Dynamic Model of a Fixed Grate Combustion in Modelica

Bed segment

Gas combustion segment

Raw Fuel

Flue Gas

Moisture
Volatiles
Char
Ash
Flue Gas

PA

PA

Ash Collector

Title Sheet

Theme: Master Thesis

Project title: Dynamic Model of a Fixed Grate Combustion in Modelica

Group: PECT10-1-F16

Semester: 10th semester

Supervisor: Kim Sørensen

Supervisor: Jens Bo Holm

University: Aalborg University Esbjerg

Company: Added Values P/S

Signature:

———————————————————

Johannes Veje

Abstract

This thesis models a single control volume for the use in a dynamic model of a fixed grate combustion in the modelling language Modelica. Different sub-models are created and validated by comparing with expected behaviour or literature data. A model for a gas combustion segment was also created using three different approaches; chemical kinetics, chemical equilibrium, and analytical.

A complete bed segment was formed from the sub-models and models from the standard Modelica library and the behaviour of the segment was tested under the five processes fuel undergoes in a fixed grate combustion process; heat up, moisture evaporation, devolatilisation, char conversion, and ash cool down. The models for the gas combustion segment from the three different approaches was compared and the model from the analytical approach was tested under a wide range of air-fuel ratios.

Both the model for the bed segment and the gas combustion segment preformed as expected under the boundary conditions given.

Preface

This report was written by Johannes Veje as a final project under the MSc in Process Engineering and Combustion Technology at Aalborg University Esbjerg in the spring of 2016. This project is part of a larger ongoing project at Added Values P/S, which was started during an internship at the company on the authors 9th semester. The 9th semester project focused on learning and being conversant with the modelling language used, Modelica, as well as creating a strong structure for the final model.

In the back of the report, Enclosure B on page 51, there is a more thorough description of the Modelica modelling language [1]. It explains parts of the syntax used in Modelica, which are also used in the report, and some of the advantages of the language. A complete read of the enclosure is not required and the most important pointers are:

- Flows are positive into the components and components by connectors.

- Modelica uses both equations and assignments.
  (a+b = x+y) and (a := x+y-b)

- The software manipulates the equations to fit the selected states. Differential equations are extracted from other differential equations.

- Contains functions for calculation of thermodynamic properties of a wide range of substances.

The author wishes to thank the people at Added Values for providing an inspiring work environment. A special thanks goes to Rene Just Nielsen for providing valuable competent feedback and always taking the time to provide help and support.

Contents

Introduction

For many years computer models have been a big part of power plant designing. Both for testing designs before new plants are constructed and to improve the design of already operational plants. Some of these models are used for CFD simulations that are focusing on the gas combustion and the temperature distribution inside the boiler. Other models are used to test how a control system will behave before it is implemented. There are also models that are used to estimate the energy output of the plant in the form of power and heat generation, that again is used on the power grid and in the district heating network.

Common for them all are that the more accurate they are, the better they preform at what they are suppose to - improving the economics. Either in the form of saving on maintenance and operational cost, or to increase the output and thereby the profit.

Previously the most used models were used to solve large steady state problems. But as the energy supply have shifted from larger centralised coal fired power plants to smaller decentralised biomass fired plants and fluctuating renewable sources such as wind and solar, the need for models have shifted from steady state to dynamic.

Dynamic models can give indications about how well a power plant can be regulated to for example counteract a decrease in wind energy. For grate-fired boilers the dynamic behaviour is slow and it is dependent on the fuel bed, which makes the control of such a boiler complicated. Fixed grate firing is one of the most used firing methods in newly constructed plants as it is flexible in regards of size distribution, particle load, and moisture content of the fuel [2]. The request for dynamic models of fixed grate combustion is therefore high.

x

CHAPTER 1

Overall Model Description

In the combustion of a solid fuel there are five steps that the fuel undergoes when entering the boiler as raw fuel to exiting as ash:

Heating up: The fuel is heated from the inlet temperature to the evaporation temperature of moisture.

Evaporation: The moisture in the fuel evaporates into steam that is carried away with the flue gas, leaving dry fuel.

Devolatilisation: The dry fuel undergoes pyrolysis where the volatile gasses in the fuel is released into the gas flow. After the devolatilisation, char remains.

Char conversion: The remaining char is converted in either a combustion process where oxygen is consumed, or a gasification process where for example carbon dioxide is consumed. The combustion process is in the order of magnitude 10^5 faster than the gasification process [3].

Ash cool down: After the char conversion, only ash remains and the last process is cooling it down as much as possible to utilise the energy.

These processes happen in a sequence and could be distributed along the grate as indicated in figure 1.1. In the figure the raw fuel enters at the left hand side where the heating up process begins, and leaves at the right hand side into the ash collector after the cool down. However, depending on the conditions in the combustion chamber and at the boundary, the dashed border lines could be located in different positions or even tilted to the right instead of the left as they are in the figure.

Since these processes can occur in different parts of the bed, it is important to not fix the zones where they can happen. For example could the evaporation process begin further up or down the grate if the temperature in the combustion chamber

1

1. Heat up
2. Moisture evaporation
3. Devolatilisation
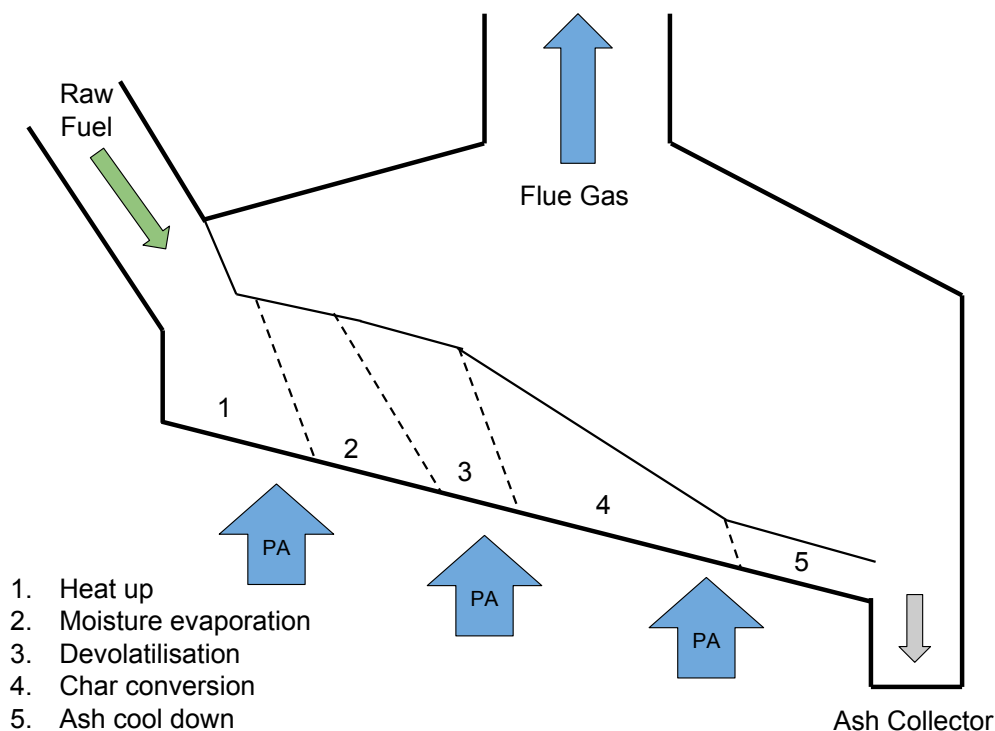4. Char conversion
5. Ash cool down

Figure 1.1: Bed processes in a fixed grate combustion.

either increases or decreases. To ensure that the model can handle this, the bed is discretised into segments which have no predetermined purposes, and the different processes can happen in all of them. This results in each segment requires a larger amount of equations and variables. To reduce the overall calculation time, the bed is discretised in just two directions; along the length of the grate and along the height of the bed. To further simplify the model, the flows of the fuel and flue gas is set to be one dimensional. The fuel's flow move horizontal along the grate and the flue gas flow move vertical from the grate through the bed.

Figure 1.2 shows how the bed segments, marked by a grey colour, could be distributed throughout the bed. All the bed segments have the same length, but not the same height. For each of the bed columns in the figure, there is also a gas combustion segment. These segments are where the volatile gases are combusted. It is therefore assumed that there is no combustion of volatile gases happening inside the bed. This assumption is expected to be reasonable since the flow of flue gas through the bed is fast. It also corresponds well with what was observed during combustion experiments conducted during a previous project [4].

However, as the flue gas composition out of each bed column is dependent on the composition of the bed and thereby different from column to column, there is a need for extra combustion segments. These segments are shown in figure 1.2,
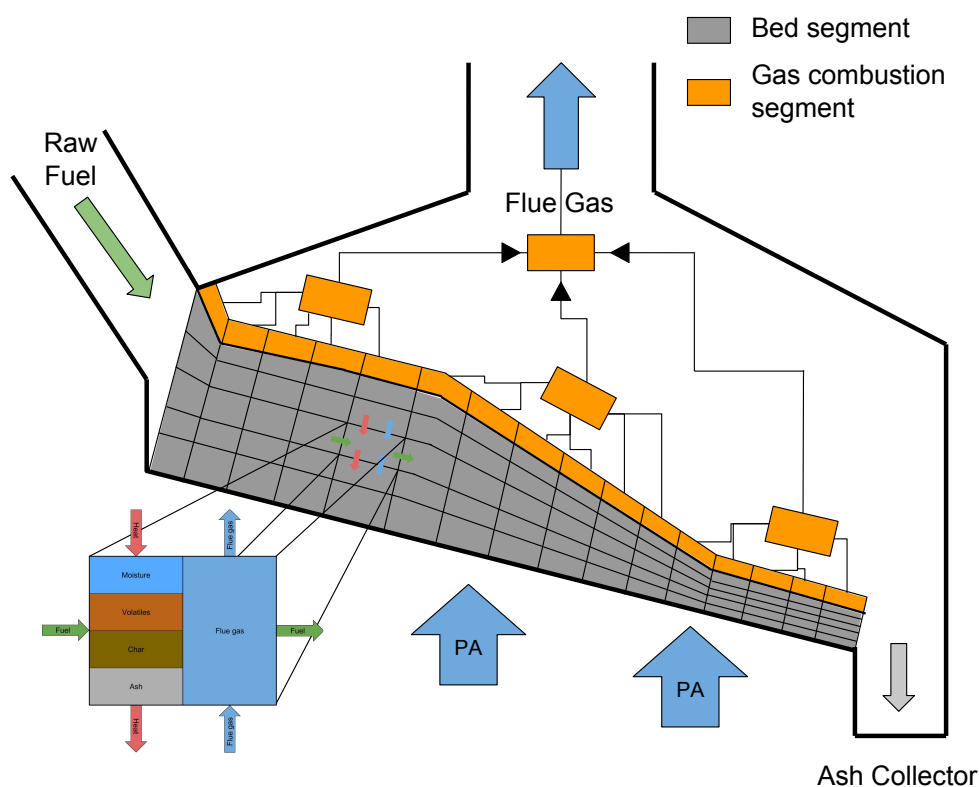
Figure 1.2: Bed and combustion segments.

where there are three extra combustion segments above the bed combustion segments. Furthermore, there is also a secondary combustion segment well above the bed. This segment is meant for the final combustion of the volatiles gases and also where secondary air can be fed into the combustion.

Each bed segment is a control volume with four surface areas, two horizontal and two vertical. Fuel enters and leaves by the vertical areas and the flue gas enters and leaves by the horizontal areas. The horizontal areas are also used for heat transfer to and from the segment. This is indicated in figure 1.3, which gives a closer look at the individual segment from figure 1.2.

Inside the bed segment, the volume is split into fuel and flue gas. Furthermore, is the fuel split into four parts; moisture, volatiles, char, and ash. Where the split between the fuel and the flue gas is a physical split between solid and gas, the split between the four parts of the fuel is merely a modelling split to separate the models.

Separating the fuel into four parts allows the smaller sub-models to be debugged and tested individually and allows for replacing of the sub-models. For example can the char conversion model be replaced with a more complex model if needed. Some of the models can also be made simpler so the simulation speed can be improved, if this can happen without compromising the accuracy or the precision of the results.

The four sub-models for the fuel part are:

- Evaporation model with absorption of heat and release of moisture.

- Pyrolysis model with release of volatile gases.

- Char conversion model with absorption of oxygen and gasification species and release of combustion products and heat.

- Ash cool down model with absorption and release of heat.

There is only one stream of flue gas moving through the bed, though it consists of multiple gas species. These gas species are assumed to be ideal and they are: N_2, H_2, CO, O_2, H_2O, CO_2, and CH_4. N_2 and O_2 are from the atmospheric air supplied as primary and secondary air to the boiler. H_2 and CH_4 are the primary volatile gases released during the pyrolysis process. H_2O is mostly from the evaporation of moisture from the fuel, but also from the combustion process. CO and CO_2 are the primary combustion products.

The modelling of the flue gas is split into two sub-models:

- Volume model with heat transfer and collection of flue gas from different sources.

- Flow model with pressure loss.

Figure 1.4 gives a graphical illustration of the interactions between the phases.

- The fuel releases moisture from the evaporation, volatiles from the pyrolysis process, and combustion products from the char conversion to the flue gas.

- The flue gas is supplying the oxygen for the char combustion and the gas species needed for the char gasification.

- Heat transfer between the fuel and flue gas. The direction of the heat transfer is dependent on the temperatures of the fuel and flue gas.

The individual fuel models will be described in chapter 2, the flue gas models in chapter 3 and chapter 5 will contain the description of the gas combustion segment.
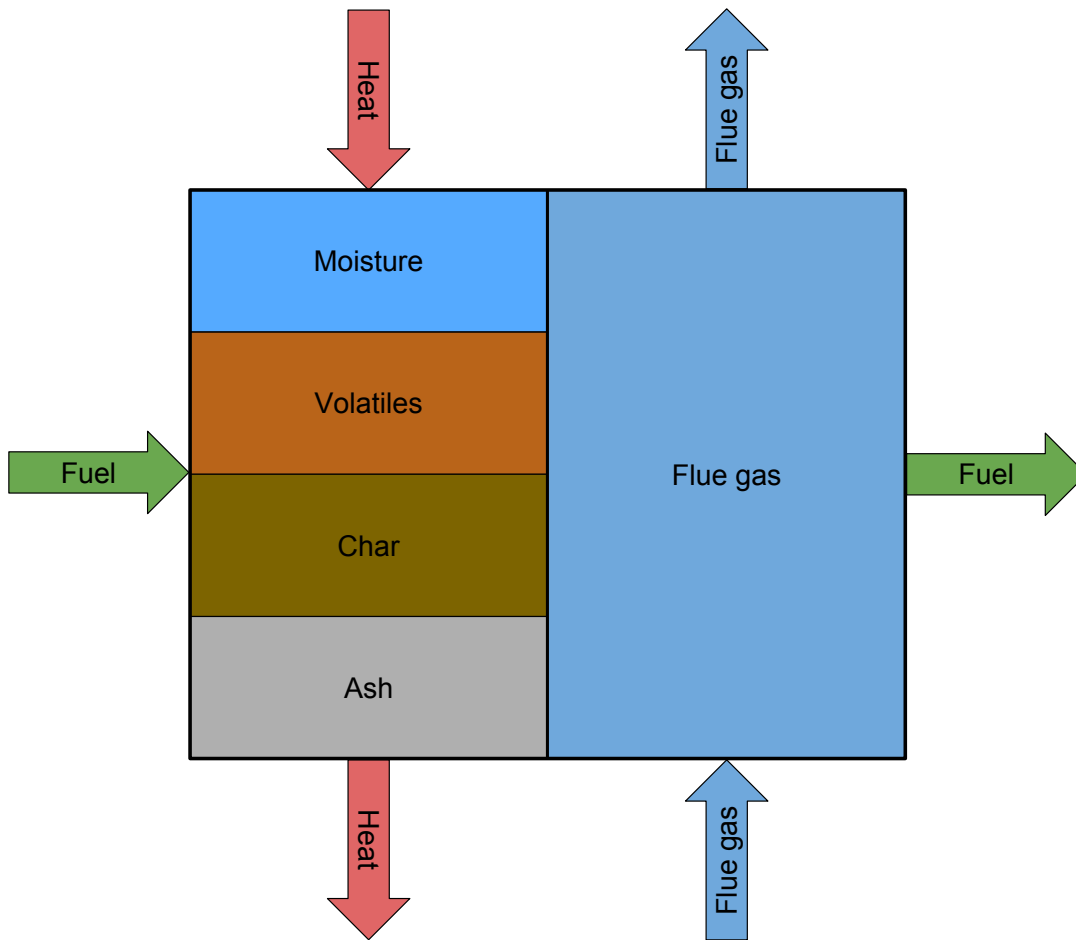
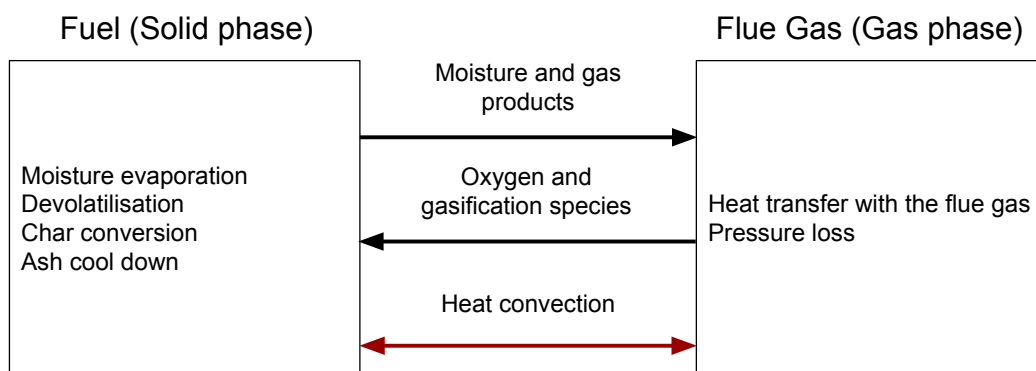Figure 1.3: Overview of a bed segment with fuel moving horizontally and flue gas moving vertically.

Figure 1.4: The fuel to flue gas interactions inside the bed segments.

1.1   Project Delimitation

Creating a working dynamic model of a fixed grate combustion is an ongoing process at Added Values P/S, and this project is a part thereof. The model specifications are listed in Enclosure A on page 49 [1].

This project is focusing on modelling a single bed segment. The bed segment should be able to handle the five processes that the fuel undergoes from raw fuel to ash; heat up, moisture evaporation, devolatilisation, char conversion, and cool down of ash. It should also be able to respond to changes in boundary conditions dynamically and behave as expected when these changes are made.

Another focus of the project is on modelling a gas combustion segment just above the bed. This is done by trying out different approaches. The segment should give indications of the flue gas composition after the combustion of the volatile gases released from the bed.

CHAPTER **2**

Fuel Models

In this chapter the four fuel models will be described and validated. Initially will the equations, that are repeated in each of the four models be described. Thereafter will the first three models be described individually. The last model, ash heating, is so simple that it will be described together with the repeating equations, and it will be used to validate these equations. After each description of a model, a validation of that model will be described. The models will be validated by looking at the expected behaviour of the models as they are sub-models of the main model. It can therefore be difficult to validate these with experiments. The four fuel models are:

- Moisture Evaporation

- Pyrolysis

- Char Conversion

- Ash Cool Down

2.1   Repeating Equations

2.1.1   Description

The first two equations, are dynamic mass and energy balances:

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \sum \dot{m} \tag{2.1}$$

$$\frac{\mathrm{d}U}{\mathrm{d}t} = \sum (\dot{m}h) + \dot{Q} \tag{2.2}$$

The mass of the segment is simply the volume of the segment multiplied by the density:

$$m = W \cdot L \cdot H \cdot \rho \tag{2.3}$$

where ρ is the density in $\frac{kg}{m^3}$.

The next equation is for determining the fuel mass flow leaving the segment. For fluids this is driven by the pressure difference or the accumulation of mass in a fixed volume, but for the solid fuel on a moving grate the mass flow is determined by the transportation velocity of the grate. It is also dependent on the amount of fuel on the grate. The mass flow is therefore taken to be the grate transportation velocity multiplied with the cross-section area of the mass in the segment and the density:

$$\dot{m}_{fuel,out} = -\underbrace{W \cdot H}_{\substack{\text{Cross-}\\ \text{section}\\ \text{area}}} \cdot \rho \cdot v \tag{2.4}$$

where v is the transport velocity of the grate in $\frac{m}{s}$ supplied as a causal input. The mass flow has a negative sign as the flow is leaving the segment.

The models also contains an equation to calculate the specific enthalpy of the segment. For the ash cool down model, this is done by a custom function described in Appendix A.2.1. That is the only extra equation in the ash cool down model.

2.1.2   Validation

The first to be validated is the mass flow entering and leaving the segment when nothing is occurring. This is the same for all the fuel models and the ash model is used for the simulation. Looking at the mass of ash in the segment there are three scenarios:

1. The mass flow entering the segment is numerical higher than the initial mass flow leaving the segment. The mass of ash should increase till the mass flow leaving is numerical equal to the mass flow entering.

2. The mass flow entering the segment is numerical equal to the mass flow leaving the segment. The mass of ash in the segment should remain constant.

3. The mass flow entering the segment is numerical lower than the initial mass flow leaving the segment. The mass of ash should decrease till the mass flow leaving is numerical equal to the mass flow entering.

Each of theses scenarios have been simulated and the left chart in figure 2.1 shows the mass of ash in the segment for each scenario. The blue graph shows scenario one; the mass of ash in the segment increases until the mass flows entering and leaving are in equilibrium. The orange graph shows scenario two; the mass of ash in the segment is constant. The yellow graph shows the third and last scenario; the mass of ash in the segment decreases until the mass flows entering and leaving are in equilibrium. These results corresponds with the expected.
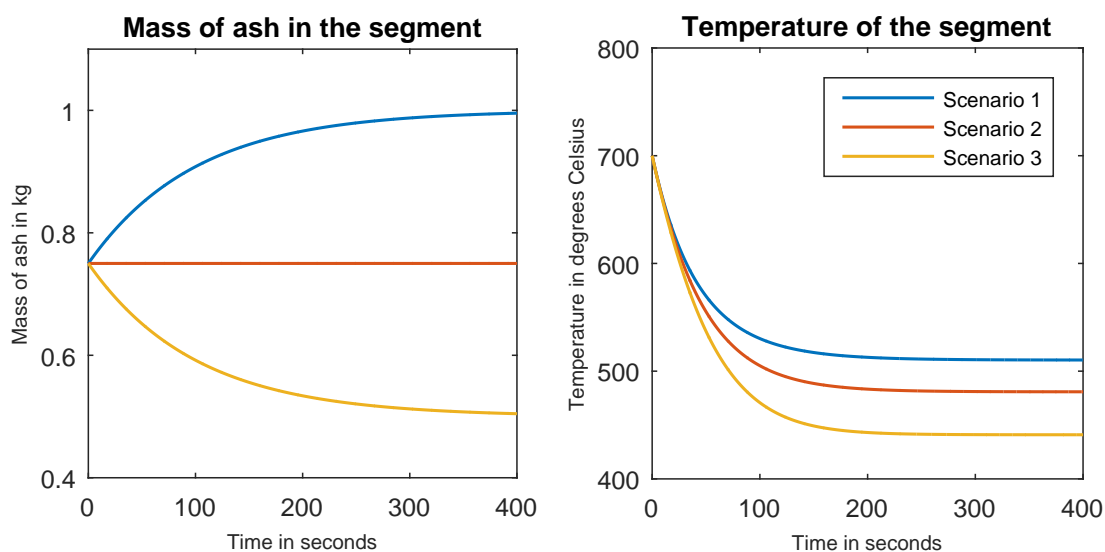
Figure 2.1: Mass and temperature of ash in the segment for the three scenarios. Left chart shows the mass, right chart shows the temperature.

The temperature development of the ash in the segment needs to be validated aswell. As heat is transferred from the segment with a temperature difference as the driving force, the temperature should decrease and then stabilise at an equilibrium point. At this point there is not enough temperature difference to reduce the temperature of the segment. The right chart in figure 2.1 shows the temperature of the ash in the previous three scenarios. The scenario with the highest mass flow and mass inside the segment, scenario one, have the highest temperature and vice versa.

2.2 Moisture Evaporation

2.2.1 Description

The moisture evaporation is modelled as a splitter, and the moisture inside is considered to be pure water. The amount that is split is determined by the vapour quality of the water in the segment. This means that there is an additional connector on the model, and the mass and energy balances, equation (2.1) and (2.2), account for this by containing an additional term. As the vapour quality is only defined at the saturation temperature, there are no steam evaporated until the saturation temperature is reached and the saturation temperature is maintained until all the moisture has evaporated.

The third connector is for the steam output, and the specific enthalpy is set to be the maximum between the specific enthalpy in the segment and the dew enthalpy

of water at the flue gas pressure.

$$h_{steam} = \text{maxSmooth}(h_{seg}, h_{dew}(p)) \qquad (2.5)$$

where

- h_{steam} is the specific enthalpy of the steam leaving the segment in $\frac{J}{kg}$

- h_{seg} is the specific enthalpy of the segment in $\frac{J}{kg}$

- $h_{dew}(p)$ is the dew enthalpy in $\frac{J}{kg}$ at flue gas pressure p in Pa

- maxSmooth is a custom function implemented as described in Appendix A.1

The specific enthalpy of the steam outlet is calculated in this way to handle that the steam flow leaving the splitter is zero before the saturation temperature is reached. It also ensures that excess heat can be utilised to overheat the steam.

The specific enthalpy of the segment is determined from the two thermodynamic properties, specific internal energy and pressure.

$$h_{seg} = h(u, p) \qquad (2.6)$$

The vapour quality of the water in the segment can be determined in a similar way:

$$x = x(h, p) \qquad (2.7)$$

As mentioned earlier, the vapour quality is used to determine the amount of steam leaving the segment. It is multiplied with the mass of moisture in the segment and with a rate constant, k, which determines the speed of which the steam leaves the segment.

$$\dot{m}_{steam} = -m \cdot x \cdot k \qquad (2.8)$$

Note that the mass flow has a negative sign since the flow is leaving the segment.

2.2.2   Validation

The validation of the model will be based on how the model reacts when heat is supplied and the water is being evaporated. As heat is applied to the segment the temperature should rise till the saturation temperature is reached and the evaporation should start. At this point the temperature should remain constant till all the moisture has been evaporated. Applying the same three scenarios as previous, the mass of the moisture in the segment should follow that of the left chart in figure 2.1. However, when the temperature reaches saturation temperature and the moisture begins to evaporate, the mass of the moisture should begin to decrease at a faster rate until the mass flows are in equilibrium.
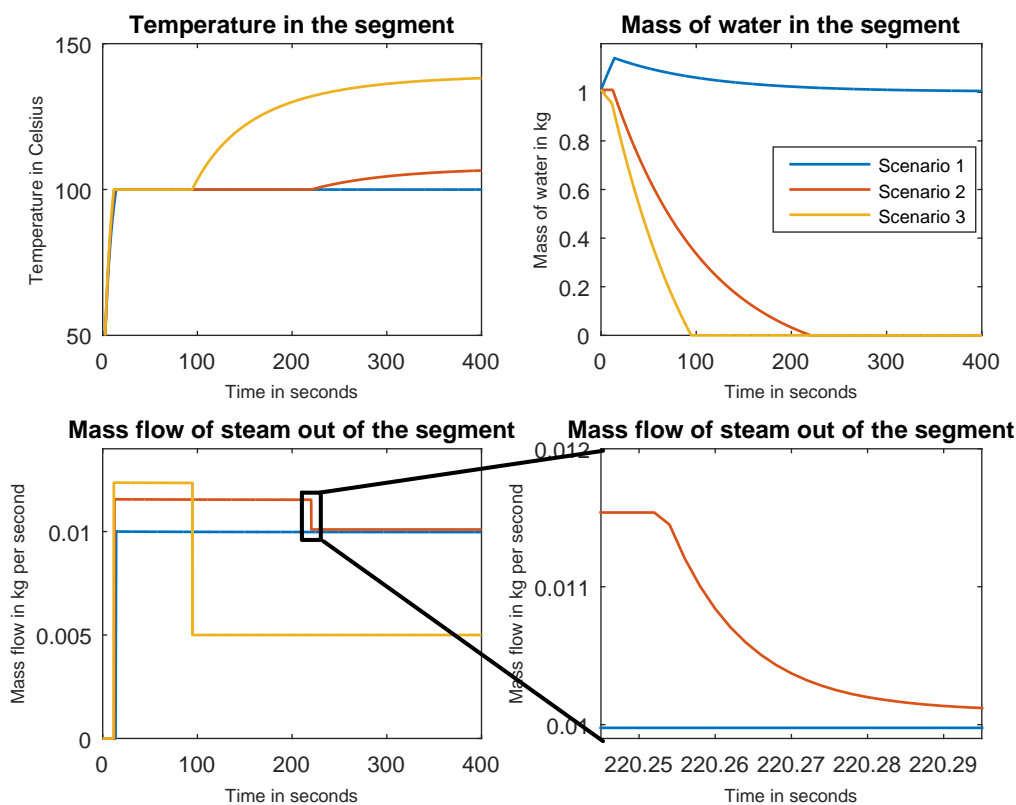
Figure 2.2: Results from the simulated model with constant heat flow.

The mass flow of steam leaving the segment should start at zero and then increase rapidly to some value when the saturation temperature is reached. This maximum value depends on how much heat is applied and on the mass flow of moisture entering the segment. The applied heat is split between heating the incoming moisture flow and vaporising the moisture already at saturation temperature. When all the moisture in the segment have been evaporated, the mass flow of steam should be numerical equal to the amount of moisture flowing in.

Figure 2.2 shows four charts with the results from a simulation with constant heat applied to the segment. The top left chart shows the temperature in the segment, and it behaves as expected. The lower inlet mass flow in scenario three results in higher final temperature and this is simply due to the fact there are less moisture to be evaporated. This chart also shows that for the first scenario, the temperature does not rise above the saturation temperature which indicates that the heat supplied is not sufficient to evaporate all of the inflowing moisture. This corresponds well with what the top right chart shows, where the mass of the moisture in the segment is shown.

The lower left chart shows the outlet flow of steam from the segment. It can be seen that the flows changes from zero at the time the saturation temperature is

reached. For scenario one, the flow stays constant for the entire simulation. In both scenario two and three the mass flow decreases to the numerical value of the inlet mass flow after all the moisture in the segment has been evaporated. The difference in the maximum mass flow between scenario two and three is due to the difference in extra inlet flow that have been heated to the saturation temperature.

The last chart, lower right, shows that the change in mass flow is not happening in steps even though it could appear that way.

2.3 Pyrolysis

2.3.1 Description

The purpose of the pyrolysis model is to estimate the composition of the volatile gases and the rate of which they are released. Like the evaporation model, there is an extra connector on the model where the volatile gases can be released to the flue gas. Again are the mass and energy balances, equation (2.1) and (2.2), extended to account for this extra connector. This connector also carries information of the mass fractions of the volatiles released.

The specific enthalpy of the volatile gases that leaves the pyrolysis model is a function of the mass fraction and the temperature:

$$h_{vol} = h(T, \vec{X})\tag{2.9}$$

where $\vec{X}$ is the mass fraction of the different species. The enthalpy of formation is included in the enthalpy as the species are created.

The rate of which the pyrolysis is occurring is as mentioned dependent on the heating rate of the biomass. However, for simplicity the rate is assumed to be a function of the mass in the segment and a temperature dependent rate constant k:

$$\dot{m}_{vol} = -m \cdot k\tag{2.10}$$

The rate constant is found from an Arrhenius expression:

$$k = A \cdot e^{\frac{-ER}{T}}\tag{2.11}$$

where the two factors, A and ER, are tuned so that the pyrolysis is starting at around $200°C$ and is finished in approximately ten seconds at $600°C$ [5].

As the pyrolysis products is assumed to consist of five gas species, H_2, CO, H_2O, CO_2, and CH_4, a total of five equations are needed. The first three equations are the elemental balances:

$$X_C = X_{CO_2,vol}X_{C,CO_2} + X_{CO,vol}X_{C,CO} + X_{CH_4,vol}X_{C,CH_4}\tag{2.12}$$

$$X_H = X_{H_2O,vol}X_{H,H_2O} + X_{CH_4,vol}X_{H,CH_4} + X_{H_2,vol}X_{H,H_2}\tag{2.13}$$

$$X_O = X_{H_2O,vol}X_{O,H_2O} + X_{CO_2,vol}X_{O,CO_2} + X_{CO,vol}X_{O,CO} \tag{2.14}$$

where $X_{i,j}$ is the mass fraction of i in j.

The fourth equation is found by setting the lower heating value (LHV) of the volatiles equal to the combined LHV of the pyrolysis products.

$$LHV_{vol} = X_{CO,vol}LHV_{CO} + X_{H_2,vol}LHV_{H_2} + X_{CH_4,vol}LHV_{CH_4} \tag{2.15}$$

Dealing with the last equation can be done in multiple ways. One of them is fixing one of the mass fractions, however this normally requires some knowledge about the fuel and can be sensitive to fuel changes. Another approach is to use an empirical correlation of some sort and using such a correlation is generally more fuel neutral than fixing a mass fraction. In this case a correlation for the ratio between two mass fractions from the devolatilisation of wood is used [6]:

$$\Omega = \frac{X_{CO}}{X_{CO_2}} = 1.94 \cdot 10^{-6}T^{1.87} \tag{2.16}$$

2.3.2　Validation

The validation of the model will be based on how the composition of the released volatiles changes with increased temperature and how well the gas composition corresponds with data from the literature [7]. There will not be made any validation regarding the rate of which the volatiles are released. A single simulation is needed for both validations and the input data listed in table 2.1 is used for this simulation. This data is for barley straw.

Table 2.1: Input data for the model.

Description	Value
Mass fraction of C	49.06%
Mass fraction of H	6.38%
Mass fraction of O	44.56%
LHV	18.5 $\frac{MJ}{kg}$

As the focus of the validation is the composition of the released volatiles, the mass and energy balances are temporally neglected. This means that there are no mass flow into or out of the segment and the temperature can be specified by a simple linear expression. Overall is this a series of steady state solutions at different temperatures. The temperature ranges from $200°C$ to $1355°C$.

Due to the nature of equation (2.16) it is expected that the fraction of CO will increase with the temperature and that the fraction of CO_2 will decrease. More H_2 should also be formed at higher temperature resulting in lower CH_4. The amount of H_2O released should not change drastic.
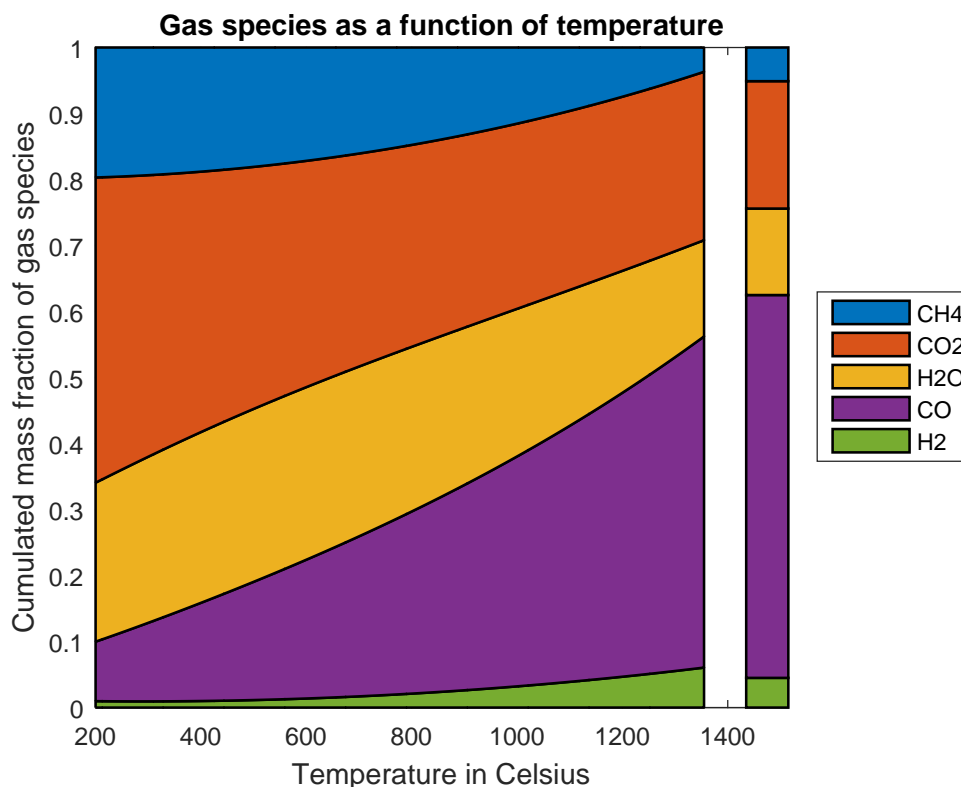
Figure 2.3: Mass fraction of pyrolysis products as a function of temperature.

The results of the simulation is shown in figure 2.3. On the x-axis is the temperature in Celsius, and on the y-axis is the cumulated mass fraction. The big square on the left shows the gas composition in the temperature range, and the bar on the right shows the composition from the literature data. The change in gas composition with temperature aligns with the expected.

The literature data corresponds well with that of the model, however there are some smaller differences and the biggest difference appears to be the ratio of CO to CO_2. This could be the result of the ratio correlation being for wood, where the data used in this simulation is for barley straw.

2.4   Char Conversion

2.4.1   Description

The char conversion model is where the char is converted by combustion and gasification. Even though this model also contains a third connector just as the evaporation and pyrolysis model, the connector for this model is different. In the two other models the flue gas only had to flow in one direction, out of the model. However, this is different for the char conversion model, as oxidiser and gasification species have to

enter the model while the products of these processes exit the model. The connector therefore contains seven individual mass flows, one for each flue gas species, and the connector also caries information regarding the concentration of the different gas species in the flue gas volume.

The conversion rate of char is controlled by two factors, diffusion and kinetics, and split into three regimes. The first regime, also referred to as zone 1, is generally occurring at low temperatures and high reactant concentration. In this zone the chemical kinetics are the limiting factor on the conversion rate. The third regime, zone 3, is at the opposite end of the spectrum. Here the diffusion of reactants to the char is the limiting factor, and this generally happens at higher temperatures and low concentration. Between those two regimes is zone 2 where the conversion rate is controlled by both diffusion and kinetics.

There are multiple reactions happening at the same time and at different rates, so only the most significant are included. Table 2.2 lists four reactions and their relative reaction rates. The data shows that combustion with O_2 is considerably faster than the gasification with CO_2 and H_2O, and that the gasification with H_2 can be excluded from the model.

Table 2.2: Relative reaction rate for four reactions involving carbon [3].

Reaction	Temperature for which products are favoured	Relative reaction rate (T=800 K, y=0.1)
$C + CO_2 \rightarrow 2CO$	T > 950 K	1
$C + H_2O \rightarrow CO + H_2$	T > 950 K	3
$C + 2H_2 \rightarrow CH_4$	T < 820 K	$3 \cdot 10^{-3}$
$C + O_2 \rightarrow CO_2$	T < 5000 K	10^5

The rates are implemented as functions that can be exchanged easily by selection from a drop-down menu. Each function can then be written in such a way that it can account for both diffusion and kinetics controlled conversion. The conversion rates are then:

$$R_i = r_i\left(T, y_i\right) \cdot m \tag{2.17}$$

where

- R_i is the absolute reaction rate with reactant i in $\frac{kg}{s}$

- r_i is the reaction rate function with reactant i in $\frac{1}{s}$

- y_i is the mole fraction of reactant i

An example of one of these functions is shown in figure 2.4, where the rate is sketched for both diffusion and kinetic controlled. The effective rate is the minimum of these two rates and there is a clear switch between the two at a temperature of approximately $950°C$.

These reaction rates are subtracted from the mass balance, equation (2.1), as char is converged from solid to gas. The mass flow of each gas species are calculated from the reaction rate of the reactions in which they are present. For N_2 and CH_4 the mass flows are zero, the other five mass flows are:

$$\dot{m}_{O_2} = \frac{M_{O_2}}{M_C} R_{O_2}$$

$$\dot{m}_{H_2O} = \frac{M_{H_2O}}{M_C} R_{H_2O}$$

$$\dot{m}_{CO_2} = \frac{M_{CO_2}}{M_C} (R_{CO_2} - R_{O_2}) \tag{2.18}$$

$$\dot{m}_{H_2} = -\frac{M_{H_2}}{M_C} R_{H_2O}$$

$$\dot{m}_{CO} = -\frac{M_{CO}}{M_C} (R_{H_2O} + 2R_{CO_2})$$

As with the pyrolysis model, there will be no validation of the rate at which the char is converted.

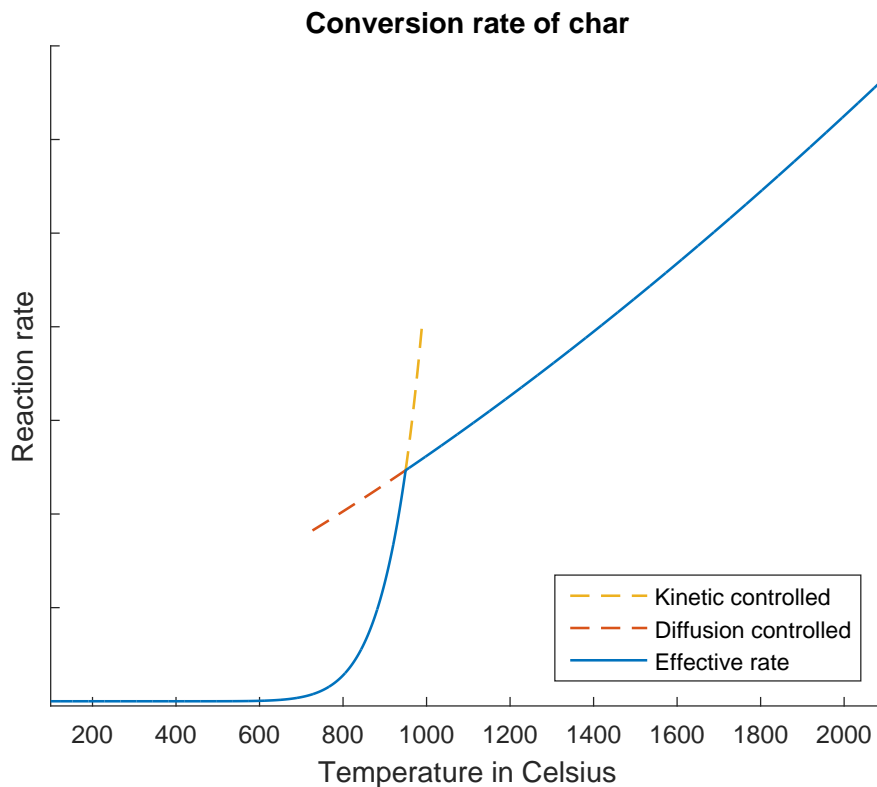This concludes the individual fuel models, and the next chapter describe the flue gas models.

Figure 2.4: Conversion rate of char as a function of temperature.

CHAPTER 3

Flue Gas Models

This chapter will describe the two flue gas models; the volume model and the flow model. The model is split into a volume and flow model, due to how the steam connectors work and how the equations are solved in Modelica. If the models where not split, it would result in high-index non-linear differential algebraic equations, which are difficult to solve [8].

3.1 Flue Gas Volume Model

The volume model is the first model the flue gas enters, here it is mixed with the steam from the evaporation and the products from the pyrolysis process. The model has a volume where mass and energy are accumulated over time, and there are multiple inputs and outputs. Mass and energy balances of this model are therefore equal to those of the fuel models:

$$\frac{\mathrm{d}m}{\mathrm{d}t} = \sum \dot{m} \tag{2.1}$$

$$\frac{\mathrm{d}U}{\mathrm{d}t} = \sum \left(\dot{m}h \right) + \dot{Q} \tag{2.2}$$

However, this model also contains a species balance for the different gas substances:

$$\frac{\mathrm{d}m_i}{\mathrm{d}t} = \sum \dot{m}_i \tag{3.1}$$

Furthermore, the model contains a static momentum equation:

$$p_{in} = p_{out} \tag{3.2}$$

The volume of the flue gas depends on the bed height and the composition of the bed. This information is relayed to the model as a causal height input from the individual bed models; evaporation, pyrolysis, char conversion, and ash cool down.

The heights are then multiplied with void fraction parameters to indicate how much each part contributes to the flue gas volume.

Using the heights from the individual bed models, results in an algebraic loop or circular dependency, as the flue gas volume is used to calculated the oxygen concentration and thereby the rate at which char is combusted and the height of the bed changes. This results in extended simulation time and in some cases unsuccessful simulations.

To circumvent this dependency, a differential decoupling was introduced:

$$\tau_H \frac{\mathrm{d}\tilde{H}}{\mathrm{d}t} = \vec{H} \bullet \vec{\alpha} - \tilde{H} \tag{3.3}$$

where

- τ_H is a time constant for the height in s

- $\vec{H}$ is a vector of size four with the actual heights of each bed model in m

- $\vec{\alpha}$ is a vector of size four with the void fractions

- $\tilde{H}$ is the decoupled height of the flue gas volume in m

The decoupled height follows the actual height, but sudden changes are smoothed out by the magnitude of the time constant. This can be seen in figure 3.1, where the blue graph is the actual value and the orange is the decoupled value. On the x-axis is the number of time constants and the y-axis shows the normalised height value. Initially, at zero time constants, the actual value is stepped from zero to one and after one time constant the decoupled value is roughly 63% of the step value. After five time constants the decoupled value is below 1% off, and after ten time constants the value is equal for practical purposes.

With the decoupled height, the flue gas volume is simply calculated as:

$$V_{gas} = L \cdot W \cdot \tilde{H} \tag{3.4}$$

where

- V_{gas} is the flue gas volume in m^3

- L and W are the length and width of the bed segment in m

3.2   Flue Gas Flow Model

After the volume model, the flue gas enters a flow model where the pressure loss of the bed layer is calculated. The flow model has only a single input and output
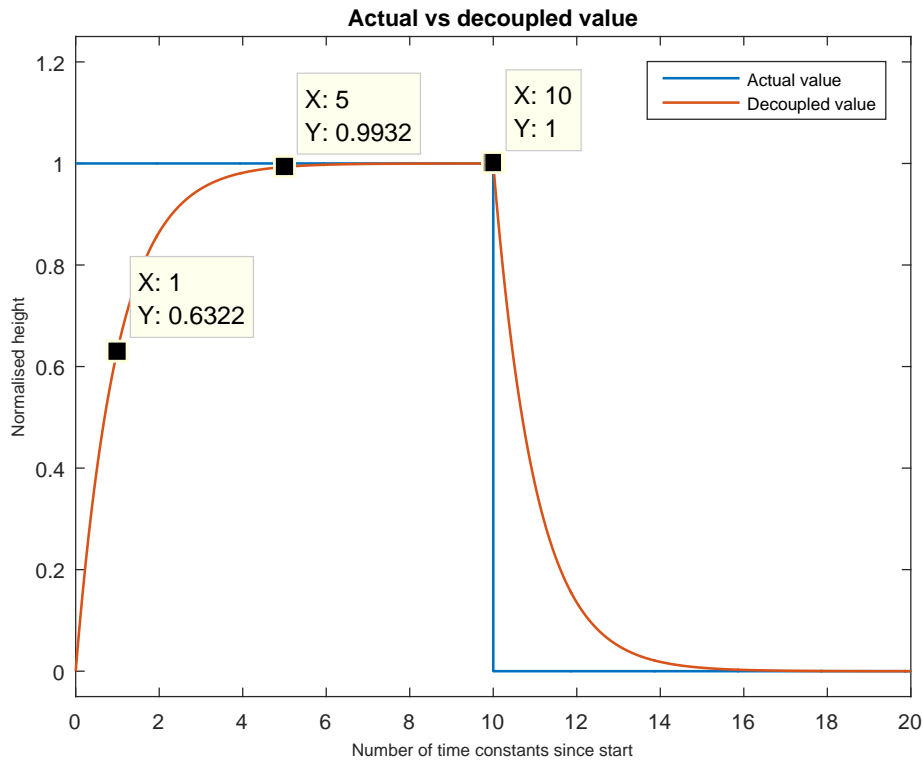
Figure 3.1: Difference between actual height and decoupled height.

and no volume, mass and energy can therefore not accumulate causing the mass and energy balances to be static:

$$0 = \dot{m}_{in} + \dot{m}_{out} \tag{3.5}$$

$$0 = (\dot{m}h)_{in} + (\dot{m}h)_{out} \tag{3.6}$$

However, the flow models contains a non-static momentum equation:

$$\Delta p = p_{in} - p_{out} \tag{3.7}$$

Where the Δp is the pressure loss over the bed height calculated with an equation of the following format:

$$\Delta p = \frac{\Delta p_n}{H_n \dot{m}_n^2} H \dot{m}^2 \tag{3.8}$$

where

- H is the bed height in m

- $\dot{m}$ is the mass flow of the flue gas in $\frac{kg}{s}$

- Subscript n indicates nominal values

An equivalent equation exist for each of the four parts of the bed, and the pressure losses from each part are summed together to one total pressure loss.

19

CHAPTER 4

Complete Segment

With the different sub-models of the segment covered, they are all combined together to form a single segment. This is done by using a few extra components, some which are custom made and some which are part of the Modelica standard library. Two of the custom components handles the splitting of the fuel into the four parts; moisture, volatiles, char, and ash. One model that split the fuel and one that combines the fuel. There is also a custom model that changes the reference of the steam leaving the moisture evaporation model, from water to ideal gas. The standard models consist of fluid and thermal connectors, as well as components for thermal resistance.

The complete segment is simulated under different scenarios. Each scenario correspond to one of the five processes that the fuel undergoes from raw fuel to ash; heat up, evaporation of moisture, devolatilisation, char conversion, and ash cool down. The different scenarios were then subjected to changes in the boundary conditions and the behaviour of the segment was observed. These changes were introduced when the segment was in steady state. For some of the simulations this could be achieved almost from the start by setting a few initial conditions, but for others a long transient period was unavoidable.

The initial boundary conditions that are supplied are:

- Steady temperature of the segment above and below the segment.

- Steady flow of fuel entering the segment.

- Steady flow of flue gas entering the segment.

4.1 Heat Up

The first scenario is the heat up scenario. In this scenario the steady state segment temperature is around $44°C$. The segment above has a temperature of $55°C$ and

the segment below has a temperature of $35°C$. Fuel is entering the segment with a temperature of $40°C$ and the flue gas is entering with a temperature of $50°C$.

After 50 seconds, the temperature of the flue gas entering the segment is ramped up by 100 degrees over 10 seconds. It is kept there for 130 seconds before it is ramped down again over the same interval. This is done to simulate the preheating of the primary air. To simulate an increase in primary air flow, the mass flow of flue gas is ramped up by 50% over 15 seconds at the 300 second marker, and down again after 120 seconds.

At the 500 second marker, the fuel flow entering the segment is ramped up by 40% over 10 seconds. After 130 seconds it is ramped down again. This is done to simulate the addition of fuel on the grate, and therefore is the temperature of the fuel also ramped down by 10 degrees in the interval.

The results from this scenario can be seen in figure 4.1 where the blue graph shows the fuel temperature of the segment and the orange graph shows the temperature of the flue gas. The black vertical lines indicates the start time of each change in boundary conditions, and the ending of those changes is not marked.

At the first change, marked by the dashed line, both the temperature of the fuel and flue gas undergoes a significant increase. This corresponds well with the idea that preheating the primary air speeds up the heating up and drying process.

The second change is marked with a dash-dot line, here the flow of flue gas entering the segment is increased. This results in an negligible increase in temperature for both the fuel and flue gas. These two different cases shows that it is considerably more efficient to add preheating rather than redistributing the primary air.

At the last change, which is marked by a dotted line, the fuel flow is increased and the fuel temperature decreased. Here there are also a change in the temperatures of fuel and flue gas, however it is a small decrease. This corresponds well with the conviction that increasing the mass flow of fuel will result in a decrease of the boiler's temperature at first.

4.2   Moisture Evaporation

The next scenario is the evaporation scenario. In this scenario the steady state segment temperature is $100°C$, which is the saturation temperature of water. The segment above has a temperature of $105°C$ and the segment below has a temperature of $100°C$. Fuel is entering the segment with a temperature of $44°C$ and the flue gas is entering with a temperature of $103°C$.

The first changes in boundary conditions are introduced at the start. This time is the temperature of the fuel entering the segment ramped up in accordance with the segment temperature of the heat up scenario, 18 degrees in 10 seconds. As with the first scenario, the temperature is ramped down again over the same interval after
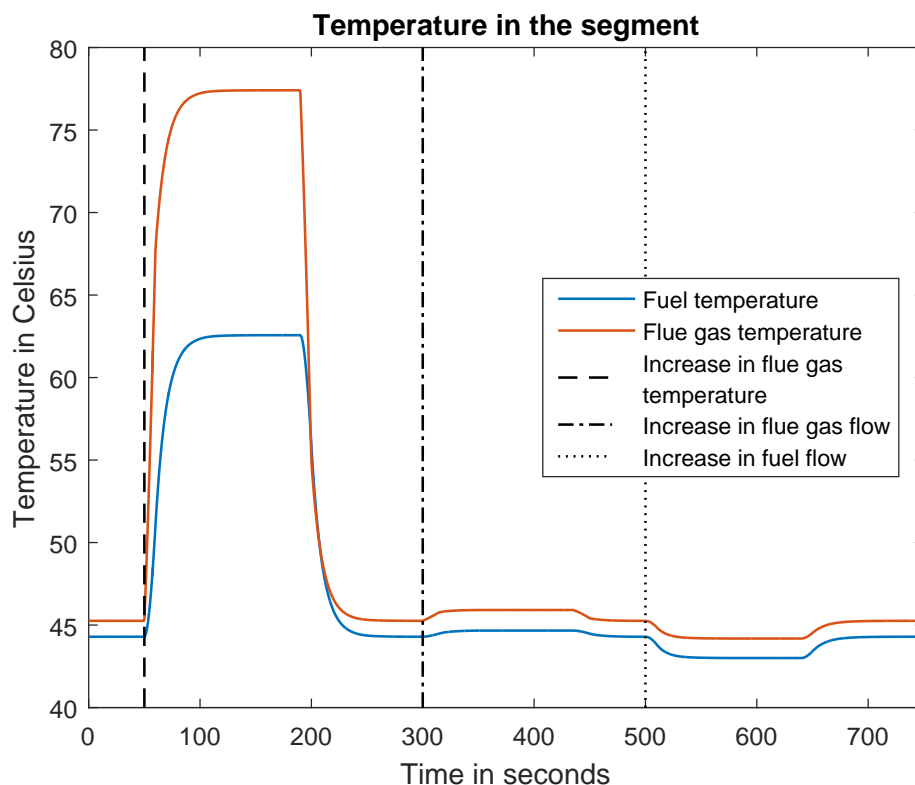
Figure 4.1: Temperature of the fuel and flue gas in the segment during the heat up scenario. Black lines mark changes in boundary conditions.

130 seconds. This simulates what happens during the evaporation process when the segment before is subjected to preheated primary air. This could be the case if the bed is divided into different zones and only one of them receives preheated air.

After 1500 seconds is the temperature of the entering flue gas ramped up by 20 degrees in 10 seconds, however it is not ramped down again. This is done to simulate what happens when the primary air is preheated in a zone where the evaporation process is ongoing.

Figure 4.2 shows the results from this scenario. Only the mass of moisture is displayed as both the temperatures of the fuel and flue gas in the segment is constant till all the moisture has evaporated. There are also no markings, as the first change is at 0 seconds and the graph does a clear drop where the second changes are introduced.

The first change in boundary conditions causes a decrease in the mass of moisture in the segment, implying that more moisture is evaporated. However, it is barely noticeable compared to the decrease at 1500 seconds where the temperature of the flue gas entering the segment is ramped up by 20 degrees. This increase in flue gas temperature causes all the moisture to evaporate and the fuel in the segment is therefore ready for the next process, the devolatilisation. As for the heat up
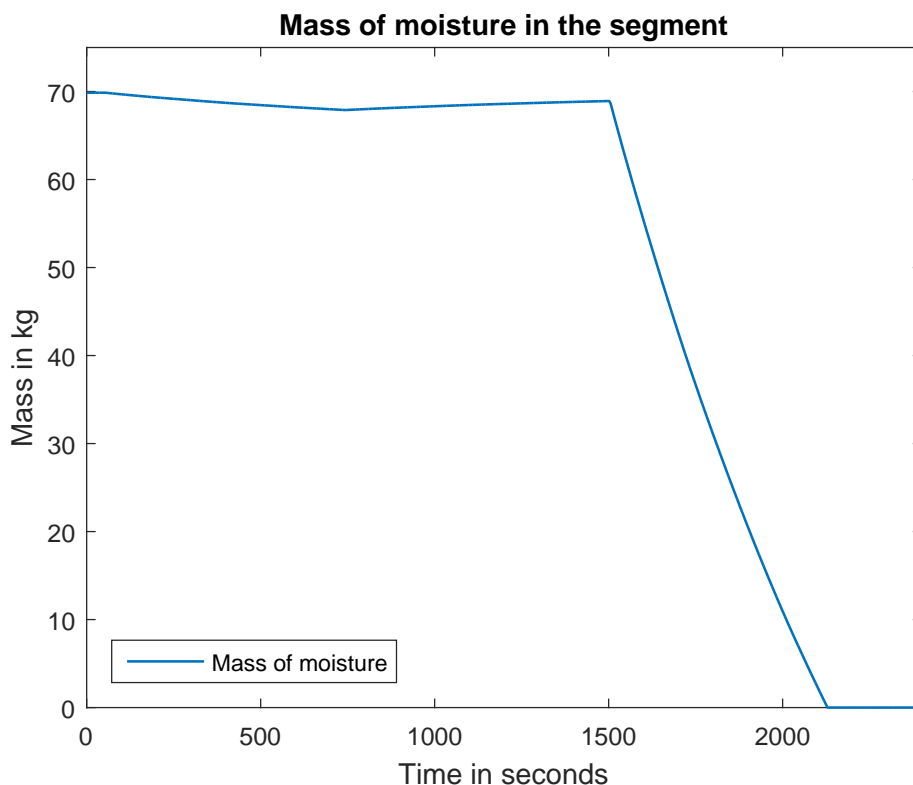
23

Figure 4.2: Mass of moisture in the segment during the evaporation scenario.

scenario, this is expected when the primary air is preheated.

4.3   Devolatilisation

The third scenario is the devolatilisation scenario. In this scenario the steady state segment temperature is around $280°C$. The segment above has a temperature of $300°C$ and the segment below has a temperature of $275°C$. Fuel is entering the segment with a temperature of $200°C$ and the flue gas is entering with a temperature of $250°C$.

There is only one change in boundary conditions in this scenario, which is the temperature of segment above being ramped up by 20 degrees over 10 seconds after 50 seconds. This is done to simulate that the temperature in the boiler increases, and thereby heating up the upper layer of the bed. The temperature inside the segment is raised about nine degrees by doing this.

Figure 4.3 shows the results from this simulation in two charts. The top chart shows how the gas composition changes with the increased temperature inside the segment. The mass fractions are normalised by the value they had before the temperature increase. The amount of H_2O and CO released is increased while the temperature is higher, and this corresponds with the results from the test of the
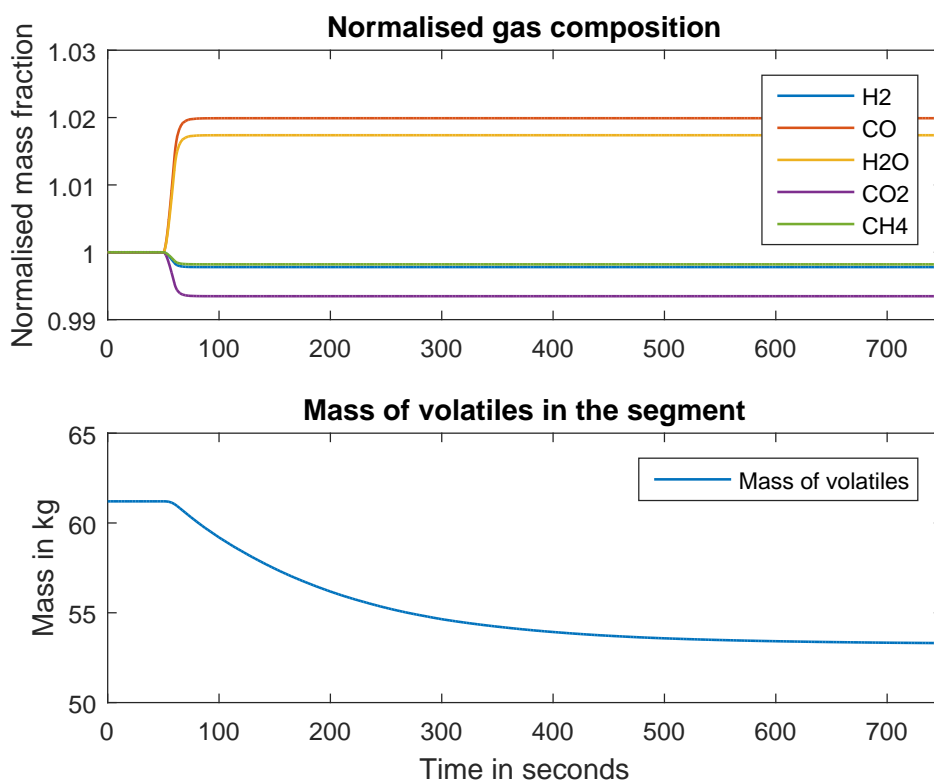
Figure 4.3: Gas compostion and mass of volatiles in the segment during the devolatilisation process.

individual model. On the bottom chart is the amount of volatiles in the segment sketched. When the temperature is increased the amount of volatiles released is also increased, and the mass of volatiles inside the segment decreases.

These results show that with an increase in boiler temperature the composition and the rate of release of volatiles are affected.

4.4 Char Conversion

The next scenario is the char conversion. In this scenario the steady state segment temperature is around $1025°C$. The segment above has a temperature of $1200°C$ and the segment below has a temperature of $1000°C$. Fuel is entering the segment with a temperature of $1000°C$ and the flue gas is entering with a temperature of $450°C$.

To simulate an increase in primary air flow, the mass flow of the flue gas is ramped up such that the mass fraction of O_2 is increased from 0.1 to 0.15. This is done over 10 seconds at the 50 seconds marker.

Figure 4.4 shows the results in two charts. The top one shows the temperature in the segment and the bottom chart shows the conversion rates, where the gasification
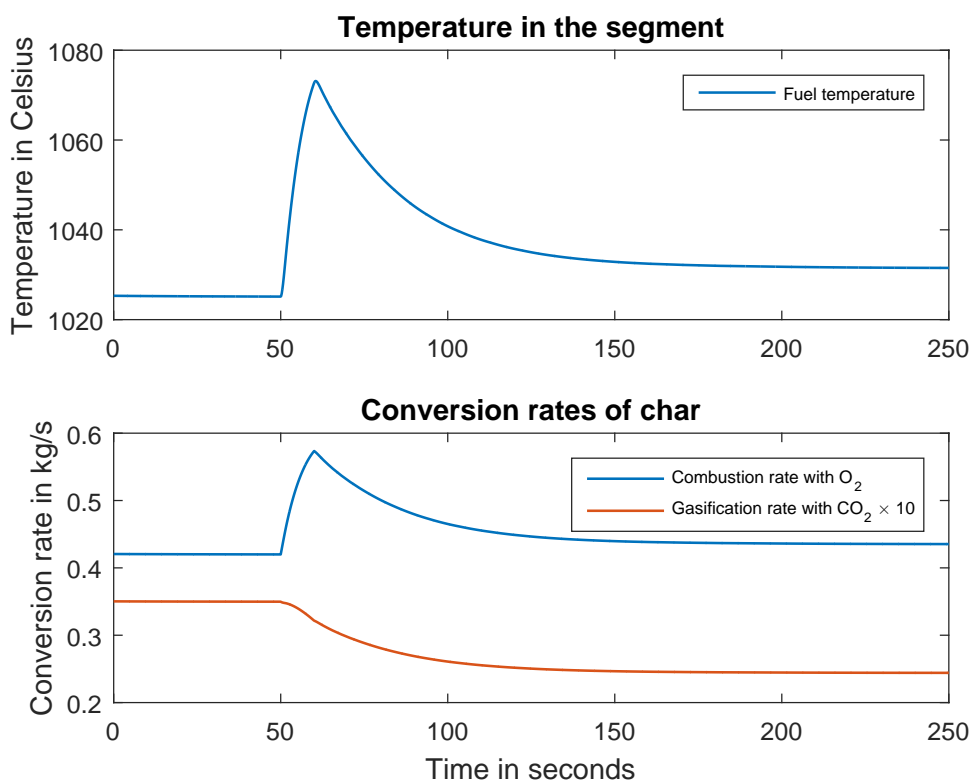
Figure 4.4: Temperature and conversion rate of char in the segment during the char conversion process.

rate have been multiplied by ten. When the mass flow of flue gas is ramped up, the combustion rate and temperature increases as well and peaks after about 10 seconds. After another 100 seconds they stabilise at a level higher than at the start.

The overall increase in combustion rate and temperature is expected as there is more O_2 available to react with the char in the segment. However, the peak behaviour is somewhat a surprise. An explanation can be that the char combustion is supplying heat faster than it can be transferred to the flue gas.

4.5   Ash Cool Down

The last scenario is the cooling down of the ash. In this scenario the steady state segment temperature is around $450°C$. The segment above has a temperature of $550°C$ and the segment below has a temperature of $450°C$. Fuel is entering the segment with a temperature of $500°C$ and the flue gas is entering with a temperature of $250°C$.

After 50 seconds the mass flow of flue gas is ramped up by 50% over 15 seconds, which simulates an increase in primary air flow. This increase in mass flow reduces the temperature of the ash as would be expected, but it also reduces the temperature
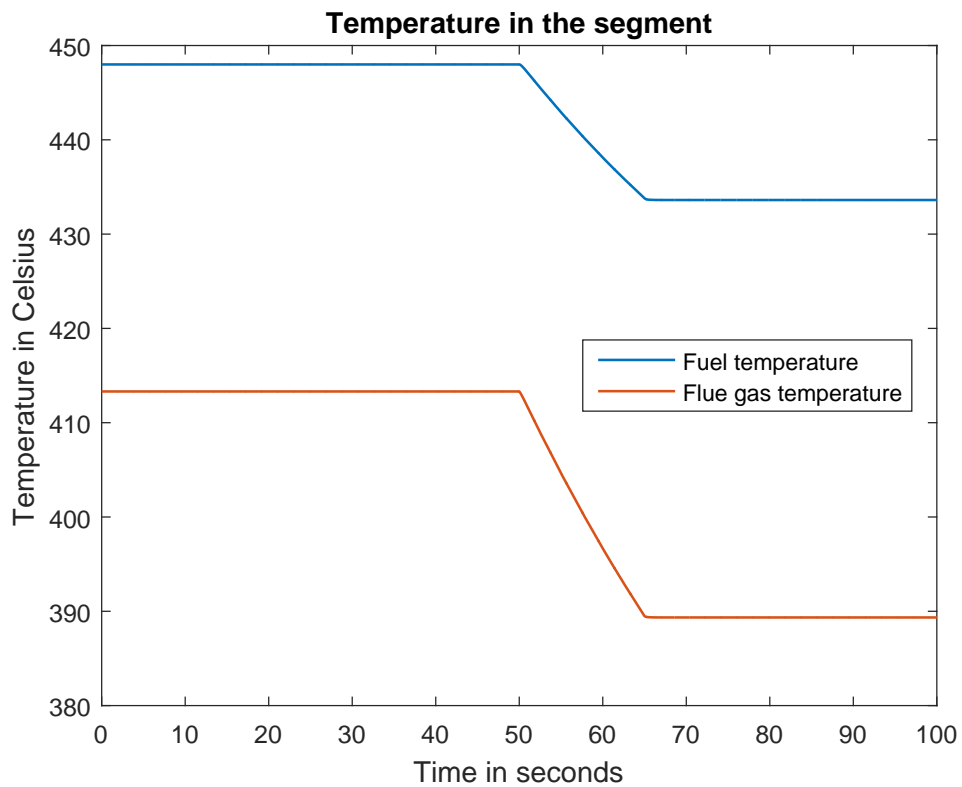
Figure 4.5: Temperature of the fuel and flue gas in the segment during the cooling of ash scenario.

of the flue gas leaving the segment. So even though more energy is recovered from the hot ash, the temperature in the boiler is still going down which could slow some of the other processes.

CHAPTER 5

Gas Combustion

As explained in chapter 1 there are gas combustion segments just above each bed column. These gas combustion segments are where the volatile gases released during the pyrolysis process are combusted. The modelling of this segment proved to be difficult and it even uncovered an error in the Modelica standard library. At first an unsuccessful approach with chemical kinetics was tried, then a somewhat successful approach with chemical equilibrium, before the last and more simplified approach with analytical combustion reactions proved to be the most successful.

5.1   Chemical Kinetics Approach

The first approach was based on chemical kinetics and it is similar to an example Michael Tiller, a member of the Modelica Association and active in the ongoing development of Modelica, described in his online book "Modelica by Example" [9]. A more complex version of the approach was also introduced at the 11th International Modelica Conference in France in September 2015 [10].

The idea was that two Modelica components was to be modelled. One to handle and keep track of the concentration of the chemical species, so-called solution component, and one to handle the reactions, so-called reaction component. Then for each reaction in the combustion process, one reaction component can be connected to the solution component, and the different parameters can be set. Figure 5.1 shows the parameter window. Here the reactants and products can be chosen, and the coefficients for an Arrhenius expression for the rate coefficient can be specified.

The Arrhenius expression for the rate coefficient is of the form:

$$k = AT^b e^{\frac{-E}{R_u T}} \tag{5.1}$$

where

Figure 5.1: Parameters for the reaction component.

- A, b, and E are experimental determined constants

- T is the temperature in K

- R_u is the universal gas constant in $\frac{J}{molK}$

Using the parameter window, the reaction can also be set to reversible and this activates the determination of the the reverse rate coefficient from the equilibrium constant which is found from the Gibbs free energy. The Modelica standard library contains data for the 1241 ideal gases listed in the "NASA Glenn Coefficients for Calculating Thermodynamic Properties of Individual Species" [11], which allows for determination of thermal properties. However, the Gibbs free energy is not included as one of those thermal properties, so custom functions had to be written and data records had to be extended. This proved to be comprehensive, since a combustion of H_2 with O_2 contains 19 reactions involving a total of eight species [12].

To test the approach a total of 21 reaction components, with the associated Arrhenius constants, was modelled. One for each of the reactions and two extra for reactions that had multiple rate coefficients. The simulation succeeded, but the results turned out not matching what was expected. Each of the custom components, functions, and data records was debugged and even though minor mistakes were found and corrected, it did not solve the problem.

The problem turned out to be in the ideal gas data where the gas species HO_2 had wrong data. The data was imported wrong from the database, as a section

which was supposed to be blank was filled with zeroes.

The approach was cancelled as the 53 species involved in the 325 reactions for the combustion of CH_4 could contain more database errors [13]. It was also a comprehensive task to create the needed reaction components.

5.2 Chemical Equilibrium Approach

A more simplified approach that does not require the kinetic rates for all the different reactions is the chemical equilibrium approach. In this approach there are no kinetics involved as it is based on the equilibrium constants for the different reactions. However, determining the equilibrium constant for each reaction is not a simple task. Instead a process known as minimisation of Gibbs free energy is used.

From the second law of thermodynamics it can be deducted that the total entropy of an isolated reactive system will increase till equilibrium is reached. The Gibbs free energy of such a system is defined from the enthalpy, entropy, and temperature:

$$G \equiv H - T \cdot S \tag{5.2}$$

where H is the total enthalpy in $\frac{kJ}{kg}$ and S is the total entropy in $\frac{kJ}{kg \cdot K}$.

Therefore will a maximisation of the entropy be equal to a minimisation of the Gibbs free energy and the system will be at equilibrium when this is reached. The total Gibbs free energy for a given system can be found from the sum of the Gibbs free energy for each species.

When solving the minimisation of the Gibbs free energy, a mathematical method called "Lagrange Multipliers" is generally applied. The method manipulates the minimisation problem by introducing a number of variables and the same number of extra equations, and changes it into an equation system.

Overall gave the chemical equilibrium approach relative good results, but at the cost of extended simulation time, as well as problems with convergence of the solution. In order to make the solving process more robust and the simulation time faster, the total number of function calls was reduced. This was done by converting the Gibbs free energy for the seven gas species into fifth degree polynomial expressions in the temperature range of 100-6000 K. However, the simulation speed was still low.
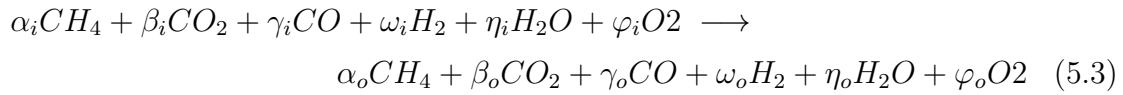
5.3 Analytical Approach

To simplify the gas combustion even further, an analytical approach was tested. This approach is based on analytical balancing of the overall chemical reaction with the following assumptions:

- Instantaneous reactions.

- CH_4 combustion occurs before H_2 combustion.

- Combustible gases can not co-exist with O_2.

These assumptions results in three cases.

The manually balancing of the reaction is an extensive task, and the analytic solution is only valid under the assumptions made and as long as no extra non-inert species are introduced.

The total reaction is:

$$\alpha_i CH_4 + \beta_i CO_2 + \gamma_i CO + \omega_i H_2 + \eta_i H_2O + \varphi_i O2 \longrightarrow$$
$$\alpha_o CH_4 + \beta_o CO_2 + \gamma_o CO + \omega_o H_2 + \eta_o H_2O + \varphi_o O2 \quad (5.3)$$

where the Greek letters indicates the mole flow of the species and the subscripts i and o specifies inlet and outlet. N_2 is assumed to be inert and does therefore not appear in the reaction.

For cases with less than the stoichiometric amount of O_2 available, the water-gas shift reaction is used to solve for the gas species. The water-gas shift reaction is:

$$CO + H_2O \rightleftharpoons CO_2 + H_2 \quad (5.4)$$
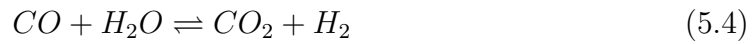
and the analytical solution to the amount of CO_2 at equilibrium is:

$$\beta_o = \frac{-b + \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \quad (5.5)$$

where a, b, and c are given by:

$$a = K_p - 1$$
$$b = K_p \left(C - O + 0.5H\right) \quad (5.6)$$
$$c = C \left(C - O\right)$$

and C, H, and O are the mole flow of carbon, oxygen and hydrogen entering the segment:

$$C = \alpha_i + \beta_i + \gamma_i - \alpha_o$$
$$H = 4\alpha_i + 2\omega_i + 2\eta_i - 4\alpha_o \quad (5.7)$$
$$O = 2\beta_i + \gamma_i + \eta_i + \varphi_i$$

lastly is K_p the equilibrium constant of the water-gas shift reaction.

5.3.1 Case 1 - Very Limited O_2

The first case is for very limited O_2. The amount of O_2 is insufficient to burn all the CH_4 and it is therefore present in the flow leaving the segment. The products from the combustion of CH_4 enters equilibrium via the water-gas shift reaction. There are no O_2 leaving and the amount of CO_2 leaving is found by equation (5.5). For the other species, the amount leaving is:

$$
\begin{aligned}
\alpha_o &= \alpha_i - \varphi_i \\
\gamma_o &= C - \beta_o \\
\omega_o &= 0.5H - \eta_o \\
\eta_o &= O - (2\beta_o + \gamma_o)
\end{aligned}
\tag{5.8}
$$

5.3.2 Case 2 - Limited O_2

In the second case there is enough O_2 for the combustion of CH_4 but still an insufficient amount to achieve a complete combustion. There are still no O_2 leaving and in this case there is also no CH_4 leaving. As in the first case, the products of the CH_4 combustion enters equilibrium via the water-gas shift reaction and the amount of CO_2 is found by equation (5.5). For the other species the amounts are:

$$
\begin{aligned}
\gamma_o &= C - \beta_o \\
\omega_o &= 0.5H - \eta_o \\
\eta_o &= O - (2\beta_o + \gamma_o)
\end{aligned}
\tag{5.9}
$$

5.3.3 Case 3 - Excess O_2

The last case is where there is an excess amount of O_2 available and the combustion is therefore complete. There are no CH_4, H_2, or CO leaving the segment. For this case the amount for the leaving species is:

$$
\begin{aligned}
\beta_o &= C \\
\eta_o &= 0.5H \\
\varphi_o &= 0.5O - \beta_o - 0.5\eta_o
\end{aligned}
\tag{5.10}
$$

5.4 Results

5.4.1 Combustion of CH_4

The model from the analytical approach was simulated with a steady flow of CH_4 and an increasing flow of air, N_2 and O_2 in the ratio 3.29:1 on mass basis. The temperature of both the CH_4 and the air were set to $25°C$ before entering the combustion segment. Figure 5.2 shows the mass fractions and the temperature of
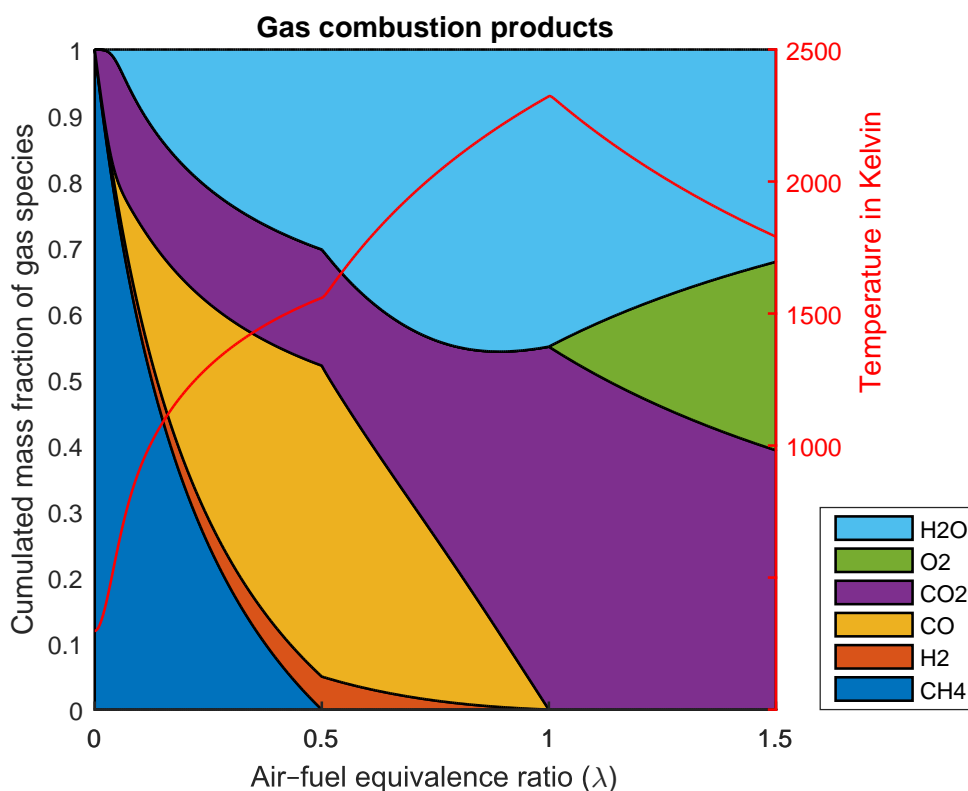
Figure 5.2: Gas species after gas combustion with analytical approach.

the products leaving the segment as a function of the air-fuel equivalence ratio. The air-fuel equivalence ratio is the ratio of actual air-fuel ratio to the stoichiometric air-fuel ratio.

Starting on the left on the x-axis there is no O_2 available and there is therefore only CH_4 leaving the segment. Adding O_2 to the segment causes other gas species to occur in the mix. At first it is mostly CO_2 and a small amount of H_2 as the temperature is still quite low, and the water-gas shift reaction is still favouring these gases. As more O_2 is added, more CH_4 is combusted and the temperature rises. With this increase in temperature the equilibrium constant of the water-gas shift reaction begins to allow for more CO and H_2O in the mix. This is indicated by a more or less constant mass fraction for CO_2 and H_2.

At an air-fuel equivalence ratio of 0.5 the combustion is moving from case 1 to case 2. There is no longer any CH_4 in the mix leaving the segment and the combustion of H_2 is now kicking off. This together with more CO being turned into CO_2 gives the temperature a boost. The amount of CO and H_2 decreases as the air-fuel equivalence ratio increases. At a ratio of 1 there are no combustible gases left and the temperature peaks at 2324 K, which is about 100 K higher than the adiabatic flame temperature.

Further increasing the amount of O_2 in the flow entering the segment, only results

	N_2	H_2	CO	O_2	H_2O	CO_2	CH_4	T
In	0.1429	0.1429	0.1429	0.1429	0.1429	0.1429	0.1429	1500
Chemkin	0.125	0.2035	0.2935	7.757E-6	0.2965	7.852E-2	7.015E-11	2285
Equilibrium	0.125	0.2035	0.2965	7.776E-6	0.2965	7.850E-2	7.070E-11	2287
Analytical	0.125	0.2030	0.2970	0	0.2970	7.802E-2	3.73E-17	2289

Table 5.1: Comparison of multiple methods for establishing gas composition.

in a decreasing temperature since more gas has to be heated.

5.4.2   Gas Mixture

The main purpose of the combustion segment is to model the combustion of the gas leaving the bed and estimate the gas composition of the products. To compare the analytical model with the equilibrium model, the model is simulated with a steady flow of a mixture of gases. The gas mixture is set to have a temperature of 1500 K, and the mole fraction of each species is set to 1/7. Table 5.1 lists the mole fractions and the results are also compared with a simulation from Chemkin, a widely used software to simulate chemical reactions, which uses chemical kinetics. There are no practical difference between the three simulations and it is therefore acceptable to use the faster analytical model.

The analytical model does produce a different result for combustion with excess air than the equilibrium model, however this is acceptable as long as the model is used just above the grate. For biomass fired fixed grate boilers the primary air does generally not contain the needed O_2, which is added with the secondary or even tertiary air. A combustion segment with the equilibrium approach could then be used further down stream.

CHAPTER 6

Conclusion

During this project a bed segment has been modelled in the modelling language Modelica. This was done by creating four sub-models for the fuel; moisture evaporation, pyrolysis, char conversion, and ash cool down and two sub-models for the flue-gas; flue gas volume and flue gas flow. The individual models were described and validated by comparing the behaviour of the model with either expected behavior or literature data.

After the sub-models were combined together to form a bed segment, the complete segment was simulated under different scenarios corresponding to different locations in the bed. For each scenario the boundary conditions was altered a number of times to observe the behaviour of the segment. The segment behaved as expected for the main part, however there were some surprises.

One example was the temperature during the char conversion. When extra flue gas, and thereby O_2, was added to the segment, the temperature spiked before settling back down again. Even though it was expected that the temperature would rise, the peak behaviour was not expected. However, as mentioned it could be caused by the heat being transferred to the flue gas slower than the combustion could supply it.

Another observation from the scenarios that is worth mentioning is the fall in flue gas temperature during the cooling of ash. The extra flue gas added to cool the ash, resulted in lower flue gas temperature out of the segment and thereby lower temperature in the boiler. This behaviour combined with the behaviour in the other scenarios, confirms the need for a fixed grate combustion model.

During this project a model for a gas phase combustion was created, with three different approaches; chemical kinetics, chemical equilibrium, and analytical. The chemical kinetics approach turned out to give the biggest flexibility, however it did require a lot of tedious and error prone work to specify the numerous coefficients for the rate constants. It was also further complicated by an error in the Modelica ideal

gas package. The chemical equilibrium approach gave practically the same results, but under more strict assumptions. One of which being the assumption of rapid reactions and/or high residence time. For the analytical approach the assumptions were even more strict, and the most significant was the assumption of no dissociation of CO_2 into CO and O_2.

Even with the strict assumptions, the results from the analytical approach matched those of the two other approaches under the circumstances present just above the bed. Furthermore, had the model from the analytical approach the fastest simulation time, which is vital as it is to be repeated above each bed column. However, one of the other two approaches should be used for modelling of the secondary combustion area.

CHAPTER 7

Future Work

A lot of work have been put into the creation of the bed segment and gas combustion segment, and even though they both behave as expected under simulations there are still many areas that can be refined. Some suggested tasks are:

Combine segments: Combining segments in length and height and test how they interact with each other is a logical extension of this project. Even though it might appear simple, the combination of multiple segments with interactions back and forth could expose hidden errors in the segment model.

Adding more complex models: Some of the sub-models in the segment are simplified, and additional models could increase the flexibility of the model.

Validation and tuning with multiple sources: Validation and tuning are only possible to a lesser extent until the model is complete, but at that point it would be preferable to validate it with multiple sources.

Bibliography

[1] Johannes Veje. Preparation for Fixed Grate Modelling in Modelica. Technical report, Aalborg University Esbjerg, 2015.

[2] Chungen Yin, Lasse Rosendahl, and Søren Knudsen Kær. Grate-firing of biomass for heat and power production. *Energy and Combustion Science*, 34:725–754, 2008.

[3] Normand M. Laurendeau. Heterogeneous kinetics of coal char gasification and combustion. *Progress in energy and combustion science*, 4:221–270, 1978.

[4] Emil K. Christensen and Johannes Veje. Co-firing of Wood Chips and Coal. Technical report, Aalborg University Esbjerg, 2015.

[5] E. Kofoed. *Pyrolyse og forgasning af halm. - 3: Varmetransport i halm.* Laboratoriet for Energiteknik, Danmarks Tekniske Universitet, 1991.

[6] Henrik Thunman, Fredrik Niklasson, Filip Johnsson, and BO Leckner. Composition of volatile gases and thermochemical properties of wood for modeling of fixed or fluidized beds. *Energy & Fuels*, 15:1488–1497, 2001.

[7] Niels Bech, Lars Wolff, and Lars Germann. Mathematical modeling of straw bale combustion in cigar burners. *Energy & Fuels*, 10:276–283, 1996.

[8] Modelica Association. *Modelica User Guide*.

[9] Michael M. Tiller. Modelica by Example, January 2016. book.xogeny.com.

[10] Marek Matejak, Martin Tribula, Filip Jezek, and Jiri Kofranek. Free modelica library of chemical and electrochemical processes. In Peter Fritzson and Hilding Elmqvist, editors, *Proceedings of the 11th International Modelica Conference*, volume 118 of *Linköping Electronic Conference Proceedings*, pages 359 – 366, Versailles, France, sep 2015. Linköping University Electronic Press, Linköpings universitet.

[11] Bonnie J. Mcbride, Sanford Gordon, and Michael J Zehe. Nasa glenn coefficients for calculating thermodynamic properties of individual species - nasa/tp-2002-211556. Technical report, NASA Glenn Research Center, 2002.

[12] Irvin Glassman and Richard A. Yetter. *Combustion*. Elsevier Inc, fourth edition, 2008.

[13] Stephen R. Turns. *An Introduction to Combustion : Concepts and Applications*. Mcgraw Hill, third edition, 2012.

[14] Michael L. Hobbs, Predrag T. Radulovic, and L. Douglas Smoot. Prediction of effluent compositions for fixed-bed coal gasifiers. *Fuel*, 71:1177–1194, 1992.

[15] Capucine Dupont, Rodica Chiriac, Guillaume Gauthier, and Francois Toche. Heat capacity measurements of various biomass types and pyrolysis residues. *Fuel*, 115:644–651, 2014.

[16] Martin Otter and Dietmar Winkler. Modelica Overview, August 2013. This slide set gives an overview about the Modelica language, including users view, libraries and a sketch of the language elements.

List of Figures

List of Tables

APPENDIX A

Utilities

A.1 Max Smooth and Min Smooth Function

The maxSmooth and minSmooth are custom functions that ensure the switch between the two values are done so that the output is two times continuously differentiable. This is done by utilising a build in Modelica operator "smooth". Furthermore, is another operator called "noEvent" used to minimise the calculation speed, as it removes the event caused by the function.

The maxSmooth function is:

$$\text{maxSmooth}(x, y) = \begin{cases} x & if & x >= y \\ y & if & x < y \end{cases} \tag{A.1}$$

The minSmooth function is similar:

$$\text{minSmooth}(x, y) = \begin{cases} x & if & x <= y \\ y & if & x > y \end{cases} \tag{A.2}$$

A.2 Specific Enthalpy Functions

A.2.1 Enthalpy of Ash

The specific enthalpy of ash is determined from the following expression:

$$h(T) = h(T_0) + \bar{c}_p \cdot (T - T_0) \tag{A.3}$$

where

- $h(T)$ is the specific enthalpy of ash in $\frac{J}{kg}$ at the temperature T in K

- T_0 is the reference temperature of 298 K or 25°C

- $\bar{c}_p$ is the average specific heat capacity in $\frac{J}{kgK}$ found from equation (A.4)

$$\bar{c}_p(T_a, T_b) = \frac{1}{T_b - T_a} \int_{T_a}^{T_b} c_p(T)\mathrm{d}T \tag{A.4}$$

The specific heat capacity can be found from different empirical expressions, the following is one example [14]:

$$c_p(T) = 594 + 0.586T \tag{A.5}$$

Substituting equation (A.4) and (A.5) into equation (A.3), setting specific enthalpy at reference temperature to zero, and rearranging results in:

$$h(T) = 0.293T^2 + 594T - 203146.87 \tag{A.6}$$

A.2.2   Enthalpy of Biomass

The specific enthalpy of biomass is determined in the same way as the ash, but the heat capacity function have different parameter [15]:

$$c_p(T) = -299 + 5.340T \tag{A.7}$$

Enclosure - Written in an earlier Project [1]

ENCLOSURE A

Model Specifications

The model have to be adaptable as the plants varies in both sizes and types. Therefore is the following a list of parameters that should be changeable in the model together with a short explanation.

Fuel types: The boiler's behaviour depends a lot on the fuel type, and it may be interesting to study how the boiler handles different fuels.

Fuel mass flow rate: Just as the fuel type, the boiler's behaviour and output is dependent on how the fuel-feeding is onto the grate.

Fuel moisture content: Again is the boiler's behaviour and output dependent on the moisture content since higher moisture content requires more heat to evaporate.

Grate velocity: The grate transportation velocity is one of the factors that determines if the fuel reaches the ash pit completely burned out.

Grate type: The type of grate is expected to have an impact on the boiler behaviour nonetheless.

Amount and distribution of primary air: The amount and the distribution of the primary air is a factor in the determination of whether or not the fuel is burned out when it reaches the ash pit.

Composition and temperature of primary air: In some of the plants, flue gas is recirculated into the bed and in other plants, the primary air is preheated.

Size and complexity: The size of the grate is an important parameter as the model should be applicable for all plant sizes, and it should also be possible to increase or decrease the complexity depending on the project at hand. For some projects, faster and more crude results might be wanted.

There are also a number of factors that the model should give some indications about. These are listed below with a short description.

Bed temperature: The temperature of the bed is important when determining the heat transfer to the boiler walls.

Composition of the flue gas: The composition of the flue gas determines if all of the combustibles have been burned.

Amount and temperature of the flue gas: The heat absorption in the economiser is dependent on both the amount and the temperature of the flue gas.

ENCLOSURE B

The Modelica Modelling Language

As mentioned in the preface this chapter contain a more thorough description of Modelica as a modelling language and will round of with a multiple domain example. The chapter was written as path of the 9th semester project [1].

Modelica is an object oriented and equation based modelling languages which can be used to simulate dynamic behaviour of technical systems [16]. These systems can consists of multiple models or components from different domains including thermal, fluid, mechanical and others. Each of these components can contain different sub-components, so the problem can be broken down to smaller and simpler models. This gives the ability to test smaller parts of the system and debug localised. Another advantage is the possibility to reuse existing components from previous systems or from one of the many libraries associated with the Modelica language.

Modelica also opens for the possibility to include units. The base SI-units and the most common derived ones are recognised by Modelica, and a warning is returned if there are not unit consistencies in the equations.

There are a number of different Modelica simulation environments. For this project the commercial software Dymola and the free software OpenModelica have been used.

B.1   Modelica Classes

The Modelica language contains different classes which each have a different purpose and the most significant are listed below. For each of these classes there will be a short description.

- Type

- Model

- Connector

- Package

B.1.1   Types

Types are used to group variables together and there are four built-in types:

Real: Used to represent real numbers, which is generally floating point numbers.

Integers: Used to represent whole numbers, which is mostly used in array indexing and in loops.

Boolean: Used to represent true and false.

String: Used to represent character strings.

Each of the built-in types can be extended to represent a more specific group of variables. For example can the real type be extended to be a diameter type with a minimum value of 0, a unit in m, and display the unit in mm.

```
type Diameter = Real (
        final quantity=''Length'',
        final unit=''m'',
        displayUnit=''mm'',
        min=0);
```

When the diameter type have been declared it can be used directly in the code. The "quantity" and the "unit" are used by Modelica to check if there are unit consistencies, and the "final" keyword is used to lock those into place. The "displayUnit" is used to specify which unit the variable will be shown in, if a conversion ratio have been specified. Lastly the "min" is used to specify that the diameter cannot have a negative value. It should be noted that "displayUnit" and "min" do not have the "final" keyword as they can be changed by the user at a later time. As with the most common derived SI-units, the most common types have been declared.

B.1.2   Models

Models are the class used for creating components. Each model must be "locally balanced", meaning that the number of equations and the number of unknowns must be equal. This allows for testing and debugging of the individual model, which reduces the debug time compared to finding errors in a big and complicated model. Furthermore, having the models locally balanced ensures that no matter how many components added to a bigger model, it will always contain the same amount of

unknowns and equations. However, this can also prove as a challenge from time to time.

The models are build up of multiple parts. Two of theses are a part containing the declaration of variables and a part containing the equations. Below is a simple example which is extended from an example in the free online book "Modelica by Example" by Michael M. Tiller [9].

```
model Example1
        parameter Real k=1;
        Real x;
equation
        der(x) = k-x;
end Example1;
```

Translated into ordinary syntax, the equation in the example is

$$\dot{x} = (k - x) \tag{B.1}$$

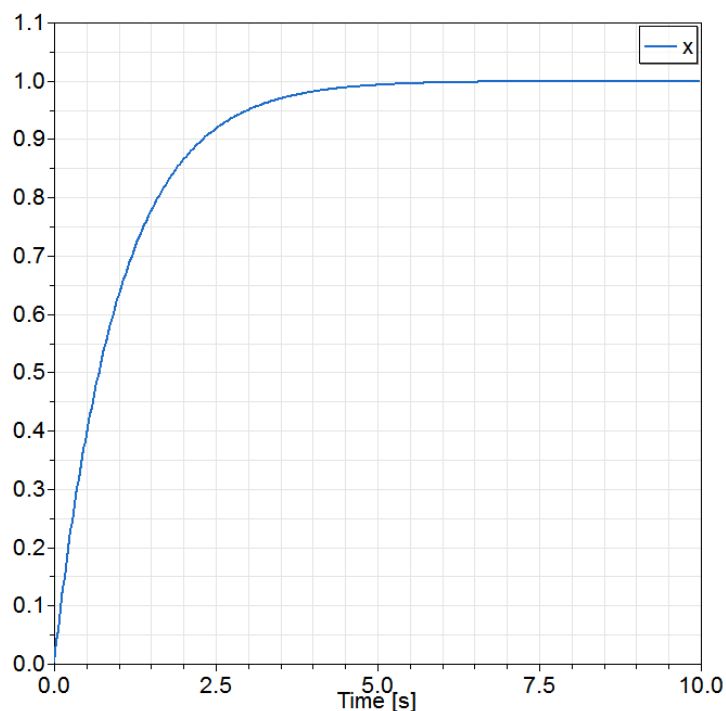Figure B.1 is the value of x as a function of time, from the simulation of the example.

Figure B.1: Simulation of equation (B.1)

The equation in the example is written explicit, but that is not a requirement in Modelica and the equation could just as easily have been written the other way around

$$k - x = \dot{x} \tag{B.2}$$

Using equation (B.2) instead of equation (B.1) would yield the same result. This is a great advantages as it allows the user to focus the time on the equations, and not on rearranging and cancelling out.

Simulation

After a model have been created and it is satisfying the criteria of having the same amount of unknown variables and equations, it can be simulated. When a model is simulated, the Modelica compiler first manipulates the equations analytical and then begins the effort to solve it numerically. The numerical solvers depends on the software used, but both Dymola and OpenModelica contains a long list of solvers each with their own strengths and weaknesses. Which solver to use is dependent on the problem, and whether or not the problem is stiff.

B.1.3   Connectors

The connection between the components are done via connectors, and each connector is dependent on the type of connection. For some of the most used connections there are predefined connectors in Modelica. A few of these are listed here:

Electrical pins: Connectors used in electrical circuits.

Heat ports: Used for one dimensional heat transfer.

Fluid port: Connector to describe fluid and gas flow.

Inputs and outputs: Used to pass control signals.

It is also possible to create custom connectors to fit specific problems. Both the predefined and the custom connectors, generally contains the following variables:

- Potential variables.

- Flow variables.

- Stream variables.

Each of these variables have a different use and behaves differently when connected together.

Potential Variables

The potential variable is the variable that drives the flow, and in each connector there should be at least one potential variable. For electrical pins, voltage is the potential variable, where in heat and fluid ports it is temperature and pressure

respectively. When the connectors are connected together, the potential variable are set equal to each other.

$$P_1 = P_2 = P_3 = \cdots = P_n \tag{B.3}$$

Flow Variables

The flow variable is the variable the describes the flow, and in each connector there can be only one flow variable. Electrical current is the flow variable in electrical pins and heat flow is in heat ports. For the fluid ports the mass flow rate is chosen over the volumetric flow rate as the flow variable. This is done as mass flow is conserved and volumetric flow is not when handling incompressible fluids. The flow variables are all assumed of entering the component and are summed to zero when connected.

$$0 = F_1 + F_2 + F_3 + \cdots + F_n \tag{B.4}$$

Stream Variables

The stream variables are properties of the flow, and there are no restrictions on the number of stream variables in each connector. In the electrical pins and heat ports there are no stream variables, where in the fluid port there are multiple. The specific enthalpy, the mass fractions of the fluid composition, and the amount of trace substances are all stream variables in the fluid connector. Furthermore, the stream variables is dependent on the direction of the flow. If the flow is positive, flowing into the component, the upstream values are chosen and if the flow is negative, flowing out of the component, the values inside the component are chosen. When the connectors are connected, the product of the stream variable and the flow variable are summed to zero.

$$0 = F_1 \cdot S_1 + F_2 \cdot S_2 + F_3 \cdot S_3 + \cdots + F_n \cdot S_n \tag{B.5}$$

B.1.4 Packages

A collection of models is called a "package", and a package can in addition to models also contain functions, connectors, and additional packages. A package is typically used to group models that are linked together in some way. For example do the standard Modelica library package contain a package named "Electrical", this package contains more packages which again contains more packages and in some of these packages are some of the models for the electrical domain. Figure B.2 shows the packages structure with the models in the package called "Basic".
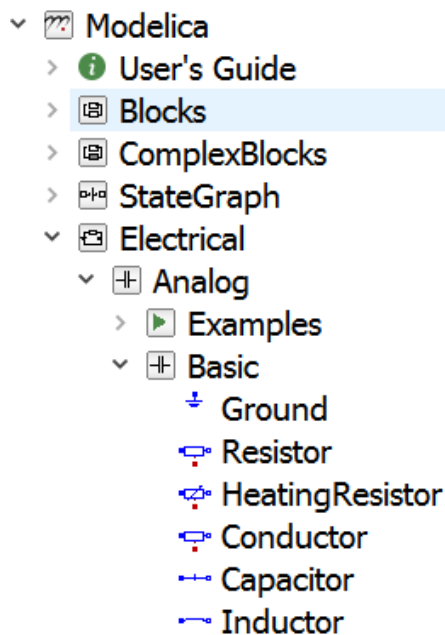
Figure B.2: Modelica models inside packages.

B.2   Extension of Classes

When creating models, connectors, and packages, Modelica offers a great tool for reusing already written code. This is done through the "extends" keyword. On figure B.2 there are a number of electrical components, and most of these all contain one port consisting of two pins. Instead of having to write the code to include these two pins in every component, a model called "OnePort" contains the basic code.

```
partial model OnePort

  SI.Voltage v "Voltage drop between the two pins (= p.v
      - n.v)";
  SI.Current i "Current flowing from pin p to pin n";
  PositivePin p "Positive pin";
  NegativePin n "Negative pin";
equation
  v = p.v - n.v;
  0 = p.i + n.i;
  i = p.i;
end OnePort;
```

The model defines the voltages drop, v, as the voltage difference between the two connectors, pin p and pin n. It also defines that the current flowing trough the component, i, is the same as the current flowing into the component on pin p and out of the component on pin n. In this model there are six unknowns and five

equations, as the two connectors each bring two unknowns and one equation. This is possible as the model got the keyword "partial" which removes the requirement of having the same number of unknowns and equations, but it also makes the model unusable as a normal model. The missing equation is supplied in the models that extends from this partial model.

Two of the simpler components that extends from this partial model are the "capacitor" and the "inductor". Both of these models include one extra parameter and one extra equation. First the capacitor

```modelica
model Capacitor "Ideal linear electrical capacitor"
  extends OnePort(v(start=0));
  parameter SI.Capacitance C(start=1) "Capacitance";
equation
  i = C*der(v);
end Capacitor;
```

On the first line the partial model "OnePort" is extended and the start value of the voltage drop is changed to zero. A parameter for the capacitance, C, have also been specified, and must be given a value priory to simulation. In the equation part of the model, the equation for the current through a capacitor is written

$$i = C \cdot \frac{\mathrm{d}V}{\mathrm{d}t} \tag{B.6}$$

The inductor is similar to the capacitor

```modelica
model Inductor "Ideal linear electrical inductor"
  extends OnePort(i(start=0));
  parameter SI.Inductance L(start=1) "Inductance";
equation
  L*der(i) = v;
end Inductor;
```

The main difference is that the current flow is given a start value of zero instead of the voltage drop and the parameter is now the inductance, L, instead. In the equation part of the model, the equation for voltage drop over an inductor is included instead

$$L \cdot \frac{\mathrm{d}i}{\mathrm{d}t} = V \tag{B.7}$$

The partial model "OnePort" is used for many more components, and it is also possible to extend from models that have been extended themselves. Furthermore, is it possible to extend from non-partial models. This extension of models reduces the amount of code that have to be maintained and it also makes it easy to create new models that fit into the existent models.

B.3   Multiple Domain Example

As mentioned, Modelica can handle models which include multiple domains. Figure B.3 shows a model with components from the three domains; Electrical, Thermal and Fluid.
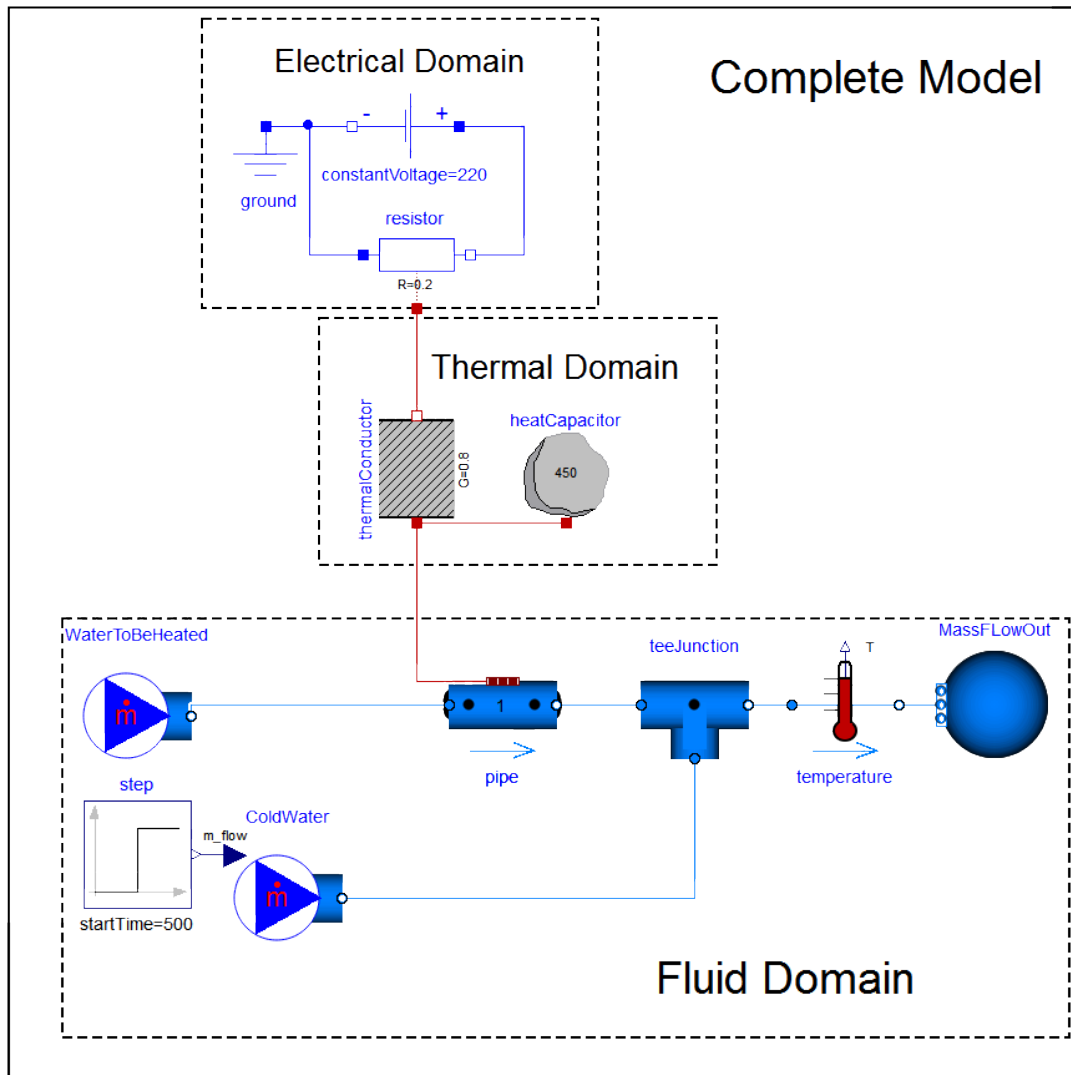

Figure B.3: Graphical interface of the multiple domain example.

A voltage source at the top of the figure is supplying a constant voltage of 220 V to a resistor with a resistance of 0.2 Ω. The voltage drop over the resistor is turned into heat which is conducted through a thermal conductor equivalent to 1 cm of iron to the fluid in the pipe in the middle of the fluid domain. In the thermal domain there is also a heat capacitor equivalent to 1 kg of iron.

The pipe in the fluid domain is, on the left hand side, connected to a flow source supplying a steady flow of fluid, which in this case is water. On the right hand side, the pipe is connected to a tee junction with a small volume where another flow source is connected. This flow source supplies a flow of non heated water. The last port in the tee junction is connected to a temperature sensor and that is connected to a pressure source, which in this case functions as a sink for the water.
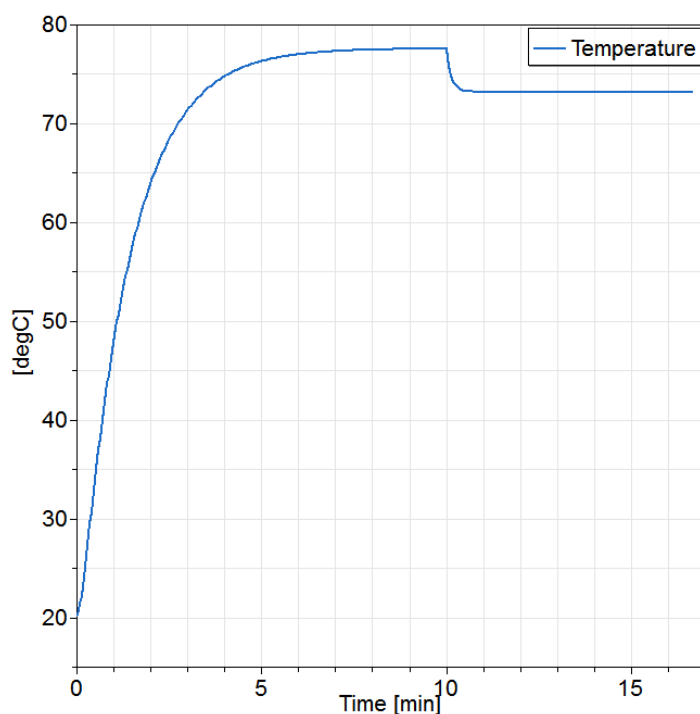

Figure B.4: Temperature of the water in the multiple domain example as a function of time.

Figure B.4 shows the temperature at the temperature sensor as a function of time. The water have a temperature of 20 degrees Celsius at the start of the simulation, and as the time goes the temperature sensor shows a rising temperature. When the simulation time reaches 600 seconds or 10 minutes, the flow of non heated water source is doubled and this can be observed as a decrease in temperature. The rise and fall of the temperature is not instantaneous as the model is taking simple dynamics into account.

The code for this example is not included as it consist only of standard components that have had some of their parameter changed to fit this model. The

equation part of the model is made of connections between the connector of the different components. All of the things done in this example can be done directly from the graphical user interface in Dymola, however if OpenModelica is used some of the parameter changing have to be done manually in the code.

The dynamic and multiple domain capabilities of Modelica combined with the ability to extend models, makes Modelica a strong choice for the model.