



Load Prediction Algorithms for Multi-Source District Heating

Endre Eisenga

IE&M Bachelor Integration Project

University of Groningen Faculty of Science and Engineering

June 2021

Supervisors: prof.dr.ir. J.M.A. (Jacquelien) Scherpen, J. (Juan)E. Machado Martinez and A. Bosch

> Student number: S3510913 E.J.R.Eisenga@student.rug.nl

Abstract

Multi-source district heating is a network of heat producers and heat consumers connected to each other by underground water pipelines. This system is used to incorporate green energy sources. However, these green energy sources output fluctuate more than the output of the coal based heat producers. In order to control this fluctuating supply in the multi-source district heating network, a controller is needed that requires short term prediction of the heat demand of the consumers. In this thesis a prediction algorithm is made based on the thermal physical system of the consumers, for example the outdoor temperature of the houses and the flow of water through the radiator. As well as the social behaviour of the consumers, for example is the consumer willing to save energy or not. In order to test the prediction algorithm designed, numerical simulations are carried out with a suggested controller. This resulted in the conclusion that when accurate values for the thermal physical system and the social behaviour of the consumer are imported, a feasible prediction for the load demand of the consumers can be made.

Contents

Abstract							
\mathbf{Li}	st of	Figures	iv				
\mathbf{Li}	st of	Tables	iv				
In	trodu	iction	1				
1	Intr	oduction	1				
	1.1	Motivation	1				
	1.2	Literature Overview	2				
	1.3	Contribution	3				
	1.4	Organisation of the thesis	3				
2	Syst	em Model	4				
	2.1	Single pipe	4				
	2.2	Heat Exchanger	6				
	2.3	Producer	9				
	2.4	Consumer	9				
	2.5	Distribution Network Control System	12				
	2.6	Overall district heating network	13				
	2.7	Heating comfort regarding social behaviour	16				
	2.8	Incorporating the social behaviour	17				
	2.9	State Space Equations	19				
	2.10	Problem formulation	21				
3	Setu	p Simulations	22				
	3.1	Identification of the Unknowns	22				
	3.2	Numerical case study	24				
	3.3	Controlling the Flow	26				
	3.4	Overall Controller	30				
4	Sim	ulations	32				
	4.1	Experiment 1, steady state test	33				
	4.2	Experiment 2, influence of the heat coefficient loss	34				
	4.3	Experiment 3, influence of the social variables	35				

5	Conclusion and Further Research								
6	Bibliography	37							
7	Appendix	40							
	7.1 Appendix A - Extra results Experiment 2	40							
	7.2 Appendix B - Code Experiment 1	41							
	7.3 Appendix C - Code Experiment 2	49							
	7.4 Appendix D - Code Experiment 3	58							

List of Figures

1	Distribution network (Machado et al. (2020))	1
2	Heat exchangers of producers and consumers (Machado et al., 2021),(Scholten	
	et al., 2015),()	4
3	Single pipe	5
4	Heat exchanger	6
5	Heat exchanger zoomed in (van der Schaft and Jeltsema, 2020) \ldots	7
6	Heat network consumer	9
7	Room from above	1
8	Junctions and pipe (Machado et al., 2020) 1	2
9	Graphs of the DH network	3
10	Interpolating linear function	8
11	Example graphs	3
12	Distribution grid	4
13	Example grid	7
14	Communication graph	9
15	Experiment 1	4
16	Experiment 2, 10% increase of λ	5
17	Experiment 3	5
18	Heat conductivity rate 10% decrease $\ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots 4$	0
19	Heat conductivity rate 5% decrease $\ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots 4$	0
20	Heat conductivity rate 5% increase $\ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots \ldots 4$	1

List of Tables

1	Overall parameters
2	Parameters of single pipe
3	Parameters heat exchanger
4	Parameters heat network consumer
5	Parameters junction and pipe

Symbol	Description	Unit
q_{in}	Flow into the pipe	$\frac{m^3}{s}$
q_{out}	Flow out of the pipe	$\frac{m^3}{s}$
q	Flow in the pipe	$\frac{m^3}{s}$
T_{in}	Temperature of the water going into the pipe	C°
T_{out}	Temperature of the water going out of the pipe	C°
T	Temperature of the water in stored	C°
M	Mass	kg
c_w	Specific heat of water	$\frac{J}{kaC^{\circ}}$
$ ho_w$	The density of water	$\frac{kg}{m^3}$
q_{hs}	Flow in the hot stream	$\frac{m^3}{s}$
q_{cs}	Flow in the cold stream	$\frac{m^3}{s}$
T_{hs}	Temperature in the hot stream	C°
T_{cs}	Temperature in the cold stream	C°
V_{hs}	Volume hot stream side	m^3
V_{cs}	Volume cold stream side	m^3
T_{hs}^{in} or T_{hs}^{out}	Temperature of the hot stream going into or out of the heat exchanger	C°
T_{cs}^{in} or T_{cs}^{out}	Temperature of the cold stream going into or out of the heat exchanger	C°
q_{cs}^{in} or q_{cs}^{out}	Flow into or out of the radiator	$\frac{m^3}{s}$
T_{cs}^{in} or T_{cs}^{out}	Temperature into or out of the radiator	C°
T_{amb}	The ambient temperature	C°
T_{rad}	Temperature in the radiator	C°
T_{room}	Temperature in the room	C°
V_{rad}	Volume of water within the radiator	m^3
V_{room}	Volume of air within the room	m^3
$q_{pipe,i}$	Flow in the pipe i	$\frac{m^3}{s}$
$T_{pipe,i}$	Temperature in the pipe i	C°
$V_{pipe,i}$	Volume in the pipe i	m^3
$T_{node,i}$	Temperature in the node i	C°
V_{node}, i	Volume in the node i	m^3

 Table 1: Overall parameters

1 Introduction

In this chapter an introduction to multi-source district heating system is given. A focus is first given on the motivation on this research. In the next section, the relevant literature is given and knowledge gaps are marked. In the last two sections the goal of this research is given and the organisation of its chapters.

1.1 Motivation

In order to reduce green houses gases, district heating networks are expected to switch from their fossil power plants to multiple green energy sources (Machado et al., 2020, Lund et al., 2014, Saletti et al., 2020, Dominković et al., 2020). As a result, the distribution network changes from a tree-like structure to a more meshed topology (Lund et al., 2014, Wang et al., 2017). District heating is capable of including multiple sources of heat producers into its system, for example geothermal or solar-thermal, which can be seen in Figure 1. With the produced heat, the system is able to supply a town, neighborhood or city. The district heating network exists out of heat producers, consumers and a distribution network. The producers generate hot water, referred to as heat, which is distributed by a network of underground pipes towards the consumers. This type of network is schematically visualised by Figure 1. Once arrived at the consumers, heat exchangers are used to transfer the heat from the closed loop distribution network to the local network of each consumer.



Figure 1: Distribution network (Machado et al. (2020))

1.2 Literature Overview

In order to regulate all the flows through the multi-source network, a control system is needed (Machado et al., 2020). Preliminary research has been done in Machado et al. (2020, 2021) to developed a control system for the distribution network. This control system needs multiple information into its system, one being the short term prediction of the heat demand. Previous studies showed that this positively affects the performance of the control system (Ma et al., 2017). This is due to the ability to deal with the time delay, because the heat needs to travel through the long pipes of the distribution network (Dotzauer, 2002, Guelpa et al., 2019*b*)

Research into prediction models proved that outdoor temperature and the behaviour of building occupants have the greatest influence on the demand (Dotzauer, 2002). Furthermore Dotzauer (2002) stated that the information acquired by a simplified short term prediction model is sufficient for the control system to function. The reason to pick simplified short term prediction model, over an extended long term prediction model, even when the information produced is the same, is the smaller computational time for the simplified models. Moreover, the short term is the hourly prediction of demand, which is needed to tackle the time delay by the distribution network.

In order to predict the heat demand, one approach suggests that the thermal physical dynamics of the consumer network is needed as well as the thermal physical dynamics of the distribution network (Ma et al., 2017, Dotzauer, 2002, Bäumelt and Dostál, 2020, Nielsen and Madsen, 2006). To model the thermal physical dynamic model of the consumer, greybox modelling is used in Bäumelt and Dostál (2020), Nielsen and Madsen (2006) to attain the model structure by physical principles and estimating the parameters by experimental data. According to Bäumelt and Dostál (2020), grey-box modelling is effective due to a relatively low time-consumption during creation, compared to other methods and exact or near physics-based interpretation. Moreover, the overall system can be described using ordinary differential equations (Nielsen and Madsen, 2006).

Existing literature does not include social aspects, which have a great influence on the demand (Dotzauer, 2002). In (Heydarian et al., 2020), a general literature study is preformed looking at how the social behaviour values influence the demand of a consumer. Since this is a literature study it focuses on a lot of different factors of the behaviour. In (Feng et al., 2020), a human physical system framework existing of human behaviour, social norms and

physical systems is developed for electrical systems. Moreover, Feng et al. (2020) focuses on energy saving behaviour and identifies two major values that affects energy saving behaviour; egoistic values and bioshperic values. These are unknown values and need to be identified, because estimating the unknown parameters of the consumers is important to make a good prediction of their demand (Namazkhan et al., 2019, Nielsen and Madsen, 2006, Pinto, 2016).

1.3 Contribution

This thesis is aimed at combining the thermal physical part discussed in Machado et al. (2020, 2021) with the social aspects of the consumers discussed in Feng et al. (2020) and Namazkhan et al. (2019), into a short term prediction model of the heat demand for district heating. This contribution is tested by carrying out a case study with numerical simulations. In this testing a special focus is put on the unknown parameters of the consumers and their influence on the prediction of the load for the consumers.

1.4 Organisation of the thesis

This paper is organized as follows. In Chapter 2, the system behind the thermal physical dynamics of the network is explained and the thermal physical dynamics of the consumer are developed. Furthermore, the social behaviour is explained and a contribution is made by incorporating this behaviour into the thermal physical dynamics of the consumer. In Chapter 3, the identification of the unknown parameters of the consumer is performed and the case study explained. In addition, a controller for this case study is suggested. In Chapter 4, the results of the simulations are presented and finally Chapter 5 the thesis ends with a conclusion and an advise for future research.

Notation: An *n*-vector of ones is written as 1_n , whereas the identity matrix of size *n* is represented by \mathcal{I}_n . Any vector $x \in \mathbb{R}^n$, in which \mathbb{R} denotes a set of real numbers, is denoted by $\langle x \rangle$ as a diagonal matrix with elements *x* in its main diagonal. For any time-varying signal w, \bar{w} represents its steady-state value, if it exists.

2 System Model

In this chapter the modelling of the district heating network and all its parts is discussed. Moreover, the social behaviour of the consumers is introduced and incorporated into state space equations to make a set of equations for the whole system. At last in the problem formulation the goal for the suggested controller and the unknown parameters are discussed.

2.1 Single pipe

A closed, water-based district heating system is considered, with multiple heat producers and consumers interconnected through a common meshed distribution network as showed in Figure 1. The system would realistically include storage tanks. These tanks store the generated heat when the production is higher than the demand, and release the heat when the demand is higher than the production. However, for simplification matters the prediction model will not include storage tanks. The consumers and producers are modelled with basic hydraulic devices as valves, pipes and pumps, as in (Machado et al., 2020). The thermal physical dynamics of the producers and consumers are modeled with a simple heat exchanger between the distribution network and network in the building of the producer or house of the consumer according to (Grassi et al., 2021). This is shown in Figure 2.



Figure 2: Heat exchangers of producers and consumers (Machado et al., 2021),(Scholten et al., 2015),()

In order to model the heat exchanger, first the model of one single pipe is created, as shown in Figure 3 with its parameters explained in Table 2. This is modeled with a basic heat equation (1), according to (Nielsen and Madsen, 2006). This equation describes the heat exchanger as a regular pipe, with the heat going into the pipe as power in, and the heat going out of the pipe as power out. The heat stored, or rather exchanged to another network, is the difference between the heat going in to the pipe and the heat going out of the pipe.



Figure 3: Single pipe

$$\frac{d(\text{Heat stored})}{dt} = \Sigma \text{Power in} - \Sigma \text{Power out}$$
(1)

The stored heat per mass unit can be described by $c_w T M$, where c_w is the specific heat of water, T is the temperature of water and M the mass of water. When substituting this in equation (1), equation (2) is determined.

$$\frac{d}{dt}(Mc_wT) = \rho_w q_{in} c_w T_{in} - \rho_w q_{out} c_w T_{out}$$
⁽²⁾

Table 2:	Parameters	of	single	pipe
----------	------------	----	--------	------

Symbo	l Description	Unit
q_{in}	Flow into the pipe	$\frac{m^3}{s}$
q_{out}	Flow out of the pipe	$\frac{m^3}{s}$
q	Flow in the pipe	$\frac{m^3}{s}$
T_{in}	Temperature of the water going into the pipe	C°
T_{out}	Temperature of the water going out of the pipe	C°
T	Temperature of the water in stored	C°
M	Mass	kg
c_w	Specific heat of water	$\frac{J}{kaC^{\circ}}$
$ ho_w$	The density of water	$\frac{kg}{m^3}$

2.2 Heat Exchanger

In order to model the heat exchanger between the distribution network and the consumer network, the heat conduction between two pipes should be modeled. This is schematically visualised in Figure 4 and with is parameters explained in Table 3. The dashed line represents the wall where heat conduction takes place between the two flows of the distribution network and the consumer network. q_{hs} and T_{hs} , represent the flow of the hot stream and the temperature of the hot stream accordingly. At the consumer side, q_{cs} and T_{cs} represent the flow of the cold stream and the temperature of the cold stream accordingly. The distribution network is the side with the hot water and the consumer network the side with the cold water. In the heat exchanger at the distribution side, a temperature of water flows in with T_{hs}^{in} and a temperature of water flows out as T_{hs}^{out} , with $T_{hs}^{in} > T_{hs}^{out}$. At the consumer side a temperature of water flows in with T_{cs}^{in} and a temperature of water flows out as T_{cs}^{out} , with $T_{cs}^{out} > T_{cs}^{in}$.



Figure 4: Heat exchanger

Symbol	Description	Unit
q_{hs}	Flow in the hot stream	$\frac{m^3}{s}$
q_{cs}	Flow in the cold stream	$\frac{m^3}{s}$
T_{hs}	Temperature in the hot stream	C°
T_{cs}	Temperature in the cold stream	C°
V_{hs}	Volume hot stream side	m^3
V_{cs}	Volume cold stream side	m^3
T_{hs}^{in} or T_{hs}^{out}	Temperature of the hot stream going into or out of the heat exchanger	C°
T_{cs}^{in} or T_{cs}^{out}	Temperature of the cold stream going into or out of the heat exchanger	C°

Table 3: Parameters heat exchange	ger
-----------------------------------	-----

In Figure 5, a heat exchanger is visualized up close with its parameters explained in Table 3. In this figure the cold stream is regarded as the consumer network, which gains heat when flowing through the heat exchanger. The hot stream is regarded as the distribution network, which loses heat when flowing through the heat exchanger. Important to notice is that the volume of at both sides of the heat exchanger remains the same level, so that: $V_{hs}^{in} = V_{hs}^{out}$, $V_{cs}^{in} = V_{cs}^{out}$. Out of which can be concluded that there is a volume for the cold stream V_{cs} and a volume for the hot stream V_{hs} .



Figure 5: Heat exchanger zoomed in (van der Schaft and Jeltsema, 2020)

Modelling the equation for the heat exchanger, a start is made with the conservation of mass. Setting up two equation, one for the cold stream and one for the hot stream. Represented by equation (3) and equation (4) accordingly.

$$\frac{dM_{cs}}{dt} = \rho_w (q_{cs}^{in} - q_{cs}^{out}) \tag{3}$$

$$\frac{dM_{hs}}{dt} = \rho_w (q_{hs}^{in} - q_{hs}^{out}) \tag{4}$$

M is the mass of water, which can be calculated by multiplying the density of water ρ_w with the volume *V*. Since the volume is constant, as well as the density of water, due to the flow being incompressible, it can stated that: $q_{hs}^{in} = q_{hs}^{out} = q_{hs} q_{cs}^{in} = q_{cs}^{out} = q_{cs} q$ is calculated by the density ρ multiplied with the the flow rate q the following equation can be obtained, in which ρ is equalised out.

$$q_{hs} = q_{hs}^{out} = q_{hs}^{in} \tag{5}$$

$$q_{cs} = q_{cs}^{out} = q_{cs}^{in} \tag{6}$$

Using these principles, the change in mass inside of the heat exchanger can be stated by the flow q_{hs} multiplied by the density of water ρ_w . The change of mass consists out of the density of water ρ_w and the volume V. For the hot stream and the cold stream the following equations are provided:

$$\rho_w V_{hs} = \rho_w q_{hs} \tag{7}$$

$$\rho_w V_{cs} = \rho_w q_{cs} \tag{8}$$

Furthermore, introducing the heat of the flow by stating the conservation of energy and setting up two equations. One for the cold stream and one for the hot stream, represented by equation (9) and equation (10) accordingly.

$$\frac{d\rho_w \, c_w \, M \, T_{cs}}{dt} = \rho_w \, c_w q_{cs}^{in} \, T_{cs}^{in} - \rho_w \, c_w \, q_{cs}^{out} T_{cs}^{out} - \frac{U A_h}{\rho_w \, c_w} (T_{hs} - T_{cs}) \tag{9}$$

$$\frac{d\rho_w \, c_w \, M \, T_{hs}}{dt} = \rho_w \, c_w \, q_{hs}^{in} T_{hs}^{in} - \rho_w \, c_w \, q_{hs}^{out} T_{hs}^{out} - \frac{U A_h}{\rho_w \, c_w} (T_{cs} - T_{hs}) \tag{10}$$

In these equation c_w is the specific heat of water and T the temperature. Multiplying these variables results in the heat gain or loss. The fraction and the difference in temperature models the heat conduction through the conduction wall. ρ_w is the density of water, this can be crossed out since it is in every part of the equation. In the fraction, A_h is the area were conduction takes place and lastly the heat conduction rate U. This equation can be simplified by stating that all of the parameters of the fraction are constant. Therefore the constant λ is introduced. This simplification results in the following equations for the cold and hot stream:

$$\frac{dc_w T_{cs}}{dt} = c_w (T_{cs}^{in} - T_{cs}^{out}) - \lambda (T_{hs} - T_{cs})$$

$$\tag{11}$$

$$\frac{d c_w T_{hs}}{dt} = c_w (T_{hs}^{in} - T_{hs}^{out}) - \lambda (T_{cs} - T_{hs})$$

$$\tag{12}$$

Substitution the equations of the flow equation (7) and equation (8) into the heat equations of the cold stream and hot stream formulates the following two equations for the cold and hot stream:

$$\rho_w c_w V_{cs} \dot{T}_{cs} = q_{cs} (T_{cs}^{in} - T_{cs}^{out}) - \lambda (T_{hs} - T_{cs})$$

$$\tag{13}$$

$$\rho_w \, c_w \, V_{hs} \dot{T}_{hs} = q_{hs} (T_{hs}^{in} - T_{hs}^{out}) - \lambda (T_{cs} - T_{hs}) \tag{14}$$

2.3 Producer

For the control system of the producer, equation (15) is taken from Scholten et al. (2015):

$$P = \frac{UA_h}{c_w \rho_w} (T_{cs} - T_{hs}) \tag{15}$$

In this equation P represents the heat transfer rate, which is a manipulable power injection from the hot stream towards the cold stream. c_w the specific heat of water, ρ_w as the density of water and U the heat conduction. The temperatures of the producers or consumers are represented here by T_e and the temperature of the network by T. Substituting the power into the conduction for the heat exchanger x, gives equation (16).

$$\rho_w \, c_w \, V_x \dot{T}_x = q_x (T_x^{in} - T_x^{out}) + P_x \tag{16}$$

2.4 Consumer

For the consumer, a contribution is made by modelling a network with a room, radiator and ambient temperature. This is schematically visualised in Figure 6, according to Grassi et al. (2021), with its parameters explained in Table 4. For the variables presented in Figure 6, T_{amb} is the ambient temperature and is viewed as a constant in the system. This due to the control system having a faster response compared to a change in the ambient temperature. Moreover, the radiator is modeled by the temperature of the radiator T_{rad} and the volume of the radiator V_{rad} . The flow within this network is constant, and since no heat loss is set in the pipe network, the temperature is the same: $T_{cs}^{out} = T_{cs,rad}^{in}$ and $T_{cs}^{in} = T_{cs,rad}^{out}$.



Figure 6: Heat network consumer

It should be noticed that, for simplifying the mathematical calculations, the house is defined as one room, with the temperature of the room T_{room} and the volume of the radiator V_{room} . Therefore, there is not heat transfer within the room. Besides this simplification, both the structure of the model as well as the qualitative properties for consumers are the same.

Symbol	Description	Unit
q_{cs}^{in} or q_{cs}^{out}	Flow into or out of the radiator	$\frac{m^3}{s}$
T_{cs}^{in} or T_{cs}^{out}	Temperature into or out of the radiator	C°
T_{amb}	The ambient temperature	C°
T_{rad}	Temperature in the radiator	C°
T_{room}	Temperature in the room	C°
V_{rad}	Volume of water within the radiator	m
V_{room}	Volume of air within the room	m^3

 Table 4: Parameters heat network consumer

In order to write the equation for the radiator, the conservation of mass and heat are used. Note however a crucial difference compared to the heat exhanger; the room is filled with air instead of a fluid. This changes the density and the specific heat, since this is differs for air and water. Therefore, the density and the specific heat can not be crossed out:

$$\rho_w c_w V_{rad} \dot{T}_{rad} = \rho_w c_w q_{cs} (T_{cs}^{out} - T_{cs}^{in}) - \lambda_2 (T_{room} - T_{rad})$$
(17)

In equation (17), ρ_w is the density of water and c_w the specific heat of water. Important to state is that λ_2 is here a different constant since the area A and the heat conduction U, are different in this radiator, in comparison to the heat exchanger. Regarding the room, it is important to take into account that the room actually has four walls. This is visualised in Figure 7, with its parameters explained in Table 4.



Figure 7: Room from above

In order to model the equation for the room, first the heat gain by the radiator is taken from equation (17). Then, the heat loss from the conduction with the ambient temperature has to be added. Combining these two additions, the heat control system for the room has been formed:

$$\rho_a c_a V_{room} \dot{T}_{room} = \lambda_2 (T_{rad} - T_{room}) - \lambda_3 (T_{amb} - T_{room})$$
(18)

In this equation, ρ_a is the density of air and c_a the specific heat of air. λ_3 is different in comparison with λ_2 , due to the different areas of the the four walls. To sum up all equations of the heat exchanger streams, the radiator and the room the following four equations are stated:

$$\rho_w \, c_w \, V_{cs} \dot{T}_{cs} = \rho_w \, c_w \, q_{cs} (T_{cs}^{in} - T_{cs}^{out}) - \lambda_1 (T_{hs} - T_{cs}) \tag{19}$$

$$\rho_w \, c_w \, V_{hs} \dot{T}_{hs} = \rho_w \, c_w \, q_{hs} (T_{hs}^{in} - T_{hs}^{out}) - \lambda_1 (T_{cs} - T_{hs}) \tag{20}$$

$$\rho_w c_w V_{rad} \dot{T}_{rad} = \rho_w c_w q_{cs} (T_{cs}^{out} - T_{cs}^{in}) - \lambda_2 (T_{room} - T_{rad})$$
(21)

$$\rho_a c_a V_{room} \dot{T}_{room} = \lambda_2 (T_{rad} - T_{room}) - \lambda_3 (T_{amb} - T_{room})$$
(22)

It can be concluded from these equations that the heat a consumer uses from the distribution network is stated by $\lambda_1(T_{cs} - T_{hs})$, which is from the distribution network side it's heat loss. Therefore in steady state the following equation (23) is stated:

$$\lambda_1(T_{cs} - T_{hs}) = \lambda_3(T_{amb} - T_{room}) \tag{23}$$

2.5 Distribution Network Control System

The distribution network exists out of nodes and pipes. In Figure 8, a node is shown at the left and a pipe at the right. The water flows in the pipe from the source node i, to the target node j. In the node of Figure 8, the variables volume $V_{node,i}$ and a temperature $T_{node,i}$ are present. In the pipe, a flow $q_{pipe,i}$, a volume $V_{pipe,i}$ and a temperature $T_{pipe,i}$ are present. Due to the water flowing from i to j, the parameters are taken from the source node.



Figure 8: Junctions and pipe (Machado et al., 2020)

Symbol	Description	Unit
$q_{pipe,i}$	Flow in the pipe i	$\frac{m^3}{s}$
$T_{pipe,i}$	Temperature in the pipe i	C°
$V_{pipe,i}$	Volume in the pipe i	m^3
$T_{node,i}$	Temperature in the node i	C°
V_{node}, i	Volume in the node i	m^3

 Table 5: Parameters junction and pipe

It is assumed that in pipe *i* the variation of temperature is uniform. Therefore, the temperature at target node *j* is equal to the temperature in the middle of the pipe: $T_{pipe,j} = T_{pipe,j}$. Using the same mass and heat balances as for the consumer, results in equation (24).

$$\rho_w c_w \dot{T}_{pipe,i} = \rho_w c_w q_{pipe,i} (T_{node,i} - T_{pipe,i})$$
(24)

In order to model the control system of the distribution network, the following equation (25) can be taken from (Machado et al., 2020). The only difference is that the ρ and c_w are not canceled out. This equation makes it possible to calculate power balance (the volume and temperature) at every node. In this equation, \exists_k is the set of edges whose streams target to node $k \in N$. According to (Machado et al., 2020): $V_{node,i} = 0$, due to the very small volume at junctions in comparison to pipelines. As a consequence, the right side of the equation can be set to zero. This is represented in equation (26).

$$\rho_w c_w V_{node,i} \dot{T}_{node,i} = \sum_{s \in \mathbf{J}_k} \rho_w c_w q_{node,j} T_{node,j} - \left(\sum_{s \in \mathbf{J}_k} \rho_w c_w q_{node,i}\right) T_{node,i}$$
(25)

$$0 = \sum_{s \in \mathbf{J}_k} \rho_w c_w q_{node,j} T_{node,j} - \left(\sum_{s \in \mathbf{J}_k} \rho_w c_w q_{node,i}\right) T_{node,i}$$
(26)

2.6 Overall district heating network

To model the overall district heating network, graph theory is used. In order to explain this an example is made. In this example, N is set as nodes and e as edges between the nodes, with e_1 and e_2 as producers, e_3 and e_4 as consumers and the other edges as distribution pipes. The total graph exists out of all nodes connected by edges. The modelling is preformed for Figure 9a, a graph with 8 nodes, 10 edges and the flows in these edges indicated by the purple lines. For simplicity the graph remains small, two producers and two consumers and pipes and valves are not included. For the flows, it is important to node that no flow dynamics are considered. This means that in the system no pressure losses are considered. However, to be able to describe all the flow directions, mass conservation dictates that some flows must be linearly dependent with respect to other flows.



Figure 9: Graphs of the DH network

The first step to identify the minimum set of independent flows is to identify the set of chords in the graph (Machado et al., 2020). This set of chords exists out of edges that needs to be removed in order to end up with a reduced graph. This reduced graph contains the same nodes as the original, but does not have any loops. The reduced graph, without the chords, is called a spanning tree of the original graph. This spanning tree may not be unique (Hillier and Lieberman, 2006), and therefore different sets of chords can exist.

To create the spanning tree the chords are selected as followed: all the consumer pipes are chords and all but one producer are chords. This is caused by the fact that all flows in the chords are viewed as control inputs. As result all the consumer flows have to be control inputs. This leads to one of the producer edges not being a chord. Moreover, it is important to note that it is assumed for simplification, that their are no loops in the cold or hot layer of the graph. Therefore, e3, e4 and e2 are selected as chords, with each flow viewed as a control input. This way the spanning tree results in Figure 9b. When adding back one of the named chords, a loop arises immediately. These loops are called fundamental loops, L.

To write this fundamental loops in a matrix, use is made of the following, e_j^+ and e_j^- . e_j^+ is given when the flow in the edge is in agreement with the orientation of the flow of the added back in chord and e_j^- is given when the flow in the edge is in disagreement with the orientation of the flow of the added back in chord. Having three chords gives three fundamental loops, stated as followed:

$$\mathbb{L}_{e,2} = \{e_2^+, e_7^+, e_6^-, e_5^-, e_1^-, e_8^-, e_9^-, e_{10}^+\}$$
$$\mathbb{L}_{e,3} = \{e_3^+, e_8^+, e_1^+, e_5^+\}$$
$$\mathbb{L}_{e,4} = \{e_4^+, e_9^+, e_8^+, e_1^+, e_3^+, e_6^+\}$$

The fundamental loop matrix, denoted by $F \in \mathbb{R}^{n_{ch} \times n_e}$ is identified by the fundamental loops. In this matrix, n_{ch} is the number of chords and n_e is the number of edges in the original graph. To create this matrix F, denoted by Machado et al. (2020) the identification of the fundamental loop matrix F_{ij} is created.

$$F_{ij} = \begin{cases} 1 & \text{if edge } e_j \text{ is in } \mathbb{L}_{e,i} \text{ and orientations agree,} \\ -1 & \text{if edge } e_j \text{ is in } \mathbb{L}_{e,i} \text{ and orientations disagree,} \\ 0 & \text{if edge } j \text{ is not in } \mathbb{L}_{e,i}. \end{cases}$$

The use of this fundamental loop matrix F is to identify the flow rate of all the edges in the graph with $q_{ch} \in \mathbb{R}^{n_{ch}}$ (Machado et al., 2020). The conservation of flows from the Kirchhoff's laws resulting in equation (27). In this equation, q_e are all the flows through all the edges $q_e = \{q_{e,1}, q_{e,2}, \dots, q_{e,n_e}\} \in \mathbb{R}^{n_{ch}}, F^T$ is the fundamental loop matrix and q_{ch} is the flow through the chords. $q_{e,n_e} = F^{\top}q_{e,n_e}$ (27)

$$q_e = F^{\top} q_{ch} \tag{27}$$

To sum up, the following node-edge incidence matrix is introduced with $\mathcal{B}_{0,ij}$ as the following and the incidence matrix concluding out of it.

$$\mathcal{B}_{0,ij} = \begin{cases} 1 & \text{if edge } e_j \text{ targets node } i, \\ -1 & \text{if edge } e_j \text{ originates from node } i, \\ 0 & \text{otherwise.} \end{cases}$$

		e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}
	N_1	1	0	0	0	-1	0	0	0	0	0
	N_2	0	0	-1	0	1	-1	0	0	0	0
	N_3	0	0	0	-1	0	1	1	0	0	0
в. —	N_4	0	1	0	0	0	0	-1	0	0	0
$\mathcal{D}_0 =$	N_5	-1	0	0	0	0	0	0	1	0	0
	N_6	0	0	1	0	0	0	0	-1	1	0
	N_7	0	0	0	1	0	0	0	0	-1	-1
	N_8	0	-1	0	0	0	0	0	0	0	1

To be able to link the thermal interaction to this graph network, a correction must be made regarding the flows, since the temperatures are related to the flow direction in the edges. In the graph in Figure 9a, the flows of e_6 and e_9 can differ when one of the producers produces more heat. In order to correct the flows in the graph system when this situation occurs, the matrix \mathcal{B}_0 is corrected by creating the matrix \mathcal{B} .

$$\mathcal{B} = \mathcal{B}_0 \cdot diag\left(sign\left(q_e^{\top}\right)\right) \tag{28}$$

When combining this graph theory with the equations 31, 21 from section 2.5, a set of all the differential equations can be written in matrix form, according to (Machado et al., 2020).

$$\begin{bmatrix} diag(V_e) & 0\\ 0 & diag(V_N) \end{bmatrix} \begin{bmatrix} \dot{T}_e\\ \dot{T}_N \end{bmatrix} = \begin{bmatrix} -diag(|q_e|) & diag(|q_e|)\mathcal{S}^\top\\ \mathcal{T}diag(|q_e|) & -diag(\mathcal{T}|q_e|) \end{bmatrix} \begin{bmatrix} T_e\\ T_N \end{bmatrix} + B_{pr}P_{pr} - B_cP_c$$

In this matrix form, B_{pr} and B_c are properly sized coefficient matrices. Moreover, \mathcal{T} represents all edges of the target node and \mathcal{S} represents all the edges of the origin of the flow. They can be identified by the following equation:

$$\mathcal{T} = \frac{1}{2} \left(\mathcal{B} + |\mathcal{B}| \right), \ \ \mathcal{S} = \frac{1}{2} \left| \mathcal{B} - |\mathcal{B}| \right|$$

2.7 Heating comfort regarding social behaviour

In the work of Feng et al. (2020), an approach is provided to write the social behaviour of consumers for power systems. This approach is used in this thesis to write the social behaviour of consumers for district heating. In order to use the method provided Feng et al. (2020), it is first analysed. Therefore, first a degree of fulfillment of the desired load is stated. This variable lies within $0 \le z_s \le 1$. With z_s being closer to 0 when the consumer is more climate consciously. The value of z_s can be multiplied with the load of the consumer. For example a consumer has a desired load of 100. However, his social behaviour affects him to consume less heat because this creates pollution. Therefore, his z_s is 0.7 giving his total load as 100 * 0.7 = 70. The change of z_s can be calculated with the following equation:

$$\dot{z}_s = a_i (p_s - z_s - h_s s_s) \tag{29}$$

In this equation, s_s is the financial incentives, for example a lower energy bill for the consumer when he uses less energy. h_s is the degree of influence of the financial incentives. This can be indicated by egoistic values, stating that a more egoistic consumer has a higher value of h_s , where $0 \le h_s \le 1$. This concludes that a more egoistic person is more affected by financial incentives. The values of s_s can variate between $0 \le s_s \le \frac{p_s}{h_s}$. The limit for s_s prevents the financial incentives at steady state to be unrealistically high. Lastly the whole right side is multiplied by a_i . This indicates the time constant of the behaviour. In this equation, p_i are the personal norms of the consumer. These norms can have values between $0 \le p_s \le 1$. The change of these personal norms are calculated with the use of egoistic and biospheric values, according equation (30). In this thesis the approach from Namazkhan et al. (2019) is used for acquiring the p_s^{ego} and p_s^{bio} with the use of questionnaires and surveys. These values are multiplied by c_i and d_i for the egoistic and biospheric values respectively. These values are take such that $c_i \ge 0$ and $d_i \ge 0$ with $c_i + d_i > 0$. This indicates that a very large value for c_i makes the consumer strongly dependent on the egoistic values, and vice versa.

$$\dot{p}_{i} = (c_{i}p_{i}^{ego} + d_{i}p_{i}^{bio})/(c_{i} + d_{i})$$
(30)

As stated in (Feng et al., 2020), the social behaviour z_s influences the load desired by the consumer. In this thesis the desired load of the consumer has a certain range, in which the consumer still thinks that the temperature is comfortable. This range is set as the following: $T_{room}^{min} \leq T_{room}^{ideal} \leq T_{room}^{max}$. In the center the most ideal temperature is represented. The T_{room}^{min} and T_{room}^{max} represents the border temperature in which the consumer is still comfortable. The border temperatures, min and max will be received by surveys or questionnaires (Namazkhan et al., 2019).

2.8 Incorporating the social behaviour

In order to be able to incorporate the social behaviour into the thermal physical system, some assumptions and simplifications need to be made. First of all the thermal physical system of the consumer needs to be presented as a producer. This will simplify the incorporation of the social behaviour, since equation (31) of the cold stream can be taken.

$$\rho_w \, c_w \, V_{hs} \dot{T}_{hs} = \rho_w \, c_w \, q_{hs} (T_{hs}^{in} - T_{hs}^{out}) - \lambda_1 (T_{cs} - T_{hs}) \tag{31}$$

In this equation, the change in temperature is the flow multiplied with the temperature difference between the incoming water and the outgoing water, minus the heat conduction. Secondly, this section elaborates on an assumption is made regarding the controller. Moreover, a range is stated in which the consumer still finds the temperature in the room as comfortable. Afterwards, the social behaviour can be incorporated

In Sleptsov et al. (2021), simulations of the control system are performed with a time step of 1 minute. As a result the control system runs every one minute. Since this is very fast the control signal in Sleptsov et al. (2021) is maintained constant. In Guelpa et al. (2019*a*) a time step of 10 minutes is used. When the proportional integral controller suggested in the system model has such a small time step, it is sufficiently fast to assume that some information inputs to the controller are constant. For the suggested design of the proportional integral controller in this thesis, two information inputs can be set constant. The first information input is the flow of the cold stream in the system of the consumer, q_{cs} . q_{cs} has been set as a control input in the system model. Since the proportional integral controller is sufficiently fast, the following can be stated regarding the flow of the cold stream:

$$q_{cs} = q_{new,cs} \frac{d}{dt} \tag{32}$$

The second information input is the temperature of the room. The temperature of the room is set the same as the temperature of the room desired, $T_{room} = T_{room}^{desired}$, since the proportional integral controller is sufficiently fast. Implementing this to the steady state equation gives the following:

$$\lambda_1(T_{cs} - T_{hs}) = \lambda_3(T_{amb} - T_{room}^{desired})$$
(33)

Substituting this into equation (31) gives the following equation, which combines all the thermal physical dynamics of the consumer:

$$\rho_w c_w V_{cs} \dot{T}_{cs} = \rho_w c_w q_{cs} (T_{cs}^{in} - T_{cs}^{out}) - \lambda_3 (T_{amb} - T_{room}^{desired})$$
(34)

In section 2.7 it is stated that the social behaviour influences the range the consumer finds temperature the room comfortable. This range was defined as the following: $T_{room}^{min} \leq T_{room}^{ideal} \leq T_{room}^{max}$. The social behaviour z_s can be defined according to section 2.7 in the following equation, with $0 \leq z_s \leq 1$:

$$z_s = \begin{cases} 1 & \text{consumer max comfort} \\ 0 & \text{consumer min comfort} \end{cases}$$
(35)

In order to link the social behaviour to the range of comfortable temperature of the consumer, an interpolating linear function is used. This method is used to define a straight line between the two points. In Figure 10, an example of this method is shown, two vectors a and b have a straight line between them with the red x as the centre of this line. x can be found by the following equation: x = tb + (1 - t)a with t as $0 \le t \le 1$.



Figure 10: Interpolating linear function

Using equation (36) for combining of the comfortable temperature of the consumer and the social behaviour. When substituting $z_s = 0$, implying that the consumer desires minimum comfort, the result will be T_{room}^{min} . Furthermore, when $z_s = 1$, implying that the consumer desires maximum comfort the result will be T_{room}^{max} .

$$T_{room}^{desired} = z_s T_{room}^{max} + (1 - z_s) T_{room}^{min}$$
(36)

Substituting this equation into equation (31) and simplifying it further gives the following equation of the thermal physical dynamics of the consumer with the social behaviour included:

$$\rho_w c_w V_{cs} \dot{T}_{cs} = \rho_w c_w q_{cs} (T_{cs}^{in} - T_{cs}^{out}) - \lambda_3 (T_{amb} - T_{room}^{min}) + \lambda_3 z_s (T_{room}^{min} - T_{room}^{max})$$
(37)

2.9 State Space Equations

In order to model the equation of the combined model of the district heating network, a state space equation is used, combing two ordinary differential equations. The state space description is defined in the standard form, which has been used in (Bäumelt and Dostál, 2020, Nielsen and Madsen, 2006, Kim et al., 2019).

$$\dot{X} = Ax(t) + Bu(t) \tag{38}$$

$$Y = Cu(t) + Du(t) \tag{39}$$

equation (38) represents a continuous time system and equation (39) a district time observation. Y is a vector of the measurement input. According to Bäumelt and Dostál (2020), the longer the time step for Y, the more complex the equation becomes. The time-varying parameter used in this state space equation is the outdoor temperature. Although multiple other parameters are used in Bäumelt and Dostál (2020) and Nielsen and Madsen (2006), this research only accounts the outdoor temperature since Dotzauer (2002) stated that this, together with the social behaviour, will give an accurate prediction.

In order to state the total open loop system of the social system and the thermal physical system in these state space equations, first all the important equations are stated.

$$\rho_w c_w V_{cs} \dot{T}_{cs} = \rho_w c_w q_{cs} (T_{cs}^{in} - T_{cs}^{out}) - \lambda_3 (T_{amb} - T_{room}^{min}) + \lambda_3 z_s (T_{room}^{min} - T_{room}^{max}) \quad (\text{Consumer})$$

$$\rho_w c_w V_x \dot{T}_x = q_x (T_x^{in} - T_x^{out}) - P_x \qquad (Producer)$$

$$\rho_w c_w \dot{T}_{pipe,i} = \rho_w c_w q_{pipe,i} (T_{node,i} - T_{pipe,i})$$
(Pipe)

$$0 = \sum_{s \in \mathbf{J}_k} \rho_w c_w q_{node,j} T_{node,j} - \left(\sum_{s \in \mathbf{J}_k} \rho_w c_w q_{node,i}\right) T_{node,i}$$
(Node)

$$\dot{z}_s = a_i(p_s - z - s - h_s s_s)$$
 (Social Component)

$$\dot{p}_s = c_i (p_s^{ego} - p_s) + d_i (p_s^{bio} - p_s)$$
(Personal Norms)

Important to state is that the social component is different for every consumer. Moreover, it is stated in Feng et al. (2020) that the social behaviour of consumer is influenced by other

consumers. In this research, influence by consumers on consumers is disregarded due to simplification of the system. In order to incorporate these equations into state space equations, sets of different parameters are made. Stating that all the temperatures are represented by $T = (T_{producer}, T_{consumer}, T_{pipes})$, where $T_{consumer}$ includes the temperature of the radiator, cold stream and ambient. In addition \dot{T} is the overall change in temperature.

All the flows in the system are included in the vector q and the prior discussed three lambda's are included in the set λ . The volumes of the model are represented by the matrix V_x . The producer is included by the power it produces from equation (17) and d_{amb} is an appropriate constant vector that codifies outdoor temperature. The matrices, A and B and the vector d will depend on the topology of the system. The social equations are taken from Feng et al. (2020). Combing these terms into one state space equation gives the following tree equations:

$$V_x \dot{X} = A(q,\lambda)(X) + B P_p + D(z) T_{room}^{desired} + d_{amb}$$

$$\tag{40}$$

$$\dot{z} = a_i(p_s - z_s - h_s s_s) \tag{41}$$

$$\dot{p} = c_i (p_i^{ego} - p_i) + d_i (p_i^{bio} - p_i)$$
(42)

2.10 Problem formulation

This thesis is concerned with the main objective of combining the thermal physical dynamics with the behaviour of the consumers to a prediction of their short term heat demand. To achieve this, the load of the consumers should get regulated to the point where the temperature in the room is equal to the desired temperature of the consumer:

$$\lim_{t \to \infty} T_{room} = T_{room}^{desired}$$

This desired temperature can be regulated through the flows of the consumers. Therefore, the flow is designed such that the consumers achieve their desired temperature. In section 2.8, the social behaviour of the consumer is incorporated into the the thermal physical dynamics, which is already a contribution to the existing knowledge.

In order to test this result a numerical case study is made. To run the simulation of the case study, a controller needs to be suggested with the goal to get the consumers the temperatures they desire. Moreover, all the unknowns from the state space equations need to be identified. This is stated as a secondary objective. The main and secondary objective both provide a contribution to heat demand prediction.

The case study exists of two producers and two consumers. The number of producers and consumer system is small and therefore simplifies the simulations. However, the same approach can be used for larger numbers of producers and consumers. In the next chapter, Chapter 3, the case study will be explained.

3 Setup Simulations

In this chapter the setup of the simulations for the case study is discussed. In the first subsection the unknown parameters are identified. The second subsection explains the taken distribution grid for the simulations and all it's thermal dynamics. In the following subsections the controller used in the simulations is explained.

3.1 Identification of the Unknowns

In this thesis multiple variables are used for modelling the thermal physical system and the social behaviour of the consumer. In order to let the model work for practical systems and make a case study to validate this, all these variables which are currently unknown must be identified. It is however important to note that the control system will run without knowl-edge of these unknowns. However, its prediction will not be that precise.

Regarding state space equations, the thermal physical system has the following unknowns stated: λ_3 and the desired temperature. However, this is affected by the social behaviour as discussed in section 2.7. λ_3 is the heat conduction between the temperature of the room T_{room} and the outside temperature T_{amb} . This value can be identified according to the study done in Nielsen and Madsen (2006), which identifies the heat dynamics of buildings using stochastic differential equations.

The steps taken in Nielsen and Madsen (2006), can be explained using an example, with the state variable $\dot{X} = ax$. In this example, *a* is the unknown that needs to be identified. This is done by looking at the measurements taken. In this thesis these measurements are preformed by temperature gauges and flow gauges, measuring the ambient, room, radiator, nodes and edges. In order to identify the *a* Nielsen and Madsen (2006) puts all his measurements in a graph, visualised by Figure 11a. Then, to identify *a*, a line is drawn which connects all the measurement points as accurately as possible, as can be seen in Figure 11b. By using this approach on every consumer, λ_3 can be identified for every house of the consumer.

For this simulation, a representative value from Pinto (2016) is taken. In this paper Pinto did a research to typical buildings in Groningen and their conductivity. This study stated that most of the buildings in Groningen are high rise buildings. Therefore, it is assumed in this thesis that a high rise building has two walls which face the outside, with each wall having a width of 5 meters and a height of 3 meters. This makes the total surface area of



Figure 11: Example graphs

the wall facing the outside 30 m^2 . In his paper, Pinto (2016) states values for the different materials and their conductivity in the exterior wall. This value is the heat transfer rate U in W/m^2K . To combine all these materials into one value for the whole wall, the following sum is made: $(\frac{1}{8.8} + \frac{1}{6.7} + \frac{1}{18.3})^{-1} = 3.14W/m^2K$. This value is then used to calculate the conductivity of the wall by multiplying it with the total area of 30 m^2 . This gives: $3.14W/m^2K * 30m^2 = 94, 2W/K$, which is the value for λ_3 .

Regarding state space equations, the social behaviour has the following unknowns p_s^{ego} and p_s^{bio} . To identify these unknowns the work of Namazkhan et al. (2019) is used. In this study it is identified, that the values of social behaviour which affect the settings of the room temperature. By using this study, the social behaviour of every consumer can be identified using surveys and questionnaires. On a scale of -1 till 7, Namazkhan et al. (2019) found a median value for p_s^{ego} and p_s^{bio} , 1.94 and 5.17 respectively. In this thesis a scale of 0 till 1 is used. Calculating the values for the scale of this thesis gives 0.37 and 0.77 respectively.

3.2 Numerical case study

In order to test the system model suggested in this thesis, a numerical simulation is carried out. The simulation exists of a district heating system which contains two heat producers $(n_{pr} = 2)$ and two consumers $(n_c = 2)$, that are interconnected by a meshed distribution grid. The distribution grid does not contain any pumps or valves to simplify the system. Figure 12 illustrates the grid for the numerical simulation.



Figure 12: Distribution grid

In Figure 12, the blue lines indicate the cold stream pipes and the red lines indicate the hot stream pipes. Almost all of the directions of the flows can be identified, since the heated water flows from the producers to the consumers. When the heated water is used by the consumer, the cold water flows back to the producer. However, the flow at q_{d2} and q_{d4} can not directly be identified. These flows can stream both ways, depending on the flows in the producers and the consumers. When producer two produces more heat than producer one and both consumers desire an equal amount of heat, the flows in q_{d2} are positive since it flows from note c to point b. For this example, the flow of q_{d4} is negative, from node b' to node c', since producers two needs a faster stream of cold water to maintain its higher production.

According to the system model q_{ch} is set as independent variables $q_{ch} = [q_{c1}, q_{c2}, p_{pr2}] = [q_{e3}, q_{e2}, q_{e4}]$. Hence, $q_{pr1} = q_{e1}$ is linearly dependent on them. Therefore, the following equation can be set:

$$q_{pr1} = q_{c1} + q_{c2} - q_{pr2} \tag{43}$$

equation (43) makes the combined flows of the producers equal to the combined flow of the consumers. Moreover, the nominal operation conditions should maintain $q_{pr1} \ge 0$. With the use of mass balances the following equations of the flow at each node are defined:

$$q_{pr1} = q_{d1} \tag{Note a}$$

$$q_{d1} + q_{d2} = q_{c1} \tag{Note b}$$

$$q_{d3} = q_{d2} + q_{c2} \tag{Note c}$$

$$q_{pr2} = q_{d3} \tag{Note d}$$

$$q_{pr1} = q_{d4} \tag{Note a'}$$

$$q_{c1} = q_{d5} + q_{d4}$$
 (Note b')

$$q_{c2} + q_{d5} = q_{d6}$$
 (Note c')

$$q_{pr2} = q_{d6} \tag{Note d'}$$

Note here that the temperatures at the nodes in the distribution grid are stated according to node a with temperature T_a . The temperature dynamics of the system can now be identified. Since q_{c1} , q_{c2} and p_{pr2} are the independent variables, all of the flows in the temperature dynamics should be stated in these three flows, using the equations of the flows in the nodes above. This gives the following equations:

$$0 = (q_{c1} + q_{c2} - q_{pr2})T_{pr1} - (q_{c1} + q_{c2} - q_{pr2})T_a$$
 (Node a)

$$0 = (q_{c1} + q_{c2} - q_{pr2})T_{pr1} + (q_{c2} - q_{pr2})T_b - q_{c1}T_{c1}$$
(Node b)

$$0 = (q_{c2} - q_{pr2})T_b + q_{c2}T_{c2} - q_{pr2}T_{pr2}$$
(Node c)

$$0 = q_{pr2}T_{pr2} - q_{pr2}T_d \tag{Node d}$$

$$0 = (q_{c1} + q_{c2} - q_{pr2})T_{pr1} - (q_{c1} + q_{c2} - q_{pr2})T_{a'}$$
(Node a')

$$0 = q_{c1}T_{c1} - (q_{c2} - q_{pr2})T_{b'} - (q_{c1} + q_{c2} - q_{pr2})T_{a'}$$
(Node b')

$$0 = q_{c2}T_{c2} + (q_{c2} - q_{pr2})T_{b'} - q_{pr2}T_{pr2}$$
 (Node c')

$$0 = q_{pr2}T_{pr2} - q_{pr2}T_{c'}$$
 (Node d')

$$V_{pr1}\dot{T}_{pr1} = (q_{c1} + q_{c2} - q_{pr2})(T_{pr1,in} - T_{pr1,out}) + P_{pr1}$$
(Producer 1)

$$V_{pr2}T_{pr2} = q_{pr2}(T_{pr2,in} - T_{pr2,out}) + P_{pr2}$$
 (Producer 2)

$$\rho_w c_w V_{c1} \dot{T}_{c1} = \rho_w c_w q_{c1} (T_{c1}^{in} - T_{c1}^{out}) - \lambda_{3,c1} (T_{c1,amb} - T_{c1,room}^{min}) + \lambda_{3,c1} z_{s,c1} (T_{c1,room}^{min} - T_{c1,room}^{max})$$
(Consumer 1)

$$\rho_w c_w V_{c2} \dot{T}_{c2} = \rho_w c_w q_{c2} (T_{c2}^{in} - T_{c2}^{out}) - \lambda_{3,c2} (T_{c2,amb} - T_{c2,room}^{min}) + \lambda_{3,c2} z_{s,c2} (T_{c2,room}^{min} - T_{c2,room}^{max})$$
(Consumer 2)

$$v_{d1}\dot{T}_{d1} = (q_{c1} + q_{c2} - q_{pr2})(T_{d1}^{in} - T_{d1}^{out})$$
(Pipe 1)

$$V_{d2}T_{d2} = (q_{c2} - q_{pr2})(T_{d2}^{in} - T_{d2}^{out})$$
(Pipe 2)

$$V_{d3}\dot{T}_{d3} = q_{pr2}(T_{d3}^{in} - T_{d3}^{out})$$
(Pipe 3)

$$V_{d4}\dot{T}_{d4} = (q_{c1} + q_{c2} - q_{pr2})(T_{d4}^{in} - T_{d4}^{out})$$
(Pipe 4)

$$V_{d5}\dot{T}_{d5} = (q_{c2} - q_{pr2})(T_{d5}^{in} - T_{d5}^{out})$$
(Pipe 5)

$$V_{d6}\dot{T}_{d6} = q_{pr2}(T_{d6}^{in} - T_{d5}^{out})$$
(Pipe 6)

Important to note is that the temperatures in the pipe is equal to the node the water is flowing from, since no heat is lost within the pipes and nodes. However, for pipe 2 and pipe 4 this will depend on the flow direction. When the flow is negative or positive, the in and out direction changes, together with the node the temperature will come from.

3.3 Controlling the Flow

This section looks into a controller for the flow. Since the numerical case study only has one of the producers flow as independent variable, the controller can be simplified to a proportional integral controller. This controller is used for controlling the network of the consumer and satisfy the desired temperature of the consumer. Since this thesis does not include a stability analysis, no real controller is designed, but a rational suggestion is introduced and used for the numerical simulations. The goal of the controller is to get the temperature of the room of the consumer close or the same as its desired temperature at all times:

$$\lim_{t \to \infty} T_{room} = T_{room}^{desired} \tag{44}$$

In order to achieve this goal, some control laws need to be suggested. These control laws can then all be added to have an overall controller. A list of the desired variables is made who need to be controlled in order to carry our the numerical simulations.

- Controlling the flow of the consumers
- Controlling the flow of the producers
- Including the variables for the social behaviour
- Controlling the power of the producers
- Controlling the power of the consumers
- Adding the control system matrix of the distribution graph network

To start with the flow of the consumers, first some assumptions need to be made. Regarding the temperature of the consumer it is stated that the inlet temperature T_c^{in} is always larger than the outlet temperature of the consumer T_c^{out} . Both of these temperatures can be decided by the consumer. In view of these assumptions, an auxiliary input variable w_c can be introduced for the consumer in equation (45) and can be substituted in the overall equation of the consumer, from the state space equations. The result of this substitution is than visualized in equation (46).

$$q_c = \frac{w_c}{T_{pr}^* - T_c^{out}} \tag{45}$$

$$\rho_w c_w V_c \dot{T}_c = w_c - \lambda_3 (T_{amb} - T_{room}^{min}) + \lambda_3 z_s (T_{room}^{min} - T_{room}^{max})$$

$$\tag{46}$$

As can be seen, one adjustment is made, T_c^{in} is changed to T_{pr}^* . This reduces the number of sensors in the controller, and simplifies the overall system. This replacement can be made with the assumption that no heat is lost in the distribution network. To explain why the neglecting of heat loss in the distribution network leads to $T_c^{in} \to T_{pr}^*$, Figure 13 is used.



Figure 13: Example grid

In this figure, it can be seen that the output of the producer P is equal to the input of the consumer C, when no heat is lost in the pipes and nodes. To control equation (45) and achieve the desired temperature, the following control law is proposed for w_c :

$$w_c = -\alpha_c (T_c - T_c^*) + Z_c \tag{47}$$

$$\dot{Z}_c = -\beta_c (T_c - T_c^*) \tag{48}$$

However, a controller is desired with q_c as variable, since this is the tool to provide the consumers with their desired temperature. Therefore, according to equation (45) it is substituted in equation (48).

$$q_c = \frac{1}{T_c - T_c^*} - \alpha_c (T_{pr}^* - T_c^{out}) + z_s$$
(49)

$$\dot{Z}_c = -\beta_c (T_c - T_c^*) \tag{50}$$

The next step is to design a control law for q_{pr} . This is done with inspiration from the results in Cucuzzella et al. (2019) for the fair load sharing in electrical micro grids. In this paper, a controller is suggested this is able to adjust the flows in the system. In the following equations this controller is set according to Cucuzzella et al. (2019). In these equations, J_{ch} is a symmetrical positive definite matrix. For simplicity, J_{ch} is taken as the identity matrix. α_{ch} , β_{ch} , γ_{ch} and d_{ch} are the tuning gains for the chords.

$$J_{ch}\dot{q}_{ch} = -diag(\alpha(q_{ch} - \phi_{ch}) + diag(d_{ch,i})\mathcal{L}_{ch}\theta_{ch}$$
(51a)

$$diag(\beta_{ch})\dot{\theta} = -\mathcal{L}_{ch}diag(d_{ch})q_{ch}$$
(51b)

$$diag(\gamma_{ch})\phi_{ch} = -\phi_{ch} + q_{ch} \tag{51c}$$

In these equations, the matrix \mathcal{L}_{ch} represents the Laplacian matrix of any undirected, connected communication graphs among devices whose flows are written in the vector q_{ch} . The usage of \mathcal{L}_{ch} is based on Simpson-Porco (Sep 2016). This communication between the cord flows, the two consumers and the second producer, is needed in order to let the consumers and producers function in one network. The communication network is visualised in Figure 14.

The matrix \mathcal{L}_{ch} is defined by the communication lines between q_{ch} . This matrix is an instrumental matrix determined by $\mathcal{L}_{ch} = \mathcal{D}_{degree} - \mathcal{A}$. \mathcal{D}_{degree} is the degree matrix of 3x3, that indicates the edges. \mathcal{A} is the adjacency matrix, that indicates which nodes are connected.

$$\mathcal{L}_{ch} = \begin{bmatrix} C_1 & C_2 & P_2 \\ 1 & 2 & -1 & -1 \\ -1 & 2 & -1 \\ 3 & -1 & -1 & 2 \end{bmatrix}$$



Figure 14: Communication graph

This instrumental matrix is useful since it has multiple fundamental properties which can be used to come up with an equation regarding the power output of both producers and their flows. The first property is that Laplacian matrices are symmetrical and positive, as can be seen by the pattern in the instrumental matrix. Therefore $\mathcal{L}_{ch} = \mathcal{L}_{ch}^T > 0$. The next property is the kernel of the Laplacian matrix. Specific matrix A has the following identity: When multiplied with its kernel the indices of matrix A become 0, $A^*Ker(A) = 0$. The kernal for the laplacian matrix is stated as, $Ker(\mathcal{L}_{ch}) = 1_n$. For which 1_n a vector of 1's with the according dimension $_n$. This indicates that the eigenvalue is equal to 0. Another property is that \mathcal{L}_{ch} is a singular matrix. Therefore there is some vector, \overline{x} for which $\overline{x} \neq 0$, that $\mathcal{L}_{ch} \overline{x} = 0$.

In order to link the communication network to the flows, \overline{x} is set as the chord flows \overline{q}_{ch} . This flow equilibrium of the independent producer is $\overline{q}_{ch} = \alpha \mathbf{1}_n$, for which α is some constant. Since $\overline{q}_{ch} = [\overline{q}_{pr_2} \ \overline{q}_{c_2} \ \overline{q}_{c_1}]$, the flow equilibrium of the independent producer \overline{q}_{pr} , can be set as equal to the same constant multiplied with vector of 1's with the according dimension n. $\overline{q}_{pr_2} = \alpha \mathbf{1}_n$. As a consequence, $\overline{q}_{pr_2} = \overline{q}_{c_2} = \overline{q}_{c_1}$.

In order to allow the equal distribution of the flows in the network between the independent chords, a diagonal matrix $\mathcal{D}_{diagonal}$ is added, in which $\mathcal{D}_{diagonal} = [d_1 \ d_2 \ d_3]$ and d_n are all diagonals for the independent chords. Therefore, $\mathcal{L}_{ch} \ \mathcal{D}_{diagonal} \ \overline{q}_{pr} = 0$. Out of this equation it can be concluded that $d_1 \overline{q}_{c_1} = d_2 \overline{q}_{c_2} = d_3 \overline{q}_{pr_2}$, since the flows are all equal to $\alpha \mathbf{1}_n$.

In conclusion, it can be assumed that $d_1 = \frac{1}{P_{c_1}^{max}}$. Therefore the following equation can be set up:

$$\frac{\bar{q}_{c_1}}{P_{c_1}^{max}} = \frac{\bar{q}_{c_2}}{P_{c_2}^{max}} = \frac{\bar{q}_{pr_2}}{P_{pr_2}^{max}}$$
(52)

It is important to note that in equation (52) the power of the consumer is equal to the power they subtracts from the network. In addition the power of the producer is the power
they add to the network. Moreover, from equation (52) it can be concluded that when the power for consumer one is larger in comparison to consumer two its flow will be larger as well.

In this thesis, the controller from Cucuzzella et al. (2019) is used to design a controller for the case study carried out in this thesis. Therefore, q_{ch} is changed to Q_{pr} since this is the only independent variable. Moreover, J_{ch} is replaced by J_{pr} , α_{ch} by $\hat{\alpha}_{pr}$, B_{pr} by $\hat{\beta}_{pr}$, d_{ch} by d_{pr} , \mathcal{L}_{ch} by \mathcal{L}_{pr} and γ_{ch} by γ_{pr} .

$$J_{pr}\dot{Q}_{pr} = -diag(\alpha_{pr})(Q_{pr} - Q_{pr}^*) + Z_{pr}$$
(53a)

$$\dot{Z}_{pr} = -\operatorname{diag}(\beta_{pr})(Q_{pr} - Q_{pr}^*)$$
(53b)

Here J_{pr} , α_{pr} , β_{pr} and Q_{pr}^* are all constant tuning parameters. Q_{pr}^* is the desired value for the flow.

3.4 Overall Controller

To conclude, the overall system dynamics in closed-loop graph network for the case study with the proposed controllers is given by the equations below. In this group of equations, the matrix (54), represents the distribution graph network, as explained in section 2.6. The equations 55, 56, 57, 58 and 59 are the controller, as mentioned the section 3.3. Equations 58 and 59 are from the social variables. Which are set up in section 2.7 and made suitable for the numerical case study of two consumer by adding a diagonal matrix. As stated in section 2.9, these equations are different for every consumer and therefore one equation is made per consumer by the diagonal matrix.

Equations 62, 63 and 64 are the flows of the consumers q_c , the power injected by the producers P_{pr} and the power consumed by the consumers P_c . The flow of the consumer is stated in section 3.3. For the power of the producers equation (16) is taken and adjusted for both producers by the diagonal matrix. For the power consumption of the consumers equation (37) is taken and made for both the consumers by the diagonal matrix. As previously mentioned, no stability analysis is carried out in this thesis. However, it can be expected that this proportional integral controller does function, since the system is simplified and based on stable controllers from Cucuzzella et al. (2019), Machado et al. (2020), Feng et al. (2020).

$$\begin{bmatrix} diag(V_e) & 0\\ 0 & diag(V_N) \end{bmatrix} \begin{bmatrix} \dot{T}_e\\ \dot{T}_N \end{bmatrix} = \begin{bmatrix} -diag(|q_e|) & diag(|q_e|)\mathcal{S}^\top\\ \mathcal{T}diag(|q_e|) & -diag(\mathcal{T}|q_e|) \end{bmatrix} \begin{bmatrix} T_e\\ T_N \end{bmatrix} + \mathcal{B}_{pr}P_{pr} - \mathcal{B}_c P_c \quad (54)$$

$$J_{pr}\dot{Q}_{pr} = -diag(\alpha(Q_{pr} - \phi_{pr}) + diag(d_{pr})\mathcal{L}_{pr}Q_{pr}$$
(55)

$$diag(\beta_{pr})\dot{\theta}_{pr} = \mathcal{L}_{pr}diag(d_{pr})Q_{pr}$$
(56)

$$diag(\gamma_{pr})\dot{\phi}_{pr} = -\gamma_{pr} + Q_{pr} \tag{57}$$

$$diag(\hat{\beta}_{pr})\dot{Z}_{pr} = -(T_{pr}^{in} - T_{pr}^{out})$$
(58)

$$\dot{Z}_c = -diag(\beta_c)(T_c - T_c^*) \tag{59}$$

$$\dot{z}_s = diag(\alpha_s)(p_s - z_s - diag(h_s)s_s) \tag{60}$$

$$\dot{p}_s = diag(c_s)(p_s^{ego} - p_s) + diag(d_s)(p_s^{bio} - p_s)$$
(61)

$$q_c = diag(T_{pr}^* - T_c^{out})^{-1}(-diag(\alpha_c)(T_c - T_c^*) + z_s)$$
(62)

$$P_{pr} = -diag(\hat{\alpha_p})(T_p - T_p^*) + Z_{pr}$$

$$\tag{63}$$

$$P_c = diag(\lambda_c)(T_{room}^{min} - T_{amb}) + diag(\lambda_c)diag(z_s)(T_{room}^{max} - T_{room}^{min})$$
(64)

For T_{pr} the temperature of both the producers and for T_c the temperature of both the consumers are put into matrix form. Recall that these temperatures can be calculated according to equations 65.

$$T_{pr} = B_{pr} \begin{bmatrix} T_e \\ T_N \end{bmatrix} \qquad T_c = B_c \begin{bmatrix} T_e \\ T_N \end{bmatrix}$$
(65)

4 Simulations

In this chapter, the performance of the closed loop district heating system model, with the suggested overall controller, is tested by numerical simulations. In the three subsections the results from the three experiments are shown.

For these simulations the data and district heating network from Chapter 3 is used, with two producers and two consumers and with the same topology as in Figure 12. The identified unknown from section 3.1 is taken for the conductivity. In this thesis, three experiments are carried out in MATLAB. The three corresponding codes are provided in the appendix, 7.2, 7.3, 7.4. The goal of these experiments are set as the following:

- 1. To see if the system goes to a steady state, even after disturbances are inserted.
- 2. To test the influence of the heat coefficient loss, λ .
- 3. To test the influence of the social variables.

In these experiments the ambient temperature develops according to a sine from 5°C at 00:00 hour to 10 degrees at 12:00 hour till 5°C at 24:00 hour. The density and specific heat values are respectively $\rho_w = 975 \ kg/m^3$ and $c_w = 4190 \ J/kg^{\circ}C$, similar to Machado et al. (2020).

As stated in 3.3 a target for the temperature of the nodes is required, which is identified by T^* . The desired temperature of the nodes can be identified using the work of Machado et al. (2020). There it is stated that the desired temperature of the nodes in the hot stream is 85°C and 55°C in the cold stream. Important to note is that the consumer does not have a temperature of 55°C in its radiator or room. The tuning gains for the controller are taken as $\hat{\alpha}_{pr} = \langle 1e^4 \rangle$ and $\hat{\beta}_{pr} = \frac{4*I_{npr}}{\langle \hat{\alpha}_{pr} \rangle^{-2}} * \langle n_{pr}, 1 \rangle$ for the producers. For the consumers, $\alpha_c = 100 * \hat{\alpha}_{pr}$ and $\beta_c = \frac{4*I_{nc}}{\langle \alpha_c \rangle^{-2}} * \langle n_c, 1 \rangle$. This section was developed using trial-and-error procedure, which had the aim to attain a fast equilibrium without much overshooting.

The coming sections provide a detailed explanation of the results of the three experiments, which are shown in Figure 15, Figure 16, Figure 17 and for experiment 2 some more figures are provided in Appendix 7.2.

4.1 Experiment 1, steady state test

In the plot of the first experiment, (Figure 15), the power subtracted by the two consumers over the time span of a week, an oscillation can be viewed. This is because of the setting of the ambient temperature which changes during day. During the night, the consumer is subtracting more heat from the network, in order to maintain its desired temperature. Important to note is that in this simulation the consumers do not put out the radiator during the night. In the legend of Figure 15, it can be seen that consumer one has higher biospheric values and consumer two higher egoistic values. The physical properties of these consumers are the same, therefore the difference caused by the social behaviour can be seen. Consumer one has an egoistic values of 0.2 and a biospheric values of 0.2. Consumer two has an egoistic value of 0.8 and a biospheric values of 0.8. It is important to remember that for biospheric a value closer to 0 indicates that this consumer has more biospheric values and for egoistic a value closer to 1 indicates that this consumer had more egoistic values. According to Figure 15, the more egoistic consumer (two), consumes more power than the more biospheric consumer(one). During the afternoon of day four, hour 108, a different incentives is modeled. At this point, the value of the incentives increased from 0.25 to 0.75. Recall that increasing the incentives results in the energy being more expensive. According to Figure 15, both of the consumers will consume less heat. The difference in power, ΔP , is however larger for consumer two in comparison to consumer one, this is in agreement with Feng et al. (2020). The percentage difference of ΔP is 7.6% and 6.1% respectively. Overall it can be seen that the controller always tries to reach the desired temperature of the consumer. Even when the disturbance s_s is applied to the model the equilibrium is again researched after a little overshoot.



Figure 15: Experiment 1

4.2 Experiment 2, influence of the heat coefficient loss

In the plot of the second experiment, (Figure 16), the same social values of experiment 1 are used for the consumers. As a result, consumer one again has a higher biospheric value and consumer two has a higher egoistic value. However, the physical properties change during the second day in the afternoon, at hour 60. In order to test the influence of the heat conductivity rate, multiple test are carried out. λ , is increased with 10% (Figure 16), 5% (Figure 20) and decreased with 10% (Figure 18) and 5% (Figure 19). The increase of the heat conductivity rate, makes the consumer lose more heat through their walls to the outside temperature. The decrease of the heat conductivity rate, makes the consumer lose less heat through their walls to the outside temperature. To these changes, both consumers respond immediately by consuming more energy for the increase and less energy for the decrease. This response is because, both consumers want to maintain their desired temperature. The effect of the λ increase of 10% is for both of the consumers an increase of $\Delta P = 10\%$. For the other tests the the same behaviour was seen, an increase of 5% is for both of the consumers an increase of $\Delta P = 5\%$, a decrease of 10% is for both of the consumers a decrease of $\Delta P = 10\%$ and a decrease of 5% is for both of the consumers a decrease of $\Delta P = 5\%$. At the afternoon of day four, hour 108, a different incentive is again modelled. The starting value of the incentives was 0.25 and this is increased to 0.75. Both consumers again react to this change similar as in experiment 1.



Figure 16: Experiment 2, 10% increase of λ

4.3 Experiment 3, influence of the social variables

In the plot of the third experiment, a change is made during the time duration in the social variables of both the consumers. All the physical properties of both of the consumers do not change compared to experiment one and two. At first, both of the consumers are equally egoistic and biospheric. During the second day in the afternoon a change is made to both the consumers. Consumer one is given more biospheric values and consumer two more egoistic values. As result, consumer one starts to consumer less power and consumer two starts to consume more power. This leads to consumer two using 10% more power than consumer one.



Figure 17: Experiment 3

5 Conclusion and Further Research

In this thesis a framework is formulated to incorporate the social behaviour of the consumer into the thermal physical system of the consumer for multi-source district heating networks. This incorporation, is done by defining state space equations, as in section 2.9. With the help of these state space equations, a case study is done, in section 3.2, on a network of two producers and two consumers. In order to control this network a suggestion is made for a controller, in section 3.3. When simulating this model, a simplified short term prediction for the consumers, in the district heating network, is made.

From results in section 4 on the numerical simulations of the case study, it can be concluded that the suggested controller makes the system, even with disturbance implemented, go to an equilibrium. Moreover, the importance of accurately defining the unknowns of the consumers is underlined. When the heat conductivity rate has an error of 10% increase, the power also has an error of 10% increase. This behaviour is tested to be linearly, so the percentage of error in heat conductivity rate is equal to the percentage of error in the power consumption. For the social behaviour of the consumers, it is again important to accurately define the unknowns. Identifying the wrong social values for a consumer can lead to 10% errors in the prediction of the heat demand. Therefore, it can be concluded that the prediction of the heat demand of the consumers, deals with large errors when their unknowns are wrongly defined. When the method of identification produces these errors, so the errors are not consumer specific, the overall prediction of the heat demand in the whole district will be way off.

For further research it would be suggested to do a stability analysis on the controller, shown in section 3.4. As of now, the system will go to an equilibrium in its current situation. However, when doing this stability analysis for the controller, stable predictions of the load demand can be guaranteed. Since the controller exists out of ordinary differential equations and should remain stable near a point of equilibrium, a Lyapunov stability analysis would be suggested. Moreover, storage tanks can be included as well as heat loss in the pipes. This would make the system more realistic and therefore the load prediction of the consumer more realistic to the real world. At last it would be advised to check whether the system will work properly on larger grids of consumers and producers as well.

6 Bibliography

- Bäumelt, T. and Dostál, J. (2020), 'Distributed agent-based building grey-box model identification', Control engineering practice 101, 104427.
 URL: http://dx.doi.org/10.1016/j.conengprac.2020.104427
- Cucuzzella, M., Trip, S., De Persis, C., Cheng, X., Ferrara, A. and van der Schaft, A. (2019),
 'A robust consensus algorithm for current sharing and voltage regulation in dc microgrids', *IEEE transactions on control systems technology* 27(4), 1583–1595.
 URL: https://ieeexplore.ieee.org/document/8362802
- Dominković, D. F., Stunjek, G., Blanco, I., Madsen, H. and Krajačić, G. (2020), 'Technical, economic and environmental optimization of district heating expansion in an urban agglomeration', *Energy (Oxford)* 197, 117243.
 URL: http://dx.doi.org/10.1016/j.energy.2020.117243
- Dotzauer, E. (2002), 'Simple model for prediction of loads in district-heating systems', Applied energy 73(3), 277–284.
 URL: http://dx.doi.org/10.1016/S0306-2619(02)00078-8
- Feng, S., Cucuzzella, M., Bouman, T., Steg, L. and Scherpen, J. M. A. (2020), 'An integrated human-physical framework for control of power grids'. URL: https://arxiv.org/abs/2012.11208
- Grassi, B., Piana, E. A., Beretta, G. P. and Pilotelli, M. (2021), 'Dynamic approach to evaluate the effect of reducing district heating temperature on indoor thermalcomfort', *Energies (Basel)* 14(25), 25. URL: https://doaj.org/article/681299b95b6b43628f909b9a07e0fab0
- Guelpa, E., Marincioni, L., Capone, M., Deputato, S. and Verda, V. (2019a), 'Demand side management in district heating networks: A real application', *Energy (Oxford)* 182, 433– 442.

URL: *http://dx.doi.org/10.1016/j.energy.2019.05.131*

- Guelpa, E., Marincioni, L., Capone, M., Deputato, S. and Verda, V. (2019b), 'Thermal load prediction in district heating systems', *Energy (Oxford)* **176**, 693–703. URL: http://dx.doi.org/10.1016/j.energy.2019.04.021
- Heydarian, A., McIlvennie, C., Arpan, L., Yousefi, S., Syndicus, M., Schweiker, M., Jazizadeh, F., Rissetto, R., Pisello, A. L., Piselli, C., Berger, C., Yan, Z. and Mahdavi, A. (2020),

'What drives our behaviors in buildings? a review on occupant interactions with building systems from the lens of behavioral theories', *Building and environment* **179**, 106928. URL: http://dx.doi.org/10.1016/j.buildenv.2020.106928

- Hillier, F. S. and Lieberman, G. J. (2006), Introduction to operations research, 8. ed., internat. ed., [nachdr.] edn, McGraw-Hill, Boston [u.a.].
- Kim, E.-J., He, X., Roux, J.-J., Johannes, K. and Kuznik, F. (2019), 'Fast and accurate district heating and cooling energy demand and load calculations using reduced-order modelling', Applied energy 238, 963–971. URL: http://dx.doi.org/10.1016/j.apenergy.2019.01.183
- Lund, H., Werner, S., Wilshire, R., Svendsen, S., Thorsen, J. E., Hvelpund, F. and Mathiesen,
 B. V. (2014), '4th generation district heating (4gdh) integrating smart thermal grids into future sustainable energy systems', *Energy (Oxford)* 68, 1–11.
- Ma, W., Fang, S., Liu, G. and Zhou, R. (2017), 'Modeling of district load forecasting for distributed energy system', *Applied energy* 204, 181–205.
 URL: http://dx.doi.org/10.1016/j.apenergy.2017.07.009
- Machado, J. E., Cucuzzella, M., Pronk, N. and Scherpen, J. M. A. (2021), 'Adaptive control for flow and volume regulation in multi-producer district heating systems'. URL: https://arxiv.org/abs/2103.06568
- Machado, J. E., Cucuzzella, M. and Scherpen, J. (2020), 'Modeling and passivity properties of district heating systems'. URL: https://arxiv.org/abs/2011.05419
- Namazkhan, M., Albers, C. and Steg, L. (2019), 'The role of environmental values, sociodemographics and building characteristics in setting room temperatures in winter', *Energy* (Oxford) 171, 1183–1192.
 URL: http://dx.doi.org/10.1016/j.energy.2019.01.113
- Nielsen, H. A. and Madsen, H. (2006), 'Modelling the heat consumption in district heating systems using a grey-box approach', *Energy and buildings* 38(1), 63–71.
 URL: http://dx.doi.org/10.1016/j.enbuild.2005.05.002
- Pinto, J. F. A. D. S. (2016), 'Refurbishment measures versus geothermal district heating for residential buildings in the netherlands'.
 URL: https://explore.openaire.eu/search/publication?articleId=od

- Saletti, C., Morini, M. and Gambarotta, A. (2020), 'The status of research and innovation on heating and cooling networks as smart energy systems within horizon 2020', *Energies* (*Basel*) 13(11), 2835. URL: https://search.proguest.com/docview/2410031202
- Scholten, T. W., De Persis, C. and Tesi, P. (2015), Modeling and control of heat networks with storage: The single-producer multiple-consumer case, EUCA, pp. 2242–2247. URL: https://ieeexplore.ieee.org/document/7330872
- Simpson-Porco, J. W. (Sep 2016), Input/output analysis of primal-dual gradient algorithms, IEEE, pp. 219–224. URL: https://ieeexplore.ieee.org/document/7852233
- Sleptsov, A., Crisostomi, E. and Bischi, A. (2021), 'Control schemes for district heating substations considering user-defined building's indoor temperature', *Building and environment* 191, 107598.
 URL: http://dx.doi.org/10.1016/j.buildenv.2021.107598
- van der Schaft, A. and Jeltsema, D. (2020), 'Limits to energy conversion'. URL: https://arxiv.org/abs/2006.15953
- Wang, Y., You, S., Zhang, H., Zheng, W., Zheng, X. and Miao, Q. (2017), 'Hydraulic performance optimization of meshed district heating network with multiple heat sources', *Energy (Oxford)* **126**, 603–621.
 URL: http://dx.doi.org/10.1016/j.energy.2017.03.044

University of Groningen

7 Appendix

consumer 1 (bio) consumer 2 (ego) $P_{\rm c}$ (W) 100 t (hour)

7.1 Appendix A - Extra results Experiment 2

Figure 18: Heat conductivity rate 10% decrease



Figure 19: Heat conductivity rate 5% decrease



Figure 20: Heat conductivity rate 5% increase

7.2 Appendix B - Code Experiment 1

```
1 %Code 1 Endre Eisenga S3510913 Bachelor IP
2 % A start of this code is from Juan, further more he helped me
     with some specific code when stated.
3
  clear all
4
5
  %% System parameters
6
7
  n_pr=2; %number of producers
8
9
  n_c=2; %number of consumers
10
11
  n_ch=3; %number of CHORDS (or number of independent flows)
12
13
  n_e=10; %number of edges
14
15
  n_N=8; %number of nodes
16
17
  rho=975; %density of water
18
19
  csh_water=4190; %Specific heat of water
20
```

```
21
  V_e=rho*csh_water*[1;1;1;1;1;1;1;1;1;1;1]; %vector of edges volumes
22
23
  V_n=zeros(n_N,1); % vector of node volumes
24
25
26
  %% define the time
27
28
  t0=0; %initial time
29
   tf = 7*(24*60*60); % final time
30
  tm=0.64*tf; 7%% this marks the transition in the incentive s s at
31
       the 5th day in the afternoon
   tu=0.357*tf; %%% this marks the point were a new value for U_c or
32
       p<sup>bio</sup> and p<sup>ego</sup> is implemented
33
   tspan = [t0 tf]; %Time span
34
35
  %% Matrixes according to graph theory
36
37
  F = [-1, 1, 0, 0, -1, -1, 1, -1, -1, 1;
38
       1.0.1.0.1.0.0.1.0.0:
39
        1,0,0,1,1,1,0,1,1,0]; %fundamental loop matrix
40
41
   calB0 = [1, 0, 0, 0, -1, 0, 0, 0, 0, 0;
42
       0, 0, -1, 0, 1, -1, 0, 0, 0, 0;
43
       0, 0, 0, -1, 0, 1, 1, 0, 0, 0;
44
       0, 1, 0, 0, 0, 0, -1, 0, 0, 0;
45
        -1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0;
46
       0, 0, 1, 0, 0, 0, 0, -1, 1, 0;
47
       0, 0, 0, 1, 0, 0, 0, 0, -1, -1;
48
       0, -1, 0, 0, 0, 0, 0, 0, 0, 1; % fixe incidence matrix
49
50
  B_pr=[eye(n_pr); zeros(n_e+n_N-n_pr, n_pr)]; %coefficient matrix
51
      P_pr
52
  B_c = [zeros(n_pr, n_c); eye(n_c); zeros(n_e+n_N-n_pr-n_c, n_c)]; \%
53
```

```
coefficient matrix of P_c
54
55
  % PI temp. reg. producer
56
  T_pr_star=85*ones(n_pr,1);
57
  alpha_pr_hat = 1e4 * [1;1];
58
  % beta pr hat=1e - 6 * [1;1];
59
  beta_pr_hat = 4*(eye(n_pr)/diag(alpha_pr_hat)^2)*ones(n_pr,1);
60
61
  %% PI flow reg. of chord producer
62
  J_pr = [1];
63
  alpha_pr = [1];
64
  beta_pr = [1];
65
  Q_pr_star=1e - 4 * [0.1];
66
67
  % PI controller flow q_c
68
  gamma_c=1e3*ones(n_c,1);
69
  T_c_star=55*ones(n_c,1);
70
  %alpha_c=1e4 * [1;1];
71
  alpha_c=100.0*alpha_pr_hat;
72
  % beta c=1e20 * [1;1];
73
  beta_c=4*(eye(n_c)/diag(alpha_c)^2)*ones(n_c,1);
74
75
  T_room_min=18*ones(n_c,1);
76
  T room \max=22*ones(n c, 1);
77
  %T_{amb} = 5 * ones (n_c, 1);
78
  T_amb=@(t) 5*sin(pi/12*(t/3600))+5; \%\% this function depends
79
      only on 't'
80
81
  %% Parameters 'social dynamics'
82
  a s=ones (n c, 1);
83
84
  % Setting a difference fot s_s for the morning and the afternoon
85
  s s=@(t)((t0 <= t)\&\&(t <= tm)) *0.25 * ones(n c, 1) + ((tm < t)\&\&(t <= tf))
86
      *0.75* ones (n c, 1);
```

```
c_s=ones(n_c,1);
88
  d\_s=ones(n\_c,1);
89
  % p ego=@(t)((t0 \le t)\&\&(t \le tu)) * [0.37; 0.37] * ones(n c, 1) + ((tu < t))
90
      \&\&(t \le t f)) * [0.9; 0.9] . * ones(n_c, 1);
91 % p_bio=@(t)((t0<=t)&&(t<=tu))*[0.77;0.77].*ones(n_c,1)+((tu<t))
      \&\&(t \le tf)) * [0.1; 0.1] . * ones(n c, 1);
   p ego = [0.2; 0.8] . * ones (n c, 1);
92
   p\_bio = [0.2; 0.8] . * ones (n\_c, 1);
93
94
   %The green lines are activated for experiment 3
95
96
   h_s = [0.25; 0.75] \cdot \text{sones}(n_c, 1); \% a more ego consumer has higher
97
      vales for h s
98
  %% Heat transfer coefficient.
99
   A_c=30*ones(n_c, 1);
100
  \% U c=Q(t) ((t0<=t) & (t<=tu)) *3.14 * ones (n c, 1) + ((tu<t) & (t<=tf))
101
      *40*ones(n c, 1);
   U c=3.14*ones(n c, 1);
102
  \% lambda c=@(t) diag (U c(t)) * (A c);
103
   lambda c=diag(U c)*A c;
104
105
   %The green lines are activated for experiment 2
106
107
108
   %% Overall closed-loop dynamics
109
   % This is made with the help of Juan
110
111
  T_e=@(x)x(1:n_e);
112
  T_N=@(x)x(n_e+1:n_e+n_N);
113
   Q pr=@(x)x(n e+n N+1:n e+n N+(n pr-1));
114
   z_pr=0(x)x(n_e+n_N+(n_pr-1)+1:n_e+n_N+2*(n_pr-1));
115
   z_c=@(x)x(n_e+n_N+2*(n_pr-1)+1:n_e+n_N+2*(n_pr-1)+n_c);
116
   z = @(x)x(n e+n N+2*(n pr-1)+n c+1:n e+n N+2*(n pr-1)+n c+n c);
117
  p = s=@(x)x(n e+n N+2*(n pr-1)+n c+n c+1:n e+n N+2*(n pr-1)+n c+n c
118
```

87

```
+n_c);
-1)+n_c+n_c+n_r);
      P_c_hat=@(x)x(n_e+n_N+2*(n_pr-1)+n_c+n_c+n_pr+1:n_e+n_N+2*(n_pr-1)+n_c+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1)+n_pr+1:n_e+n_N+2*(n_pr-1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n_pr+1)+n
120
                 n_{pr-1}+n_{c+n_{r-1}}+n_{r-1}+n_{r-1};
121
        n x=n e+n N+2*(n pr-1)+n c+n c+n pr+n c; \%size of the state
122
                  variable
123
124
        %% The auxiliary definitions according to the overall controller
125
        %This is made with the help of Juan
126
127
        T_pr=@(x) [eye(n_pr) zeros(n_pr, n_e-n_pr)] * T_e(x);
128
129
        T_c=@(x) [zeros(n_pr) eye(n_c) zeros(n_c, n_e-n_pr-n_c)] * T_e(x);
130
131
        T_c_i = 0(x) [0, 1, 0, 0, 0, 0, 0, 0;
132
                      [0, 0, 1, 0, 0, 0, 0, 0] *T N(x);
133
134
         q = (x)(eve(n c)/diag(rho*csh water*(T pr star-T c(x))))*(-diag(rho*csh water*(T pr star-T c(x))))*(-diag(rho*csh water*(T pr star-T c(x)))))*(-diag(rho*csh water*(T pr star-T c(x)))))*(-diag(rho*csh water*(T pr star-T c(x))))))
135
                  alpha c) *(T c(x) - T c star) + z c(x));
136
         P_pr=@(x)-diag(alpha_pr_hat)*(T_pr(x)-T_pr_star)+z_pr_hat(x);
137
138
        P_c=@(t,x) diag(lambda_c)*(T_room_min-[T_amb(t); T_amb(t)])+diag(
139
                 lambda_c) * diag(z_s(x)) * (T_room_max-T_room_min);
140
        P_ch=0(t,x) [[0 \ 1]*P_p(x);P_c(t,x)];
141
142
        q_ch=0(x) [Q_pr(x); q_c(x)];
143
144
        q_e=0(x)F'*q_ch(x);
145
146
         calB=@(x) calB0 * diag(sign(q e(x))); \% flow-adjusted incidence
147
                  matrix
```

```
148
   calT=@(x) 0.5*(calB(x)+abs(calB(x))); %target nodes - edges matrix
149
150
   calS=@(x) 0.5*abs(calB(x)-abs(calB(x)));%sources nodes - edges
151
       matrix
152
   %% Overall function according to the overall controller
153
   %This is made with the help of Juan
154
155
   f=@(t,x)[[[-rho*csh_water*diag(abs(q_e(x)))*T_e(x)+rho*csh_water*
156
       diag(abs(q e(x))) * calS(x) '*T N(x);
        rho*csh\_water*calT(x)*diag(abs(q_e(x)))*T_e(x)-rho*csh\_water*
157
            diag(calT(x)*abs(q_e(x)))*T_N(x)]+B_pr*P_pr(x)-B_c*P_c(t,x)
            )];
        -\operatorname{diag}(\operatorname{alpha}_{\operatorname{pr}}) * (Q_{\operatorname{pr}}(x) - Q_{\operatorname{pr}}\operatorname{star}) + z_{\operatorname{pr}}(x);
158
        -(Q_pr(x)-Q_pr_star);
159
        -(T_c(x)-T_c_star);
160
        diag(a_s)*(p_s(x)-z_s(x)-diag(h_s)*s_s(t));
161
        \operatorname{diag}(c_s) * (p_{ego}-p_s(x)) + \operatorname{diag}(d_s) * (p_{bio}-p_s(x));
162
        -(T_pr(x)-T_pr_star);
163
        -(P c hat(x)-P c(t,x))];
164
165
   M=blkdiag(diag(V_e), diag(V_n), diag(J_pr), diag(beta_pr), diag(
166
       beta_c), eye(n_c), eye(n_c), diag(beta_pr_hat), diag(gamma_c));
167
   options=odeset('Mass',M);
168
169
   %% Initial conditions:
170
171
   T_e0=20*ones(n_e,1);
172
   T_N0=20*ones(n_N,1);
173
   Q_pr0=0.5*[Q_pr_star];
174
   z_pr0 = [0.0];
175
   z_c0=zeros(n_c,1);
176
   z s0=zeros(n c, 1);
177
  p s0=zeros(n c, 1);
178
```

```
z_pr_hat0=zeros(n_pr,1);
179
   P_c_hat0=zeros(n_c,1);
180
181
   x0 = [T_0; T_N0; Q_pr0; z_pr0; z_c0; z_s0; p_s0; z_pr_hat0; P_c_hat0];
182
183
   %% The ODE solver
184
185
   [t_sol,x_sol]=ode15s(f,tspan,x0,options);
186
187
   %% Extract the useful data
188
189
   data_T_e = [];
190
   data_T_N = [];
191
   data_q_ch = [];
192
   data_q_e = [];
193
   data_P_pr = [];
194
   data_P_c = [];
195
   data_P_c_hat = [];
196
197
   for i=1:size(t_sol,1)
198
        data T e(i, :) = T e(x \text{ sol}(i, :));
199
        data T N(i, :) = T N(x \text{ sol}(i, :));
200
        data_q_ch(i,:) = q_ch(x_sol(i,:));
201
        data_q_e(i,:) = q_e(x_sol(i,:));
202
        data_P_pr(i,:)=P_pr(x_sol(i,:)');
203
        data_P_c(i,:) = P_c(t_sol(i), x_sol(i,:));
204
        data_P_c_hat(i,:) = P_c_hat(x_sol(i,:));
205
206
   end
207
208
209
   %% Plots
210
211
   figure()
212
   plot(t sol/3600, data T e)
213
   ylabel('$T_{\mathrm{e}}$ ($^\circ$C)', 'Interpreter', 'Latex');
214
```

```
xlabel('$t$~(hour)', 'Interpreter', 'Latex');
215
   grid off
216
   lgd=legend({ 'edge 1', 'edge 2', 'edge 3', 'edge 4'}, 'Interpreter', '
217
      Latex');
   lgd.FontSize = 8;
218
   lgd.Location='NorthEast';
219
220
   figure()
221
   plot(t_sol/3600,data_T_N)
222
   ylabel('$T_{\mathrm{N}}$ ($^\circ$C)', 'Interpreter', 'Latex');
223
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
224
   grid off
225
   lgd=legend({ 'node 1', 'node 2', 'node 3', 'node 4'}, 'Interpreter', '
226
      Latex');
   lgd.FontSize=8;
227
   lgd.Location='NorthEast';
228
229
   figure()
230
   plot (t_sol/3600,data_q_e(:,1),t_sol/3600,data_q_e(:,2))
231
   ylabel((\$q_{\mathrm{pr}})  (m\$^3, 'Interpreter', 'Latex');
232
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
233
   grid off
234
   lgd=legend({ '(edge 1)', 'edge 2'}, 'Interpreter', 'Latex');
235
   lgd.FontSize=8;
236
   lgd.Location='NorthEast';
237
238
   figure()
239
   plot(t sol/3600, data q ch)
240
   ylabel('$q_{\mathrm{ch}}$ (m$^3$/s)', 'Interpreter', 'Latex');
241
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
242
   grid off
243
   lgd=legend({ 'producer 2 (edge 2) ', 'consumer 1 (edge 3) ', 'consumer
244
       2 (edge 4) '}, 'Interpreter', 'Latex');
   lgd.FontSize=8;
245
   lgd.Location='NorthEast';
246
247
```

```
figure()
248
   plot (t_sol/3600,data_P_pr)
249
   ylabel('$P_{\mathrm{pr}}$ (W)', 'Interpreter', 'Latex');
250
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
251
   grid off
252
   lgd=legend({ 'producer 1 (edge 1) ', 'producer 2 (edge 2) '}, '
253
      Interpreter ', 'Latex');
   lgd. FontSize=8;
254
   lgd.Location='NorthEast';
255
256
   figure()
257
   plot(t_sol/3600,data_P_c_hat)
258
   ylabel('$P_{\mathrm{c}}$ (W)', 'Interpreter', 'Latex');
259
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
260
   grid off
261
   lgd=legend({ 'consumer 1 (bio)', 'consumer 2 (ego)'}, 'Interpreter',
262
      'Latex');
   lgd.FontSize=8;
263
   lgd.Location='NorthEast';
264
265
   % Save the relevant data.
266
267
   delete data_sim.mat
268
   save('data_sim.mat')
269
   disp('The data from the simulations has been correctly saved.')
270
```

7.3 Appendix C - Code Experiment 2

```
1 %Code 2 Endre Eisenga S3510913 Bachelor IP
2 %A start of this code is from Juan, further more he helped me
with some specific code when stated.
3
4 clear all
5
6 %% System parameters
7
8 n_pr=2; %number of producers
```

```
9
  n_c=2; %number of consumers
10
11
  n_ch=3; %number of CHORDS (or number of independent flows)
12
13
  n_e=10; %number of edges
14
15
  n N=8; %number of nodes
16
17
  rho=975; %density of wate
18
19
  csh_water=4190; %Specific heat of water
20
^{21}
  V_e=rho*csh_water*[1;1;1;1;1;1;1;1;1;1;1]; %vector of edges volumes
22
23
  V_n=zeros(n_N,1); % vector of node volumes
24
25
26
  %% define the time
27
28
  t0=0; %initial time
29
  tf = 7*(24*60*60); % final time
30
  tm=0.64*tf; 7%% this marks the transition in the incentive s_s at
31
       the 4th day in the afternoon
  tu=0.357*tf; %%% this marks the point were a new value for U_c at
32
       experiment 2 or p<sup>bio</sup> and p<sup>ego</sup> at experiment 3 is
     implemented, day 2 in the afternoon
33
  tspan = [t0 tf]; %Time span
34
35
  %% Matrixes according to graph theory
36
37
  F = [-1, 1, 0, 0, -1, -1, 1, -1, -1, 1;
38
       1, 0, 1, 0, 1, 0, 0, 1, 0, 0;
39
       1,0,0,1,1,1,0,1,1,0]; %fundamental loop matrix
40
41
```

```
calB0 = [1, 0, 0, 0, -1, 0, 0, 0, 0, 0;
42
        0, 0, -1, 0, 1, -1, 0, 0, 0, 0;
43
        0, 0, 0, -1, 0, 1, 1, 0, 0, 0;
44
        0, 1, 0, 0, 0, 0, -1, 0, 0, 0;
45
        -1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0;
46
        0, 0, 1, 0, 0, 0, 0, -1, 1, 0;
47
        0, 0, 0, 1, 0, 0, 0, 0, -1, -1;
48
        0, -1, 0, 0, 0, 0, 0, 0, 0, 1]; % fixe incidence matrix
49
50
  B_pr=[eye(n_pr); zeros(n_e+n_N-n_pr,n_pr)]; %coefficient matrix
51
      P_pr
52
  B_c = [zeros(n_pr, n_c); eye(n_c); zeros(n_e+n_N-n_pr-n_c, n_c)]; \%
53
      coefficient matrix of P c
54
  % PI temp. reg. producer
55
  T_pr_star=85*ones(n_pr,1);
56
   alpha_pr_hat=1e4 * [1;1];
57
  \%beta_pr_hat=1e-6*[1;1];
58
   beta_pr_hat = 4*(eye(n_pr)/diag(alpha_pr_hat)^2)*ones(n_pr,1);
59
60
  %% PI flow reg. of chord producer
61
  J_pr = [1];
62
  alpha_pr = [1];
63
  beta_pr = [1];
64
   Q_pr_star=1e - 4 * [0.1];
65
66
  %% PI controller flow q c
67
  gamma_c=1e3*ones(n_c,1);
68
  T_c_star=55*ones(n_c,1);
69
  \%alpha_c=1e4 * [1;1];
70
   alpha c = 100.0 * alpha pr hat;
71
  \%beta_c=1e20 * [1;1];
72
   beta_c=4*(eye(n_c)/diag(alpha_c)^2)*ones(n_c,1);
73
74
  T room \min=18* \operatorname{ones}(n c, 1);
75
```

```
T room max=22*ones(n c,1);
76
  \%T_{amb} = 5 * ones(n_c, 1);
77
   T_amb=@(t) 5*\sin(pi/12*(t/3600))+5; %%% this function depends
78
      only on 't'
79
80
   %% Parameters 'social dynamics'
81
   a s=ones (n c, 1);
82
83
  % Setting a difference fot s_s for the morning and the afternoon
84
   s s=@(t)((t0 <= t)\&\&(t <= tm)) *0.25 * ones(n c, 1) + ((tm < t)\&\&(t <= tf))
85
      *0.75*ones(n_c, 1);
86
   c_s=ones(n_c,1);
87
   d\_s=ones(n\_c,1);
88
  \% p_ego=@(t)((t0<=t)&&(t<=tu))*[0.37;0.37].*ones(n_c,1)+((tu<t))
89
      \&\&(t \le tf)) * [0.9; 0.9] . * ones(n_c, 1);
90 % p_bio=@(t)((t0<=t)&&(t<=tu))*[0.77;0.77].*ones(n_c,1)+((tu<t))
      \&\&(t \le t f)) * [0.1; 0.1] . * ones(n_c, 1);
   p_ego = [0.2; 0.8] . * ones (n_c, 1);
91
   p\_bio = [0.2; 0.8] . * ones (n\_c, 1);
92
93
   %The green lines are activated for experiment 3
94
95
   h_s = [0.25; 0.75] \cdot \text{sones}(n_c, 1); \% a more ego consumer has higher
96
      vales for h s
97
  % Heat transfer coefficient.
98
   A c=30*ones(n c, 1);
  U_c=@(t)((t0 \le t)\&\&(t \le tu)) *3.14 * ones(n_c, 1) + ((tu \le t)\&\&(t \le tf))
100
      *3.45*ones(n_c, 1);
  %For the 10% decrease, 3.45 is changed to 2.83.
101
  %For the 5% decrease, 3.45 is changed to 2.98.
102
  % For the 5% increase, 3.45 is changed to 3.29.
103
  \% U c=[3.14;3.14].*ones(n c,1);
104
   lambda c=@(t) \operatorname{diag}(U c(t)) * (A c);
105
```

```
\% lambda c=diag(U c)*A c;
106
107
108
  %% Overall closed-loop dynamics
109
  %This is made with the help of Juan
110
111
  T e=@(x)x(1:n e);
112
  T N=@(x)x(n e+1:n e+n N);
113
  Q pr=@(x)x(n e+n N+1:n e+n N+(n pr-1));
114
  z_pr=0(x)x(n_e+n_N+(n_pr-1)+1:n_e+n_N+2*(n_pr-1));
115
  z = @(x)x(n = +n N+2*(n pr-1)+1:n = +n N+2*(n pr-1)+n c);
116
  z_s=@(x)x(n_e+n_N+2*(n_pr-1)+n_c+1:n_e+n_N+2*(n_pr-1)+n_c+n_c);
117
  p_s=@(x)x(n_e+n_N+2*(n_pr-1)+n_c+n_c+1:n_e+n_N+2*(n_pr-1)+n_c+n_c
118
      +n c;
  z pr hat=@(x)x(n e+n N+2*(n pr-1)+n c+n c+n c+1:n e+n N+2*(n pr))
119
      -1)+n c+n c+n pr);
  120
      n_{pr-1}+n_{c+n_{r-1}}+n_{pr+n_{r-1}};
121
  n_x=n_e+n_N+2*(n_pr-1)+n_c+n_c+n_pr+n_c; %size of the state
122
      variable
123
124
  %% The auxiliary definitions according to the overall controller
125
  %This is made with the help of Juan
126
127
  T_pr=@(x) [eye(n_pr) zeros(n_pr, n_e-n_pr)] * T_e(x);
128
129
  T_c=@(x) [zeros(n_pr) eye(n_c) zeros(n_c, n_e-n_pr-n_c)] * T_e(x);
130
131
   T_c_i = @(x) [0, 1, 0, 0, 0, 0, 0, 0;
132
       [0, 0, 1, 0, 0, 0, 0, 0] *T N(x);
133
134
   q_c=@(x)(eye(n_c)/diag(rho*csh_water*(T_pr_star-T_c(x))))*(-diag(rho*csh_water*(T_pr_star-T_c(x)))))*(-diag(rho*csh_water*(T_pr_star-T_c(x)))))
135
      alpha c) *(T c(x) - T c star) + z c(x));
```

136

```
P_pr=@(x)-diag(alpha_pr_hat)*(T_pr(x)-T_pr_star)+z_pr_hat(x);
137
138
   P_c=@(t,x) diag(lambda_c(t)) * (T_room_min-[T_amb(t); T_amb(t)]) +
139
       diag(lambda_c(t)) * diag(z_s(x)) * (T_room_max-T_room_min);
140
   P_ch=@(t,x)[[0,1]*P_pr(x);P_c(t,x)];
141
142
   q_ch=0(x) [Q_pr(x); q_c(x)];
143
144
   q_e=@(x)F'*q_ch(x);
145
146
   calB=@(x) calB0 * diag(sign(q_e(x))); \% flow-adjusted incidence
147
       matrix
148
   calT=@(x) 0.5*(calB(x)+abs(calB(x))); %target nodes - edges matrix
149
150
   calS=@(x) 0.5*abs(calB(x)-abs(calB(x))); %sources nodes - edges
151
       matrix
152
153
   %% Overall function according to the overall controller
154
   %This is made with the help of Juan
155
156
   f=@(t,x)[[[-rho*csh_water*diag(abs(q_e(x)))*T_e(x)+rho*csh_water*
157
       diag(abs(q_e(x))) * calS(x) * T_N(x);
         rho*csh\_water*calT(x)*diag(abs(q_e(x)))*T_e(x)-rho*csh\_water*
158
            \operatorname{diag}\left(\operatorname{calT}\left(x\right)*\operatorname{abs}\left(q_{e}(x)\right)\right)*T_N(x)]+B_{pr}*P_{pr}(x)-B_{c}*P_{c}(t,x)
            )];
        -\operatorname{diag}(\operatorname{alpha}_{\operatorname{pr}}) * (Q_{\operatorname{pr}}(x) - Q_{\operatorname{pr}}\operatorname{star}) + z_{\operatorname{pr}}(x);
159
         -(Q_pr(x)-Q_pr_star);
160
         -(T_c(x)-T_c_star);
161
         diag(a \ s) * (p \ s(x) - z \ s(x) - diag(h \ s) * s \ s(t));
162
         diag(c_s)*(p_ego-p_s(x))+diag(d_s)*(p_bio-p_s(x));
163
         -(T_pr(x)-T_pr_star);
164
         -(P c hat(x)-P c(t,x))];
165
166
```

```
<sup>167</sup> M=blkdiag (diag (V_e), diag (V_n), diag (J_pr), diag (beta_pr), diag (
      beta_c), eye(n_c), eye(n_c), diag(beta_pr_hat), diag(gamma_c));
168
   options=odeset('Mass',M);
169
170
   %% Initial conditions:
171
172
   T e0=20*ones(n e, 1);
173
   T_N0=20*ones(n_N,1);
174
   Q_pr0=0.5*[Q_pr_star];
175
   z_pr0 = [0.0];
176
   z_c0=zeros(n_c,1);
177
   z_s0=zeros(n_c,1);
178
   p_s0=zeros(n_c,1);
179
   z_pr_hat0=zeros(n_pr,1);
180
   P_c_hat0=zeros(n_c,1);
181
182
   x0 = [T_0; T_N0; Q_pr0; z_pr0; z_c0; z_s0; p_s0; z_pr_hat0; P_c_hat0];
183
184
   %% The ODE solver
185
186
   [t_sol,x_sol]=ode15s(f,tspan,x0,options);
187
188
   %% Extract the useful data
189
190
   data_T_e = [];
191
   data_T_N = [];
192
   data q ch = [];
193
   data_q_e = [];
194
   data_P_pr = [];
195
   data_P_c = [];
196
   data_P_c_hat = [];
197
198
   for i=1:size(t_sol,1)
199
        data T e(i, :) = T e(x \text{ sol}(i, :));
200
        data T N(i, :) = T N(x \text{ sol}(i, :));
201
```

```
data_q_ch(i, :) = q_ch(x_sol(i, :));
202
        data_q_e(i, :) = q_e(x_sol(i, :));
203
        data_P_pr(i,:) = P_pr(x_sol(i,:));
204
        data P c(i,:) = P c(t \text{ sol}(i), x \text{ sol}(i,:)');
205
        data_P_c_hat(i,:)=P_c_hat(x_sol(i,:)');
206
207
   end
208
209
210
  %% Plots
211
212
   figure()
213
   plot (t_sol/3600,data_T_e)
214
   ylabel('$T_{\mathrm{e}}$ ($^\circ$C)', 'Interpreter', 'Latex');
215
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
216
   grid off
217
   lgd=legend({ 'edge 1', 'edge 2', 'edge 3', 'edge 4'}, 'Interpreter', '
218
      Latex');
   lgd. FontSize = 8;
219
   lgd.Location='NorthEast';
220
221
   figure()
222
   plot (t sol/3600, data T N)
223
   ylabel('$T_{\mathrm{N}}$ ($^\circ$C)', 'Interpreter', 'Latex');
224
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
225
   grid off
226
   lgd=legend({ 'node 1', 'node 2', 'node 3', 'node 4'}, 'Interpreter', '
227
      Latex');
   lgd.FontSize=8;
228
   lgd.Location='NorthEast';
229
230
   figure()
231
   plot (t_sol/3600,data_q_e(:,1),t_sol/3600,data_q_e(:,2))
232
   ylabel((\$q_{\mathrm{pr}})  (m\$^3, 'Interpreter', 'Latex');
233
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
234
   grid off
235
```

```
lgd=legend({ '(edge 1)', 'edge 2'}, 'Interpreter', 'Latex');
236
   lgd.FontSize=8;
237
   lgd.Location='NorthEast';
238
239
   figure()
240
   plot (t_sol/3600,data_q_ch)
241
   ylabel('$q {\mathrm{ch}}$ (m$^3$/s)', 'Interpreter', 'Latex');
242
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
243
   grid off
244
   lgd=legend({ 'producer 2 (edge 2) ', 'consumer 1 (edge 3) ', 'consumer
245
       2 (edge 4)'}, 'Interpreter', 'Latex');
   lgd. FontSize = 8;
246
   lgd.Location='NorthEast';
247
248
   figure()
249
   plot (t_sol/3600,data_P_pr)
250
   ylabel('$P_{\mathrm{pr}}$ (W)', 'Interpreter', 'Latex');
251
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
252
   grid off
253
   lgd=legend({ 'producer 1 (edge 1) ', 'producer 2 (edge 2) '}, '
254
      Interpreter ', 'Latex');
   lgd. FontSize=8;
255
   lgd.Location='NorthEast';
256
257
   figure()
258
   plot(t_sol/3600,data_P_c_hat)
259
   ylabel('$P_{\mathrm{c}}$ (W)', 'Interpreter', 'Latex');
260
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
261
   grid off
262
   lgd=legend({ 'consumer 1 (bio)', 'consumer 2 (ego)'}, 'Interpreter',
263
      'Latex');
   lgd. FontSize = 8;
264
   lgd.Location='NorthEast';
265
266
   %% Save the relevant data.
267
268
```

```
269 delete data_sim.mat
270 save('data_sim.mat')
271 disp('The data from the simulations has been correctly saved.')
```

7.4 Appendix D - Code Experiment 3

```
1 %Code 3 Endre Eisenga S3510913 Bachelor IP
<sup>2</sup> %A start of this code is from Juan, further more he helped me
     with some specific code when stated.
3
  clear all
4
5
  % System parameters
6
7
  n_pr=2; %number of producers
8
9
  n_c=2; %number of consumers
10
11
  n_ch=3; %number of CHORDS (or number of independent flows)
12
13
  n_e=10; %number of edges
14
15
  n_N=8; %number of nodes
16
17
  rho=975; %density of water
18
19
  csh_water=4190; %Specific heat of water
20
21
  V_e=rho*csh_water*[1;1;1;1;1;1;1;1;1;1;1]; %vector of edges volumes
22
23
  V_n=zeros(n_N,1); % vector of node volumes
24
25
26
  %% define the time
27
28
  t0=0: %initial time
29
  tf = 7*(24*60*60); % final time
30
```

```
tm=0.64*tf; %%% this marks the transition in the incentive s_s at
31
       the 4th day in the afternoon
   tu=0.357*tf; %%% this marks the point were a new value for U_c at
32
       experiment 2 or p<sup>bio</sup> and p<sup>ego</sup> at experiment 3 is
      implemented, day 2 in the afternoon
33
   tspan=[t0 tf]; %Time span
34
35
  % Matrixes according to graph theory
36
37
  F = [-1, 1, 0, 0, -1, -1, 1, -1, -1, 1;
38
        1, 0, 1, 0, 1, 0, 0, 1, 0, 0;
39
        1,0,0,1,1,1,0,1,1,0]; %fundamental loop matrix
40
41
   calB0 = [1, 0, 0, 0, -1, 0, 0, 0, 0, 0;
42
        0, 0, -1, 0, 1, -1, 0, 0, 0, 0;
43
        0, 0, 0, -1, 0, 1, 1, 0, 0, 0;
44
        0, 1, 0, 0, 0, 0, -1, 0, 0, 0;
45
        -1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0;
46
        0, 0, 1, 0, 0, 0, 0, -1, 1, 0;
47
        0, 0, 0, 1, 0, 0, 0, 0, -1, -1;
48
        0, -1, 0, 0, 0, 0, 0, 0, 0, 1]; % fixe incidence matrix
49
50
51
  B_pr = [eye(n_pr); zeros(n_e+n_N-n_pr, n_pr)]; \% coefficient matrix
52
      P_pr
53
  B_c = [zeros(n_pr, n_c); eye(n_c); zeros(n_e+n_N-n_pr-n_c, n_c)]; \%
54
      coefficient matrix of P_c
55
56
  % PI temp. reg. producer
57
  T pr star=85*ones (n pr, 1);
58
  alpha_pr_hat = 1e4 * [1;1];
59
  beta\_pr\_hat=4*(eye(n\_pr)/diag(alpha\_pr\_hat)^-2)*ones(n\_pr,1);
60
61
```

```
%% PI flow reg. of chord producer
62
           J_pr = [1];
63
            alpha_pr = [1];
64
            beta_pr = [1];
65
             Q_pr_star=1e - 4 * [0.1];
66
67
           %% PI controller flow q c
68
           gamma_c=1e3*ones(n_c,1);
69
             T c star=55*ones(n c,1);
70
           \%alpha_c=1e4 * [1;1];
71
             alpha c = 100.0 * alpha pr hat;
72
           \%beta_c=1e20 * [1;1];
 73
             beta_c=4*(eye(n_c)/diag(alpha_c)^2)*ones(n_c,1);
74
75
           T room_min=18 \times \text{ones}(n_c, 1);
76
           T room \max=22*ones(n c, 1);
77
           %T_amb = 5 * ones (n_c, 1);
78
           T_amb=@(t) 5*\sin(pi/12*(t/3600))+5; %%% this function depends
79
                            only on 't'
80
81
           %% Parameters 'social dynamics'
82
             a_s=ones(n_c,1);
83
84
           % Setting a difference fot s s for the morning and the afternoon
 85
             s_s=0(t)((t_0<=t)\&\&(t<=t_m))*0.25*ones(n_c,1)+((t_m<t)\&\&(t<=t_f))
86
                            *0.75*ones(n_c, 1);
87
             c_s=ones(n_c,1);
88
             d\_s=ones(n\_c,1);
 89
                  p_ego=@(t)((t0 <= t)\&\&(t <= tu)) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t)\&\&(t) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t)\&\&(t) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu < t) & (tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) = tu) * [0.37; 0.37] . * ones(n_c, 1) + ((tu) 
90
                                 t \le tf)  = tf)  = [0.37; 0.9] . * ones (n c, 1); 
                  p\_bio=@(t)((t0 \le t)\&\&(t \le tu)) * [0.77; 0.77] * ones(n\_c,1) + ((tu \le t)\&\&(t \le tu)) * [0.77; 0.77] * ones(n\_c,1) + ((tu \le t)\&\&(t \le tu)) * [0.77; 0.77] * ones(tu) * [0.77; 0.
91
                                 t \le tf) ) * [0.1; 0.77]. * ones (n_c, 1);
         % p ego = [0.2; 0.8] * ones (n c, 1);
92
_{93} % p bio = [0.2; 0.8].* ones (n c, 1);
```

```
94
  h_s=0(t)((t_0<=t)\&\&(t<=t_u))*[0.5;0.5].*ones(n_c,1)+((t_u<t)\&\&(t<=t_f))
95
     )) *[0.25; 0.75]. * ones (n_c, 1);
  % a more ego consumer has higher vales for h s
96
97
  % Heat transfer coefficient.
98
  A c=30*ones(n c, 1);
99
  \% U c=@(t)((t0<=t)&&(t<=tu))*3.14*ones(n c,1)+((tu<t)&&(t<=tf))
100
      *40*ones(n c, 1);
  U c=3.14*ones(n c, 1);
101
  \% lambda c=@(t) diag (U c(t)) * (A c);
102
  lambda_c=diag(U_c)*A_c;
103
104
  %The green lines are activated for experiment 2
105
106
107
  %% Overall closed-loop dynamics
108
  This is made with the help of Juan
109
110
  T_e=0(x)x(1:n_e);
111
  T_N=@(x)x(n_e+1:n_e+n_N);
112
  Q pr=@(x)x(n e+n N+1:n e+n N+(n pr-1));
113
  z pr=@(x)x(n e+n N+(n pr-1)+1:n e+n N+2*(n pr-1));
114
  z_c=@(x)x(n_e+n_N+2*(n_pr-1)+1:n_e+n_N+2*(n_pr-1)+n_c);
115
  z_s=@(x)x(n_e+n_N+2*(n_pr-1)+n_c+1:n_e+n_N+2*(n_pr-1)+n_c+n_c);
116
  p_s=@(x)x(n_e+n_N+2*(n_pr-1)+n_c+n_c+1:n_e+n_N+2*(n_pr-1)+n_c+n_c
117
     +n_c);
  z pr hat=@(x)x(n e+n N+2*(n pr-1)+n c+n c+n c+1:n e+n N+2*(n pr))
118
     -1)+n c+n c+n pr);
  119
     n_{pr-1}+n_{c+n_{r-1}}+n_{pr+n_{r-1}};
120
  n x=n e+n N+2*(n pr-1)+n c+n c+n pr+n c; \%size of the state
121
      variable
122
123
```

```
%% The auxiliary definitions according to the overall controller
124
   %This is made with the help of Juan.
125
126
   T_pr=@(x) [eye(n_pr) zeros(n_pr, n_e-n_pr)] * T_e(x);
127
128
   T_c=@(x) [zeros(n_pr) eye(n_c) zeros(n_c, n_e-n_pr-n_c)] * T_e(x);
129
130
   T_c_i = @(x) [0, 1, 0, 0, 0, 0, 0, 0;
131
        0, 0, 1, 0, 0, 0, 0, 0] *T N(x);
132
133
   q_c=0(x)(eye(n_c)/diag(rho*csh_water*(T_pr_star-T_c(x))))*(-diag(rho*csh_water*(T_pr_star-T_c(x)))))*(-diag(rho*csh_water*(T_pr_star-T_c(x)))))
134
      alpha_c) * (T_c(x)-T_c_star)+z_c(x));
135
   P_pr=@(x)-diag(alpha_pr_hat)*(T_pr(x)-T_pr_star)+z_pr_hat(x);
136
137
   P_c=@(t,x) diag(lambda_c)*(T_room_min-[T_amb(t); T_amb(t)])+diag(
138
      lambda_c) * diag (z_s(x)) * (T_room_max-T_room_min);
139
   P_ch=@(t,x)[[0,1]*P_pr(x);P_c(t,x)];
140
141
   q_ch=0(x) [Q_pr(x);q_c(x)];
142
143
   q = @(x)F'*q ch(x);
144
145
   calB=@(x) calB0 * diag(sign(q_e(x))); \% flow-adjusted incidence
146
      matrix
147
   calT=@(x)0.5*(calB(x)+abs(calB(x))); %target nodes - edges matrix
148
149
   calS=@(x) 0.5*abs(calB(x)-abs(calB(x))); %sources nodes - edges
150
      matrix
151
152
   %% Overall function according to the overall controller
153
   %This is made with the help of Juan
154
155
```

```
f=@(t,x)[[-rho*csh_water*diag(abs(q_e(x)))*T_e(x)+rho*csh_water*
156
       diag (abs(q_e(x))) * calS(x) * T_N(x);
        rho*csh\_water*calT(x)*diag(abs(q_e(x)))*T_e(x)-rho*csh\_water*
157
            diag(calT(x) * abs(q_e(x))) *T_N(x) = B_pr *P_pr(x) - B_c *P_c(t, x)
            )];
        -\operatorname{diag}(\operatorname{alpha}_{pr}) * (Q_{pr}(x) - Q_{pr}\operatorname{star}) + z_{pr}(x);
158
        -(Q_pr(x)-Q_pr_star);
159
        -(T c(x)-T c star);
160
        diag(a_s) * (p_s(x) - z_s(x) - diag(h_s(t)) * s_s(t));
161
        diag(c s) * (p ego(t)-p s(x))+diag(d s) * (p bio(t)-p s(x));
162
        -(T \operatorname{pr}(x)-T \operatorname{pr} \operatorname{star});
163
        -(P_c_hat(x)-P_c(t,x))];
164
165
   M=blkdiag(diag(V_e), diag(V_n), diag(J_pr), diag(beta_pr), diag(
166
       beta_c), eye(n_c), eye(n_c), diag(beta_pr_hat), diag(gamma_c));
167
   options=odeset('Mass',M);
168
169
   %% Initial conditions:
170
171
   T e0=20*ones(n e, 1);
172
   T N0=20*ones(n N,1);
173
   Q_pr0=0.5*[Q_pr_star];
174
   z_pr0 = [0.0];
175
   z_c0=zeros(n_c,1);
176
   z_s0=zeros(n_c,1);
177
   p_s0=zeros(n_c,1);
178
   z pr hat0=zeros(n pr, 1);
179
   P_c_hat0=zeros(n_c,1);
180
181
   x0 = [T_0; T_N0; Q_pr0; z_pr0; z_c0; z_s0; p_s0; z_pr_hat0; P_c_hat0];
182
183
   %% The ODE solver
184
185
   [t sol, x sol]=ode15s(f, tspan, x0, options);
186
187
```

```
%% Extract the useful data
188
189
   data_T_e = [];
190
   data T N = [];
191
   data_q_ch = [];
192
   data_q_e = [];
193
   data P pr = [];
194
   data_P_c = [];
195
   data_P_c_hat = [];
196
197
   for i=1: size(t sol, 1)
198
        data_T_e(i, :) = T_e(x_sol(i, :));
199
        data_T_N(i, :) = T_N(x_sol(i, :));
200
        data_q_ch(i,:) = q_ch(x_sol(i,:));
201
        data_q_e(i, :) = q_e(x_sol(i, :));
202
        data_P_pr(i,:)=P_pr(x_sol(i,:)');
203
        data_P_c(i,:) = P_c(t_sol(i), x_sol(i,:));
204
        data_P_c_hat(i,:)=P_c_hat(x_sol(i,:)');
205
206
   end
207
208
209
   %% Plots
210
211
   figure()
212
   plot (t_sol/3600,data_T_e)
213
   ylabel('$T_{\mathrm{e}}$ ($^\circ$C)', 'Interpreter', 'Latex');
214
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
215
   grid off
216
   lgd=legend({ 'edge 1', 'edge 2', 'edge 3', 'edge 4'}, 'Interpreter', '
217
      Latex');
   lgd.FontSize=8;
218
   lgd.Location='NorthEast';
219
220
   figure()
221
   plot (t_sol/3600,data_T_N)
222
```

```
ylabel('$T_{\mathrm{N}}$ ($^\circ$C)', 'Interpreter', 'Latex');
223
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
224
   grid off
225
   lgd=legend({ 'node 1', 'node 2', 'node 3', 'node 4'}, 'Interpreter', '
226
      Latex');
   lgd.FontSize=8;
227
   lgd.Location='NorthEast';
228
229
   figure()
230
   plot (t sol/3600, data q e(:, 1), t sol/3600, data q e(:, 2))
231
   ylabel('$q {\mathrm{pr}}$ (m$^3$/s)', 'Interpreter', 'Latex');
232
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
233
   grid off
234
   lgd=legend({ '(edge 1)', 'edge 2'}, 'Interpreter', 'Latex');
235
   lgd.FontSize=8;
236
   lgd.Location='NorthEast';
237
238
   figure()
239
   plot (t_sol/3600,data_q_ch)
240
   ylabel('$q_{\mathrm{ch}}$ (m$^3$/s)', 'Interpreter', 'Latex');
241
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
242
   grid off
243
   lgd=legend({ 'producer 2 (edge 2) ', 'consumer 1 (edge 3) ', 'consumer
244
       2 (edge 4)'}, 'Interpreter', 'Latex');
   lgd.FontSize=8;
245
   lgd.Location='NorthEast';
246
247
   figure()
248
   plot (t_sol/3600,data_P_pr)
249
   ylabel('$P_{\mathrm{pr}}$ (W)', 'Interpreter', 'Latex');
250
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
251
   grid off
252
   lgd=legend({ 'producer 1 (edge 1) ', 'producer 2 (edge 2) '}, '
253
      Interpreter ', 'Latex ');
  lgd.FontSize=8;
254
  lgd.Location='NorthEast';
255
```
```
256
   figure()
257
   plot(t_sol/3600,data_P_c_hat)
258
   ylabel('$P_{\mathrm{c}}$ (W)', 'Interpreter', 'Latex');
259
   xlabel('$t$~(hour)', 'Interpreter', 'Latex');
260
   grid off
261
   lgd=legend({ 'consumer 1 (gets bio) ', 'consumer 2 (gets ego) '}, '
262
      Interpreter ', 'Latex ');
   lgd.FontSize = 8;
263
   lgd.Location='NorthEast';
264
265
   %% Save the relevant data.
266
267
   delete data_sim.mat
268
   save('data_sim.mat')
269
   disp('The data from the simulations has been correctly saved.')
270
```