost function, J , must be defined as a sufficiently smooth convex function. The cost function defined in (7.2) is convex and smooth because it is a sum of convex quadratic and linear functions. The definition of convex functions is explained in further detail in Appendix C. In this thesis a sufficiently smooth function means all the partial derivatives, that will appear later on, are defined and continuous.

In order to shape the optimization problem to fit the control requirements, specified by Table 6.2, the optimization problem is reformulated in (7.4). The optimization problem is inherently constrained by system dynamics and actuator constraints and it is therefore necessary to solve the problem without

violating any of them. It is also necessary to solve the optimization problem each discrete time step (online) because it is important to change the set point values to ensure optimal operation conditions, at all times.

The cost function of (7.2) is now reformulated into (7.4) in order to fit it to the MPC structure which had been chosen in Section 6.2. All terms e , y , x , Δu , u , and U are vectors given at time k containing i elements.

$$\min_{U \in \mathbb{R}^{mN}} \sum_{i=0}^{N-1} \mathbf{e}[k+i]^T \mathbf{Q}_e \mathbf{e}[k+i] + \mathbf{q}_y^T \mathbf{y}[k+i] + \mathbf{q}_{\Delta u}^T \Delta \mathbf{u}[k+i] \quad (7.4)$$

In (7.4) the time step k is modified to contain all inputs over the prediction horizon $i = 1, \dots, N$ for a new vector $U \in \mathbb{R}^{mN}$ and $U = [\mathbf{u}[k]^T \mathbf{u}[k+1]^T, \dots, \mathbf{u}[k+N-1]^T]^T$. N denotes the total length of the window horizon.

Designing the Cost Function

The main control objective is to meet the demand of district heating and doing this in an optimal economic way while also satisfying the remaining control requirements in Table 6.2. This is done by composing the cost function of three terms which describes different minimization objectives. Each term is posed in a quadratic or linear manner.

The first term is the cost function J_e seen in (7.5). This term is designed to minimize the error between current output matrix $\mathbf{Y} = [\mathbf{y}[k]^T \mathbf{y}[k+1]^T, \dots, \mathbf{y}[k+N-1]^T]^T$ and the reference matrix $\mathbf{r} = [\mathbf{r}[k]^T \mathbf{r}[k+1]^T, \dots, \mathbf{r}[k+N-1]^T]^T$. The reference vectors denotes optimal operating conditions required to meet district heating demands. In the implementation this is done for the terminal pressure difference with a quadratic weight, \mathbf{Q}_e . The pressure difference can be seen as a consumer comfort and it is very important that the pressure is maintained near the reference set point, not to damage pipes. Using the quadratic function punishes the trajectory pressure error with a large weight if the pressure moves far away from the reference. Small variations from the reference is on the other hand more acceptable. This is the design feature of quadratic function as previously seen in Figure 6.3.

$$J_e(k, u) = \sum_{i=0}^{N-1} (\mathbf{r}[k+i] - \mathbf{y}[k+i])^T \mathbf{Q}_e (\mathbf{r}[k+i] - \mathbf{y}[k+i]) = \sum_{i=0}^{N-1} \mathbf{e}[k+i]^T \mathbf{Q}_e \mathbf{e}[k+i] \quad (7.5)$$

The second term has to minimize the fuel costs of meeting consumer demands. The term (7.6) in is chosen to be linear because it fits well with a linear increase in cost for more used input fuel. This linear cost could be varying in time to model changes in fuel prices using price forecasts. In the implementation the input heat is measured for each unit and multiplied with the fuel costs.

$$J_y(k, u) = \sum_{i=0}^{N-1} \mathbf{q}_y^T \mathbf{C} \mathbf{x}[k+i] = \sum_{i=0}^{N-1} \mathbf{q}_y^T \mathbf{y}[k+i] \quad (7.6)$$

The third objective of (7.7) is to ensure changes in the input signal Δu not being under damped. This is done by adding a cost to input change. Changing inputs/control moves does not have direct cost but have a cost because components are faster worn out and therefore requires more maintenance costs. This is especially true for district heating pipes because too large varying temperature stresses the pipe material and significantly reduces lifespan.

$$J_{\Delta u}(k, u) = \sum_{i=0}^{N-1} \mathbf{q}_{\Delta u}^T (\mathbf{u}[k+i] - \mathbf{u}[k-1+i]) = \sum_{i=0}^{N-1} \mathbf{q}_{\Delta u}^T \Delta \mathbf{u}[k+i] \quad (7.7)$$

In the implementation this is done for the difference pressure, the supply temperatures, and the mass flow set points.

The total cost function in (7.4) is now constructed in (7.8) from summing the cost functions (7.5), (7.5), and (7.7).

$$J(k, u) = J_e(k, u) + J_y(k, u) + J_{\Delta u}(k, u) \quad (7.8)$$

Choice of Weight Matrices

When designing the optimization problem in a quadratic and linear manner the weights, Q_e , q_y and $q_{\Delta u}$, are free design parameters. Choosing a weight is about balancing terms e.g. between good minimization of an error against the required cost to achieve this. The linear weight is a vector with real values. The quadratic weigh is a square, diagonal, symmetric matrices, where each element in the diagonal contains a real value. This works as a relative punishment of deviations of the states from reference. The diagonal matrix Q is defined in (7.9) as being semi positive definite and all entries not in the diagonal is zero.

$$\mathbf{Q}_e = \{ \mathbf{Q} \geq 0 : \mathbf{Q}(i, j) = 0 \mid i \neq j \} \quad (7.9)$$

The choice of weights in the diagonal are the max values of the states because these values are known.

$$\mathbf{Q}_e(i, j) \begin{cases} \frac{1}{x_{max}^2} & \text{if } i = j \quad \forall i, j \in \{1, 2, \dots, n\} \\ 0 & : \text{ Otherwise} \end{cases} \quad (7.10)$$

$$(7.11)$$

The linear weights are defined as the following.

$$\mathbf{q}_y(i) = \frac{1}{y_{max}} \quad \forall i \in \{1, 2, \dots, r\} \quad (7.12)$$

$$\mathbf{q}_{\Delta u}(i) = \frac{1}{\Delta u_{max}} \quad \forall i \in \{1, 2, \dots, r\} \quad (7.13)$$

Each of the terms in the cost function will be scaled by their maximum values which keeps them in the $[0;1]$ interval. This is done to weight different terms equal in the cost function e.g. if terms have different units. An alternative is to design the weights from data statistics and include information about the standard deviation and variance of the distributions. This can be done by designing the weights from by Standard Score.

7.2 Model Predictive Control

In this section the total MPC controllers optimization problem is defined with previously defined cost functions and upcoming explained constraints. This is seen in (7.14) and each type of constraints is explained in the subsections below, see reference to the left of the constraint.

$$\min_{\mathbf{U} \in \mathbb{R}^{mN}} \sum_{i=0}^{N-1} \mathbf{e}[k+i]^T \mathbf{Q}_e \mathbf{e}[k+i] + \mathbf{q}_y^T \mathbf{y}[k+i] + \Delta \mathbf{u}[k+i]^T \mathbf{Q}_{\Delta u} \Delta \mathbf{u}[k+i] \quad (7.14a)$$

Subject to

$$\mathbf{x}[k+i+1] = \mathcal{A}\mathbf{x}[k+i] + \mathcal{B}\mathbf{u}[k+i] \quad \textit{Subsection 7.3.1} \quad (7.14b)$$

$$\mathbf{e}[k+i] = \mathcal{C}\mathbf{z}[k+i] \quad (7.14c)$$

$$\mathbf{y}[k+i] = \mathbf{C}\mathbf{x}[k+i] \quad (7.14d)$$

$$\mathbf{U}_{min} \leq \mathbf{u}[k+i] \leq \mathbf{U}_{max} \quad \textit{Subsection 7.3.2} \quad (7.14e)$$

$$\mathbf{Y}_{min} \leq \mathbf{y}[k+i] \leq \mathbf{Y}_{max} \quad (7.14f)$$

$$\mathbf{X}_{min} \leq \mathbf{x}[k+i] \leq \mathbf{X}_{max} \quad (7.14g)$$

$$\Delta \mathbf{U}_{min} \leq \Delta \mathbf{u}[k+i] \leq \Delta \mathbf{U}_{max} \quad \textit{Subsection 7.3.3} \quad (7.14h)$$

7.3 Model Constraints

In this section is the constraints chosen, for the optimization problem in (7.14), described. The constraints originates primarily from individual production unit requirements, operating conditions and physical limitations of equipment. The constraints include dynamic model, min/max bounds and rate of change constraints.

7.3.1 Prediction Model Dynamics

The model dynamics represent the linearised plant state space model from Section 5.8. This model is used to propagate state trajectories over the predict horizon, given the temperature forecasts data in Figure 5.8. In order to track a reference, which is required by the cost function, the linear dynamic prediction model is extended with a reference model. This is done by following the theory of classical

optimal control [Sørensen and Andersen, 2010].

$$\text{System Model: } \mathbf{x}_s(k+1+i) = \mathbf{A}_s \mathbf{x}_s(k+i) + \mathbf{B}_s \mathbf{u}_s(k+i) \quad (7.15a)$$

$$\mathbf{y}_s(k+i) = \mathbf{C}_s \mathbf{x}_s(k+i) + \mathbf{D}_s \mathbf{u}_s(k+i) \quad (7.15b)$$

$$\text{Reference Model: } \mathbf{x}_r(k+1+i) = \mathbf{A}_r \mathbf{x}_r(k+i) \quad (7.15c)$$

$$\mathbf{r}(k+i) = \mathbf{H}_r \mathbf{x}_r(k+i) \quad (7.15d)$$

The augmented model is now redefined to (7.16) to be as stated in (7.14b).

$$\underbrace{\begin{bmatrix} \mathbf{x}_s(k+1+i) \\ \mathbf{x}_r(k+1+i) \end{bmatrix}}_{\mathbf{x}(k+1+i)} = \underbrace{\begin{bmatrix} \mathbf{A}_s & | & \mathbf{0} \\ \mathbf{0} & | & \mathbf{A}_r \end{bmatrix}}_{\mathcal{A}} \underbrace{\begin{bmatrix} \mathbf{x}_s(k+i) \\ \mathbf{x}_r(k+i) \end{bmatrix}}_{\mathbf{x}(k+i)} + \underbrace{\begin{bmatrix} \mathbf{B}_s \\ \mathbf{0} \end{bmatrix}}_{\mathcal{B}} \mathbf{u}_s(k+i) \quad (7.16a)$$

$$\mathbf{e}(k+i) = \underbrace{\begin{bmatrix} -\mathbf{C}_s & | & -\mathbf{D}_s & | & \mathbf{C}_r \end{bmatrix}}_{\mathcal{C}} \underbrace{\begin{bmatrix} \mathbf{x}_s(k+i) \\ \mathbf{u}_s(k+i) \\ \mathbf{x}_r(k+i) \end{bmatrix}}_{\mathbf{z}(k+i)} \quad (7.16b)$$

$$\mathbf{y}_s(k+i) = \underbrace{\begin{bmatrix} \mathbf{C}_s & | & \mathbf{D}_s \end{bmatrix}}_{\mathcal{C}} \underbrace{\begin{bmatrix} \mathbf{x}_s(k+i) \\ \mathbf{u}_s(k+i) \end{bmatrix}}_{\mathbf{x}_y} (k+i) \quad (7.16c)$$

7.3.2 Min/Max Bounds

For each production unit there exists physical limitations of the inputs, outputs, and states operating bounds.

$$\mathbf{U}_{\min} \leq \mathbf{u}[k+i] \leq \mathbf{U}_{\max} \quad \mathbf{U} \in \mathbb{R}^{m \times N} \quad (7.17)$$

$$\mathbf{Y}_{\min} \leq \mathbf{y}[k+i] \leq \mathbf{Y}_{\max} \quad \mathbf{Y} \in \mathbb{R}^{r \times N} \quad (7.18)$$

$\mathbf{U}_{\min, \max}$ and $\mathbf{Y}_{\min, \max}$ are matrices of minimum/maximum operating set points. In the implementation the supply temperature set points are bounded in between 40 and 100 °C in order to give MPC more freedom in the control signal. Doing this it is necessary to bound the the supply temperature output between 75 and 82 °C. The difference pressure set point can vary between 0.1 to 15 bars. The measured output pressure must remain between the specified 0.3 - 0.6 bars. The consumer valve opening is bounded in order to ensure that the opening degree remains, within the physical limitation, between 0-1.

7.3.3 Ramping Limitations

Each production unit ability to increase or decrease heat output from time period k to $k + 1$ is limited. These rate of change bounds or ramping limitations are expressed as (7.19).

$$\Delta U_{\min} \leq \Delta \mathbf{u}[k+i] \leq \Delta U_{\max} \quad (7.19)$$

Where $\Delta \mathbf{u}(k+i)$ is defined as $\Delta \mathbf{u}(k+i) = \mathbf{u}(k+i) - \mathbf{u}(k-1+i)$. ΔU_{\min} and ΔU_{\max} are minimum and maximum ramping limits for each input. The same for the output.

In the implementation the $\Delta \mathbf{u}$ term is applied on changes in the accumulator charge signal, supply temperatures, and the mass flow set points. This is done to counteract hard and quick set point changes. The pumps are assumed to have the gradient capacity of changing 1 % of maximum load per second. For the supply set points it is assumed that the burner can change 1 °C per minute. For the accumulator tank the gradient bounds are chosen so that it can vary with 5-10 % of maximum capacity during an hour.

The economic production balance controller has now been designed and developed. In Chapter 8 the controller is implemented together with the DH plant model within Simulink. The controller and the plant model is to be tested to evaluate the performance and behaviours of the controller.

Implementation and Results

This chapter is about the implementation and tests of the balance controller and DH model. The implementation details and considerations are described Section 8.1 and the tests in Section 8.2. During the implementation several modifications to the optimization problem have been done. The tests are conducted to evaluate if the current performance and behaviours of the controller is as design from Chapter 7.

8.1 General Simulation Implementation

The implementation is done within the Matlab Simulink and is seen in Figure 8.1. The MPC controller is implemented using an embedded Matlab function block. This is the left block in Figure 8.1. The DH model is implemented using a FMI-block from the FMI-toolbox. The FMI-toolbox gives an interface block within Simulink in order to use Dymola exported Modelica models within Simulink. This is the right block in Figure 8.1. Outputs and states are measured from the DH model and fed back to the controller function block with a small sampling delay. The simulation sampling times are explained in Subsection 8.1.2. The controller block generates small signal set points for the DH model. The input operating points are added to the small signal set points as explained in Subsection 8.1.1. During the solving of the optimization the chosen Mosek solver sometimes had numerical issues due to large values of heat resulting in stall problem. This is explained in Appendix D.

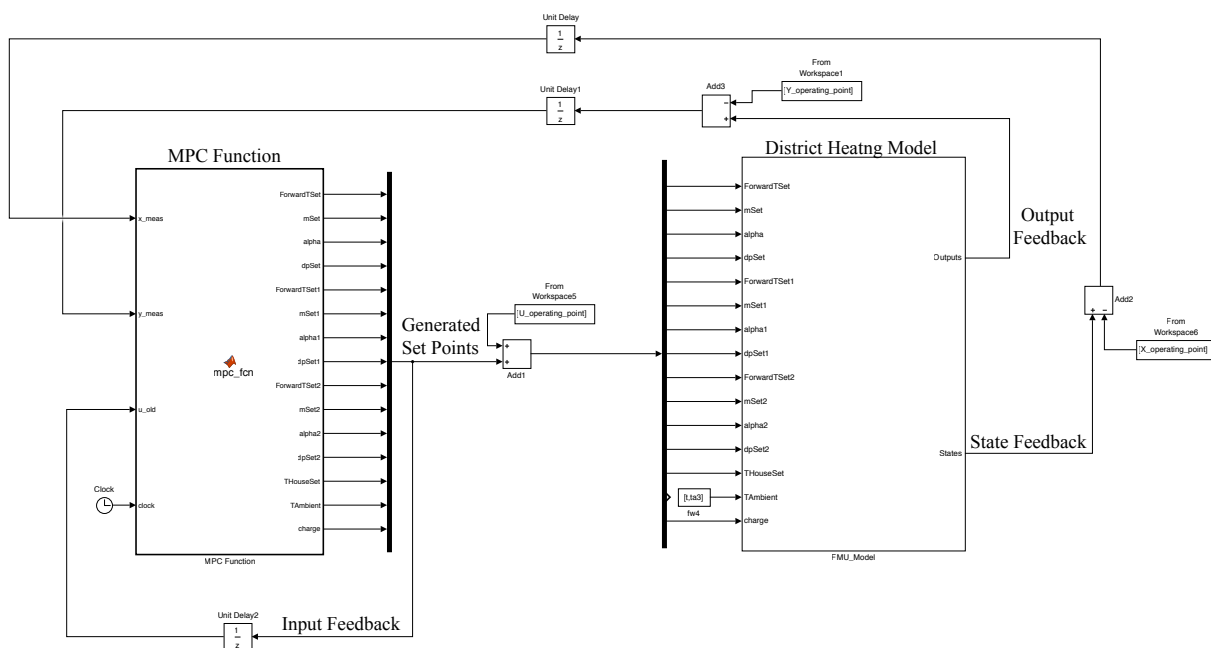


Figure 8.1: Implementation of the controller in an embedded Matlab function block and DH model in a FMI interface block within in Simulink.

8.1.1 Small Signal Implementation

In order to maintain the optimization of the linear model in small signals the input OPs are added to the generated set points. The state OPs are subtracted from the state feedback as illustrated in Figure 8.2. In this way the optimization problem is calculated with small variations around the OPs of interest.

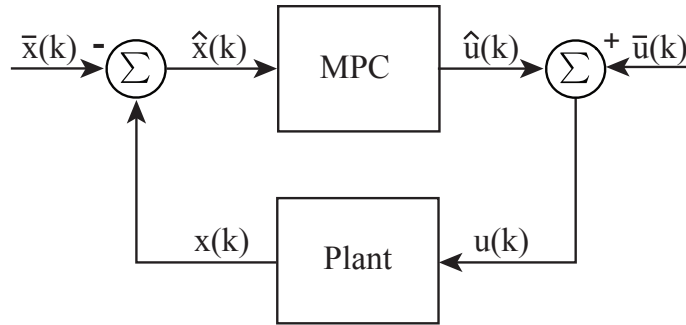


Figure 8.2: Adding input and subtracting state OPs ensuring small signal for the linear model optimization problem.

When using the linearized model with small signals it is important to rewrite all the constraints to include information about the used OPs. An example for this is seen in (8.1) where the input min/max bound is rewritten to.

$$\mathbf{U}_{\min} - \bar{U} \leq \mathbf{u}[k+i] \leq \mathbf{U}_{\max} - \bar{U} \quad (8.1)$$

where

\bar{U} is the steady-state operating point for the input signal.

8.1.2 Sample Times

The DH plant is sampled every 3.6s in order to get the newest state measurements from the plant. In the input, state, and output feedback paths there are implemented unit delay blocks to avoid algebraic loop problems. The delay blocks are sampled every 3.6s and this will introduce a 3.6s delay to the state measurements given to the MPC. This is deemed a very short and acceptable delay due to the relative slow dynamics of the plant i.e. the measurement will not differ a lot over 3.6s compared to the measurement with no delay. The embedded Matlab function block is updated with measurements from the plant every 360s. This is to only update MPC with initial state conditions which are steady-state state measurements. New MPC set points are generated every 3600s. The Simulink solver is defined as discrete and with a variable sample time in order to increase simulation speed.

8.1.3 Prediction Model Deviation

The prediction model is having a hard time predicting the pressure. This is in particular the case for the pressure controlling unit which job is to deliver the missing mass flow to obtain ΔP across the producers terminals. If predictions are deviating too much the rapid increases or decreases in the pressure occurs and the nonlinear FMU DH model reports an error which stops the simulation.

In order to accommodate for this a set of modifications to the cost function and measurement data is done in the sections below.

Temperature Data Modification

The ambient temperature data is multiplied by a scaling factor of 0.5 which decreases the size of variations in the temperature measurements. The consequence of this can be seen in Figure 8.3. This is done to limit the overall variations in consumption which reduces the variations in production. This is because the most prediction deviations occurs when during large changes in consumption.

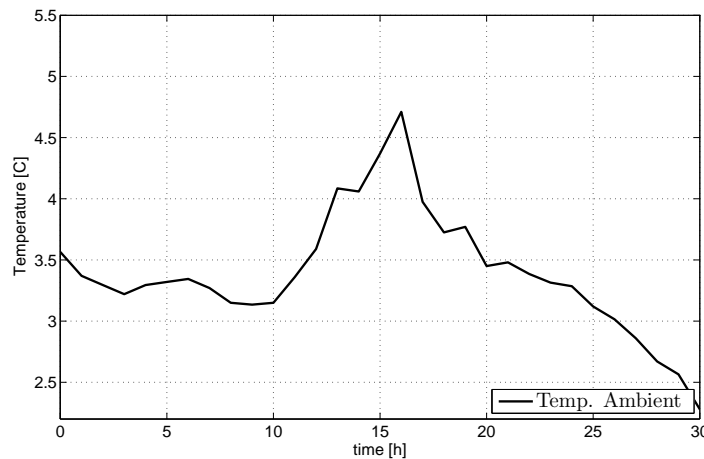


Figure 8.3: The ambient temperature data scaled by a factor of 0.5.

The prediction model is given the mean over an hour of the measurement data rather than a single measurement. This is done for the whole prediction window. The measurements are given in seconds i.e. therefore the mean temperature for an hour calculated from 3600 data samples.

Soft-Constraints

Another method used is to implement soft-constraints term in the pressure and mass flow constraint. This is done to give the solver more freedom in order to avoid infeasibility problems and to give better conditions for the plant FMU model. The soft-constraint added to the output terminal pressure and one for the output mass flow of the pressure controlling unit. This is implemented in the constraints as (8.2),

(8.3) and in the cost function as (8.4).

$$\mathbf{Y}_{\Delta P, \min} - \alpha_{\Delta P} \leq \mathbf{y}[k+i] \leq \mathbf{Y}_{\Delta P, \max} + \alpha_{\Delta P} \quad (8.2)$$

$$\mathbf{Y}_{m, \min} - \alpha_m \leq \mathbf{y}[k+i] \leq \mathbf{Y}_{m, \max} + \alpha_m \quad (8.3)$$

The cost function is modified to contain two additional quadratic terms, one weight for each specified alpha. In general this gives the option to the optimization problem to raise the pressure or lower the pressure out of the bounds (0.3-0.6 bar) if really necessary. The cost have to be a numerically large value to ensure usage only in extreme cases.

$$J_\alpha(k, u) = \sum_{i=0}^{N-1} \alpha_{\Delta P}[k+i]^T \mathbf{Q}_{\alpha_{\Delta P}} \alpha_{\Delta P}[k+i] + \alpha_m[k+i]^T \mathbf{Q}_{\alpha_m} \alpha_m[k+i] \quad (8.4)$$

The mass flow soft-constraint is added to maintain the mass flow, of the pressure controlling unit, in the proximity of the operation point to decrease prediction errors. The minimum and maximum bounds for the pressure controlling unit have been chosen as the values in (8.5).

$$10 - \alpha_m \leq \mathbf{y}[k+i] \leq 40 + \alpha_m \quad (8.5)$$

This maintains the predicted output near to the operating point of 27 kg/s and to some degree also the measure output. $Y_{\Delta P, \max}$ and $Y_{\Delta P, \min}$ is chosen away from the true maximum bounds (0-60 kg/s) to give the controller margin which it can cross if really necessary.

FMU Model Start-Up

In order to start up the FMU model it is beneficial but not necessary to ease some of the constraints. In particular the mass flow in the first hour of operation to quick make the pressure reach the set point value. This is done simply by introducing an if-statement. The conditional statement evaluates the sample clock and check then the first hour is over. A constant value ϵ has been used. This is seen in the pseudo-code example in Example 8.1).

Example 8.1 (Start-Up Constraints)

```

If clock >  $\epsilon_{hour}$ 
     $Y_{m, \min} = 0;$ 
     $Y_{m, \max} = 100;$ 
else
     $Y_{m, \min} = 10;$ 
     $Y_{m, \max} = 40;$ 
end

```

Designed Weights Values

The values picked for the weights are shown and reasoned for in Table 8.1. The Δu weights are to decrease rapid variations in the set points. Too large set point changes is in a practical manner not wanted and this also difficult for the simulation model to handle. The $Q_{e,dp}$ is quite large because it is important to maintain the pressure across the consumers. The choice of fuel costs $q_{y_{heat},unit1}$, $q_{y_{heat},unit2}$, $q_{y_{heat},unit3}$ are described in the next section.

MPC Weights			
Weight	Value	Design Reasoning	Unit
$Q_{e,dp}$	1.67	Relatively high to be the maintain the pressure set point objective compared to minimization fuel costs.	
$q_{\Delta u,Supply}$	2.43	Chosen as a relative high value to decrease the size of set point changes due to lack of additional PI-Controllers at different mass flow loads.	
$q_{\Delta u,Charge}$	100	High value to avoid abrupt charge/discharge changes.	
$q_{\Delta u,dP}$	0.12195	Chosen as a low value just to associate a cost with set point changes.	
$q_{\Delta u,mSet}$	0.0025	Chosen as a low value just to associate a cost with set point changes.	
$q_{y_{heat},unit1}$	0.0008	The cost of an expensive fuel source, e.g. natural-gas.	[DKK/Wh]
$q_{y_{heat},unit2}$	0.00002	The cost of a cheap fuel source e.g. waste.	[DKK/Wh]
$q_{y_{heat},unit3}$	0.0008	The cost of an expensive fuel source.	[DKK/Wh]

Table 8.1: The values and design reasons for the cost function weights.

8.2 MPC Tests

In this section multiple tests are made in order to evaluate the functionalities of the controller on the DH Modelica models, within Simulink. For each test there is given a portfolio of three producer units with the specifications specified in Subsection 8.2.1. The test specification for the consumers is outlined in Subsection 8.2.2. Each test sections follows this structure: Firstly the motivation for the test presented, then the test configuration explained, small discussion of expected Results, results evaluation, and reflection on results.

The following two test cases have been conducted.

- Test Scenario 1 - 3 Producers and Scaled Consumer: One pressure controlling, two mass flow control units and a set of consumers are connected in parallel to the DH distribution network. The producer units have been designed to have different fuel costs. The two mass flow controlled units will have the same dynamics. The pressure controlling unit has faster dynamics in order maintain the pressure even if the demand is not fully matched by the MPC. This is also done while excluding the accumulator tank.
- Test Scenario 2 - Including Accumulator Tank: This test will be similar to test 1 expect that the accumulator tank is connected to the portfolio.

8.2.1 Producer Test Unit Specifications

In the two tests conducted there are three producer units with the test specifications seen in Table 8.2.

Producer Specifications			
Unit Type	Fuel Costs [DKK/MWh]	Max Output Heat [MW]	Max Massflow [kg/s]
Unit 1 - Pressure control (Alpha = 1)	800	10e6	60
Unit 2 - Mass flow control (Alpha = 0)	20	10e6	40
Unit 3 - Mass flow control (Alpha = 0)	800	10e6	40

Table 8.2: Producer Unit Specifications

These producer units or boiler tanks are designed to deliver 10 MW at temperature difference of 40 which gives a mass flow of 60 kg/s. Due to model problems the ΔT is 57 and therefore the producer units delivers 10 MW at approximately 40 kg/s. The 10 MW is design freedom, though designed from a real boiler tank at SDH. The pressure controlling unit is parametrized with a maximum mass flow of 60 kg/s to have more influence to fix pressure prediction errors.

The fuel costs are chosen to have one expensive and one cheap fuel source. The expensive fuel source could be gas/oil and the cheap fuel costs could be the cost associated with of burning waste. The direct waste fuel costs are normally negative but becomes positive when including the maintenance costs, operator costs, ect.

8.2.2 Consumer Test Specifications

The test specifications for the consumer are the following: `n_scale` is specified to be 27.000 in order to have a realistic demand relative to the production capabilities specified in Table 8.2.

This gives a total consumption of $1400 \cdot 27.000 = 37.80MW$ at the $-12^{\circ}C$. This should be enough to evaluate the controllers performance. It is important that the consumption is dimensioned realistic relative to the production in order to test the controller. Of course in reality it is not possible to change the consumption relative to the production capabilities but this is done for simulation purposes.

8.3 Test Case 1 - 3 Producers and Scaled Consumer

Motivation

The goal of this test is to see if the controller can minimize the fuels costs while meeting the consumer demands, given a portfolio of committed production units. Three units are set in a parallel configuration together with a scaled consumer similar to the standard configuration previously seen in Figure 5.29. The accumulator tank is not included for this test. For all the test results the prediction are plotted with a low resolution to avoid overlap of markers. A much higher resolution of sample data is available.

Simulation Configuration

- Simulation duration - 30 hour period.
- 3 producer units.

- 27.000 household units.

Expected Results

It is expected that the optimization problem is solved and set points are generated. The set points will extensively enable the cheapest unit compared to the more expensive unit. The controller should use the expensive unit just enough to maintain the terminal pressure difference. The pressure control unit should have a mass flow varying with a high frequency in order to maintain the peaks in the consumption.

Results

In Figure 8.4 are the outlet mass flows seen for each production unit and total both for the measured and the predicted. The predictions are the dashed lines with markers and the plant measurements are the solid lines. In the first hour the mass flows are going out of the normal operating bounds because in the first hour relaxed constraints are enabled to ensure plant start-up. At hour 2 the mass flow of the cheap unit is immediately increase to full capacity and maintained there for the whole 30 hour simulation. This mass flow only diverges very little from maximum capacity during the ambient temperature peak in hour 15 as seen in Figure 8.3. This is too lower the production when the consumption decreases though MPC do not like to decrease the production from unit 1 too much because it is cheap.

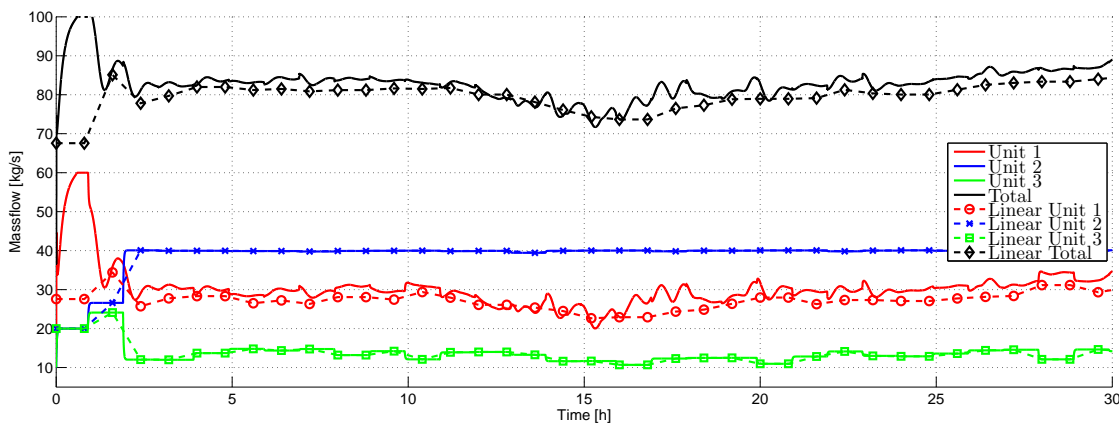


Figure 8.4: Measure and predicted mass flows for each production unit and for the total mass flows.

The expensive fuelled unit 3 is running at a low capacity of approximately 25 %. This agrees with the expected results, that the cheap unit will be used much more than the expensive one. This is a good sign of the balance controller current behaviour. The goal for unit 1 is to maintain pressure while maintaining the mass flow as close to its linearisation point as possible. This is to decrease potential mass flow prediction errors. Too wrong mass flow predictions ultimately results in a quick increase or decrease in pressure. If the prediction only deviates a little it is possible for the pressure controlling unit to repair it. If this occurs the plant (FMU model) often reports that an error has occurred and the simulation stops. When maintaining the temperature changes low and keeping the mass flow of unit 1 near the operating point the predictions are as seen in unit 1 of Figure 8.4.

In Figure 8.5 the supply temperature for each unit and the mixed is plotted. The predictions of the supply temperature are very accurate for unit 2 and 3. For unit 1 the predictions are a bit off as seen from the red dashed line with circle markers and the solid red line. This is most likely a consequence of the deviating

mass flow predictions of unit 1 in Figure 8.4. Again here the solid lines are measurements from the nonlinear plant and the dashed lines are the linear model prediction outputs.

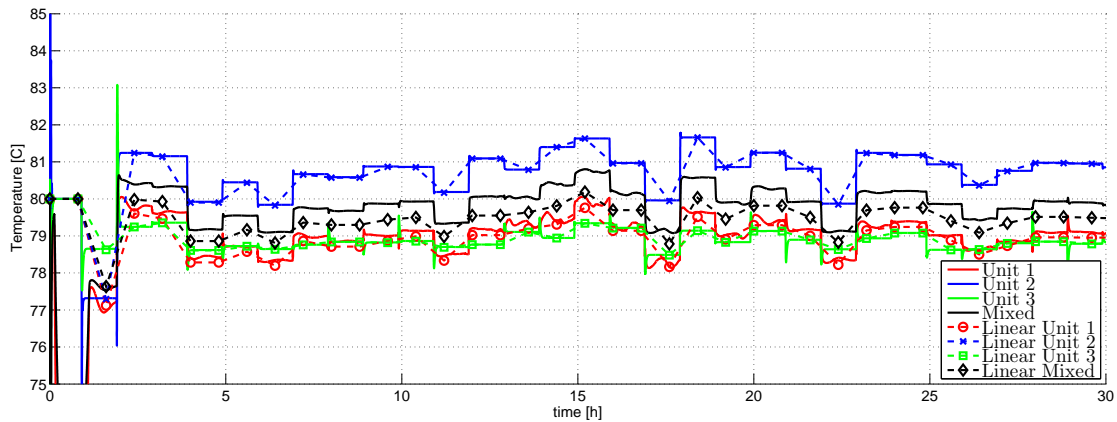


Figure 8.5: Measure and predicted supply temperatures for each production unit and for the mixed water.

The temperature drops in the first hour. This occurrence is intended in order to decrease the restrictions on MPC during start-up. The temperatures are maintained for most cases within the supply temperature requirement (between 75-82 °C) as specified in Section 6.1. When mass flow set point changes happens spikes in the supply temperature often occurs. The temperature spikes normally last around 5 minutes before the internal PI-Controller temperature controller catches up. A large mass flow set point decrease is seen in hour 2 for the expensive unit 3 and therefore the supply temperature spikes up to 83 °C for a short period of time. The spikes could be decreased by spending more time on tuning the internal temperature controllers.

In Figure 8.6 the terminal pressure difference, ΔP , is plotted. The pressure predictions are almost on top of the measured values except for the small spikes. These pressure spikes are not estimated by the linear model which could be the reason for the occurrence of FMU errors when the predictions are too of and the pressure quickly changes and goes out of bounds.

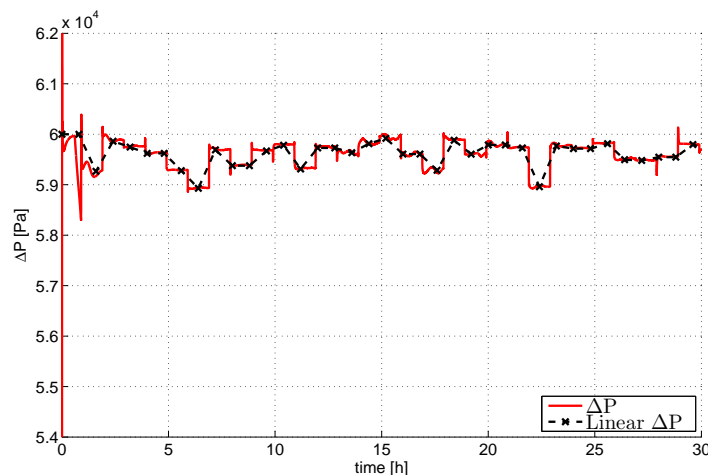


Figure 8.6: The measured and predict terminal pressures differences.

In Figure 8.7 the supply and demand curves are plotted. In general the consumption demand curve is met by the production supply curve. It can be noticed that in peaks the supply is often a bit larger than and demand. For large quick spikes in the demand the supply tends to be deviating more. This suggest that the prediction model often over estimates of the consumption.

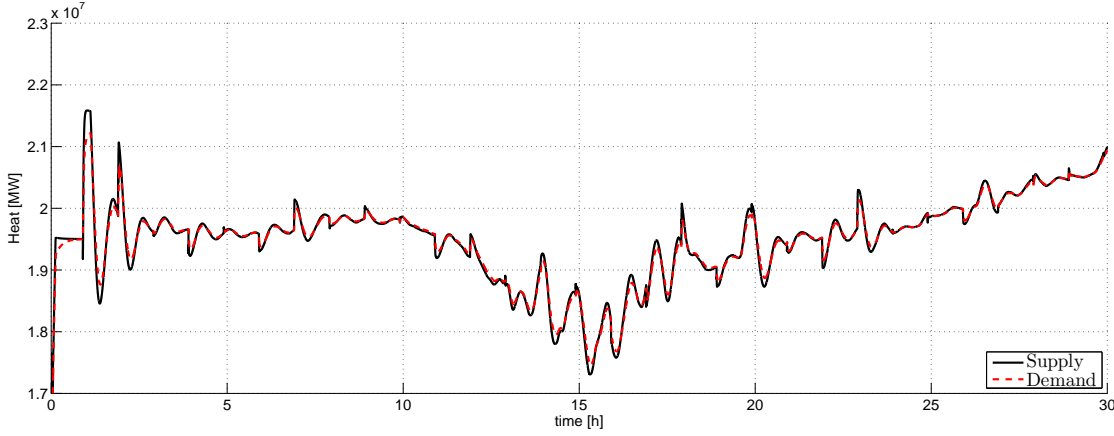


Figure 8.7: The consumer heat demand curve and the production supply curve.

In order to get a numerical value describing how much the two curves deviates from each other a Root Mean Square Deviation (RMSD) and Average Deviation (AD) is calculated.

$$RMSD_{Supply,Demand} = \sqrt{\frac{\sum_{k=1}^S (y_s(k) - y_d(k))^2}{S}} = 138.6\text{kW} \quad (8.6)$$

$$AD_{Supply,Demand} = \frac{\sum_{i=1}^S |(y_s(k) - y_d(k))|}{S} = 44.02\text{kW} \quad (8.7)$$

where

- y_s is the output of the supply curve [W]
- y_d is the output of the demand curve [W]
- S is the number of samples [-]

The average demand is 19.4 MW which the RMSD calculation deviate with 0.72 %, and the AD with 0.23 % relative to the mean production. The most significant difference in the two expression for deviations are that RMSD punishes larger deviations more than AD. This argues that the RMSD is the most informative deviation measure since it is important to maintain the peak loads. It is preferable that the deviation to the demand curve is above rather than below to maintain the consumers warm.

In Figure 8.8 is the indoor consumer house temperature as plotted. The measured temperature is maintained at the 20 °C set point. The predictions are also quite decent only diverges with a maximum of 0.1 °C from the measured value. The predictions are not perfect, they have a constant offset and make relative large temperature changes compared to the measured values. This again reflects back to the nonlinear and linear model differences.

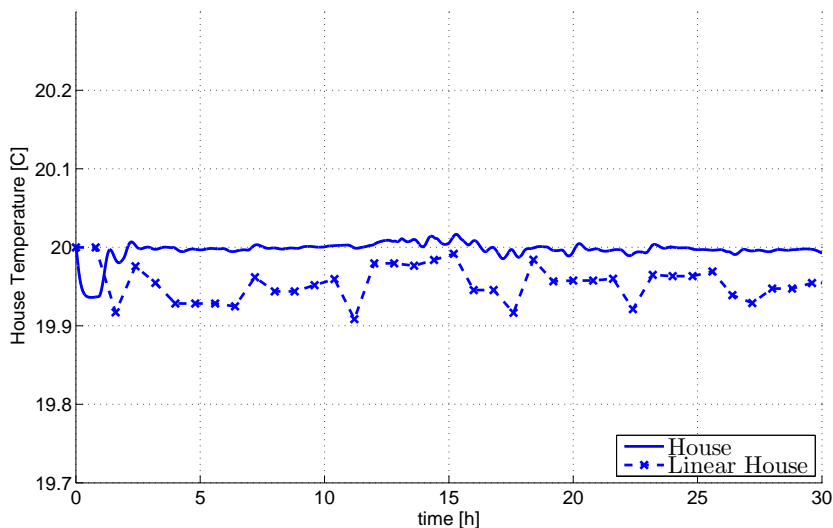


Figure 8.8: The measured and linear model predictions of the indoor house temperatures.

In Figure 8.9 are the accumulated fuel costs for each production and the total fuel costs plotted. It is the fuel accumulated costs for the three units under an operation period of 30 hours. The costs are largest for the cheap unit 2 because it is producing heat at maximum capacity for the whole duration. The total costs for the production reaches approximately 0.47 mio. DKK delivering an average of 19.4 MW over the period.

The accumulated cost for each unit are: unit 1 = 0.1673 mio. DKK/MW, unit 2 = 0.2243 mio. DKK/MW, unit 3 0.07637 mio. DKK/MW and the total costs of 0.46797 mio. DKK/MW.

Reflection

In general the balancing controller is doing a good job of choosing set points for the cheap and expensive production units. If large changes in the demand occurs the predictions are deviating more from the true values. The heat demands are met by the production expect in extremities where the supply tends to exceed the demand. The indoor house temperature is also maintained with a small offset. Additional tuning of the internal PI-Controllers could be made in order to give MPC a more critical damped system to interact with. In general it seems that there are problems with the linearised prediction model especially if the temperature is too high or.

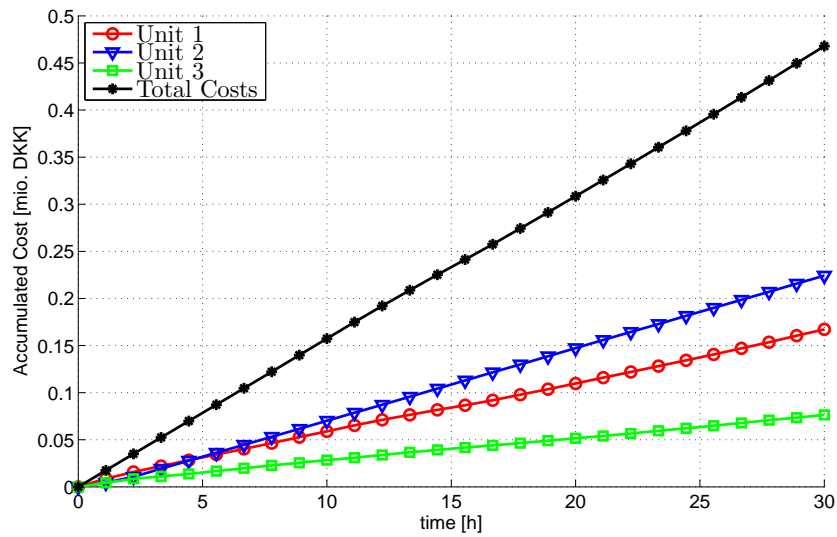


Figure 8.9: The accumulated costs for the 30 hour periods for each unit and the total.

8.4 Test Case 2 - With an Accumulator Tank

Motivation

In this test an accumulator tank is included in the portfolio to see how the balancing controller behaves with this additional tool at hand. The overall goal is still to minimize the costs of meeting consumer demands while using the same configured production three units as from Section 8.3.

Simulation Configuration

- Simulation duration - 30 hour period.
- 3 producer units.
- 27.000 household units.
- 1 accumulator tank with maximum storage capacity of 100 MW. Charge/discharge gradient of 5 % per hour of max storage capacity.

Expected Results

The controller will still use the cheap unit extensively and the expensive units less. It is expected that the tank will be charged/discharged to avoid using the expensive units and even less than compared to test 1 in Section 8.3.

Results

In Figure 8.10 is the measure and predicted mass flow output plotted. In general there are larger mass flow set point changes occurring. Unit 1 and unit 2 also changes locations much more often than without the tank. It seems like there are a lot of prediction errors from hour 10-22. This can be concluded because unit 1 has to do a lot of correction to maintain the pressure. In general the less mass flow changes for unit is when the prediction are deviating a lot.

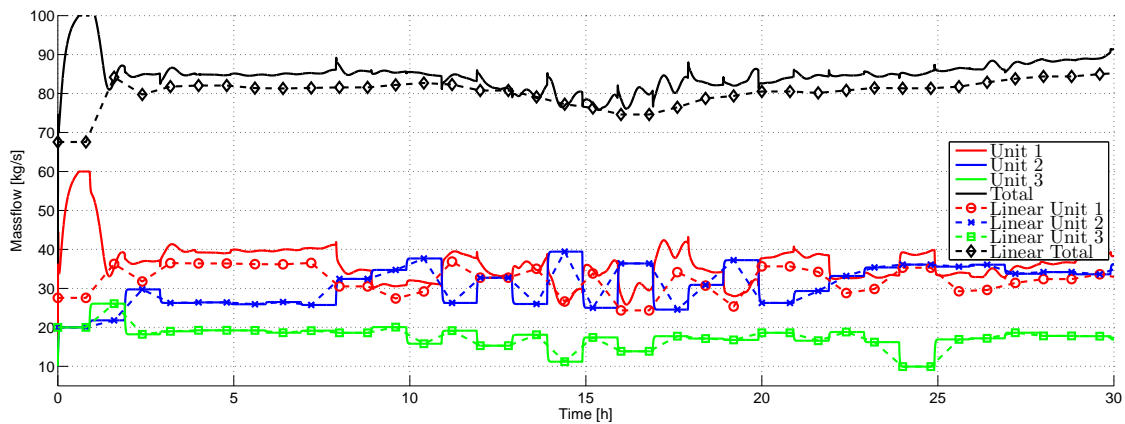


Figure 8.10: Measure and predicted mass flows for each production unit and for the total mass flows.

In Figure 8.11 is the pressure shown again and the spikes in pressure have become much larger. This is likely due to the increased mass flow variations. When the predictions are deviating too much increases in the pressure occurs until the pressure controlling unit will be able to fix it. At hour 15-16 the predicted mass flow for unit 2 goes from 40 to 28 kg/s. This choice is only partially correct and the pressure control unit 1 has to increase the mass flow to maintain the pressure. This gives the pressure drop seen at hour 15 in Figure 8.11. Too large pressure spikes are very unwanted and problematic.

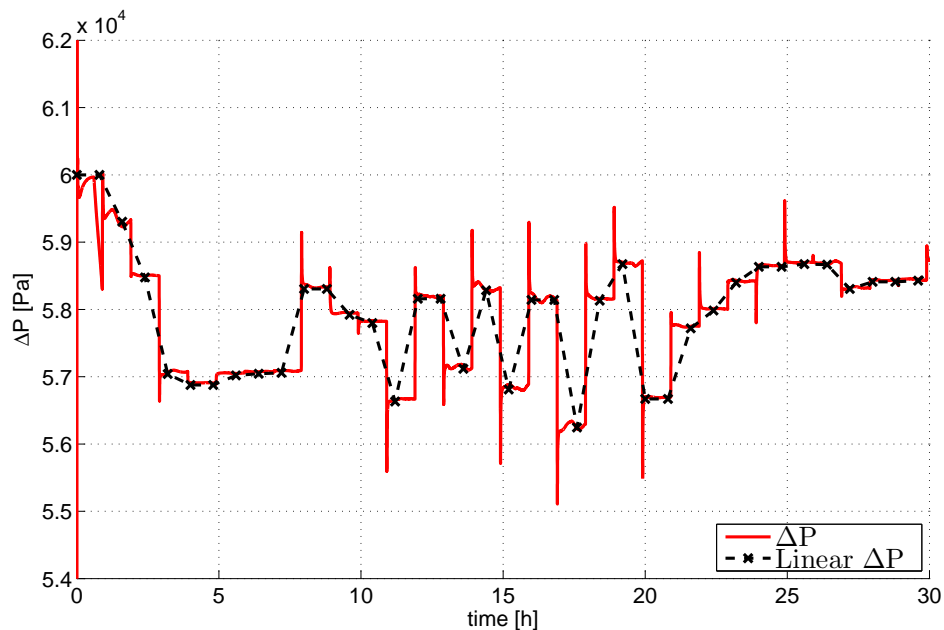


Figure 8.11: The measured and predict terminal pressures differences.

The stored heat in the accumulator tank is plotted in Figure 8.12. The tank has an initial 10 MW stored. The tank is rapidly discharged over the first 5 hours and starts to level out around 2.5 MW storage. This is likely due to the high cost on maintaining the pressure difference. MPC finds it beneficial to discharge the tank rather than charging it except at few periods.

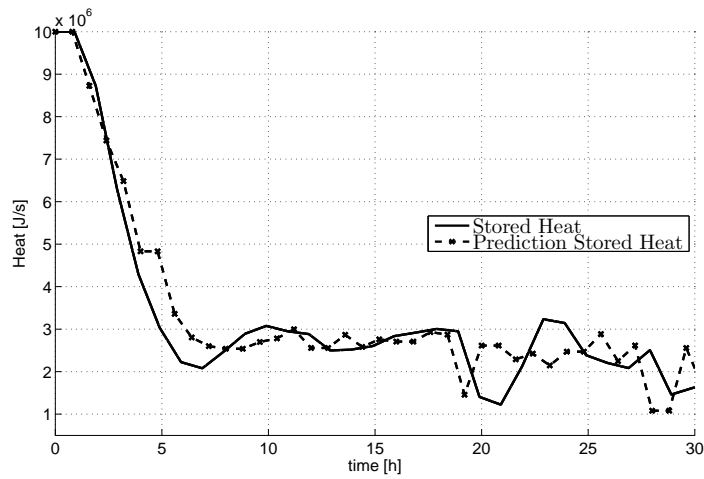


Figure 8.12: Measured and prediction storage of heat in the accumulator tank.

In Figure 8.13 the tans charge input signal to plotted. In hour 21, 22, and 23 a large charging occurs. This is because the pressure increases, as seen from Figure 8.11, and MPC finds it profitable to charge the tank when the pressure is maintained.

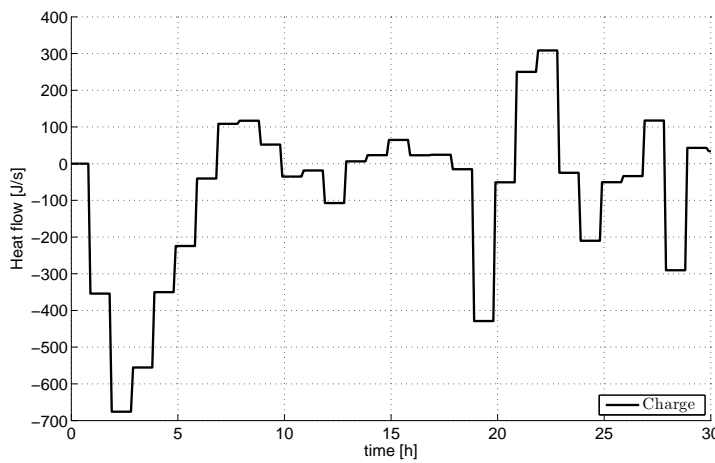


Figure 8.13: The generated charge input from MPC.

The house temperature is now without the offset as seen in Figure 8.14. This suggests that the accumulator tank helps correct some of the mismatches of meeting the exact consumer demands. Before the temperature was 0.1 °C the mean error between the two temperature are 0.0038 °C.

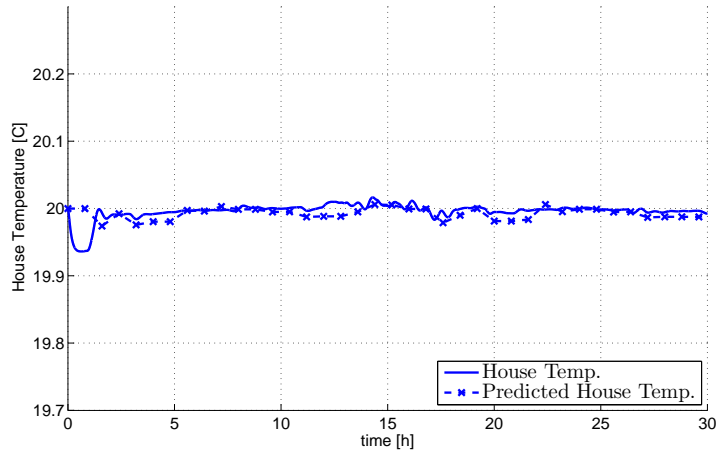


Figure 8.14: The generated charge input from MPC.

In Figure 8.15 is the supply and demand curves plotted. It is hard to see much difference but now the supply and demand curves but again taking the RMSD and AD gives the insight of the size of deviations.

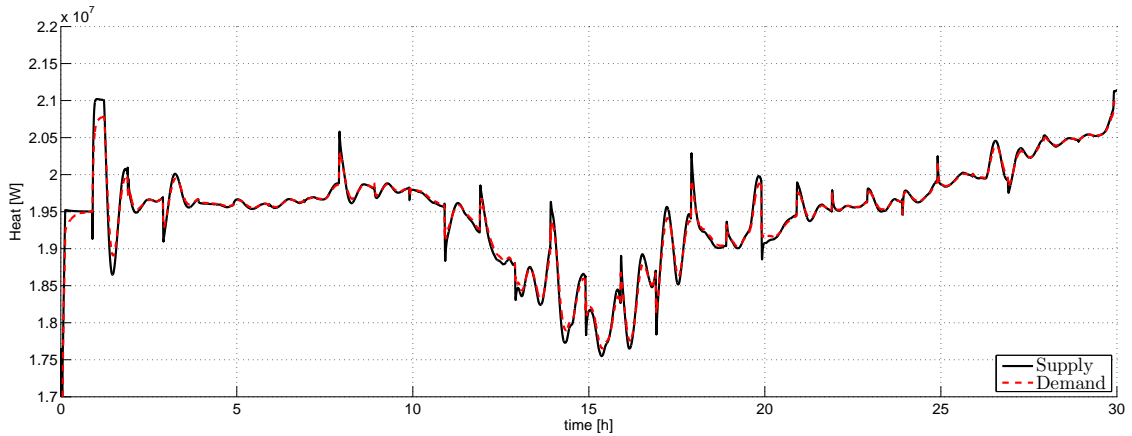


Figure 8.15: The generated charge input from MPC.

$$RMSD_{supply,Demand} = \sqrt{\frac{\sum_{k=1}^S (y_s(k) - y_d(k))^2}{S}} = 185.5 \text{ kW} \quad (8.8)$$

$$AD_{Supply,Demand} = \frac{\sum_{i=1}^S |(y_s(k) - y_d(k))|}{S} = 45.04 \text{ kW} \quad (8.9)$$

Judging from the deviations values with and without the accumulator tank then the tank decreases the average deviation but also introduces a lot of variations. This is why the RMSD is increased a lot with

the accumulator tank. In general it is best to see if the demand is matched if the house temperature is at the reference value.

The accumulated costs, seen in plot Figure 8.16, have not been reduced by introducing the accumulator tank, at this current time. The accumulated costs for unit 1 and 2 have also been switched because unit 1 is a lot more expensive and is used a lot more now.

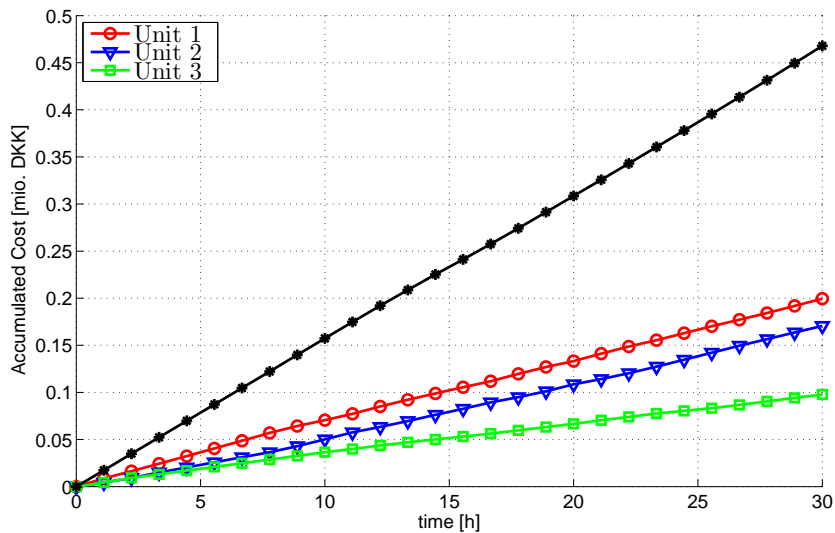


Figure 8.16: The generated charge input from MPC.

The accumulated cost for unit 1 = 0.1996 mio.DKK/MW, unit 2 = 0.1705 mio.DKK/MW, unit 3 0.09778 mio.DKK/MW and the total costs of 0.46788. The expensive unit is still the least which is as wanted. The total accumulated costs are close to identical for both cases which suggest for further development of the controller.

Reflection

These initial accumulator test shows that the accumulator can be an additional tool to met the demand curve. Usage of the accumulator tank has potential of decreasing the over all production costs though the results do not indicate this at the current development state. The accumulated costs are close to identical which indicates that MPC does not use the tank to reduce costs but rather favours meeting the production demands without regard for costs. This is most likely a weighting issue.

PART
IV

**EVALUATING THE PERFORMANCE OF THE
BALANCE CONTROLLER**

Acceptance Test

In this chapter is the requirement specification for the controller in Section 6.1 evaluated. This is to see which requirements are completed and which are not. It is important that the controller live up to these requirements to be a feasible solution and potential an alternative to the current control method existing at DH producers and SDH. All of this in the Table 9.1 together with a verdict of the requirement.

Status of Thesis Objectives

- ✓ Understanding the system
- ✓ Develop Simulation Models
- ✓ Develop Controller:
- ✓ Controller Tests:
 - Evaluate Controller Performance: The developed controller must be evaluated on its ability to fulfill the controller requirements defined in Table 6.2.

Controller Requirement Evaluation Table.

Controller Requirement	Requirement Status
The forward supply temperatures must be within 75 - 82 °C	There are some spikes in the supply temperatures during mass flow set point changes. The spikes last for very short periods approximately a tenth of an hour. This is difficult to avoid though maybe with better tuning of the temperature PI-Controllers status: partially completed.
The TPD must be kept within 0.3-0.6 Bar.	From all tests this pressure requirement is satisfied and only small high frequency spikes in the pressure occurs. status: completed.
Optimal operation set points must be calculated at each sample time for all production units.	Each sample near-optimal set points calculated. This is due to numerical issues in the optimization problem and Mosek stall problem as described in Appendix D. status: partially completed.
Minimization of the input heat energy must be possible.	In general MPC chooses to use the least costly production units to deliver the most heat. The Mosek solver does not always report back that is it an optimal solution. status: partially completed.
Physical plant limitations and constraints must be included in the optimization.	All known physical limitations have been added as constraints in the optimization problem status: completed.
A wanted consumer household temperature must be maintain within ± 1 °C of the set point.	From the two tests the house indoor temperature requirement is satisfied with a maximum RMSD of 0.01 °C. status: completed.
The consumer heat demand must always be met by the DH production units. This is closely related to the indoor household temperature requirement.	Test results show that the supply and demand curves are very close and deviating in average with 44-45 kW from each other e.g. seen in Figure 8.7 status: Partially completed.

Table 9.1: Requirement specification evaluation.

Conclusion

Now all thesis objectives have been completed and in this chapter there will be concluded the problem statement defined in the start of the thesis in Chapter 3 and the additional work done during the thesis. This involves looking back at previous chapters and sections to conclude upon: the control implementation results, the developed modelica models, the software structure, and controller design.

The outlined problem statement.

The Problem Statement

How is a controller designed which can economically minimize the cost of meeting district heating consumer demands.

The controller must be able to tackle two essential problem: one to match the consumption with production and one doing it least costly. One way of doing this is to design a cost function which consist of the two control objectives. The matching objective must always be favoured over the cost minimization but not too much. MPC offers schemes to predict future consumption and in this way schedule economic optimal production for a portfolio of arbitrary production unit.

In general the developed controller satisfies several of the specified requirements, some to more than other, as seen in the Chapter 9. This is a promising starting point for the controller. In general the balancing controller is choosing high load set points for the cheap units and low load set points for the expensive production units. This is indeed wanted a very feature of the balancing controller. The controller also ensures the terminal pressure to be maintained even though prediction errors do occur. The supply temperatures are mostly maintained within the operating bounds expect when sudden large changes in mass flow occurs. All heat demands are met by the production expect in extremities. The indoor house temperature is also maintained. Additional tuning of internal model PI-Controllers could be made in order to give MPC a more critical damped system to interact with.

It is also possible to include additional production units within the optimization problem. This makes the controller versatile to be adapted to control any producers portfolio of production units. In general the MPC structure can be extended to include several more and different types of unit and can generate set points to them all.

During implementation it was discovered that the predicted output deviated a lot from the measured values. This made it difficult for MPC to generate the correct set points. This was often corrected by the pressure controlling unit. If too large changes in the demand occurred the predictions are deviating

too much. This would make the pressure rapidly increases and decreases which created problems for simulation and FMU mode. The prediction model is not very reliable and it has a hard time predicting when not being in the linearisation point. A few methods have been used to try and push for smaller prediction errors.

Implementation of the accumulator tank within the MPC structure is in the early stages. It is necessary to be further developed upon to achieve the full potential of including an accumulator tank. The tank is primary used to maintain the pressure rather than to reduce the costs. This could be solve with modifying the weights.

It if possible import the complex nonlinear modelica plant model into Simulink using FMU and FMI. All features existing inside Dymola is available in an easy accessible way from Simulink as well. The FMU block delivers a very detailed plant model due to the advanced modelling tools available within Modelica. All derived modelica models are designed and tested as functions as intended.

Discussion

This chapter discusses and reflects over the design choices and selected tools during the thesis work. Also possible project extensions and further development ideas is discussed.

The concept of the economic balancing controller could be extended much further. At the moment the focus is on minimizing costs while producing heat, but as started in the initial phase of the thesis, this could also be done for the production of power. CHPs have a very cross coupled heat and power production therefore it makes a lot of sense to do this. Additional costs could also be included in the cost function e.g. start up and shut down costs for each available production unit. This is very relevant for production turbines which require maintenance after a number of start ups. Also maintenance cost on the distribution pipes and costs associated to transmission losses could be included. This could be done by evaluating the temperatures and pressures ensures the least wear on the system and heat losses during transmissions. Maintenance costs of the distribution network can become very costly if the system is operated under bad conditions. Also the very relevant cost of storing water inside the tank because of to heat losses over time. This would be an relative easy extension to this thesis. This term would inform the controller not to store water in the tank for too long.

In order to accommodate for the deviation in the prediction model one possible solution could be to a bank of linear models and use different linear models relative to the production loads. This could also just be one in 25 %, 50 % and 75 % and switch between them during predictions to get a better estimate. Using nonlinear control methods could also an option to make nonlinear model seem as a model linear from MPCs point of view. Another way to solve this problem is to not use liner MPC but rather use nonlinear MPC. Nonlinear MPC has been utilized in this context before with success and also using FMU models. Nonlinear MPC is when the prediction model is nonlinear.

It is difficult to design the weights in the cost function and it should be kept in mind that a simpler cost function seems to be better than a very complex one.

The software structure could be improved as well. The interface between Modelica models and Matlab Simulink is far from ideal. Matlab is not very good at reporting back errors and often an error results in a Matlab crash. This hinders the development speed and debugging. The alternative is to use Python instead of Matlab where both FMU and CVX is supported. In Python the nonlinear and linear optimization problem solver Jmodelica is also available.

The Modelica models could be develop much further to include all kind of production units, CHPs and stochastic production units. One very sensible extension could be to include the developed AHP model which Rene Just Nielsen and Kasper Vinther have developed under the Greenflex project. Modelling of distributions networks and model the true different unit relative placements could be done in Modelica with some effort.

Bibliography

- Palle Andersen and Tom S. Pedersen. Modeldannelse - Electronics and Information Technology, Aalborg University. Note, March 2006. URL <http://www.control.aau.dk/~dimon/edu/grundfos/lecture1/modelnote.pdf>.
- Ashfaq, Khan, hej, hej, and eh, May 2014.** Ahsan Ashfaq, Akif Zia Khan, hej, hej, and eh. *Optimization of Economic Load Dispatch Problem by Linear Programming Modified Methodology*. 2nd International Conference on Emerging Trends in Engineering and Technology ICETET, 2014. URL <http://iieng.org/siteadmin/upload/3540E0514510.pdf>.
- Blochwitz, Otter, Arnold, Bausch, Clauß, Elmqvist, Junghanns, Mauss, Monteiro, Neidhold, Neumerkel, Olsson, Peetz, and Wolf.** T. Blochwitz, M. Otter, M. Arnold, C. Bausch, C. Clauß, H. Elmqvist, A. Junghanns, J. Mauss, M. Monteiro, T. Neidhold, D. Neumerkel, H. Olsson, J.-V. Peetz, and S. Wolf. *The Functional Mockup Interface for Tool independent Exchange of Simulation Models*. URL https://svn.modelica.org/fmi/branches/public/docs/Modelica2011/The_Functional_Mockup_Interface_paper.pdf.
- Boyd and Vandenberghe, 2004.** Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- Casella.** Francesco Casella. *Object-Oriented Modelling and Simulation, 01 Introduction Modelica*. Lecture 1 slides.
- Cassioli, June 2014.** Andrea Cassioli. *What if the solver stalls?* web - blog, June 2014. URL <http://blog.mosek.com/2014/06/what-if-solver-stall.html>. Mosek blog.
- CVX Research, May 2015.** CVX Research. *CVX Research*. web, May 2015. URL <http://cvxr.com>.
- DANSK INGENIØRFORENING, 1968.** DANSK INGENIØRFORENING. *REGLER FOR BEREGNING AF BYGNINGERS VARMETAB*. TEKNISK FORLAG, 3. edition, 1968.
- Dassault Systèmes AB, September 2014.** Dassault Systèmes AB. *User Manual Volume 2 - Dymola Dynamic Modeling Laboratory*. September 2014.
- dbdh.dk, May 2015a.** dbdh.dk. *District heating history*. web, May 2015a. URL <http://dbdh.dk/district-heating-history/>.

- dbdh.dk, May 2015b.** dbdh.dk. *District Heating Characteristics*. Web, May 2015b. URL <http://dbdh.dk/characteristics/>.
- Dogra, Gupta, and Saroa, January 2014.** Rahul Dogra, Nikita Gupta, and Harsha Saroa. *Economic Load Dispatch Problem and Matlab Programming of Different Methods*. International Conference of Advance Research and Innovation (ICARI), 2014. URL <http://www.ijari.org/currentissue/icari2014/icari-ps-14-02-102.pdf>. ISBN 978-93-5156-328-0.
- Edlund, Bendtsen, Børresen, and Mølbak, July 2008.** Kristian Edlund, Jan Dimon Bendtsen, Simon Børresen, and Tommy Mølbak. *Introducing Model Predictive Control for Improving Power Plant Portfolio Performance*. Proceedings of the 17th World Congress. The International Federation of Automatic Control Seoul, Korea, 2008.
- Elmqvist and Mattsson, October 1997.** Hilding Elmqvist and Sven Erik Mattsson. *AN INTRODUCTION TO THE PHYSICAL MODELING LANGUAGE MODELICA*. Proceedings of the 9th European Simulation Symposium, ESS'97, Passau, Germany, 1997.
- energiforskning.dk, September 2014.** energiforskning.dk. *Green Flex Ø Mobilisering af driftsflexibilitet i grøn energiproduktion*. Webpage, September 2014. URL <http://www.energiforskning.dk/da/project/green-flex-mobilisering-af-driftsflexibilitet-i-groen-energiproduktion>.
- Energinet.dk, January 2015.** Energinet.dk. *Wind turbines reached record level in 2014*. web, January 2015. URL <http://www.energinet.dk/EN/El/Nyheder/Sider/Vindmoeller-slog-rekord-i-2014.aspx>.
- Energinet.dk, January 2014.** Energinet.dk. *2013 was a record-setting year for Danish wind power*. web, January 2014. URL <http://www.energinet.dk/EN/El/Nyheder/Sider/2013-var-et-rekordaar-for-dansk-vindkraft.aspx>.
- Energinet.dk, January 2013.** Energinet.dk. *Fossil fuels: Coal, natural gas and oil will continue to be used in the Danish energy system for a number of years*. 2013. URL <http://energinet.dk/EN/KLIMA-OG-MILJOE/Energi-og-klima/Forskning-i-vedvarende-energi/Sider/Fossile-braendsler.aspx>.
- energitilsynet.dk, April 2014.** energitilsynet.dk. *Kort om fjernvarmesektorens økonomiske omfang*. pdf - web, April 2014. URL http://energitilsynet.dk/fileadmin/Filer/0_-_Nyt_site/VARME/Fjernvarmestatistik/ny1.1_Fjernvarmesektorens_oekonomiske_omfang.pdf. Fjernvarmestatistik.
- energitilsynet.dk, Marts 2015.** energitilsynet.dk. *Energitilsynets prisstatistik for fjernvarmeområdet*. pdf - web, Marts 2015. URL http://energitilsynet.dk/fileadmin/Filer/0_-_Nyt_site/VARME/Prisstatistik/2015/2Marts_2015_.pdf.

- Energytransition.de, December 2013.** Energytransition.de. *Flexible power production (no more baseload)*. web, December 2013. URL
<http://energytransition.de/2012/10/flexible-power-production-no-more-baseload/>.
- engineeringtoolbox.com, May 2015.** engineeringtoolbox.com. *Thermal Conductivity of some common Materials and Gases*. web, May 2015. URL
http://www.engineeringtoolbox.com/thermal-conductivity-d_429.html.
- EURELECTRIC.** Flexible generation: Backing up renewables. Report, October 2011. URL
http://www.eurelectric.org/media/61388/flexibility_report_final-2011-102-0003-01-e.pdf.
- Francesco Casella.** Dipartimento di Elettronica e Informazione Politecnico di Milano
Francesco Casella. *Object-Oriented Modelling and Simulation, 03 Symbolic Manipulation*. Lecture 3 slides.
- geotermi.dk, april 2013.** geotermi.dk. *Sønderborg Geotermi*. web, april 2013. URL
<http://www.geotermi.dk/geotermiske-anlaeg/soenderborg>.
- Haynes, June 2014.** William M. Haynes. *CRC Handbook of Chemistry and Physics 95th Edition*. CRC Pressm, Taylor & Francis Group, June 2014. ISBN 9781482208672.
- Johansen, October 2014.** Jørgen Johansen. *Chief Engineer at Sønderborg District Heating*. Phone correspondence with SDH., October 2014.
- Santos Coelho, June 2008.** Chu-Sheng Lee Leandro dos Santos Coelho. *Solving economic load dispatch problems in power systems using chaotic and Gaussian particle swarm optimization approaches*. Electrical Power and Energy Systems 30, 2008.
- Petersen and Gram-Hanssen, 2005.** Kirstine Nærvig Petersen and Kirsten Gram-Hanssen. *Husholdningers energi- og vandforbrug*. SBI 2005:09. Statens Byggeforskningsinstitut, 1. edition, 2005. ISBN: 87-563-1231-8, (pp. 10, pp. 62).
- Serway and John W. Jewett, 2010.** Raymond A. Serway and Jr. John W. Jewett. *PHYSICS FOR SCIENTISTS AND ENGINEERS with Modern Physics*. Mary Finch, 2010.
- Sønderborg Fjernvarme, October 2014.** *Sønderborg Fjernvarme*. web, October 2014. URL
http://www.sonderborg-fjernvarme.dk/?page_id=7.
- Sønderborg Fjernvarme. Årsrapport 2014 Sønderborg Fjernvarme.
- Ole Sørensen and Palle Andersen. Optimal Control - Electronics and Information Technology, Aalborg University. Lecture Note, February 6 2010.
- Stéphane Velut, Per-Ola Larsson, Johan Windahl, Linn Saarinen, March 2014.** Katatrina Boman Stéphane Velut, Per-Ola Larsson, Johan Windahl, Linn Saarinen. *Short-term production planning for district heating networks with JModelica.org*. Proceedings of the 10th International Modelica Conference, 2014.

Wang and Boyd, MARCH 2010. Yang Wang and Stephen Boyd. *Fast Model Predictive Control Using Online Optimization*. IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, VOL. 18, NO. 2, 2010. URL http://stanford.edu/~boyd/papers/pdf/fast_mpc.pdf.

Winther, Hammer, Ulbjerg, Larsen, and Frederiksen, 2012. Niels Winther, Flemming Hammer, Flemming Ulbjerg, Jesper Møller Larsen, and John Frederiksen. *Varme Ståbi*, volume 6. Nyt Teknisk Forlag, 2012. Fjernvarme Kapitel pp. 93.

Yang and He, Aug 2013. Xin-She Yang and Xingshi He. *Firefly Algorithm: Recent Advances and Applications*. Vol. 1, 2013. URL <http://arxiv.org/pdf/1308.3898.pdf>.

Introduction to Modelica and Dymola

In the following appendix a general introduction to Modelica and Dymola is given and to the most relevant subjects.

Modelica is an object-orientated modelling language for modelling of multi-domain complex physical systems. The Modelica language is open source and is developed by the non-profit Modelica Association. The main goal with modelica is to make it easy to exchange models and model libraries. This is done by object-oriented constructs to facilitate reuse of already developed models. Modelica works as a non-causal modelling language with true ordinary differential and algebraic equations. [Elmqvist and Mattsson, 1997]. The overall Modelica and Dymola O-O environment is seen in Figure A.1. A model description is developed in the Modelica language together with some test description and is simulation within Dymola.

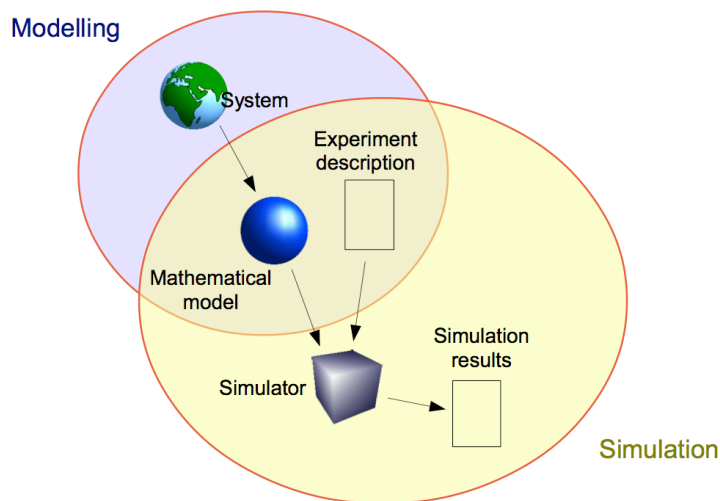


Figure A.1: Overview diagram of the Modelica and Dymola as a combination [Casella]

Causal and A-Causal Approaches

Causal: One method of modelling and simulating of physical systems is through a causal or also known as signal flow approach. Signals are transmitted between blocks transferring information of variables from outputs of the current block to inputs of the next block. In this way a set of aggregated models can be constructed to simulate a real system. This method designs a system which is relatively easy to solve by first propagating values and solving with numerical integration e.g. using ODE solvers like Euler or

Runge-Kutta integration [Casella]. An example of a simulation environment which uses causal models is Simulink which is a tool often used at AAU.

Causal Pros and Cons

- Pro: Easy to solve and simulated with well established ODE solvers.
- Con: Hard to reconfigure and hard to understand for others.
- Con: Implemented models can only be reused for similar context systems.
- Con: For more complex systems the modelled physical reality disappears using this approach

A-causal: The other method of modelling and simulating approach is using a-causal models. Each model is described in a context-independent form and interaction between models is formalized as connector equations, not to be mistake for signal. [Casella].

A-Causal Pros and Cons

- Pro: Basic models are easily reused and modified.
- Pro: Models are easy to understand for others.
- Con: Models are not tailored for a solution algorithm which makes it difficult to find numerical solutions.

Connectors / Ports

All O-O models are connected with connectors also refereed to as ports. A connector defines at least two variables: a potential/effort and flow variable. For different modelling disciplines connectors are defined as in Table A.1 [Casella].

Variable	Electrical	Mechanical	Hydraulic	Thermal
Potential/Effort	Voltage	Displacement/Angle	Pressure	Temperature
Flow	Current	Force/Torque	Volume flow	Heat flow

Table A.1: Connectors effort and flow variables for different modelling disciplines.

The connectors are not predefined as input or outputs and can be both during simulation. An example of code for a connector is seen in Section A [Casella].

Connectors

```

1 connector Port "Model of an Hydraulic Port"
2 Pressure P; "Pressure at Port"
3 flow v; "Volumetric flow rate through Port"
4 end Port;
```

Code snip A.1: Hydraulic Connector

In the stream connector stream variables are included. Stream connectors carries medium properties e.g. specific enthalpy.

Stream Connectors

```

1 connector FluidPort
2 replaceable package Medium = Modelica.Media.Interfaces.PartialMedium;
3 "Mass flow rate from the connection point into the component"
4 flow Medium.MassFlowRate m_flow;
5 "Thermodynamic pressure in the connection point"
6 Medium.AbsolutePressure p;
7 "Specific enthalpy close to connection point if m_flow < 0"
8 stream Medium.SpecificEnthalpy h_outflow;
9 "Independent mixture mass fractions close to connection point if m_flow < 0"
10 stream Medium.MassFraction Xi_outflow[Medium.nXi];
11 "Properties c_i/m close to the connection point if m_flow < 0"
12 stream Medium.ExtraProperty C_outflow[Medium.nC];
13 end FluidPort;

```

Code snip A.2: Source code for the Modelica accumulator tank model

All connector equations e.g. $\text{pump.port.b.h_flow} = \text{pipe.port.a.h_flow}$, no change in energy between the two ports, are included in the system Differential Algebraic Equation (DAE) in order describe how the individual models are connected.

Figure A.2 shows the following: how a specified medium model can be chosen and how connectors combines model components.

- Fluid medium: a specified medium model is chosen which include medium properties and a set of components independent variables.
- Ideal mixing: when two fluid lines are connected to each other ideal mixing occurs as in the blue node in Figure A.2.
- Reversible flow: it is also possible to have reverse flows e.g. a reversed flow coming from Pipe Component C if there was a higher pressure on *PipeC.port.b*.

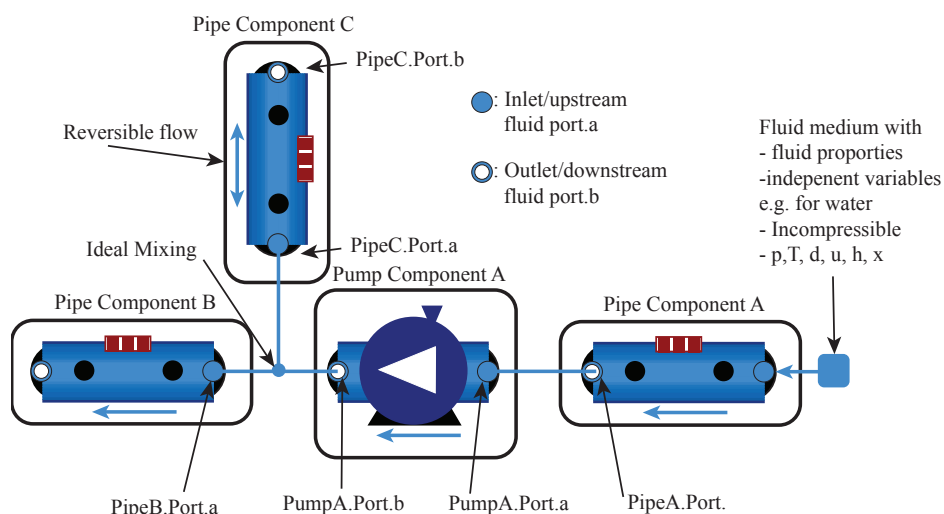


Figure A.2: This shows some of the conventions used within dymola.

Linearization of System Models

Nonlinear models can be linearized by use of a first order Taylor expansion series and higher orders terms are neglected. A nonlinear system with n state variables, m input variables, and r output variables is described by the set of differential equations:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, \dots, x_n, u_1, \dots, u_m) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, \dots, x_n, u_1, \dots, u_m) \\ y_1 &= g_1(x_1, \dots, x_n, u_1, \dots, u_m) \\ &\vdots \\ y_r &= g_r(x_1, \dots, x_n, u_1, \dots, u_m) \end{aligned}$$

or in the equivalent matrix form:

$$\dot{\mathbf{x}} = \mathbf{f}(x, u) \tag{B.3a}$$

$$\dot{\mathbf{y}} = \mathbf{g}(x, u) \tag{B.3b}$$

where

$\mathbf{x} \in \mathbb{R}^n$ is the system states.

$\mathbf{u} \in \mathbb{R}^m$ is the system input.

$\mathbf{y} \in \mathbb{R}^r$ is the system outputs.

The first order Taylor expansion series for a n state variables system is given as (B.4b).

$$\dot{x} \approx f(\bar{x}, \bar{u}) + \left. \frac{\partial f}{\partial x_1} \right|_{x=\bar{x}} \hat{x}_1 + \left. \frac{\partial f}{\partial x_2} \right|_{x=\bar{x}} \hat{x}_2 + \dots + \left. \frac{\partial f}{\partial x_n} \right|_{x=\bar{x}} \hat{x}_n + \tag{B.4a}$$

$$\left. \frac{\partial f}{\partial u_1} \right|_{u=\bar{u}} \hat{u}_1 + \left. \frac{\partial f}{\partial u_2} \right|_{u=\bar{u}} \hat{u}_2 + \dots + \left. \frac{\partial f}{\partial u_m} \right|_{u=\bar{u}} \hat{u}_m \tag{B.4b}$$

where

- $\hat{x} = x - \bar{x}$ is the small signal perturbation vector
- \bar{x} is the steady-state operating point vector
- x is the state vector

$$\bar{x} = \begin{bmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_n \end{bmatrix} \quad x = \begin{bmatrix} \hat{x}_1 \\ \vdots \\ \hat{x}_n \end{bmatrix} = \begin{bmatrix} x_1 - \bar{x}_1 \\ \vdots \\ x_n - \bar{x}_n \end{bmatrix} \quad (\text{B.5})$$

For the linearization at the steady-state operating point.

$$f_1(\bar{x}_1, \dots, \bar{x}_n, \bar{u}_1, \dots, \bar{u}_m) \quad \dots \quad f_n(\bar{x}_1, \dots, \bar{x}_n, \bar{u}_1, \dots, \bar{u}_m) = 0 \quad (\text{B.6a})$$

$$g_1(\bar{x}_1, \dots, \bar{x}_n, \bar{u}_1, \dots, \bar{u}_m) \quad \dots \quad g_n(\bar{x}_1, \dots, \bar{x}_n, \bar{u}_1, \dots, \bar{u}_m) = \bar{y} \quad (\text{B.6b})$$

The linearisation matrices or Jacobian matrices for the system is then given as:

$$A_{ij} = \left. \frac{\partial f_i}{\partial x_j} \right|_{x=\bar{x}, u=\bar{u}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \quad (\text{B.7})$$

$$B_{ij} = \left. \frac{\partial f_i}{\partial u_j} \right|_{x=\bar{x}, u=\bar{u}} \quad C_{ij} = \left. \frac{\partial f_i}{\partial x_j} \right|_{x=\bar{x}, u=\bar{u}} \quad D_{ij} = \left. \frac{\partial f_i}{\partial u_j} \right|_{x=\bar{x}, u=\bar{u}} \quad (\text{B.8})$$

In short vector notation the full linearised system on state space form is formulated as.

$$\dot{\hat{x}} = A_{ij}\hat{x} + B_{ij}\hat{u} \quad (\text{B.9a})$$

$$\hat{y} = G_{ij}\hat{x} + D_{ij}\hat{u} \quad (\text{B.9b})$$

Convex Sets and Functions

The definition of convex sets and functions are described in this appendix.

Convex Sets

A set C is convex if the line segment between any two points in C lies in C . This means that for any x_1 and $x_2 \in C$ and θ with $0 \leq \theta \leq 1$ [Boyd and Vandenberghe, 2004].

$$x = \theta x_1 + (1 - \theta)x_2 \in C \tag{C.1}$$

(Every affine and linear subset is convex.)

A convex combination of any points $x_1 \cdots x_n$ is given as:

$$x = \theta_1 x_1 + \theta_2 x_2 \cdots \theta_n x_n \tag{C.2}$$

with $\theta_1 + \cdots + \theta_n = 1, \quad \theta_i \geq 0$

Convex Functions

A function f is convex if $f : \mathbb{R}^n \rightarrow \mathbb{R}$ if domain of f is a convex set and for all x_1 and $x_2 \in \text{dom } f$ and θ with $0 \leq \theta \leq 1$ and with (C.3) [Boyd and Vandenberghe, 2004].

$$f(\theta x_1 + (1 - \theta)x_2) \leq \theta f(x_1) + (1 - \theta)f(x_2) \tag{C.3}$$

f is strictly convex if $\text{dom } f$ is convex and with (C.4) and for $x_1, x_2 \in \text{dom } f, x_1 \neq x_2$ and $0 < \theta < 1$ [Boyd and Vandenberghe, 2004].

$$f(\theta x_1 + (1 - \theta)x_2) < \theta f(x_1) + (1 - \theta)f(x_2) \tag{C.4}$$

An illustration of a convex function is seen in Figure C.1(a).

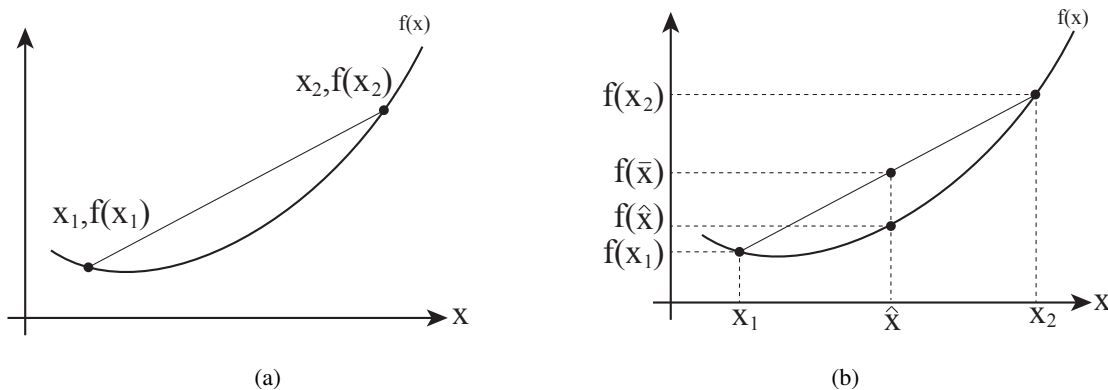


Figure C.1: (a) A convex function with a line segment between two points being above the function itself.: (b):

The convex combination of point denotes all points in the interval x_1 to x_2 . For $\theta = 1 : \hat{x} = x_1$ and for $\theta = 0 : \hat{x} = x_2$. The convex combination \hat{x} and the function values of \hat{x} is given as.

$$\hat{x} = \theta x_1 + (1 - \theta)x_2 \tag{C.5a}$$

$$f(\hat{x}) = f(\theta x_1 + (1 - \theta)x_2) \tag{C.5b}$$

The fact that and the property of linearity.

$$f(\hat{x}) < f(\bar{x}) \Leftrightarrow f(\theta x_1 + (1 - \theta)x_2) < \theta f(x_1) + (1 - \theta)f(x_2) \tag{C.6a}$$

Convex functions on \mathbb{R}

- Affine functions: $ax+b$ on $\mathbb{R} \forall x_1, x_2 \in \mathbb{R}$
- Exponential functions: e^ax for any $x_1 \in \mathbb{R}$
- Power or absolute values $|x|^p$ on \mathbb{R} for $p \geq 1$

Convex functions on \mathbb{R}^n

- Affine functions: $f(x) = a^T x + b$
- All norms $\|x\|_p = \left(\sum_{i=1}^n |x_i|^p \right)^{1/p}$ for $p \geq 1$; $\|x_{\text{inf}}\| = \max_k |x_k|$
- Max functions $f(x) = \max x_1, \dots, x_n$.

[Boyd and Vandenberghe, 2004].

Stall of Mosek Solver

During the simulation tests the Mosek solver sometimes stalls which means that it can not make sensible progress towards a feasible solution. Due to numerical issues the Mosel solver may not be able to compute the next search direction and if this leads to an extremely short step towards the optimal solution then Mosek stops and writes Stalled.

$$x^+ = x + \alpha d_x \tag{D.1}$$

where

- x Is the current guess for an optimal solution.
- x^+ is the next guess for an optimal solution.
- d_x is the step direction decided in some way by Mosek.
- α is a non-negative step size.

The iteration procedure of the Mosek solver is seen in Figure D.1. By iteration the solver goes towards the optimal solution.

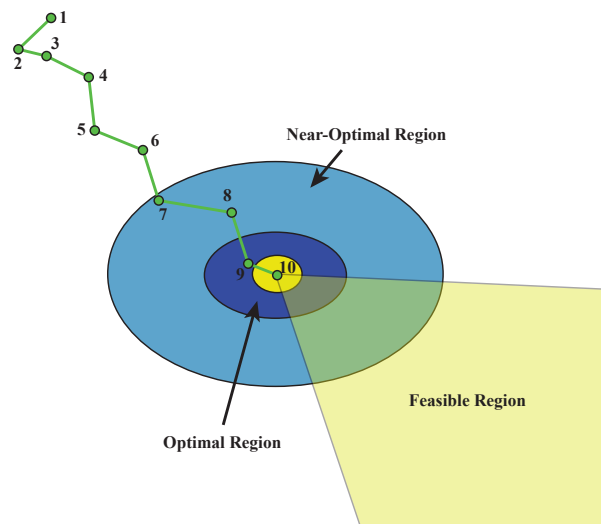


Figure D.1: Mosek’s solver iterates towards an optimal solution.

Even though the Mosek solver stalls the solution is still meaningful since the solver is able to check the feasibility and residual values. The solution could therefore be a near-optimal solution [Cassioli, 2014].

Matlab MPC Function Source Code

```

1 function [ForwardTSet,mSet,alpha,dpSet,ForwardTSet1,mSet1,alpha1,dpSet1,...
2         ForwardTSet2,mSet2,alpha2,dpSet2,THouseSet,TAmbient,charge]...
3         = cvx_function2(x_meas,y_meas,u_old,clock)
4
5 %% Inital Simulation Parameters
6 Ts = 3600;
7 t_scale = 0.5;
8 t_delay = 0;
9
10 %% Import Linear Prediction Model
11 [A, B, C, D, xName, uName, yName] = tloadlin('FullModel_18_05','quiet');
12 sys_c = ss(A,B,C,D);
13
14 %% Discretize Linear Model
15 rsys_d = c2d(sys_c,Ts);
16 Ad = rsys_d.a;
17 Bd = rsys_d.b;
18 Cd = rsys_d.c;
19 Dd = rsys_d.d;
20
21 %% Desfine system matrix sizes
22 n = size(Ad,1); % nr. of states
23 p = size(Bd,2); % nr. of inputs
24 q = size(Cd,1); % nr. of outputs
25
26 %% Define Control/Predict Horizon
27 N = 48; % Horizon
28
29 %% MPC Reference Extended Model
30 Hr = eye(q,q);
31 Ae = [Ad zeros(n,q); zeros(q,n) Hr];
32 Be = [Bd; zeros(q,p)];
33 Ce = [-Cd -Dd Hr];
34
35 %% Producer Model Constraints
36 % Mass Flows Inputs
37 mMinP = 0*ones(1,N);
38 mMaxP = 200*ones(1,N);
39
40 % Mass Flows Outputs
41 if clock < Ts
42 mMin = 0*ones(1,N);
43 mMin2 = 10*ones(1,N);
44 mMax = 100*ones(1,N);

```

```

45 mMax2 = 100*ones(1,N);
46 else
47 mMin = 0*ones(1,N);
48 mMin2 = 10*ones(1,N);
49 mMax = 40*ones(1,N);
50 mMax2 = 60*ones(1,N);
51 end
52
53 % Supply Temperature Inputs
54 TFPmin = 40*ones(1,N);
55 TFPmax = 100*ones(1,N);
56
57 % Supply Temperature Outputs
58 TFmin = 75*ones(1,N);
59 TFmax = 82*ones(1,N);
60
61 % Pressure Difference Input
62 dpPmax = 15e4*ones(1,N);
63 dpPmin = 1e1*ones(1,N);
64
65 % Pressure Difference Output
66 if clock < Ts
67 dpMax = 15e4*ones(1,N);
68 dpMin = 1e4*ones(1,N);
69 else
70 dpMax = 6e4*ones(1,N);
71 dpMin = 3e4*ones(1,N);
72 end
73
74 %% Tank Capacity Constraints
75 accMin = 1e6*ones(1,N);
76 accMax = 100e6*ones(1,N);
77
78 %% Consumer Valve Constraints
79 vo_min = 0*ones(1,N);
80 vo_max = 1*ones(1,N);
81
82 %% Producer Alpha Values
83 alpha = 1*ones(1,N);
84 alpha1 = 0*ones(1,N);
85
86
87 %% Slew Rate Constraints
88 % Producers mass flow
89 u_m_p2min = -(40/50)*60*60;
90 u_m_p2max = (40/50)*60*60;
91 u_m_p2min2 = -60;
92 u_m_p2max2 = 60;
93
94 % Temp
95 u_temp_min = -60;
96 u_temp_max = 60;
97
98
99 % Slew rate Acc. Tank
100 u_acc_min = (-10e6/3.6/Ts);
101 u_acc_max = (10e6/3.6/Ts);
102

```

```

103 u_acc_min_P = (-5e6)/Ts*ones(1,N);
104 u_acc_max_P = (5e6)/Ts*ones(1,N);
105
106
107 %% Input Operating Points
108 FTemp_op = 80*ones(1,N);
109 dp_op = 6e4*ones(1,N);
110 massflow_op = 20*ones(1,N);
111 massflow_op1 = 27.57*ones(1,N);
112 tambient_op = 6*ones(1,N);
113 TFP_op = 80*ones(1,N);
114 vo_op = 0.136347*ones(1,N);
115 accTank_op = 10e6;
116
117 %% Not-square weights
118 % Temperature
119 Q1T = 200/(82);
120 Q1T1 = 100;
121
122 Q1P = 10/(82);
123
124 % Pressure Weight
125 Q1P1 = 1e5/(6e4);
126
127 % Mass flow
128 Qump1 = 10/(4000);
129
130
131 if clock < Ts
132 % Heat costs
133 Q_heat1 = 8.0000e-12;
134 Q_heat2 = 8.0000e-12;
135 Q_heat3 = 8.0000e-12;
136 else
137 % Heat costs
138 Q_heat1 = 8.0000e-04;
139 Q_heat2 = 2.0000e-04;
140 Q_heat3 = 8.0000e-04;
141 end
142
143 % Large alpha cost value
144 Qmsc = 1e4;
145
146 %% Load Ambient Temperature Data
147 Tf = 200e3;
148 ta2 = zeros(1,N);
149
150 load('T_Ambient_18_03_2014_to_18_03_2015_snip.mat');
151 ta3 = ta3'*t_scale;
152
153 % Find the mean for each hour for the entire window
154 for i = 1:N
155 ta2(i) = mean(ta3(clock+1+Ts*(i-1)+(t_delay*Ts):clock+Ts*i+(t_delay*Ts)));
156 end
157
158 %% Old Variables
159 x_m = x_meas;
160 u_o = u_old;

```

```

161 y_m = y_meas;
162
163
164 %% CVX Optimization
165 cvx_begin
166     cvx_precision
167         variables u(p,N) x(n,N) e(q,N) y(q,N) xr(q,N) alpha(1,N) alpha1(1,N)
168
169         J = sum(...
170             ... % dP Comfort Cost
171             square_pos(norm(Q1P1*(e(4,1:N))))...
172             ... % Fuel Output Costs
173             + Q_heat1*(ones(1,N)*y(2,1:N))'...
174             + Q_heat2*(ones(1,N)*y(7,1:N))'...
175             + Q_heat3*(ones(1,N)*y(11,1:N))'...
176             ... % Delta U Mass flow Costs
177             + Qump_p*ones(1,N)*([y(3,1) y(3,1:N-1)]-y(3,1:N))'...
178             + Qump1*ones(1,N)*([u_o(6,1) u(6,1:N-1)]-u(6,1:N))'...
179             + Qump1*ones(1,N)*([u_o(10,1) u(10,1:N-1)]-u(10,1:N))'...
180             ... % Delta U Temperature Costs
181             + Q1T*ones(1,N)*([u_o(1,1) u(1,1:N-1)]-u(1,1:N))'...
182             + Q1T*ones(1,N)*([u_o(5,1) u(5,1:N-1)]-u(5,1:N))'...
183             + Q1T*ones(1,N)*([u_o(9,1) u(9,1:N-1)]-u(9,1:N))'...
184             ... % Delta Terminal Pressure Difference Costs
185             + Q1P*ones(1,N)*([u_o(4,1) u(4,1:N-1)]-u(4,1:N))'...
186             ... % Soft-constraints
187             + square_pos(norm(Qmsc*(alpha)))...
188             + square_pos(norm(Qmsc*(alpha1)))...
189             ... % Delta Accumulator Tank Charge
190             + Q1T1*ones(1,N)*([u_o(15,1) u(15,1:N-1)]-u(15,1:N))');
191
192
193         minimize(J);
194
195     subject to
196
197 % System Dynamics with Reference Model
198     [x(:,1:N); xr(:,1:N)] == Ae*[x_m(:,1) x(:,1:N-1)]; xr(:,1:N)]+ Be*[u(:,1:N)];
199     e(:,1:N) == Ce*[x(:,1:N); u(:,1:N); xr(:,1:N)];
200     y(:,1:N) == Cd*[x(:,1:N)] + Dd*[u(:,1:N)];
201
202 % Slew-Rate Constraints
203     % Producer Mass Flow
204     repmat(u_m_p2min,1,N) <= [u_o(6,1) u(6,1:N-1)] - u(6,1:N) <= repmat(u_m_p2max,1,N);
205     repmat(u_m_p2min,1,N) <= [u_o(10,1) u(10,1:N-1)] - u(10,1:N) <= repmat(u_m_p2max,1,N);
206     % Producer Supply Temperatures
207     repmat(u_temp_min,1,N) <= [u_o(1,1) u(1,1:N-1)] - u(1,1:N) <= repmat(u_temp_max,1,N);
208     repmat(u_temp_min,1,N) <= [u_o(5,1) u(5,1:N-1)] - u(5,1:N) <= repmat(u_temp_max,1,N);
209     repmat(u_temp_min,1,N) <= [u_o(9,1) u(9,1:N-1)] - u(9,1:N) <= repmat(u_temp_max,1,N);
210     % Accumulator Tank Charge Inputs
211     repmat(u_acc_min,1,N) <= [u_o(15,1) u(15,1:N-1)] - u(15,1:N) <= repmat(u_acc_max,1,N);
212
213 % Input Constraints
214     % Producer Forward Temperature Inputs
215     TFPmin-TFP_op <= u(1,1:N) <= TFPmax-TFP_op; % Forward Temp. P1 bounds
216     TFPmin-TFP_op <= u(5,1:N) <= TFPmax-TFP_op; % Forward Temp. P1 bounds
217     TFPmin-TFP_op <= u(9,1:N) <= TFPmax-TFP_op; % Forward Temp. P1 bounds
218

```

```

219 % Producer Mass Flow inputs
220 u(2,1:N) == 0;
221 mMinP-massflow_op <= u(6,1:N) <= mMaxP-massflow_op; % Massflow control set P2
222 mMinP-massflow_op <= u(10,1:N) <= mMaxP-massflow_op; % Massflow control set P3
223
224 % Producer Pressure Difference Inputs
225 dpPmin-dp_op <= u(4,1:N) <= dpPmax-dp_op;
226 u(8,1:N) == 0;
227 u(12,1:N) == 0;
228
229 % Producer Alphas
230 u(3,1:N) == 0; % Alpha1
231 u(7,1:N) == 0; % Alpha2
232 u(11,1:N) == 0; % Alpha3
233
234 % Consumer House Inputs
235 u(13,1:N) == THouseSet; % Consumesr Indoor temp set point.
236 u(14,1:N) == ta2-tambient_op; % Outdoor Abient temperature
237
238 % Accumulator Tank Charge Input
239 u_acc_min_P <= u(15,1:N) <= u_acc_max_P;
240
241 % Reference vector
242 xr(1:3,1:N) == 0;
243 xr(4,1:N) == u(4,1:N);
244 xr(5:end,1:N) == 0;
245
246
247 % Output Constraints
248 % Producer Supply Temperatures.
249 TFmin-(TFP_op) <= y(5,1:N) <= TFmax-(TFP_op);
250 TFmin-(TFP_op) <= y(10,1:N) <= TFmax-(TFP_op);
251 TFmin-(TFP_op) <= y(14,1:N) <= TFmax-(TFP_op);
252
253 % Valve Opening
254 vo_min-vo_op <= y(15,1:N) <= vo_max-vo_op;
255
256 % Producer dP
257 (dpMin)-(dp_op)-alpha1 <= y(4,1:N) <= (dpMax)-(dp_op)+alpha1;
258
259 % Producer Mass flow
260 mMin2-(massflow_op1)-alpha <= y(3,1:N) <= mMax2-(massflow_op1)+alpha;
261 mMin-massflow_op <= y(8,1:N) <= mMax-massflow_op;
262 mMin-massflow_op <= y(12,1:N) <= mMax-massflow_op;
263
264 % Accumulator Tank
265 accMin-accTank_op <= y(16,1:N) <= accMax-accTank_op;
266
267 cvx_end
268 %% Update the next step output of function, input to plant.s
269 ForwardTSet = full(u(1,1));
270 mSet = full(u(2,1));
271 alpha = full(u(3,1));
272 dpSet = full(u(4,1));
273
274 ForwardTSet1 = full(u(5,1));
275 mSet1 = full(u(6,1));
276 alpha1 = full(u(7,1));

```

```
277 dpSet1 = full (u(8,1));
278
279 ForwardTSet2 = full(u(9,1));
280 mSet2 = full (u(10,1));
281 alpha2 = full (u(11,1));
282 dpSet2 = full (u(12,1));
283
284 THouseSet = full(u(13,1));
285 TAmbient = full (u(14,1));
286 charge = full (u(15,1))
```

Code snip E.1: Source code for the Modelica accumulator tank model

Attached CD

MATLAB Scripts

Simulink Simulation Files

Modelica District Heating Library

Litterature

Digital Copy of the report