Strategies for co-operated wood chip fired and municipal waste fired combined heat and power plants

Alexander Taylor
Abstract

The Brista 1 plant is a wood chip-fired combined heat and power (CHP) plant located near Märsta, northwest of Stockholm, Sweden. The primary purpose of the plant is to supply heat to the northwest district heating grid. In order to meet increasing demand for district heating, Fortum Heat is constructing a second CHP plant next to Brista 1. The Brista 2 plant will use a mixture of municipal and industrial waste as fuel.

Due to changes in the European Green Certificate program, the fuel subsidies for wood chips will be significantly reduced. This will cause the Brista 1 plant to incur significantly increased operating costs. The Brista 2 plant, however, will not be affected by these changes and will therefore be much cheaper to run than Brista 1. However, due to the large demand for district heating it will be necessary to run both plants in parallel at certain times in order to meet the heating demand and/or maximize revenue during periods of high electricity demand.

A computer program has been constructed using MATLAB which simulates the Brista 1 and 2 plants and their combined operation in both backpressure and direct condensing mode. The results show that the optimum allocation of heat production does not seem to be affected by electricity price assuming both plants are operated in backpressure mode. The reason for this would seem to be that the production costs (fuel, emissions, O&M) are unaffected by the electricity price. Therefore, the allocation which maximizes electrical power production, and thus revenue from electricity sales, will always be favored.

In certain cases, it is more profitable to run the Brista 1 plant in direct condensing mode. The reason for this would seem to be that the thermal efficiency is somewhat higher, and that at low electricity prices the revenues from electricity sales do not offset the cost of the reduced heat production.

Keywords: combined heat and power, CHP, co-operated power plants, municipal waste, wood chips, operational strategy, MATLAB
# Table of Contents

Abstract .......................................................................................................................... 2

1 Glossary of terms ....................................................................................................... 5

2 List of figures ............................................................................................................. 6

3 List of variables ......................................................................................................... 7

4 Introduction ................................................................................................................ 8

4.1 Background ............................................................................................................. 8

4.2 Method .................................................................................................................... 8

4.3 Objectives ............................................................................................................... 8

4.4 Constraints ............................................................................................................ 8

5 The Plants .................................................................................................................. 9

5.1 Brista 1 .................................................................................................................. 9

5.1.1 Overview .......................................................................................................... 9

5.1.2 Plant layout ...................................................................................................... 9

5.1.3 Operating modes ............................................................................................. 10

5.1.4 Operating range ............................................................................................... 11

5.2 Brista 2 .................................................................................................................. 11

5.2.1 Overview .......................................................................................................... 11

5.2.2 Plant layout ...................................................................................................... 11

5.2.3 Operating modes ............................................................................................. 12

5.2.4 Operating range ............................................................................................... 12

6 Process modeling ........................................................................................................ 13

6.1 Mathematical modeling ....................................................................................... 13

6.1.1 Boiler thermal power output and fuel input ..................................................... 13

6.1.2 Energy and mass balances ............................................................................. 13

6.1.3 Approximating temperatures and pressures .................................................... 20

6.1.4 Economic modeling ......................................................................................... 21

6.2 Computer model ................................................................................................... 22

6.2.1 Programming approach .................................................................................. 23

6.2.2 Structure of the computer program ................................................................. 24

7 Results ....................................................................................................................... 26

8 Conclusion .................................................................................................................. 35

8.1 Recommendations ............................................................................................... 35

9 Discussion ................................................................................................................... 36

10 Bibliography ............................................................................................................ 37

11 Appendix 1: MATLAB Code .................................................................................. 38
11.1 Notes on MATLAB code ...........................................................................................................38
11.2 Economics.m ..........................................................................................................................38
11.3 B1_BP.m .................................................................................................................................43
11.4 B1_DC.m .................................................................................................................................47
11.5 B2_BP.m .................................................................................................................................50
11.6 ComputeFlueGas.m ..................................................................................................................53
11.7 ComputeEnthalpies_BP.m ........................................................................................................55
11.8 ComputeEnthalpies_DC.m ........................................................................................................56
11.10 ComputeEnthalpies_BP_B2.m ...............................................................................................57
# 1 Glossary of terms

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>B1</td>
<td>Brista 1 Combined Heat and Power Plant</td>
</tr>
<tr>
<td>B2</td>
<td>Brista 2 Combined Heat and Power Plant</td>
</tr>
<tr>
<td>BP</td>
<td>Backpressure mode</td>
</tr>
<tr>
<td>CFB</td>
<td>Circulating Fluidized Bed</td>
</tr>
<tr>
<td>CHP</td>
<td>Combined Heat and Power</td>
</tr>
<tr>
<td>DC</td>
<td>Direct Condensing mode</td>
</tr>
<tr>
<td>DHW</td>
<td>District Heating Water</td>
</tr>
<tr>
<td>FGC</td>
<td>Flue Gas Condensation</td>
</tr>
<tr>
<td>HHV</td>
<td>Higher Heating Value (of a given fuel)</td>
</tr>
<tr>
<td>O&amp;M</td>
<td>Operations &amp; Maintenance</td>
</tr>
</tbody>
</table>
2 List of figures

Figure 1: Basic layout of Brista 1 plant in BP mode with FGC ................................................................. 9
Figure 2: Basic layout of Brista 1 plant in DC mode with FGC ............................................................... 10
Figure 3: Basic layout of Brista 2 plant in BP mode with FGC ................................................................. 11
Figure 4: Brista 1 in BP mode with stream numbers denoted for mass and energy balances .................. 14
Figure 5: Brista 1 in DC mode with stream numbers denoted for mass and energy balances .................. 15
Figure 6: Brista 2 in BP mode with stream numbers denoted for mass and energy balances .................. 16
Figure 7: Hierarchical diagram of the computer model including the dependencies of each component ..... 24
Figure 8: Grid load 80 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, high electricity price 27
Figure 9: Grid load 80 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, low electricity price 27
Figure 10: Grid load 80 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, high electricity price 28
Figure 11: Grid load 80 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, low electricity price 28
Figure 12: Grid load 100 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, high electricity price 29
Figure 13: Grid load 100 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, low electricity price 29
Figure 14: Grid load 100 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, high electricity price 30
Figure 15: Grid load 100 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, low electricity price 30
Figure 16: Grid load 120 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, high electricity price 31
Figure 17: Grid load 120 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, low electricity price 31
Figure 18: Grid load 120 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, high electricity price 32
Figure 19: Grid load 120 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, low electricity price 32
Figure 20: Grid load 140 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, high electricity price 33
Figure 21: Grid load 140 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, low electricity price 33
Figure 22: Grid load 140 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, high electricity price 34
Figure 23: Grid load 140 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, low electricity price 34
3 List of variables

This is a list of variables which will be used in the equations in the following chapters.

\( m \) Total steam mass flow rate
\( m_x \) Steam mass flow rate in steam extraction \( x \)
\( m_{\text{dhw}} \) Mass flow rate of district heating water
\( m_{\text{air}} \) Mass flow rate of combustion air for preheating (Brista 2)
\( h_x \) Enthalpy of stream \( x \)
\( h_{\text{in}} \) Enthalpy of district heating water entering first condenser
\( h_{\text{int}} \) Enthalpy of district heating water leaving first condenser
\( h_{\text{out}} \) Enthalpy of district heating water leaving second condenser
\( d t_x \) Temperature difference of stream \( x \)
\( c_{px} \) Specific heat of stream \( x \)
\( p_x \) Pressure at point \( x \)
\( T_x \) Temperature at point \( x \)
\( \eta_x \) Efficiency of component \( x \)
\( Q_x \) Thermal power input/output to/from component \( x \)
\( P_x \) Electrical power input/output to/from component \( x \)
\( M_x \) Molar mass of chemical species \( x \)
\( C_x \) Concentration of chemical species \( x \)
\( p_x \) Average spot price of \( x \) during a given hour
\( c_x \) Total operating cost of \( x \) during a given hour
4 Introduction

4.1 Background

The Brista 1 plant is a wood chip-fired combined heat and power (CHP) plant with a maximum thermal power output of 130 MW and a maximum electrical power output of 42 MW located near Märsta, northwest of Stockholm, Sweden. The primary purpose of the plant is to supply heat to the northwest district heating grid.

In order to meet increasing demand for district heating, Fortum Heat is constructing a second CHP plant next to Brista 1. The Brista 2 plant will have a thermal power output of 60 MW and an electrical power output of 20 MW using a mixture of municipal and industrial waste as fuel.

Due to changes in the European Green Certificate program, the fuel subsidies for wood chips will be significantly reduced. This will cause the Brista 1 plant to incur significantly increased operating costs. The Brista 2 plant, however, will not be affected by these changes and will therefore be much cheaper to run than Brista 1. However, due to the large demand for district heating it will be necessary to run both plants in parallel at certain times in order to meet the heating demand and/or maximize revenue during periods of high electricity demand.

In order to run the plants in parallel as profitably as possible, Fortum Heat have expressed interest in modeling the two power plants and conducting computer simulations in order to formulate an operational strategy, taking part-load operation and the costs related to starting and stopping the plants into consideration.

4.2 Method

A computer model has been programmed based on the operating characteristics of the two CHP plants. The computer model takes production costs and potential revenues from electricity production into account and optimizes the parallel operation of the two plants based on multiple constraints for different likely operating scenarios.

4.3 Objectives

This project has studied the parallel operation of the Brista 1 and Brista 2 CHP plants from a production cost perspective. This involved thermodynamic and economic modeling of the two plants and programming an optimization program to determine the best possible operating strategy for their parallel operation.

4.4 Constraints

The modelling has been limited to the Brista 1 and Brista 2 plants. The rest of the northwestern district heating grid is not a part of the model.
5 The Plants

5.1 Brista 1

5.1.1 Overview

The Brista 1 plant features a CFB boiler fueled by wood chips with a rated power output of 122 MW. This produces steam at a temperature of 540°C and a pressure of 140 bar with a maximum steam flow rate of 50 kg/s. The steam is fed through high- and low pressure turbine stages which both have multiple steam extractions. The high-pressure steam extractions are used for feedwater preheating, process steam and boiler inlet preheating. The low-pressure steam extractions are used for preheating condensate and to provide steam for two separate condensers which operate at different pressures.

The plant has also been retrofitted with a flue gas condensation unit, which provides an additional 30 MW of thermal power output at maximum steam load.

5.1.2 Plant layout

5.1.2.1 Backpressure mode

![Diagram of Brista 1 plant in BP mode with FGC](image)

Figure 1: Basic layout of Brista 1 plant in BP mode with FGC
5.1.2.2 **Direct condensing mode**

![Diagram of the Brista 1 plant in DC mode with FGC](image)

**Figure 2:** Basic layout of Brista 1 plant in DC mode with FGC

### 5.1.3 Operating modes

The plant can be run in three different modes – backpressure mode, direct condensing mode and mixed mode. In backpressure mode, the steam from the boiler is fed through both turbines and then passes through condenser 1 and 2. The condensers operate at two different pressure levels, and for this reason the steam reaching condenser 2 is extracted from the low-pressure turbine prior to full expansion. In this mode, the plant produces both electricity and district heating.

In direct condensing mode, the steam from the boiler completely bypasses the turbines and is fed into condenser 3. Before it can be fed into the condenser, however, the steam is expanded to a lower pressure and water is injected in order to lower the temperature to an acceptable level for condenser operation. This water is extracted between two stages in the feedwater pump at a pressure of around 40 bar. The increased mass flow of lower-pressure steam creates a larger mass flow of steam through the condenser than in backpressure mode. In this mode, the plant produces no electricity but produces significant amounts of district heating.

In mixed mode, part of the steam is fed through the turbines and part is fed directly to condenser 3. This can be viewed as a hybrid of the two previously described operating modes. In this mode, the plant produces both electricity and district heating; however, the electricity production is lower than it would be in backpressure mode.

Additionally, all of the above modes can be run with or without the flue gas condensation module. In the Brista 1 plant, the flue gas condensation and flue gas cleaning systems are separate.
5.1.4 Operating range

The maximum plant thermal power output is 113 MW and the minimum stable thermal power output is 43 MW. In addition, condenser 3 is limited to a thermal power output of 85 MW and the two turbines have a combined maximum electrical power output of 42 MW and a minimum stable electrical power output of 12.6 MW. The flue gas condensation system has a maximum thermal power output of 30 MW and a minimum of 15 MW.

5.2 Brista 2

5.2.1 Overview

The Brista 2 plant features a grate fired boiler fueled by a mix of industrial and municipal waste with a rated power output of 80 MW. This produces steam at a temperature of 415°C and a pressure of 60 bar with a maximum steam flow rate of 34 kg/s. The steam is fed through a single turbine which has up to 4 steam extractions depending on operating mode. The high-pressure steam extractions are used for combustion air preheating and boiler feedwater preheating. The low-pressure steam extractions are used to provide steam for two separate condensers which operate at different pressures. The plant also features an integrated flue gas condensation unit which provides an additional 12 MW of thermal power output at maximum steam load.

5.2.2 Plant layout

Figure 3: Basic layout of Brista 2 plant in BP mode with FGC
5.2.3 Operating modes

The Brista 2 plant can, similarly to the Brista 1 plant, be run in three different modes – backpressure mode, direct condensing mode and mixed mode. In backpressure mode, the steam from the boiler is fed through the turbine and then passes through condenser 1 and 2. Steam is extracted at a higher pressure for condenser 2, in order to facilitate the two-stage heating process. In this mode, the plant produces both electricity and district heating.

In direct condensing mode, the steam from the boiler completely bypasses the turbines and is fed into condenser 3. Before it can be fed into the condenser, however, the steam is expanded to a lower pressure and water is injected in order to lower the temperature to an acceptable level for condenser operation. This water is extracted between two stages in the feedwater pump at a pressure of around 40 bar. The increased mass flow of lower-pressure steam creates a larger mass flow of steam through the condenser than in backpressure mode. In this mode, the plant produces no electricity but produces significant amounts of district heating.

In mixed mode, part of the steam is fed through the turbines and part is fed directly to condenser 3. This can be viewed as a hybrid of the two previously described operating modes. In this mode, the plant produces both electricity and district heating; however the electricity production is lower than it would be in backpressure mode.

The flue gas condensation unit in the Brista 2 plant is integrated into the flue gas cleaning system; this means that it cannot be disconnected. All of the above modes must be run with flue gas condensation.

5.2.4 Operating range

The maximum boiler thermal power output is 80 MW, and the minimum output for stable operation is 20 MW. The turbine has a maximum electrical power output of 26 MW and a minimum stable electrical power output of 4 MW.
6 Process modeling

In order to keep the size of the project manageable, it was decided that the only operating modes that should be considered are:

1) Brista 1, Backpressure mode
2) Brista 1, Direct Condensing mode
3) Brista 2, Backpressure mode

The reason for this was that mixed mode was found to be very hard to model and that running Brista 2 in direct condensing mode was determined to be an unlikely scenario.

6.1 Mathematical modeling

6.1.1 Boiler thermal power output and fuel input

Since the total district heating load is an input parameter, the required heat output of each plant is known beforehand. Using a known boiler efficiency and heating value for the fuel, it is therefore possible to compute the necessary boiler thermal power output for a given steam flow independently of the mass/energy balances for the steam system. This, in turn, allows the fuel flow to be computed using the higher heating value (HHV).

Using arbitrary condenser and boiler efficiencies $\eta_{\text{cond}}$ and $\eta_{\text{boil}}$ the boiler thermal power output can be computed as follows:

$$Q_{\text{boil}} = \frac{1}{\eta_{\text{cond}}} \cdot m \cdot (h_{\text{boil, out}} - h_{\text{boil, in}})$$  \hspace{1cm} (1)

Where $m$ is the total steam mass flow rate and $h_{\text{out}}$ and $h_{\text{in}}$ are the enthalpies at boiler outlet and inlet, respectively. The required thermal energy input from the fuel can be computed using the following equation.

$$Q_{\text{fuel}} = \frac{1}{\eta_{\text{boil}}} Q_{\text{boil}}$$  \hspace{1cm} (2)

The corresponding fuel mass flow rate can be determined using the fuels higher heating value.

$$m_{\text{fuel}} = \frac{Q_{\text{fuel}}}{HHV_{\text{fuel}}}$$  \hspace{1cm} (3)

6.1.2 Energy and mass balances

Based on the flow sheets illustrated in Figure 1, Figure 2 and Figure 3, sets of mass and energy balances were constructed for each operating mode. The purpose of these balances is to compute the steam flow in each branch of the system, i.e. the size of the steam extractions. This is necessary in order to determine the turbine power output and the thermal power output from the condensers. The thermal power contribution from the flue gas condensation system is treated independently of these mass/energy balances. This is explained further in section 6.1.2.4.

The number of energy and mass balance equations for each operating mode is kept at a minimum in order to ensure that the equation systems they produce are as small as possible. The reason for this is that the
A computer program which solves the equation systems does so iteratively, so larger systems will result in significantly longer computation times.

### 6.1.2.1 Brista 1, Backpressure mode

![Diagram of Brista 1 in BP mode with stream numbers denoted for mass and energy balances]

**Figure 4:** Brista 1 in BP mode with stream numbers denoted for mass and energy balances

#### Energy balances over preheaters:

\[
(h_{1a} - h_{1aa})m_{1a} = (h_8 - h_7)m \quad (4)
\]

\[
(h_{1aa} - h_{1ab})m_{1a} = (h_7 - h_6)m \quad (5)
\]

#### Energy balances over condensers:

\[
(h_{2b} - h_{4a})m_{2b} = (h_{out} - h_{int})m_{dhw} \quad (6)
\]

\[
(h_3 - h_{4b})m_3 = (h_{int} - h_{in})m_{dhw} \quad (7)
\]

#### Mass balances over entire system:

\[
m_{1a} + m_{1b} + m_{1c} + m_2 = m \quad (8)
\]

\[
m_{2a} + m_{2b} + m_3 = m_2 \quad (9)
\]
When Brista 1 is run in backpressure mode at low boiler loads, a situation can occur where one of the steam extractions must be shut off. In this case, an energy balance over the feedwater tank is also used:

**Energy balance over feedwater tank:**

\[ m_{1a}h_{1ab} + m_{1b}h_{1ab} + m_{1c}h_{1c} + m_{2b}h_{4a} + m_{3b}h_{4b} + m_{2a}h_{2a} = mh_a \]  

(10)

The set of energy and mass balance equations form a linear equation system which can be rewritten in matrix form and solved using Gaussian elimination.

### 6.1.2.2 Brista 1, Direct Condensing mode

In direct condensing mode, the mass and energy balances are significantly simpler than in backpressure mode. This is because there is only one condenser in direct condensing mode and because there is no turbine work there are no steam extractions at varying pressures. The complicating factor when modeling the direct condensing mode is the water injection before the condenser and the feedwater tank. In order to solve this problem, we disregard the water injection during the calculations.

The purpose of the water injection is to reduce the pressure of the water from the boiler to a level which the condenser can handle. Effectively, we obtain a significantly larger mass flow rate of steam at a much lower pressure. This is due to practical considerations such as material fatigue in the condenser. However, from an energy standpoint, a smaller mass flow rate of higher pressure steam contains the same amount of energy as a larger mass flow rate of lower pressure steam. Reducing the pressure simply spreads the same amount of energy throughout a larger volume of water.

Therefore, in order to simplify the simulation, we can disregard the water injection and simply view the amount of energy in the high pressure steam as equal in magnitude. This allows us to remove the entire water injection loop from the equation. The model will not be able to solve the actual mass flow rate through the condenser, but it will be able to solve the heat flow rate through the condenser accurately.

![Figure 5: Brista 1 in DC mode with stream numbers denoted for mass and energy balances](image_url)
Using this approach requires an energy balance over the entire system and energy balances over the condenser and feedwater tank.

**Energy balance over system**

\[ m_2 h_1 + m_3 h_1 = m_1 h_1 \]  \hspace{1cm} (11)

**Energy balance over condenser**

\[ m_2 = \frac{m_{dhw} \cdot (h_{out} - h_{in})}{(h_1 - h_4)} \]  \hspace{1cm} (12)

**Energy balance over feedwater tank**

\[ m_2 h_1 + m_3 h_4 = m h_5 \]  \hspace{1cm} (13)

### 6.1.2.3 Brista 2, Backpressure mode

In the Brista 2 plant, steam extractions are used to preheat the combustion air entering the boiler before preheating the boiler feedwater. The thermal energy lost through this process must be incorporated into the energy balances. The inlet and outlet temperature of the combustion air is known, and the mass flow rate of air can be approximated using least-squares interpolation from known data points, creating a function which is dependent on the steam mass flow rate. The specific heat of the air is assumed to be the average value between the inlet and outlet air temperatures.

![Diagram of Brista 2 in BP mode](image)

**Figure 6:** Brista 2 in BP mode with stream numbers denoted for mass and energy balances
For the Brista 2 plant in backpressure mode, a mass and energy balance over the entire system and energy balances over the air preheaters and condensers are used.

**Mass balance over system**

\[(m_{1a} + m_{1b}) + m_{1c} + m_2 = m_1\]  \(14\)

**Energy balance over system**

\[(m_{1a} + m_{1b}) \cdot (h_{1a} + h_{1b}) + m_{1c}h_{1c} + m_2h_2 + P_f = m_1h_1\]  \(15\)

**Energy balance over air preheaters**

\[(m_{1a} + m_{1b}) \cdot ((h_{1a} - h_{1aa}) + (h_{1b} - h_{1ab})) = m_{air} (c_{pair} \cdot dt_{air}) + m \cdot (h_6 - h_5)\]  \(16\)

**Energy balances over condensers**

\[(h_{1c} - h_{3a})m_{1c} = (h_{out} - h_{int})m_{dhw}\]  \(17\)

\[(h_2 - h_{3b})m_2 = (h_{int} - h_{in})m_{dhw}\]  \(18\)

These equations form a linear equation system which can be solved using conventional techniques, for example Gaussian elimination.

### 6.1.2.4 Condenser thermal power output

The thermal power output is computed using an energy balance over the condenser(s) after the mass and energy balances have been solved for. The reason for this is that all of the steam extractions have to be taken into account in order for the steam mass flow rate over the condenser(s) to be known.

For Brista 1 operating in backpressure mode, the thermal power output is computed as:

\[Q_{cond1} = m_{2c}(h_{2b} - h_{4a})\]  \(19\)

\[Q_{cond2} = m_3(h_3 - h_{4b})\]  \(20\)

For Brista 1 in direct condensing mode, the thermal power output is computed as:

\[Q_{cond3} = m_3(h_3 - h_{4})\]  \(21\)

And for Brista 2 in backpressure mode, the thermal power output is computed as:

\[Q_{cond1} = m_{1c}(h_{1c} - h_{3a})\]  \(22\)

\[Q_{cond2} = m_3(h_3 - h_{3b})\]  \(23\)

In order to compute the total thermal power output of the plant, the contribution from the flue gas condensation has to be added. This computation is performed independently, see section 6.1.2.6.
6.1.2.5 Turbine electrical power output

The turbine electrical power output is computed using an energy balance over the turbine(s). Since the turbine(s) feature multiple steam extractions, these must be taken into account as they effectively reduce the mass flow through certain sections of the turbine and contribute to a reduced power output.

Since Brista 1 features two turbines – one high pressure and one low pressure – it requires two separate energy balances.

\[ P_{HP} = \eta_T(m_1(h_1 - h_{1a}) + (m_1 - m_{1a})(h_{1a} - h_{1b}) + (m_1 - m_{1a} - m_{1b})(h_{1b} - h_{1c}) + (m_1 - m_{1a} - m_{1b} - m_{1c})(h_{1c} - h_2)) \]

\[ P_{LP} = \eta_T(m_2(h_2 - h_{2a}) + (m_2 - m_{2a})(h_{2a} - h_{2b}) + (m_2 - m_{2a} - m_{2b})(h_{2b} - h_{2c}) + (m_2 - m_{2a} - m_{2b} - m_{2c})(h_{2c} - h_3)) \]

\[ P_T = P_{HP} + P_{LP} \]

The Brista 2 plant features a single turbine with multiple steam extractions. The energy balance is as follows:

\[ P_T = \eta_T(m_1(h_1 - h_{1a}) + (m_1 - m_{1a})(h_{1a} - h_{1b}) + (m_1 - m_{1a} - m_{1b})(h_{1b} - h_{1c}) + (m_1 - m_{1a} - m_{1b} - m_{1c})(h_{1c} - h_2)) \]

6.1.2.6 Combustion and flue gas condensation

The flue gas condensation was treated as an entirely separate system. Since the boiler steam load is used as a control variable, it is known before the steam system balances have been solved. This means that the boiler power output can also be computed independently of the steam system mass/energy balances. Since the boiler power output can be computed, the fuel flow rate and corresponding flue gas flow can also be computed.

Once the flue gas flow rate has been computed, the amount of heat available in it can also be computed, and the corresponding flow rate of district heating water which can be heated to the desired temperature can be computed. This value can then be deducted from the total required flow rate of district heating water, and the remainder used in the steam system mass/energy balances. This approach avoids nonlinearities which would otherwise be present and allows the economic benefits of the flue gas condensation system to be taken into account.

The total district heating water flow rate is effectively split into two parts – one which is covered by the thermal energy in the flue gases and one which is covered by the combined thermal power output of the condensers.

Given a dry, ash-free chemical composition and the moisture content of the fuel in question, the flue gas flow rate can be computed using the following method.

The dry weight percentage of a chemical species \( X \) is first converted to a wet-weight percentage by multiplying it by the moisture content.

\[ X_{dry \cdot moist} = X_{wet} \]
The number of moles per kg fuel of each species can be computed using their respective molar masses.

\[ X \left[ \frac{mol}{kg \ fuel} \right] = 1000 \cdot X_{wet} \cdot \frac{1}{M_x} \quad (29) \]

Each chemical species requires a different amount of oxygen to combust. These are computed based on the following combustion reactions:

\[ C + O_2 \rightarrow CO_2 \quad (30) \]

\[ H_2 + \frac{1}{2} O_2 \rightarrow H_2O \quad (31) \]

\[ S + O_2 \rightarrow SO_2 \quad (32) \]

Nitrogen present in the fuel is considered inert and oxygen present in the fuel gives a corresponding negative contribution to the total oxygen demand. The total oxygen demand is computed as the sum of each chemical species individual demand. The amount of flue gases produced is computed based on the same reactions, with the fuel moisture content added to the total water and the nitrogen present in the combustion air added to the total nitrogen.

The final step is to use the excess air factor and add the corresponding amount of oxygen and nitrogen to the flue gas composition. Thus, the total gas amount is:

\[ TG = H_2O + CO_2 + N_2 + SO_2 + N_{2(excess)} + O_{2(excess)} \quad (33) \]

This is the total amount of moles of flue gas produced per kg fuel. In order to compute the mass flow rate of the fuel, this must be converted from normal cubic meters \((Nm^3)\) into kilograms.

\[ m_{flue} = 0.0224 \cdot TG \quad (34) \]

This value represents the amount of flue gases produced per kilogram of fuel burned. It is therefore multiplied by the fuel flow to give the total flow rate of flue gases at a given operating point.
6.1.3 Approximating temperatures and pressures

Both of the plants and their operating modes were modeled using an approach based on energy- and mass balances combined with extrapolation of known operational data. Energy and mass balances were constructed for each operating mode and turned into a linear equation system. However, certain parameters within the model are dependent on external influences such as the temperature difference of the district heating water and the boiler steam load. These parameters affect the pressures and temperatures of the streams within the plant.

In order to account for this, several data points were obtained from the operational data of the Brista 1 plant, several years of which was made available by Fortum. For the Brista 2 plant, design estimates were obtained from the contractors and manufacturers responsible for the construction of the power plant components.

Using these data points and a known quantity such as the boiler steam load or the district heating water temperatures, least-squares approximation can be used to construct functions which compute the unknown quantity (pressures, temperatures) as a function of the known quantities (boiler steam load, DHW temperature). Using quadratic least-squares approximation, this gives an equation of the form:

\[ p = ax^2 + bx + c \]  
(35)

Since several of the pressures and temperatures are dependent on both steam load and district heating temperature, it is necessary to construct equations which take this multiple-variable dependency into account. Assuming that the quantities follow a somewhat “smooth” relationship, we can use a two-step quadratic least squares approximation. Effectively, first constructing an equation of the type described above for one of the dependencies (either steam load or district heating temperature) and then performing the same type of quadratic least squares approximation on each of the factors \( a, b \) and \( c \). This yields a set of 4 equations which need to be solved in order to determine the pressure or temperature of interest at the current load point and district heating temperature.

The first three equations are used to determine the factors \( a, b \) and \( c \) in the final equation. Denoting the steam mass flow rate as \( m \) and the district heating temperature difference as \( dt \) yields:

\[ a = a_1 m^2 + b_1 m + c_1 \]  
(36)

\[ b = a_2 m^2 + b_2 m + c_2 \]  
(37)

\[ c = a_3 m^2 + b_3 m + c_3 \]  
(38)

\[ p = a(dt)^2 + b(dt) + c \]  
(39)

Where \( p \) is the pressure or temperature of interest. The factors \( a_{1-3}, b_{1-3} \) and \( c_{1-3} \) are determined using the known data points, and thus this system can be solved rapidly in order to determine the correct pressures and temperatures in each branch of the system.

The method is of course sensitive to input data, but if it can be reasonably assumed that any operating point to be simulated falls within the range specified by the data used to construct the equations, then it would seem reasonable to assume that the equations will return a valid result since this will effectively be a case of interpolation rather than extrapolation.
6.1.4 Economic modeling

The economic modeling is done by taking the results of the thermodynamic computations - the required fuel input, the resulting flue gas flow, the electrical power output from the turbine(s) and the internal power consumption – and computing their related costs. This requires fuel prices for both plants, electricity price, average NOx emissions per Nm³ flue gas and emission tariffs for said emissions to be known. For reasons of corporate confidentiality, the values used in the simulations are not presented here, but the general method through which the total cost of operation can be computed is.

The cost is computed in the unit kr/h. The reason for this is that the price of electricity tends to fluctuate, so the output of the plants may need to be adjusted on an hourly basis in order to compensate for this and maintain profitability.

Any and all taxes and other applicable fees for each term are incorporated into the spot price in this model. This is done in order to reduce the number of variables required for the equations. This means, for example, that taxes related to the electricity grid such as grid fees are incorporated into the electricity price, while for example CO₂ taxes are incorporated into the fuel price.

6.1.4.1 Electricity revenue

In the computer model, we are interested in minimizing the cost of operation. Due to this, the revenue from sales of electricity is viewed as a negative cost. Assuming that the spot price of electricity on the market is given in the unit kr/MWh electrical power output and that both the electrical power output of the turbine(s) and the internal electrical power consumption is given in the unit MW, the revenue from electricity sales can be computed as follows:

\[ p_{el}P_{int} - p_{el}P_T = c_{el} \]  
(40)

In the case of BP operation, this term will always be negative, as the turbine power output is greater than the internal power consumption. In the case of DC operation, this term will be positive as the turbine power output is zero. The internal electrical power consumption is computed based on a quadratic least-squares approximation which is based on operational data from the Brista 1 plant. In the case of combined operation where one of the plants operates in direct condensing mode, the other plant could potentially cover the internal electricity consumption of both plants.

6.1.4.2 Fuel costs

Assuming a fuel price given in the unit kr/MWh fuel power input and a fuel power input given in the unit MW, the cost of fuel for a given hour can be computed as:

\[ p_{fuel}Q_{fuel} = c_{fuel} \]  
(41)

This term will always be positive, as fuel is always required in order for the plant to operate.

6.1.4.3 Emissions

The computer model only takes NOx emissions into account. This is done using an average yearly value for NOx emissions per kg of flue gas measured in the stack of the Brista 1 plant. Assuming such a value with the units mg/kg flue gas, an associated NOx tax (cost) in the units kr/kg and a flue gas flow rate in the units kg/s, the following expression is obtained:

\[ m_{flue} \cdot (C_{NOx} \cdot 10^{-6}) \cdot p_{NOx} \cdot 3600 = c_{NOx} \]  
(42)
6.1.4.4 Operations & Maintenance

The O&M cost is given in the units kr/MWh thermal power output and includes salaries for maintenance personnel, cost of replacement parts, insurance and all other similar costs. To compute the total cost of O&M, the following equation is used:

\[ p_{O&M}Q_{\text{cond}} = c_{O&M} \]  

(43)

6.1.4.5 Total cost of operations

Putting all of these equations together yields the equation for total cost of operation.

\[ p_{el}P_{\text{int}} - p_{el}P_{e} + p_{\text{fuel}}Q_{\text{fuel}} + m_{\text{flue}} \cdot (C_{N_{\text{OX}}} \cdot 10^{-6}) \cdot p_{NO_{\text{X}}} \cdot 3600 + p_{O&M}Q_{\text{cond}} = c_{\text{tot}} \]  

(44)

The total cost of operations is given in the unit kr/h.

6.2 Computer model

Based on the equations and mathematical methods described in the previous section, a computer model was constructed using the MATLAB programming language. This language was chosen mostly because of my own familiarity with it, but also because of its excellent support and simple syntax for vector- and matrix operations which make the economic calculations simple and fast. Furthermore, the XSteam library, a freeware MATLAB and MS Excel implementation of the IAPWS IF97 steam and water properties for industrial use (www.x-eng.com, 2012), makes the computation of pressures, temperatures and enthalpies simple and straightforward.

Previous research (Lahdelma & Rong, Efficient algorithms for combined heat and power production planning under the deregulated electricity market, 2007) suggests that several CHP plants operating in parallel have an “operational envelope”, i.e. a range of possible heat and electricity production values with associated production costs that can be visualized as a 3D figure. Essentially, one constructs a function which computes the cost of production. This function is dependent on two variables – the heat and electricity production of the plants. Constraints are then applied to this function such as the minimum and maximum production of each plant. In order to find the minimum production cost, one simply follows the edge of the function and its given constraints and eventually the minimum will be found, as it must lie on the edge of the function space. This is an implementation of the Simplex algorithm for mathematical optimization.

This method was found to be unnecessarily complex, since it is designed for problems with a larger number of production sites and a varying grid load. However, the idea of constructing an operational envelope was used in constructing this computer model.

In essence, the purpose of the computer model is to first compute every possible combination of B1 and B2 that gives a certain user-defined total thermal power output. Once this is done, the mass and energy balances for each of these combinations are solved. These balances yield the thermal power output, electrical power output, fuel flow and flue gas flow of the plants. Once these are known, the total cost of operating the plants in each combination can be computed.

The operational cost is of course contingent on the electricity price, however, since the mass and energy balances are not dependent on the electricity price, they need only be solved once. Afterwards, the operating costs can be solved for any desired range of electricity prices.
6.2.1 Programming approach

The computer model was designed using a modular approach since it was not clear at the start of the project how many operating modes would need to be incorporated. Programming each different operating mode as a separate module made the program scalable and easier to troubleshoot.

Based on experiences from previous research (Lahdelma & Rong, An efficient linear model and optimization algorithm for multi-site combined heat and power production, 2006), it was initially thought that a significant amount of optimization programming would be required in order to reduce the amount of computations needed to solve the model. However, this research focused on much larger district heating grids with many more production sites.

Other sources (Dotzauer, 2002) suggested that for a small number of production sites it may be possible to solve the system iteratively within an acceptable timeframe. The reasoning behind this was that the purpose of optimization programming is simply to reduce the computation time. If the computation time required to iteratively solve for all possible solutions is not unreasonably long, then there is no reason to use an optimization algorithm.

While developing the computer model for the Brista 1 plant, the computation time was not particularly long. Therefore, it was found to be unnecessary to use any optimization algorithms – an iterative approach was sufficiently fast, assuming that the chosen step sizes are reasonably large.
6.2.2 Structure of the computer program

The computer program is constructed modularly using a number of functions. These functions are called in a hierarchical fashion. The top-level program is Economics, which calls the simulation programs for each individual operating mode (B1_BP, B1_DC and B2_BP). These, in turn, call other functions which compute the relevant temperatures, pressures, enthalpies, flue gas condensation data, turbine efficiencies and district heating water flow rates. Some of these functions, in turn, call the XSteam library in order to compute enthalpies, temperatures and pressures based on other inputs. This calling sequence is illustrated in Figure 7.

Figure 7: Hierarchical diagram of the computer model including the dependencies of each component

6.2.2.1 Economics

The purpose of the main program (Economics) is data aggregation, economic calculations and data plotting. This is where the input variables are defined. The user must input a total heat load for the district heating grid, the incoming and outgoing DHW temperatures for the grid and the range of electricity prices.

Given a total heat load, the program computes which combinations of B1 and B2 will yield said total load given the minimum and maximum loads of each individual plant. The user can control which step size should be used for the heat load and the electricity price and can also control the minimum and maximum constraints for each plant.

Once the range of combinations has been computed, it is stored as a vector. The corresponding operating mode function (B1_BP, B1_DC or B2_BP) is then called iteratively for each value in this vector and the corresponding fuel flow, thermal power output, electrical power output and flue gas flow is computed.
6.2.2.2  **B1_BP, B1_DC and B2_BP**

These functions are the simulators for each individual operating mode. They all work in a similar fashion. The simulation starts at a steam mass flow rate determined by the user and computes the corresponding steam extraction flow rates, condenser thermal power output, turbine electrical power output, flue gas condensation thermal power output and fuel flow rate. If the sum of the thermal power outputs matches the desired plant load, the function terminates and returns the corresponding values to the calling program. If it does not match, the program increases the steam mass flow rate until the thermal power outputs match.

6.2.2.3  **ComputeFlueGas**

This function performs a basic combustion calculation given the fuel composition for the plant in question, and returns the corresponding flue gas flow rate.

6.2.2.4  **FindTurbineEfficiency and ComputeDHWFlow**

FindTurbineEfficiency computes the efficiency of the turbine at part load conditions. This is based on quadratic least squares interpolation of known data points for each of the plants, using the same method described in section 6.1.2.4. ComputeDHWFlow uses the same method to compute the DHW flow rate for the Brista 1 plant when it operates in backpressure mode.
7 Results

Due to the sensitive nature of certain information involved in this project, the results have been normalized. All of the graphs are plotted with the fraction of the maximum cost of operations on the X-axis and the fraction of the total heat load on the Y-axis. The red line represents the Brista 1 plant and the blue line represents the Brista 2 plant. The upper graph represents both plants working in backpressure mode, whilst the lower graph represents B1 working in direct condensing mode. A negative fractional cost implies that the plants have gone from a net loss to a net profit.

The region of interest for simulation is between the two plants combined minimum load and the two plants combined maximum load. Therefore, a starting point of 80 MW and an end point of 140 MW have been chosen, with a 20 MW step size. Furthermore, each case has been simulated with two different district heating temperature combinations (80/45°C and 55/115°C) and two different electricity prices – one high and one low.
Figure 8: Grid load 80 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, high electricity price

Figure 9: Grid load 80 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, low electricity price

The figures clearly show that at a high electricity price, it is significantly more profitable to run both plants in backpressure mode rather than running B1 in direct condensing mode. In either case, an approximately 50%-50% split between the two plants, i.e. B1 producing 40 MW and B2 producing 40 MW, is the most cost-efficient allocation of heat production.
Figure 10: Grid load 80 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, high electricity price

Figure 11: Grid load 80 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, low electricity price

There seems to be no significant difference in heat allocation due to the higher DHW temperature, and the same allocation as in the 80/45°C case is preferred. However, the potential profits at high electricity prices are significantly lower. This is likely due to the higher DHW temperature requiring a higher exit pressure from the low-pressure turbine stage, causing a lower pressure drop over the turbine. This reduces the electricity production, which in turn reduces the amount of electricity available to sell.
When the grid load increases to 100 MW the optimum allocation changes accordingly. At this grid load, a distribution of about 60% to B2 and 40% to B1 is preferred. When the electricity price is lower, it becomes significantly cheaper to run B1 in direct condensing mode.
There still seems to be no significant difference in optimum allocation depending on the DHW temperature, and the same trend holds true here. Increased DHW temperature leads to lower revenues, most likely due to reduced turbine power output. Interestingly, it is not significantly cheaper to run B1 in direct condensing mode at low electricity prices – the cost is approximately the same.
Figure 16: Grid load 120 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, high electricity price

Figure 17: Grid load 120 MW, inlet DHW temperature 45°C, outlet DHW temperature 80°C, low electricity price

At 120 MW, the optimum allocation reverts to 50%-50%. This is most likely because we approach the maximum load of B2 as defined in the simulations. The revenues at high electricity prices continue to be much higher when running both plants in backpressure mode, and the cost continues to be significantly lower at low electricity prices when running B1 in direct condensing mode.
Figure 18: Grid load 120 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, high electricity price

Figure 19: Grid load 120 MW, inlet DHW temperature 55°C, outlet DHW temperature 115°C, low electricity price

At 120 MW, the same trend which was previously only observed at 80/45°C starts to show itself at low electricity prices. Running B1 in direct condensing mode becomes less costly than running both plants in backpressure mode. The revenues at high electricity prices continue to be significantly lower at higher DHW temperatures.
At 140 MW, B2 is operating at full load but only covering about 45% of the total grid load. B1 must make up the rest. This causes profits to drop slightly from the 120 MW case at high electricity prices. At low electricity prices, direct condensing mode continues to be cheaper than backpressure mode for B1.
The same trend is observed at higher DHW temperatures. The potential profits are lower than in the 120 MW case, however, the margin is much lower at higher DHW temperatures. This means that the minimum electricity price required in order to make a profit is higher at high DHW temperatures than at low ones. This is also due to the reduced turbine power output.
8 Conclusion

The optimum allocation of heat production seems to be the same regardless of the temperatures in the district heating grid. It is most likely more dependent on the total grid load and each plant's minimum and maximum load. However, the increase in revenue is much greater when the DHW temperatures are lower. This makes sense because increasing the DHW temperature requires a larger outlet pressure from the turbine(s), which reduces the pressure drop over the turbine and thus reduces the amount of electrical power that is produced.

Furthermore, the optimum allocation does not seem to be affected by electricity price assuming both plants are operated in backpressure mode. The reason for this would seem to be that the production costs (fuel, emissions, O&M) are unaffected by the electricity price. Therefore, the allocation which maximizes electrical power production, and thus revenue from electricity sales, will always be favored.

An interesting twist here is that operating Brista 1 in direct condensing mode can be more profitable than running both plants in backpressure mode at low electricity prices. The reason for this would seem to be that the thermal efficiency is higher in direct condensing mode than backpressure mode, and that the revenues from electricity sales at low prices do not cover the difference in fuel costs between the two modes.

Since the only revenue stream in the cost calculations is the electrical power output, anything that reduces this will increase costs. Another effect of this is that the total operational cost is highly dependent on the electricity price.

At high grid loads (160 MW), there are not really that many combinations that are possible, since it is close to the maximum load of both plants. However, there is still a significant difference in operating cost between the different combinations.

8.1 Recommendations

Based on these results, the electricity production from the Brista 2 plants should be maximized at all times. The operation of the Brista 1 plant should be changed depending on the current price and price forecasts of electricity. If the electricity price is expected to be low, then it will be more profitable to run Brista 1 in direct condensing mode. Once the electricity price is predicted to increase beyond a certain point, the plant should be switched into backpressure mode.


9 Discussion

The results obtained from this computer model are only as good and accurate as the data which the simulations are based on. Even though there are several years of operational data available for the Brista 1 plant, it has mostly been run at full load in backpressure mode or mixed mode. The data on direct condensing mode and part load behavior is very scarce. Because of this, the part-load and direct condensing mode behavior has had to be extrapolated from design studies by the manufacturers of the power plants components. This is not ideal, as I would much prefer to base the simulations on actual operational data.

Furthermore, all of the data concerning the Brista 2 plant is speculative at this point, as the plant is still under construction. Once the plant comes online and actual operational data becomes available, the accuracy of this model could be improved significantly.

The modeling of the individual plants could probably be done more easily and accurately using software such as ASPEN (ASPEN Technology, 2010) or ProSim (ProSim SA, 2012), but I think the economic modeling would be a lot harder in those environments. I chose to use MATLAB in order to make the program modular and easily expandable. However, ASPEN or ProSim would likely be much better at resolving nonlinearities in the mass/energy balance, as they already have built-in optimized solvers for these types of problems. In this project, I have instead made simplifications in order to avoid the nonlinearities occurring in the first place; however, this has most likely reduced the accuracy of the model. More accurate and thorough mass and energy balances along with handling of nonlinearities could of course be incorporated into a MATLAB model as well, but it would be a much larger project.
10 Bibliography


11 Appendix 1: MATLAB Code

11.1 Notes on MATLAB code

Certain modules have been redacted from the code presented here as they contain material which is sensitive to the interests of Fortum Heat Scandinavia AB. The modules which compute pressures, temperatures and turbine efficiencies all follow the same general structure, however, which is described in section 6.1.3. Furthermore, some of the variables in the programs presented here have been left blank for the same reasons of confidentiality.

The XSteam library is not included in the appendix as it contains many thousands of lines of code. It is, however, available free of charge at (www.x-eng.com, 2012).

11.2 Economics.m

% MJ211X Degree Project in Thermal Engineering, KTH 2012
% Thesis title: "Strategies for co-operated wood chip fired and municipal
% waste fired combined heat and power plants"
% Written by Alexander Taylor 860909-7491
% Version dated: 2012-05-29
% This program was written on assignment from Fortum Heat Scandinavia AB

% This is the main control program for simulating the combined operation of
% the Brista 1 and 2 plants. It requires the following modules in the same
% folder in order to run:

% B1_BP.m
% B1_DC.m
% B2_BP.m
% ComputeDHWFlow_BP.m
% ComputeEnthalpies_BP.m
% ComputeEnthalpies_BP_B2.m
% ComputeEnthalpies_DC.m
% ComputeFlueGas.m
% ComputePressures_BP.m
% ComputePressures_BP_B2.m
% ComputeTemperatures_BP.m
% ComputeTemperatures_BP_B2.m
% FindTurbineEfficiency.m
% FindTurbineEfficiency_B2.m
% XSteam.m

% The XSteam library is used with kind permission from www.x-eng.com

% This program computes the operating costs for the combined operation of
% the two plants at a user-defined grid load and user-defined range of
% electricity prices and plots the results.

clear all
close all
clc

% Set which operating modes should be included; BP = Backpressure, DC =
% Direct condensing, B1 = Brista 1, B2 = Brista 2
BP_B1 = 1;
BP_B2 = 1;
DC_B1 = 1;
% Main program to compute economics of plant operation

% Set range of electricity prices [kr/MWh] and step size. Make sure step
% size is an even multiple of the min/max price.
el_price_min =
el_price_max =
el_step = 100;

% Define max and min heat loads for both plants
heat_min_b1 = 40000;
heat_max_b1 = 105000;

heat_min_b2 = 20000;
heat_max_b2 = 60000;

% Define price and load vectors [kr/MWh]
el = el_price_min:el_step:el_price_max;
fuel_price_B1 = 226;
fuel_price_B2 = -122;

% Define other costs
% Operation & maintenance [kr/MWh]
op_maint =

% NOx related: Year-average nox production [mg/Nm3], cost [kr/kg]
average_nox =
cost_nox =

% Compute relevant combinations of b1 and b2 that fulfill the load
% requirement given min/max constraints. Make sure heat_step is an even
% multiple of heat_min and heat_max for both plants.
heat_step = 1000;

% Vectors of all possible values of heat production for b1 and b2 given an
% arbitrary step size

heat_b1 = heat_min_b1:heat_step:heat_max_b1;
heat_b2 = heat_min_b2:heat_step:heat_max_b2;

% Set goal value for heat production - the algorithm will compute all
% combinations of b1 and b2 that return this goal value. Make sure that the
% goal value is a multiple of the heat step size!
heat_goal =

% Computes all possible combinations of b1 and b2 with the given heat step
% size that return the heat goal value. Returns these values as a 2xn
% matrix where column 1 is B1 output and column 2 is B2 output.
combinations = [];
kk = 1;
for ii = 1:length(heat_b1)
    for jj = 1:length(heat_b2)
        if (heat_b1(ii) + heat_b2(jj)) == heat_goal
            combinations(kk,1) = heat_b1(ii);
            combinations(kk,2) = heat_b2(jj);
            kk = kk + 1;
        end
    end
end

-39-
kk = kk + 1;
end
end
disp(['Current heat load yields ' num2str(length(combinations)) ' possible
combinations of B1 and B2']);
disp('Running simulations, please wait...');
fprintf('
');
costs_BP_B1 = zeros(length(combinations),length(el));
costs_BP_FG_B1 = zeros(length(combinations),length(el));
costs_DC_B1 = zeros(length(combinations),length(el));
costs_DC_FG_B1 = zeros(length(combinations),length(el));
costs_BP_B2 = zeros(length(combinations),length(el));
costs_BP_B2_tot = zeros(length(combinations),length(el));
costs_DC_B1_tot = zeros(length(combinations),length(el));
hours = zeros(2,length(el));

% Preallocate data vectors
B1_BP_data = zeros(length(combinations),5);
B1_BP_FG_data = zeros(length(combinations),5);
B1_DC_data = zeros(length(combinations),6);
B1_DC_FG_data = zeros(length(combinations),6);
B2_BP_data = zeros(length(combinations),5);

% Set relevant input data & constraints
t_dhw_in = 45;
t_dhw_out = 80;
mass_step = 1;
mass_start = 10;

% Compute power production, fuel flow, etc for each possible combination
for ii = 1:length(combinations)
    if BP_B1 == 1
        B1_BP_data(ii,:) = B1_BP(combinations(ii,1),0,mass_start,mass_step,t_dhw_in,t_dhw_out);
        B1_BP_FG_data(ii,:) = B1_BP(combinations(ii,1),1,mass_start,mass_step,t_dhw_in,t_dhw_out);
    end
    if DC_B1 == 1
        B1_DC_data(ii,:) = B1_DC(combinations(ii,1),0,mass_start,mass_step,t_dhw_in,t_dhw_out);
        B1_DC_FG_data(ii,:) = B1_DC(combinations(ii,1),1,mass_start,mass_step,t_dhw_in,t_dhw_out);
    end
    if BP_B2 == 1
        B2_BP_data(ii,:) = B2_BP(combinations(ii,2),mass_start,mass_step,t_dhw_in,t_dhw_out);
    end
end
% Compute costs for each scenario
% Cost matrices will be constructed with each row corresponding to the
% total cost of one scenario and each column corresponding to the total
% cost of each item, i.e. electricity revenue, fuel costs, etc.

% Compute internal electricity consumption
internal_B1 =
internal_B2 =
for tt = 1:length(el)
% Compute total cost as fuel cost - electricity revenue + op/maint cost +
% nox cost + internal electricity consumption cost
if BP_B1 == 1
  costs_BP_B1(:,tt) = B1_BP_data(:,3)*(fuel_price_B1/1000) +
  B1_BP_data(:,1)*(-el(tt)/1000)
  + B1_BP_data(:,2)*(op_maint/1000) +
  B1_BP_data(:,5)*average_nox*(10^-6)*cost_nox*3600 ... 
  + internal_B1*el(tt);
end
if DC_B1 == 1
  costs_DC_B1(:,tt) = B1_DC_data(:,4)*(fuel_price_B1/1000) +
  B1_DC_data(:,3)*(op_maint/1000) + ... 
  + internal_B1*el(tt);
end
if BP_B2 == 1
  costs_BP_B2(:,tt) = B2_BP_data(:,3)*(fuel_price_B2/1000) +
  B2_BP_data(:,1)*(-el(tt)/1000) ...
  + B2_BP_data(:,2)*(op_maint/1000) +
  B2_BP_data(:,5)*average_nox*(10^-6)*cost_nox*3600 ... 
  + internal_B2*el(tt);
end

% Compute total cost of each combination of operating modes
if BP_B1 && BP_B2 == 1
  costs_BP_B2_tot(:,tt) = costs_BP_FG_B1(:,tt) + costs_BP_B2(:,tt);
end
if DC_B1 && BP_B2 == 1
  costs_DC_B1_tot(:,tt) = costs_DC_FG_B1(:,tt) + costs_BP_B2(:,tt);
end

disp('Simulation complete!');
fprintf('
');

% Convert load of both plants into fractions of 100%
fractions = combinations./heat_goal;

% Plotting
figure(tt)
subplot(2,1,1)
plot(costs_BP_B2_tot(:,tt),fractions(:,1),'-r',costs_BP_B2_tot(:,tt),fractions(:,2),'-b')
ylabel(['Fraction of total heat load [' num2str(heat_goal/1000) ' MW]'])
xlabel('Total cost [kr/h]')
title(['Operating cost for B1 Backpressure and B2 Backpressure at p_el = ' num2str(el(tt)) ' kr/MWh']);
legend('Brista 1','Brista 2')

subplot(2,1,2)
plot(costs_DC_B1_tot(:,tt),fractions(:,1),'-r',costs_DC_B1_tot(:,tt),fractions(:,2),'-b')
ylabel(['Fraction of total heat load [' num2str(heat_goal/1000) ' MW]'])
xlabel('Total cost [kr/h]')
title(['Operating cost for B1 Direct Condensing and B2 Backpressure at p_el = ' num2str(el(tt)) ' kr/MWh']);
legend('Brista 1','Brista 2')

% Compute start/stop costs

cost_warm =
cost_cold =

revenue_BP = min(costs_BP_B2_tot(:,tt));

if revenue_BP <= 0
    hours_warm = cost_warm/revenue_BP;
    hours_cold = cost_cold/revenue_BP;
    disp(['Number of hours revenue required to reach warm start cost: ' num2str(round(hours_warm))]);
    disp(['Number of hours revenue required to reach cold start cost: ' num2str(round(hours_cold))]);
else
    hours_warm = cost_warm/revenue_BP;
    hours_cold = cost_cold/revenue_BP;
    disp(['Number of hours loss required to reach warm start cost: ' num2str(round(hours_warm))]);
    disp(['Number of hours loss required to reach cold start cost: ' num2str(round(hours_cold))]);
end

hours(1,tt) = hours_warm;
hours(2,tt) = hours_cold;
revenue(tt) = revenue_BP;
end

% Plotting
disp('Plotting results...');
fprintf('n')
function [output] = B1_BP( Q_heat, FlueGasFlag, m, m_step, t_dhw_in, t_dhw_out )

% Set boiler power threshold [kW]
Q_boil_max = 122000;

% Set steam flow threshold [kg/s];
m_max = 50;

% Set turbine power thresholds
P_T_min = 4000;
P_T_max = 43000;

% Preallocate warning flags
flag = 0;

% Preallocate heating variables to make while-loop argument valid
Q_cond1 = 0;
Q_cond2 = 0;
Q_fg = 0;
gasflow = 0;

while (Q_cond1 + Q_cond2 + Q_fg) <= Q_heat
% Total steam load through boiler
m = m + m_step;

% Ensure that steam load does not exceed threshold
if m >= m_max
    break
end

% Incoming and outgoing DHW temperature, pressure
p_dhw = 6;
dt = t_dhw_out - t_dhw_in;

% Known/assumed efficiencies
n_boil =
n_is_hp =
n_is_lp =
   n_t = FindTurbineEfficiency(m, dt);
   n_cond =

% Compute pressures, temperatures and enthalpies. These vectors have the % following structures:
% Pressures: [p1 p1a p1b p1c p2 p2a p2b p3 p6 p7]
% Temperatures: [t1 t1aa t1ab t7 tint]
% Enthalpies: [h1 h1a h1b h1c h1aa h1ab h2 h2a h2b h3 h4a h4b h6 h7]
Pressures = ComputePressures_BP(m, dt);
Temperatures = ComputeTemperatures_BP(m, dt);
Enthalpies = ComputeEnthalpies_BP(Temperatures, Pressures, n_is_hp, n_is_lp);

% Compute DHW data - mass flow, intermediate temperature, enthalpies and % heat load
h_dhw_in = XSteam('h_pT', p_dhw, t_dhw_in);
h_dhw_out = XSteam('h_pT', p_dhw, t_dhw_out);
m_dhw = ComputeDHWFlow_BP(m, dt);

% Compute boiler load
Q_boil = (1/n_cond)*m*(Enthalpies(1) - Enthalpies(15));
Q_fuel = Q_boil/n_boil;

% Ensure that boiler power does not exceed threshold!
if Q_boil >= Q_boil_max
    break
end

% Compute combustion data
% Convert LHV from MJ/kg to kJ/kg
HHV =
fuelflow = Q_fuel/HHV;

% Flue gas condensation computations
if FlueGasFlag == 1
    C =
    H =
    O =
    N =
    S =
    moist =
gasflow = ComputeFlueGas( C,H,N,O,S,moist);
gasflow = gasflow*fuelflow;
% Based on maximum flue gas condensation heat
% output of 30 MW @ 50kg/s steam load
dh_gas = 591.1439;
Q_fg = gasflow*dh_gas;
m_dhw_fg = (gasflow*dh_gas)/(h_dhw_out - h_dhw_in);
% Total dhw flow is the sum of the boiler load flow and the flue gas
% contribution
m_dhw_tot = m_dhw + m_dhw_fg;
else
    m_dhw_tot = m_dhw;
end

% Compute intermediate dhw data
t_dhw_int = XSteam('Tsat_p',Pressures(8)) - 1;
h_dhw_int = XSteam('h_pT',p_dhw,t_dhw_int);

% Construct linear equation system. Structure depends on whether stream 2a
% has to be closed or not (i.e. if p2b > p2a).
if Enthalpies(8) > 0
    PrintFlag = 0;
    A = [(Enthalpies(2)-Enthalpies(5)) 0 0 0 0 0; (Enthalpies(5)-Enthalpies(6)) (Enthalpies(3)-Enthalpies(6)) 0 0 0 0 0; 1 1 1 1 0 0 0; 0 0 0 -1 1 1 1; 0 0 0 0 0 (Enthalpies(9)-Enthalpies(11)) 0; 0 0 0 0 0 (Enthalpies(10)-Enthalpies(12));]
    b = [m*(Enthalpies(15) - Enthalpies(14)); m*(Enthalpies(14) - Enthalpies(13)); m; 0; (1/n_cond)*m_dhw*(h_dhw_out - h_dhw_int); (1/n_cond)*m_dhw*(h_dhw_int - h_dhw_in);];
else

    PrintFlag = 1;
    A = [(Enthalpies(2)-Enthalpies(5)) 0 0 0 0 0;
         (Enthalpies(5)-Enthalpies(6)) (Enthalpies(3)-Enthalpies(6)) 0 0 0
         0; 1 1 1 0 0;
         0 0 0 -1 1 1;
         Enthalpies(6) Enthalpies(6) Enthalpies(4) 0 Enthalpies(11)
         Enthalpies(12); 0 0 0 0 (Enthalpies(9)-Enthalpies(11)) 0;
         0 0 0 0 (Enthalpies(10)-Enthalpies(12));
    ];

    b = [m*(Enthalpies(15) - Enthalpies(14));
         m*(Enthalpies(14) - Enthalpies(13));
         m;
         0;
         m*Enthalpies(13);
         (1/n_cond)*m_dhw*(h_dhw_out - h_dhw_int);
         (1/n_cond)*m_dhw*(h_dhw_int - h_dhw_in);]
    ];

end

% Solve linear equation system
sol = A\b;

% Compute power output of turbines
P_HP = (m*(Enthalpies(1) - Enthalpies(2)) + (m-sol(1))*(Enthalpies(2) - Enthalpies(3))... + (m - sol(1) - sol(2))*(Enthalpies(3) - Enthalpies(4)) + (m - sol(1) - sol(2) ... - sol(3))*(Enthalpies(4) - Enthalpies(7)))*n_t;

% LP turbine expansion line depends on whether steam extraction at point 2a
% is performed or not, therefore the power equation must change accordingly
if Enthalpies(8) > 0
    P_LP = (sol(4)*(Enthalpies(7) - Enthalpies(8)) + (sol(4) - sol(5))*(Enthalpies(8) - ... Enthalpies(9)) + (sol(4) - sol(5) - sol(6))*(Enthalpies(9) - Enthalpies(10)))*n_t;
    Q_cond1 = n_cond*sol(6)*(Enthalpies(9)-Enthalpies(11));
    Q_cond2 = n_cond*sol(7)*(Enthalpies(10)-Enthalpies(12));
else
    P_LP = (sol(4)*(Enthalpies(7) - Enthalpies(9)) + (sol(4) - sol(5))*(Enthalpies(9) - ... Enthalpies(10)))*n_t;
    Q_cond1 = n_cond*sol(5)*(Enthalpies(9)-Enthalpies(11));
    Q_cond2 = n_cond*sol(6)*(Enthalpies(10)-Enthalpies(12));
end

% Output returned as vector with [Electrical power output, thermal power % output, fuel heat input, fuel mass flow]
% Check all thresholds!
if (P_HP+P_LP) > P_T_max
    disp('WARNING: Turbine power exceeds maximum value!')
    flag = 1;
end
if (P_{HP}+P_{LP}) < P_{T_{min}}
    disp('WARNING: Turbine power lower than minimum value!')
    flag = 1;
end

if Q_{boil} > Q_{boil_{max}}
    disp('WARNING: Boiler power exceeded threshold!')
    flag = 1;
end

if m > m_{max}
    disp('WARNING: Steam flow exceeded threshold!')
    flag = 1;
end

if flag == 0
    disp('Simulation completed with no errors!')
    fprintf('
')
else
    disp('There were one or more warnings during the simulation.')
    fprintf('
')
end

output = [(P_{HP}+P_{LP}) (Q_{cond1} + Q_{cond2} + Q_{fg}) Q_{fuel} fuelflow gasflow];
end
11.4 B1_DC.m

```matlab
function [output] = B1_DC(Q_heat, FlueGasFlag, m, m_step, t_dhw_in, t_dhw_out)

% Set condenser heat load threshold
Q_cond_limit = 85000;

% Set boiler power threshold [kW]
Q_boil_limit = 122000;

% Set steam flow threshold
m_max = 50;

% Preallocate heating variables to make while-loop argument valid
Q_cond = 0;
Q_fg = 0;
gasflow = 0;

% Preallocate error flag
flag = 0;

while (Q_cond + Q_fg) <= Q_heat
% Total steam load through boiler
m = m + m_step;

p_dhw = 6;

% Known/assumed efficiencies
n_boil =
n_cond =

% Compute enthalpies.
% Enthalpies: [h1 h2 h2a h3 h3a h4 h5 h6]
Enthalpies = ComputeEnthalpies_DC();

% Compute DHW data - mass flow, intermediate temperature, enthalpies and
% heat load
h_dhw_in = XSteam('h_pT', p_dhw, t_dhw_in);
h_dhw_out = XSteam('h_pT', p_dhw, t_dhw_out);

% DHW flow computed this way since it is a direct function in direct
% condensing mode, unlike backpressure mode!
m_dhw = Q_heat/(h_dhw_out - h_dhw_in);

% Compute boiler load
Q_boil = (1/n_cond)*m*(Enthalpies(1) - Enthalpies(9));
Q_fuel = Q_boil/n_boil;

% Compute combustion data
HHV =
fuelflow = Q_fuel/HHV;

% Flue gas condensation computations
if FlueGasFlag == 1
    C =
    H =
    O =
    N =
```
S = moist =
gasflow = ComputeFlueGas(C,H,N,O,S,moist);
gasflow = gasflow*fuelflow;
  % Based on maximum flue gas condensation heat output of 30 MW @ 50kg/s steam load
dh_gas = 591.1439;
Q_fg = gasflow*dh_gas;
m_dhw_fg = (gasflow*dh_gas)/(h_dhw_out - h_dhw_in);
  % Total dhw flow is the sum of the boiler load flow and the flue gas contribution
m_dhw_tot = m_dhw + m_dhw_fg;
else
  m_dhw_tot = m_dhw;
end

% Construct linear equation system
A = [Enthalpies(1) Enthalpies(1);
   0 1;
   Enthalpies(1) Enthalpies(6)];
b = [m*Enthalpies(1);
   n_cond*((m_dhw*(h_dhw_out-h_dhw_in))/(Enthalpies(1)-Enthalpies(6)));
   m*Enthalpies(7)];

% Solve linear equation system
sol = A\b;

% Compute steam flow over condenser
Q_cond = sol(2)*n_cond*(Enthalpies(1)-Enthalpies(6));
end

% Output returned as vector with [condenser heat output, flue gas condensation heat output, total heat output, fuel energy input, fuel mass flow]

% Threshold checks
if Q_boil > Q_boil_limit
  disp('WARNING: Boiler power exceeded threshold!')
  flag = 1;
end

if m > m_max
  disp('WARNING: Steam flow exceeded threshold!')
  flag = 1;
end

if Q_cond > Q_cond_limit
  disp('WARNING: Condenser load exceeds threshold!')
  flag = 1;
end

if flag == 0
  disp('Simulation completed with no errors!')
  fprintf('
')
else
  disp('There were one or more warnings during the simulation.')
end

-48-
fprintf('
')
end

output = [Q_cond Q_fg (Q_cond+Q_fg) Q_fuel fuelflow gasflow];
end
11.5 B2_BP.m

function [output] = B2_BP(Q_heat,m,m_step,t_dhw_in,t_dhw_out)

% Set boiler power threshold [kW]
Q_boil_max = 80000;

% Set steam flow threshold [kg/s];
m_max = 34;

% Set turbine power limits
P_T_min = 4187;
P_T_max = 25791;

% Preallocate warning flags
flag = 0;

% Preallocate heating variables to make while-loop argument valid
Q_cond1 = 0;
Q_cond2 = 0;
Q_fg = 0;

while (Q_cond1 + Q_cond2 + Q_fg) <= Q_heat
% Total steam load through boiler
m = m + m_step;

% Break loop if steam load exceeds threshold
if m >= m_max
    break
end

p_dhw = 9;

% Known/assumed efficiencies
n_boil =
n_is_t =
n_cond =
n_t = FindTurbineEfficiency_B2(m);

% Since the turbine efficiency "flips signs", this needs to be adjusted
% for. The efficiency term is actually the ratio between the value from the
% Siemens documents and the value that this model would give without the
% factor. It essentially corrects the power output.
if n_t >= 1
    n_t = 1/n_t;
end

% Compute pressures, temperatures and enthalpies. These vectors have the
% following structures:
% Pressures: [p1 p1a p1b p1 p2 p5 p6]
% Temperatures: [t1 t1a t1b t5]
% Enthalpies: [h1 h1a h1b h1c h2 h3a h3b h5 h6]
Pressures = ComputePressures_BP_B2(m,t_dhw_in,t_dhw_out);
Temperatures = ComputeTemperatures_BP_B2(m);
Enthalpies = ComputeEnthalpies_BP_B2(Temperatures,Pressures,n_is_t);
% Compute DHW data - mass flow, intermediate temperature, enthalpies and heat load
h_dhw_in = XSteam('h_pT',p_dhw,t_dhw_in);
h_dhw_out = XSteam('h_pT',p_dhw,t_dhw_out);
m_dhw = Q_heat/(h_dhw_out - h_dhw_in);

% Compute boiler load
Q_boil = (1/n_cond)*m*(Enthalpies(1) - Enthalpies(11));
Q_fuel = Q_boil/n_boil;

% Break loop if boiler power exceeds threshold
if Q_boil >= Q_boil_max
    break
end

% Compute combustion data
% Use higher heating value to determine actual mass flow of fuel
HHV =
fuelflow = Q_fuel/HHV;

% Compute flow of combustion air, based on maximum boiler load of 33.72 kg/steam/s and maximum air flow of 45610 m3/h using three point quadratic least squares approximation.
m_air =

% Specific heat of air determined as average between 25 C and 170 C. The combustion air is preheated from 25C to 170C in the air preheaters.
cp_air = 1.01;
dt_air = 170 - 25;

% Flue gas condensation computations
C =
H =
O =
N =
S =
moist =
gasflow = ComputeFlueGas(C,H,N,O,S,moist);
gasflow = fuelflow*gasflow;
% Based on maximum flue gas condensation heat output of 12 MW @ 34kg/s steam load
dh_gas = 350.6593;
Q_fg = gasflow*dh_gas;
% Compute DHW enthalpy and temperature after FG condensation
m_dhw_fg = (gasflow*dh_gas)/(h_dhw_out - h_dhw_in);
% Total dhw flow is the sum of the boiler load flow and the flue gas contribution
m_dhw_cond = m_dhw - m_dhw_fg;

% Compute intermediate dhw data, assume 1 degree pinch point temp difference
t_dhw_int = XSteam('Tsat_p',Pressures(7)) - 1;
h_dhw_int = XSteam('h_pT',p_dhw,t_dhw_int);

% Construct linear equation system.
A = 
    [((Enthalpies(2)-Enthalpies(3))+(Enthalpies(4)-Enthalpies(5))) 0 0 0 ;
     1 1 1 0 ;
     0 (Enthalpies(6)-Enthalpies(8)) 0 0 ;
     0 0 (Enthalpies(7)-Enthalpies(9)) 0 ;
     (Enthalpies(2)+Enthalpies(4)) Enthalpies(6) Enthalpies(7) 1 ];
\[ b = \left( (m_{\text{air}} \times c_{p_{\text{air}}} \times \Delta t_{\text{air}}) + m \times (\text{Enthalpies}(11) - \text{Enthalpies}(10)) \right) \]
\[ m_{\text{dhw\_cond}} \times \left( \frac{1}{n_{\text{cond}}} \times (h_{\text{dhw\_out}} - h_{\text{dhw\_int}}) \right) \]
\[ m \times (\text{Enthalpies}(1)) \];

% Solve linear equation system
sol = \text{A}\backslash b;

% Compute power output of turbines and condensers
% Assume that mass flow \( m_{\text{la}} \) is 0.381 kg/s, as this is the case for almost all of the given simulations. \( \text{sol}(1) \) is then equal to \( m_{\text{la}} + m_{\text{lb}} \).
\( m_{\text{la}} = 0.381; \)
\( P_{\text{T}} = n_{\text{t}} \times (m \times (\text{Enthalpies}(1) - \text{Enthalpies}(2)) + (m - m_{\text{la}}) \times (\text{Enthalpies}(2) - \text{Enthalpies}(4)) \ldots + (m - \text{sol}(1)) \times (\text{Enthalpies}(4) - \text{Enthalpies}(6)) + (m - \text{sol}(1) - \text{sol}(2)) \ldots \)
\[ (\text{Enthalpies}(6) - \text{Enthalpies}(7)) \);
\( Q_{\text{cond\_1}} = n_{\text{cond}} \times \text{sol}(2) \times (\text{Enthalpies}(6) - \text{Enthalpies}(8)); \)
\( Q_{\text{cond\_2}} = n_{\text{cond}} \times \text{sol}(3) \times (\text{Enthalpies}(7) - \text{Enthalpies}(9)); \)

end

% Check all thresholds!
if \( P_{\text{T}} > P_{\text{T\_max}} \)
\( \text{disp}('\text{WARNING: Turbine power exceeds maximum value!}') \)
\( \text{flag} = 1; \)
end

if \( P_{\text{T}} < P_{\text{T\_min}} \)
\( \text{disp}('\text{WARNING: Turbine power lower than minimum value!}') \)
\( \text{flag} = 1; \)
end

if \( Q_{\text{boil}} > Q_{\text{boil\_max}} \)
\( \text{disp}('\text{WARNING: Boiler power exceeded threshold!}') \)
\( \text{flag} = 1; \)
end

if \( m > m_{\text{max}} \)
\( \text{disp}('\text{WARNING: Steam flow exceeded threshold!}') \)
\( \text{flag} = 1; \)
end

if \( \text{flag} == 0 \)
\( \text{disp}('\text{Simulation completed with no errors!}') \)
\( \text{fprintf}('\n') \)
else
\( \text{disp}('\text{There were one or more errors during the simulation.}') \)
\( \text{fprintf}('\n') \)
end

output = [\( P_{\text{T}} \) (\( Q_{\text{cond\_1}} + Q_{\text{cond\_2}} + Q_{\text{fg}} \)) \( Q_{\text{fuel}} \) \text{fuelflow} \text{gasflow}];
end
11.6 ComputeFlueGas.m

function [ m_gases ] = ComputeFlueGas( C,H,N,O,S,moist )
% Function to compute flue gas flow given a certain fuel composition, 
% moisture content and fuel flow.

% Molar masses
M_C = 12.01;
M_H2O = 18.02;
M_H2 = 2.02;
M_O2 = 32.00;
M_N2 = 28.01;
M_S = 32.06;

% Analysis based on 1kg dry, ash-free fuel. Convert given dry weight 
% percentages to wet weight percentages.
wet_C = C*moist;
wet_H2 = H*moist;
wet_N2 = N*moist;
wet_O2 = O*moist;
wet_S = S*moist;

N_in_air = 3.77;
excess_air = 1.03;

% Amounts per kg fuel
C_molkg = 1000*wet_C*(1/M_C);
H2_molkg = 1000*wet_H2*(1/M_H2);
O2_molkg = 1000*wet_O2*(1/M_O2);
N2_molkg = 1000*wet_N2*(1/M_N2);
S_molkg = 1000*wet_S*(1/M_S);
moist_molkg = 1000*moist*(1/M_H2O);

% Required oxygen
req_C = C_molkg;
req_H2 = (1/2)*H2_molkg;
req_O2 = -O2_molkg;
req_N2 = 0;
req_S = S_molkg;

sum_O2 = req_C + req_H2 + req_O2 + req_N2 + req_S;

% Produced flue gases
CO2 = C_molkg;
H2O = H2_molkg + moist_molkg;
N2 = N2_molkg + N_in_air*sum_O2;
SO2 = S_molkg;

sum_dry_air = N2 + sum_O2;

nitrogen_excess = (excess_air*N2) - N2;
oxygen_excess = (sum_dry_air*excess_air) - nitrogen_excess - sum_dry_air;

% Total gas amounts
tg_H2O = H2O;
tg_CO2 = CO2;
tg_N2 = N2 + nitrogen_excess;
tg_SO2 = SO2;
tg_O2 = oxygen_excess;
real_total_air = sum_dry_air * excess_air;
real_total_gas = tg_H2O + tg_CO2 + tg_N2 + tg_SO2 + tg_O2;

% Convert from mol/kg fuel to m3n/kg fuel
m_air = 0.0224*real_total_air;
m_flue = 0.0224*real_total_gas;

m_gases = m_flue;
end
11.7 ComputeEnthalpies_BP.m

function [ Enthalpies ] = ComputeEnthalpies_BP( Temperatures, Pressures, n_is_hp, n_is_lp )
% Extract pressures & temperatures from input vectors
p1 = Pressures(1);
p1a = Pressures(2);
p1b = Pressures(3);
p1c = Pressures(4);
p2 = Pressures(5);
p2a = Pressures(6);
p2b = Pressures(7);
p3 = Pressures(8);
p6 = Pressures(9);
p7 = Pressures(10);
t1 = Temperatures(1);
t1aa = Temperatures(2);
t1ab = Temperatures(3);
t7 = Temperatures(4);
t8 = Temperatures(5);

h1 = XSteam('h_pT',p1,t1);
h1aa = XSteam('h_pT',p1a,t1aa);
h1ab = XSteam('h_pT',p1b,t1ab);

% Compute h1a, h1b, h1c, h2 based on isentropic turbine expansion line
s1 = XSteam('s_pT',p1,t1);
h2s = XSteam('h_ps',p2,s1);
h2 = h1 - n_is_hp*(h1-h2s);
s2 = XSteam('s_ph',p2,h2);
m_1 = (s1-s2)/(p1-p2);
s1a = m_1*(p1a-p1) + s1;
s1b = m_1*(p1b-p1) + s1;
h1a = XSteam('h_ps',p1a,s1a);
h1b = XSteam('h_ps',p1b,s1b);
h1c = XSteam('hV_p',p1c);

% Compute h2a, h2b, h3 based on isentropic turbine expansion line
h3s = XSteam('h_ps',p3,s2);
h3 = h2 - n_is_lp*(h2-h3s);
s3 = XSteam('s_ph',p3,h3);
m_2 = (s2-s3)/(p2-p3);
s2a = m_2*(p2a-p2) + s2;
s2b = m_2*(p2b-p2) + s2;
if p2a <= p2b
    h2a = 0;
else
    h2a = XSteam('h_ps',p2a,s2a);
end
h2b = XSteam('h_ps',p2b,s2b);
h4a = XSteam('hL_p',p2b);
h4b = XSteam('hL_p',p3);
h6 = XSteam('hL_p',p6);
h7 = XSteam('h_pT',p7,t7);
h8 = XSteam('h_pT',p7,t8);

Enthalpies = [h1 h1a h1b h1c h1aa h1ab h2 h2a h2b h3 h4a h4b h6 h7 h8];
function [ Enthalpies ] = ComputeEnthalpies_DC()
% Compute enthalpies in direct condensation mode. In this mode, all
% pressures and temperatures are fixed since they are not dependant on any
% turbine characteristics.

p1 = 140;
p2a = 5;
p3a = 10;
p5 = 10;
p5a = 40;
p6 = 168;

t1 = 540;
t6 = 155;
t2a = 200;
t3a = 180;
t4 = 80;

t5a = 155;

h1 = XSteam('h_pT',p1,t1);
h2 = h1;
h3 = h1;

h2a = XSteam('h_pT',p2a,t2a);
h3a = XSteam('h_pT',p3a,t3a);
h4 = XSteam('h_pT',p3a,t4);
%h4 = XSteam('hL_p',p3a);
h5 = XSteam('hL_p',p5);
h5a = XSteam('h_pT',p5a,t5a);
h6 = XSteam('h_pT',p6,t6);

Enthalpies = [h1 h2 h2a h3 h3a h4 h5 h5a h6]';
end
function [ Enthalpies ] = ComputeEnthalpies_BP_B2( Temperatures,Pressures,n_is_t )
% Extract pressures & temperatures from input vectors
p1 = Pressures(1);
p1a = Pressures(2);
p1aa = Pressures(3);
p1b = Pressures(4);
p1ab = Pressures(5);
p1c = Pressures(6);
p2 = Pressures(7);
p5 = Pressures(8);
p6 = Pressures(9);

t1 = Temperatures(1);
t1aa = Temperatures(3);
t1ab = Temperatures(5);
t4 = Temperatures(6);
t6 = Temperatures(7);

h1 = XSteam('h_pT',p1,t1);

% Compute h1a, h1b, h1c, h2 based on isentropic turbine expansion line
s1 = XSteam('s_pT',p1,t1);
h2s = XSteam('h_ps',p2,s1);
h2 = h1 - n_is_t*(h1-h2s);
s2 = XSteam('s_ph',p2,h2);
m_1 = (s1-s2)/(p1-p2);
s1a = m_1*(p1a - p1) + s1;
s1b = m_1*(p1b - p1) + s1;
s1c = m_1*(p1c - p1) + s1;
h1a = XSteam('h_ps',p1a,s1a);
h1b = XSteam('h_ps',p1b,s1b);
h1c = XSteam('h_ps',p1c,s1c);

h1aa = XSteam('h_pT',p1aa,t1aa);
h1ab = XSteam('h_pT',p1ab,t1ab);

h3a = XSteam('hL_p',p1c);
h3b = XSteam('hL_p',p2);
h4 = XSteam('h_pT',p5,t4);
h6 = XSteam('h_pT',p6,t6);

Enthalpies = [h1 h1a h1aa h1b h1ab h1c h2 h3a h3b h4 h6];
end