Applying the Power Plant Library ClaRa for Control Optimisation

Friedrich Gottelt[1]   Timm Hoppe[1]   Lasse Nielsen[2]

[1]XRG Simulation GmbH, Germany, `{gottelt, hoppe}@xrg-simulation.de`
[2]TLK Thermo GmbH, Germany, `l.nielsen@tlk-thermo.com`

Abstract

This paper presents the current state of the open-source Modelica library *ClaRa* which enables the user to solve complex power plant simulation tasks. The library can be used to help control experts to develop and test controllers without disturbing the daily operation of the applying power plant. This reduces project risks and costs significantly. As a use case the analysis and optimisation of a Benson boiler power control is presented. The presented solution reduces the impact of soot blowing on the power output by 57 %.

Keywords: Power plant, Clausius-Rankine cycle, open-source library, control optimisation, soot blowing, simulation

1 Introduction

The global energy markets are in a phase of significant changes due to increasing power production from renewable sources like solar and wind power, see (International Energy Agency, 2015). These renewables are usually fluctuating energy inputs as they depend on local weather. In order to ensure a stable grid operation conventional power suppliers have to outbalance these fluctuations. This introduces new modes of operation to new and existing power plants. The questions and challenges that arise from these new operation modes can be solved at low costs with the help of system simulation. Therefore, it is very likely that activities in this field will grow in both, university-based and industry research and development.

The most recent Modelica library in this field is the library *ClaRa* which was developed from 2011 in a German collaboration[1] of Hamburg University of Technology, TLK-Thermo GmbH and XRG Simulation GmbH. Its first official release of version 1.0.0 dates from March 2015 (see (Brunnemann et al., 2012; Gottelt et al., 2012) for an introduction to ClaRa and a control-related application). The aim of this development was to provide a library that is both, suitable for beginning and advanced researchers in the field of simulation with Modelica. This leads to special requirements with respect to usability and flexibility which are described in section 2. Currently the library is under ongoing development within the follow up project Dynstart[2]. New features, for example a parameter based

initialisation concept, have been added and the library is going to be enhanced to handle extra-low-loads, start-up and shut-down processes. To illustrate the potential of the library, the challenging task of a soot blowing simulation is presented as use case in this paper.

2 Overview of the Library ClaRa

2.1 Scope of Library

The library *ClaRa* was created to enable simulation of large steam power plants with coal dust firing. At the heating side the library comprises models for the complete fuel handling process from the grinding via the fuel combustion to the flue gas cleaning. At the water steam side the library features models for the cooling of the combustion chamber to the electrification of the steam in the turbo generator.

The library is intended to be the centre of a whole family of Modelica libraries in the field of electricity production and consumption. Any extension of *ClaRa* will be handled in a separate, so-called *ClaRa_AddOn*. This avoids the constant growing of the library, limits its scope and reduces the effort of library maintenance since not all add-ons must be fully compatible to each other. Especially the last mentioned aspect eases the library development by external developers. Two *ClaRa_AddOns* are currently under development in close collaboration with the *ClaRa* developers: *ClaRa_Control* supplies blocks for an efficient implementation of state of the art power plant process control including the start-up and shut-down processing. *Transient* widens the scope of *ClaRa* by the energy distribution and allows for the simulation of strongly coupled electric grids, gas grids and district heating grids as well as its economic evaluation (Andresen et al., 2015).

Consequently, also the *ClaRa* library makes use of two external libraries, namely the open source library *Fluid-Dissipation* (Vahlenkamp and Wischhusen, 2008, 2009) providing functional descriptions of pressure losses and heat transfer and the free-of-charge version of the *TILMedia* (Schulze, 2014) providing media data for water, CO_2 and gaseous flue gas and air mixtures.

2.2 Structure of Library

The structure of the library follows a functional approach rather than structuring according to the used media. For

example, a steam-heated shell-and-tube heat exchanger for the preheating of feedwater will be found in the same package `HeatExchanger` as a flue gas to fresh air heat exchanger.

Table 1 gives an overview of the top level package structure. Herein, the `UsersGuide` provides a brief introduction to the library as well as developer's contact data and information on the license model and the revisions. The package `Examples` provides a number of introductory examples making new users familiar with the capabilities of the library. The package `Basics` contains basic models and other internally used codes like functions, records and interfaces. Most users will not get in touch often with this package. In contrast, the package `Components` contains all the component models required to build up a power plant model.

Table 1. ClaRa library structure

[i]	**UsersGuide**	Information on basic modeling concepts, contact and license
▣	**Examples**	Introducing examples
▣	**Basics**	Basic models and informatics
▬	**Components**	Models for turbo machines and electrical machines, connection pipings, heat exchanger, mass storage and steam separation, valves, coal grinding, furnace, flue gas cleaning, and sensors, i.e. "the core of the library"
▟	**SubSystems**	Conceptual package aiming at supporting team work
▦	**Visualisation**	Auxiliaries for the visualisation of results
▣	**StaticCycles**	Static models for the calculation of consistent initial guess values

The package `SubSystems` contains some examples for the definition of subsystems. This package is still somewhat conceptual since it does not feature a complete set of reasonable sub systems but rather aims at introducing ideas for an efficient team work. The package `Visualisation` provides means to visualise the results using both, Modelica-based annotations and third party post processing tools. The section 2.5 gives a brief overview of the options. Finally, the package `StaticCycles` contains simplified, static and parameter-based models of most of the power plant components. Details on this package and the idea for its application within *ClaRa's* dynamic modelling approach are discussed in section 2.6

2.3 Overview of Component Modelling

One aspect of the *ClaRa* modelling concept is to provide as much physics as applicable to achieve a close to reality model behaviour of single components as well as whole power plant models. Different modelling approaches are used throughout the library, for example a finite volume approach for pipes, flow models for valves, zonal models for reservoirs and characteristic maps for compressors and pumps while almost all components are using balance equations.

The different models are reasonably modular in structure avoiding doubled code and providing high code transparency. Another aspect in model structuring is to gain flexibility in the model application. The structuring is done according to the following concepts:

Models at different levels of detail are provided as separate classes, they can be exchanged in many cases (e.g. via Dymola's context menu "change class...") since they have equal interfaces. However, its internal structure is significantly different providing a differently detailed view into the component.

For instance, a pipe can be modelled applying an integrated, slim set of balance equations (as described in (Velut and Tummescheit, 2011)) or its balance equations can be discretised applying a finite volume approach. The first mentioned approach will be very efficient for the simulation of water hammer effects but loosing information about the internal, local fluid temperature and pressure distribution. The latter mentioned approach is very robust for reverse flow and off-design conditions and gives detailed information about local and temporal effects like steaming but will be less efficient for very long pipes.

Physical effects at different levels of insight are provided to allow to apply a single class to be instantiated in different contexts. For instance, the pressure loss of a valve might be calculated either considering supercritical or subcritical flow conditions. In other cases, if the valve's specific behaviour is not of particular interest it might be sufficient to simply consider a linear hydraulic correlation. All this is handled in different *replaceable* models.

Basic models are reused by instantiation to form new component models. For instance, a shell-and-tube heat exchanger model is a compound of the shell's volume, the tube's volumes and the separating wall. ClaRa provides basic models for these sub-compounds which makes the models very easy to understand and to maintain and allows users to easily create new component models.

The component list is complete for most tasks around evaluation of the normal power plant operation of both once-through boilers and circulation boilers. Heating can be realised either by coal dust firing or heat recovery (other heat sources like solar energy or biomass firing will be subject to *ClaRa_AddOns*). However, future releases will increase this content further, e.g. there will be a cooling tower enabling studies on the performance of the cold side

of the process.

2.4 Media Supported

ClaRa is shipped with a non-profit version of the *TIL-Media* featuring four different media types[3]. For pure mediums like water/steam there are table based and spline interpolated data available which are very encouraging concerning simulation speed and simulation stability, see (Schulze, 2013). The flue gas is described by a gas-vapour mixture with eight substances similar to humid air. A mixture of real fluids for application in CO_2 separation processes is supported as well as pure CO_2 for ORC applications.

The calculation of substance properties within *TILMedia* is applied by external C code for faster simulation speed and reusability. The robust media formulation has also a positive effect on the initialisation of *ClaRa* models because media evaluation outside of the range of validity is handled by extrapolation or limitation to numerical range of validity to prevent division by zero or infinity. Significant less crashes were experienced compared to work with other media libraries.

2.5 Visualisation and Usability

The usability is a key point to ease new users the introduction to the library. Therefore, *ClaRa* component models feature well-structured parameter dialogues that make extensive use of Modelica's dialogue structuring annotations like "tab" and "group". This helps the user to distinguish between expert settings and fundamental, geometric settings of a component. Where applicable, descriptive figures help to understand the technical context.

Modelica libraries often provide only brief documentation thus implying that the source code is self-declarative. Although this might be true for certain libraries available it is often an obstacle for those who are less familiar either with commonly used modelling techniques or the programming language used. *ClaRa* provides a comprehensive documentation for these users aiming at deepening the understanding of the work and improving the confidence in the library by granting a maximum of transparency. The documentation gives detailed insight into the underlying theory and explains expert user settings and spent validation efforts.

In order to help users to keep track of very complex power plant process designs a number of visualisers can be included into the models, see Figure 1. These items help to better understand the current state of the process by visualising important process variables (pressure, temperature, spec. enthalpy and mass flow and levels) and highlighting critical values like negative or zero flows.

2.6 Global Initialisation

The initialisation of a large differential-algebraic system of equations is a challenge that has been discussed in
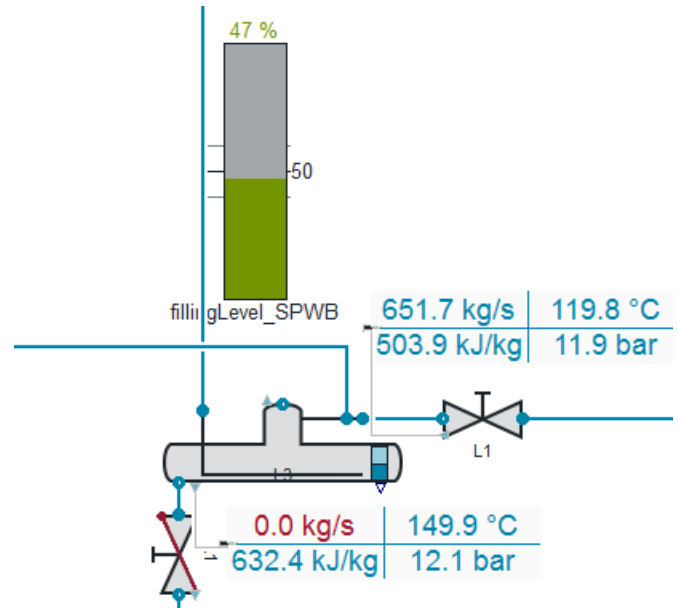

Figure 1. Screenshot of *ClaRa* visualisation

the Modelica community from the very beginning, e.g. (Mattsson et al., 2002; Bachmann et al., 2006; Najafi, 2008). Since initialisation can often be a time-consuming and frustrating task, especially for beginners, a library's quality can also be measured in its features that support the user in getting robustly and reproducibly initialising models. For *ClaRa's* implementation five aspects are considered to ease the initialisation process: First, a new user will expect to get some kind of guidance in the task of initialisation, i.e. it should be clear for which variables the user should provide estimation values and for which not. Second, the library should support the homotopy concept that was proved to be advantageous according to (Casella et al., 2011). Third, the available initialisation options shall provide a flexibility to initialise models in arbitrary combinations. Fourth, the initialisation should be reliable, i.e. it should not be sensitive to smaller model changes. This point also refers to the accuracy that estimated values must have. Finally, taking the system topology into account is a valuable feature that would significantly improve the initialisation process. The last point is not straightforward since it counteracts the modular modelling principles Modelica is based on.

If we take a look on a dynamic, 0D T-join for example. The volume normally is initialised by applying estimation values for the state variables, i.e. pressure and specific enthalpy provided by the user as parameters. In principle, these parameters are defined locally and they are completely independent from its neighbouring components. However, from a technical point of view it is clear that it would be useful to take the neighbours into account since the mixer's specific enthalpy will be the weighted mixing enthalpy of the two inlet enthalpy flows coming from the upstream components.

The *ClaRa* approach to take this kind of informa-

---

[3]There are more media available at costs for a commercial *TILMedia* license

tion about the system topology into account is the `StaticCycle` package. This package contains a set of stationary models which can be used to create a simplified, static and parameter-based mimic of the dynamic cycle. The result is a consistent, load depending set of parameters for mass flow, pressure and specific enthalpy or temperature for the complete cycle. Linking the results of this static cycle to the respective initial guess values in the dynamic cycle allows to give flexible and consistent initial values at all dynamic components considering system topology and the possibility to use a cascaded initialisation with values of upstream neighbouring components. This feature allows for a very robust initialisation with automated adaptation on varying design points.

Overview of the `StaticCycle` package features:

- parameter based

- signal connectors

- different input/output combinations for different assembly yield four differently coloured connectors. Equally coloured connects match

- connectors are error proof with respect to wrong connections (e.g. outlet-outlet connection, blue-red connection)

- functionality for load dependent initialisation (table based)

Figure 2 shows an exemplary cut-out of a `StaticCycle` circuit with visualised signal flow directions and corresponding parameters. Blue arrows indicate the local flow direction. In this example the heat exchanger calculates for the outlet at position 1 the enthalpy according to the energy balance, passes through the mass flow from the inlet and receives a pressure value at the blue outlet connector. In contrast, at the red steam inlet connector (position 2) mass flow and pressure are defined via user input and passed over to the upstream valve. The red inlet connector receives a value for the enthalpy. In the red-connected steam valve a nominal pressure drop is assigned so that the component passes pressure and mass flow to the red outlet connector of the T-split which itself provides a value for the enthalpy. The green outlet connector of the T-split provides values for mass flow, enthalpy and pressure for the top valve component which serves as a pressure break because this components also receives a pressure signal at its blue inlet connector while sending parameters for mass flow and enthalpy for the downstream component.

In the following an example of a simple tube is used to illustrate the `StaticCycle`'s work with fixed=false parameters to determine parameters that are passed over from neighbouring components.
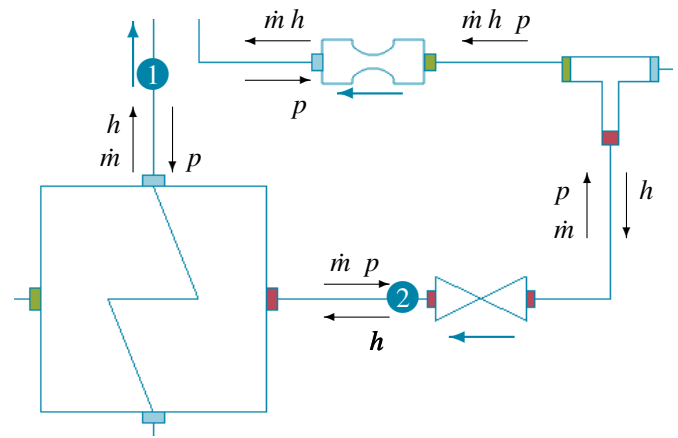

Figure 2. `StaticCycle` component example cut-out. Blue arrows indicate mass flow direction, black arrows indicate signal flow directions

```
model tube
  blueConnector_inlet inlet(p=p_in); //
      send pressure to upstream component
  blueConnector_outlet outlet(m_flow=
      m_flow, h=h_in); // send mass flow
      and enthalpy to downstream component
  parameter Real Delta_p_nom; // nominal
      pressure loss

  final parameter Real m_flow(fixed=false);
  final parameter Real h_in(fixed=false);
  final parameter Real p_in = p_out+
      Delta_p_fric;
  final parameter Real Delta_p_fric =
      m_flow / m_flow_nom * Delta_p_nom; //
      pressure loss calculation

  initial equation
    outlet.p=p_out; // get pressure from
        downstream neighbour
    inlet.m_flow=m_flow;
    inlet.h=h_in; //get the enthalpy and
        mass flow from the upstream
        neighbour
  end tube;
```

As can be seen, parameters calculated or set by the component are set via the connector's modifier, e.g. inlet.p. In contrast, parameters that are set by neighbouring components are made available by parameters that apply the fixed = false feature. The corresponding internal values are set in an initial equation environment, ensuring that the results can be used as initial values for the dynamical simulation of the main model.

3 Use Case for Control Analysis and Improvement

3.1 Scope of Work

The idea of this generic use case is to illustrate *ClaRa's* capabilities to model power plants at the state of the art including the full complexity of common condensation power plants due to extensive feedwater preheating and
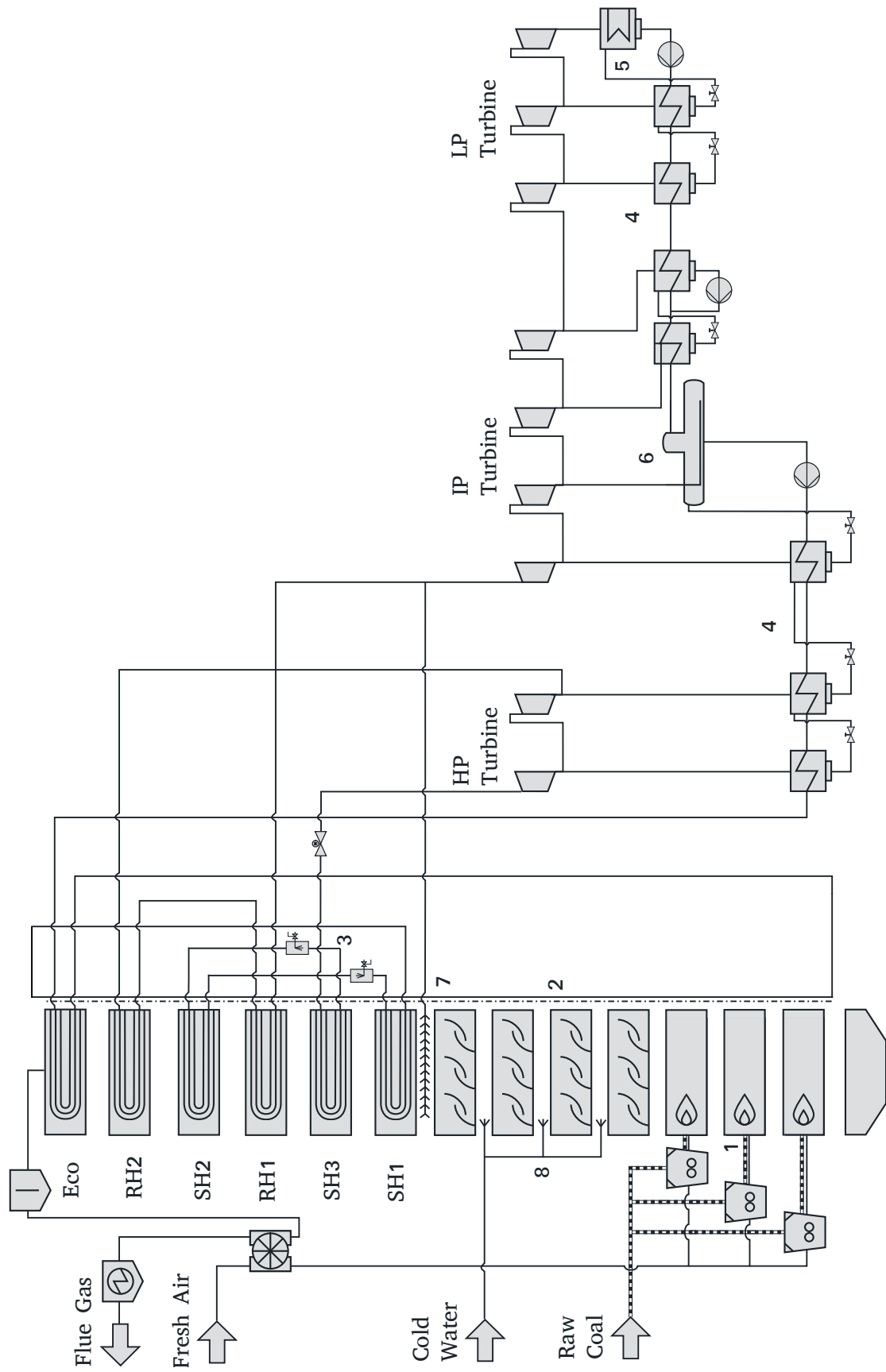
Figure 3. P&ID of simulation model

exhaust gas cleaning and cooling technologies. The use case also aims at proving Modelica's power to analyse complex processes at a physical level and its interaction with control structures. System simulation tools are used to analyse and improve a power plant's unit control to better handle disturbances due to soot blowing.

3.2 Model Description

The dynamic power plant model analysed in this paper is a generic once-through steam generator and has a nominal electrical power output of 600 MW, see Figure 3. The design life steam pressure is about 270 bar at a temperature of 600 °C. The combustion chamber has an overall height of approx. 80 m and is heated with three coal dust-fired burners (1) at three different burner levels. The chamber's walls are cooled with evaporating water which is superheated in three convective heating surfaces (SH1-SH3) to the live steam conditions. The layout comprises one reheating (RH1-RH2) to 520 °C at 52 bar after the high pressure turbine stage. The model discretises the boiler in height by using an arrangement of 14 different components including the hopper, the burners, the radiation zone (2), the convective heat exchangers and the spray injectors (3) for steam temperature control. Each of these models is parametrised with detailed geometry information and connected to neighbouring components according to the piping and instrument diagram. The heat exchange between combustion chamber and walls is implemented with detailed correlations for radiative and/or convective heat transfer. The flue gas is post processed with respect to NO_x, SO_x and ash and is used to preheat the fresh air carrying the coal dust from the mills. In order to reproduce the impact of soot blowing to the process at several positions steam and cool water can be introduced to the combustion chamber.

The steam is expanded in nine turbine stages with bleeds for several preheaters (4). The models for condenser (5) and feed water tank (6) are taking non-ideal phase separation into account and the resulting water levels are controlled by valves and pumps applying PI controllers.

The controlling system is build up applying the upcoming ClaRa_AddOn *ClaRa_Control* and based on a German guideline for unit control of conventional and nuclear steam power plants, (VDI/VDE, 2003). This guideline, defining a baseline for the power plant control, is the starting point for numerous implementation in German power plants.

The purpose of the model discussed in this paper is to analyse soot blowing of the evaporator furnace and its impact on the controlling system. The soot blowers (7,8) are modelled as a dynamical gas volume with an additional connector for water/steam inlet. The inflowing water is mixed ideally with the gas mass flow and the enthalpy of evaporation is considered by the dynamical energy balance. An effect of the soot blowing on the fouling coefficient, which is a parameter that reduces the heat transfer

to the heating surfaces, is not considered. An approach to model variable fouling factors which are affected by soot blowing is presented in (Gierow et al., 2015). This simplification is acceptable since the focus of the investigation is on the short-term energy and mass transfer of steam and water from the water steam cycle to the furnace and its impact on the control performance of the power plant. The medium-term improvement of the local efficiency of the heat transfer is of minor interest here.

The resulting power plant model consists of 9212 components with 28264 equations and 1606 differentiated variables.
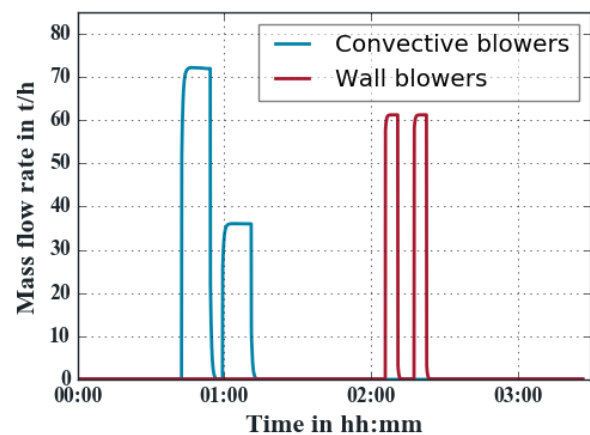
3.3 Scenario Description

Figure 4. Soot blower injection mass flows

The underlying scenario to analyse the improved control strategy comprises two soot blowing events occurring during normal power plant operation. Heating surfaces in coal fired power plants tend to foul due to the high ash and slag content of the combustion air. Particles stick to the heating surfaces and cause a rising heat transmission resistance over operation time. Thus, the heat transfer to the water steam cycle is reduced. Soot blowing is a measure to clean the tube bundles and evaporator walls during operation by spraying steam or water through special lances onto the heating surfaces. This measure uses the combined effect of a thermal shock and the kinetic energy of the water/steam jet to reduce fouling and improve the heat transfer.

In general, two different soot blower types can be distinguished: The soot blowers for the tube bundles are fed with steam which is extracted ahead the intermediate pressure turbine. This rededication of steam has a direct impact on the electric power production and its control. In contrast, the soot blowers for the evaporator walls are fed with external water at 20° C which has a more indirect effect on the power production as the water injection cools down the furnace and thus reduces the steam production.

In figure 4 the soot blower injection mass flows are

shown. The convective blowers are active two times over a timespan of 15 min with a mass flow of 20 $\frac{kg}{s}$ and 10 $\frac{kg}{s}$, respectively while the wall blowers are active for time spans of 5 min with a mass flow of 17 $\frac{kg}{s}$.

3.4 Analysis and Improvement of the Control Strategy

The plant is run in "steam generator control" mode and natural sliding pressure which means that the power output controller acts on the fuel mass flow and the turbine valve is fully opened. Therefore, deviations in fuel mass flow or disturbances on the steam generation have direct impact on pressure and generator output. Compensation of the generator power output by the fuel mass flow only takes effect with a delay.

A basic model based unit control strategy according to the German guideline for power plant unit control VDI 3508 (VDI/VDE, 2003) is implemented. The control is sketched in Figure 5, the basic control in black, the additions of the improved control in green. In the figure three sub-figures may be distinguished, starting from top to bottom we find the unit feed forward control and the process predictor in sub-figure a), the feedback controllers in sub-figure b) and the process itself in sub-figure c). Furthermore, three different line types are used: solid lines for internal control signals, dashed lines for measurement signals and dot-dashed lines for process input signals. For the sake of simplicity only the turbo generator power output control and the soot blowing control are sketched in the figure.

In the following the basic control set up is explained in detail. The turbine valve is fully opened by setting the turbine valve opening set value $y_{T,set}$ to 100 %. In subfigure a) the target power output is fed into the output limiter which applies a limiting according to static and dynamic limits. From that value the fuel feedforward block generates a load dependent value of the firing power $\dot{Q}_{F,FF}$. The firing power is input to the simplified process model, referred to as the predictor in the further course, and to the process itself via a recalculation of the firing power into a corresponding fuel mass flow $\dot{m}_F$. The predictor calculates a corresponding expectation value of the generator power $\hat{P}_G$. In detail the predictor works as follows. From the fuel forward signal $\dot{Q}_{F,FF}$ a transfer function calculates the expected steam generation of the boiler $\hat{\dot{m}}_{St,G}$. The expected steam mass flow to the turbine $\hat{\dot{m}}_T$ is subtracted from that value and the result serves as input to an integrator block from which the expected live steam pressure $\hat{p}_{LS}$ is obtained. By multiplication with the set value of the turbine valve opening $y_{T,set}$ the steam mass flow to the turbine is calculated. The turbine is modelled with a first order transfer function block, thus receiving the expected power output $\hat{P}_G$.

However, the real process will be disturbed e.g. by fouling of heating surfaces which leads to a deviation between the ideal and the real process. In subfigure b) it can be seen that this deviation in power generation is calculated

from the difference of the expected value $\hat{P}_G$ and the measured value P_G. The difference is input to the feedback controller of the power output which is a conventional PI-controller. It corrects the feed forward value $\dot{Q}_{F,FF}$ by the value deviation $\Delta\dot{Q}_F$.

Furthermore, in subfigure b) the feedback controller of the convective tube bundles soot blowing mass flow can be found. From the required soot blowing mass flow $\dot{m}_{SB,set}$ the measured soot blowing mass flow $\dot{m}_{SB}$ is subtracted. The difference is input to the soot blowing mass flow controller which is a conventional PI controller.

Applying the concept of a model based unit control the process will be controlled in open loop as long as the process reacts as predicted. Thus, the control effort of the feedback control is significantly reduced compared to a conventional control system in which load changes are acting on the set value of a feedback controller. However, a disturbance like soot blowing has to be outbalanced by the power feedback controller. In case of the tube bundle soot blowing, intermediate pressure steam is consumed. Thus, the steam mass flow to the turbine is reduced and the generated electric power drops. The power feedback controller raises the fuel mass flow to compensate the power generation drop, but it takes effect with a delay. Poor control accuracy is the result.

To improve the behaviour of the model based unit control during soot blowing of the convective tube bundles some additions have been made, marked green in Figure 5. The set value of the required soot blowing mass flow $\dot{m}_{SB,set}$ is introduced as positive disturbance signal to the predictor's expected steam mass flow to turbine $\hat{\dot{m}}_T$. To overcome the delayed reaction of the boiler on changes of the fuel mass flow, the disturbance signal to the predictor is activated prior to the starting time of the soot blowing process. This is done by delaying the soot blowing set point signal to the soot blowing mass flow controller with a delay block. The predictor reacts on the mass flow disturbance signal and raises the expected power output. In consequence, the feedback control outbalances the disturbance and raises the control signal. Thus, the coal mass flow rises and at the start time of the soot blowing additional steam is produced, which then can be consumed for the soot blowing. A drop of the produced power can be compensated.

The presented modifications acting on the feedback control are very efficient and easy to implement. A physical signal, the soot blowing steam mass flow, can be used directly as a disturbance signal without complicated recalculation. It is also independent of the load point as the signal is added to the boiler predictor of the unit control. The predictor's output is a load dependent rise of the generator power. The approach to derive an improvement of an existing control system from a profound analysis of the system of question rather than applying purely control-theoretical approaches has the advantage that the ideas can be easily understood and discussed with the power plant's operating personnel. This eases the acceptance of
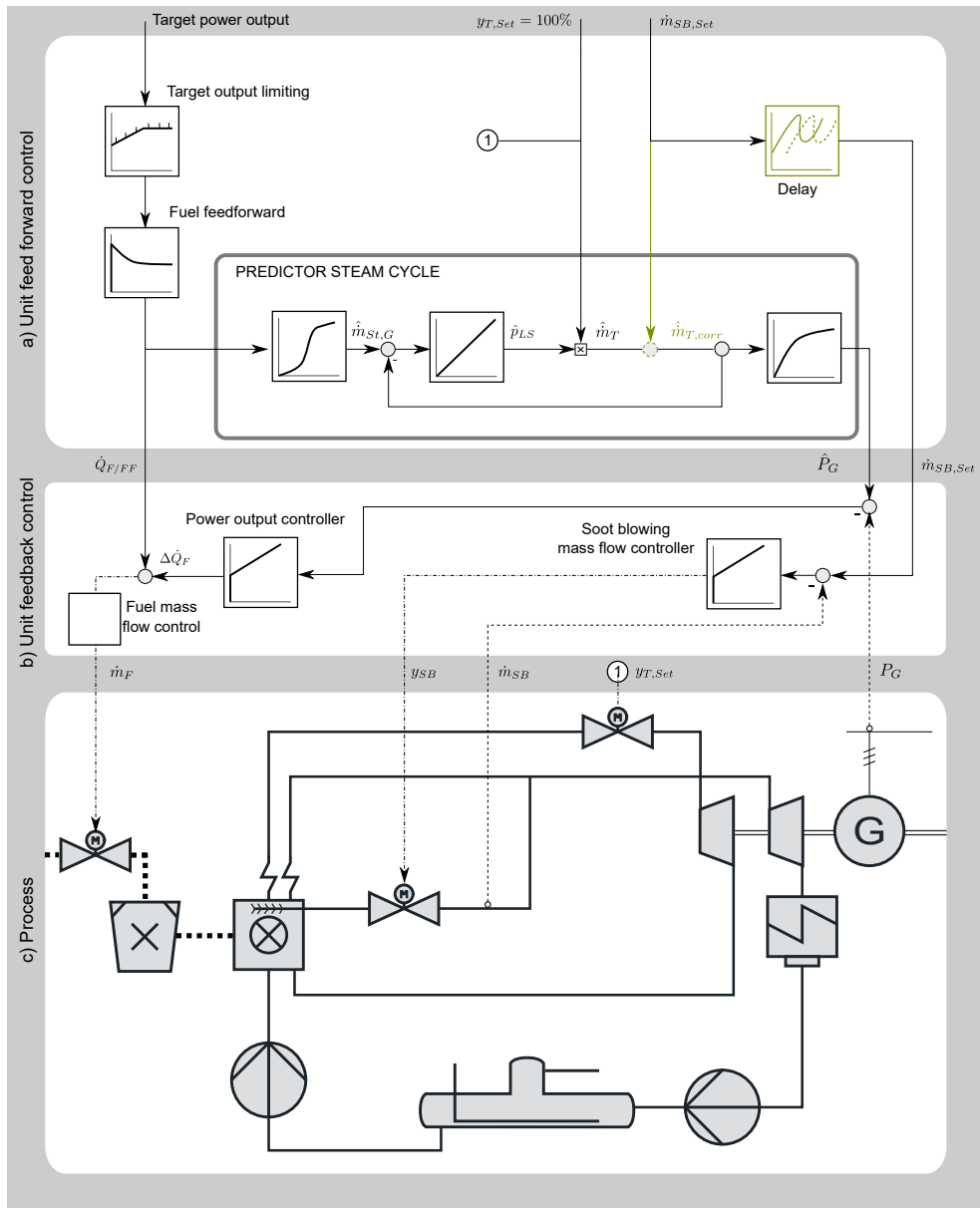
Figure 5. Model-based unit control and convective sootblowing control

such optimisation projects. System simulation prior to the commissioning can help to reduce technical and economic risks.

For the sake of completeness it shall be mentioned that an implementation of the disturbance signal in the feedforward control would require a load dependent recalculation of the physical signal, the steam mass flow rate, to a corresponding firing power signal.

Analogous additions also improve the control behaviour during steam generator wall soot blowing. As cold water is injected by the wall blowers the mass flow of cold injection water has to be expressed as a corresponding reduction of the steam mass flow to turbine. This is done by multiplication of the cold water mass flow with a load independent factor. The result is input as a disturbance signal to the predictor in the previously described manner.

In the following the simulation results are discussed. Figure 6 shows the flue gas outlet temperatures of the first superheater bundle during the soot blowing events for both control strategies. As can be seen, the impact of the soot blowing on the flue gas temperatures is higher for the wall blowers compared to the convective blowers. The reason behind this is that the convective blowers are fed with hot steam taken from the hot reheating pipe and the wall blowers are fed with cold water from an external reservoir. Due to the huge amounts of evaporating water the flue gas temperature is being reduced which results in less steam production. In comparison to the original control strategy, the improved one shows a significant reduction of flue gas temperature oscillations during wall soot blowing.

Figure 7 shows the intermediate pressure turbine mass flow during the soot blowing process with and without the
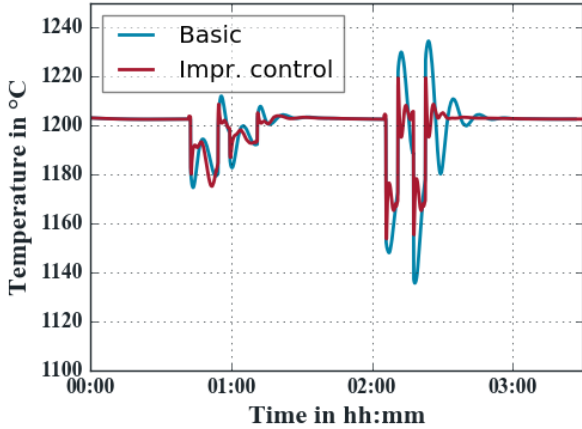
Figure 6. Impact of soot blowing on flue gas temperatures

Figure 8. Impact of improved control on generator output during soot blowing. The results corresponds to a reduction of primary control from 10.7 MWh to 4.6 MWh.

improvement of the controller system. The steam for soot blowing of convective heating surfaces is taken directly from before the intermediate pressure turbine. The control improvement raises the steam generation such that we see less reduction of steam mass flow through the turbine. The soot blowing induced mass flow oscillations are of lower amplitude and the system reaches a steady state faster with the modified control strategy especially during the soot blowing of the furnace walls with cold water, because a swing up of the turbine mass flow is prevented.
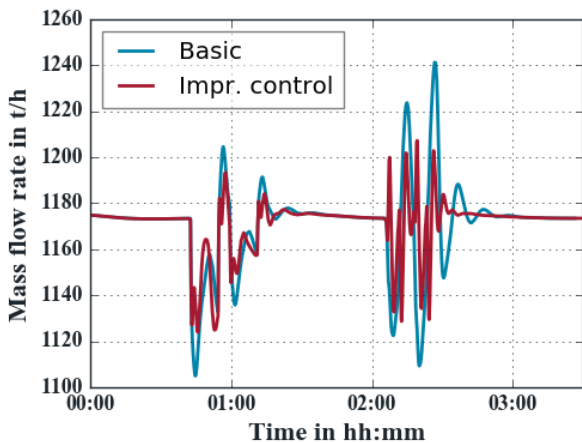

Figure 7. Impact of improved control on IP turbine flow during soot blowing

In Figure 8 the generator output for both control strategies are shown. The improvement in quality of control by the additional disturbance value is visible here too and results in lower amplitudes and a faster reaching of a stable state. The greater impact of the wall soot blowing with cold water in comparison to the convective soot blowing with hot steam manifests in higher amplitudes.

In the following, simulation results of the wall soot blowing process with cold water are discussed in detail,
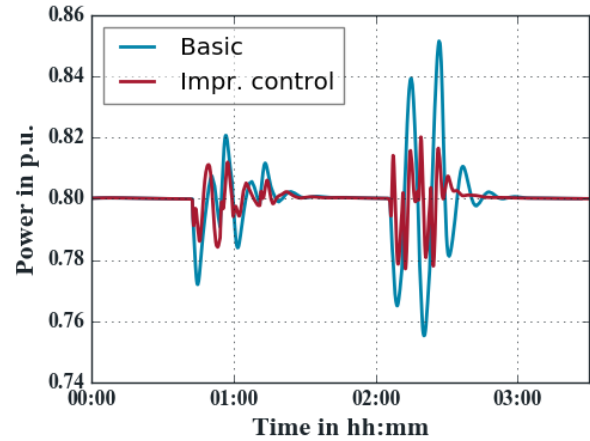
which is significantly improved by the alternative control strategy. Figure 9 shows the fuel mass flows of mills and burners, soot blowing mass flow and first superheater flue gas outlet temperature during the period of wall soot blowing without control improvement. When the soot blowing process is started, the flue gas temperature inside the furnace reacts with an initial drop due to the fed in cold soot blowing water and its evaporation, which can also be seen in the superheater outlet temperature. This results in a reduced mass flow inside the turbine forcing the power controller to increase the fuel mass flow at mill inlet. The mass flow at burner inlet reacts time shifted due to mass storage effects in the mill. When the increased but delayed fuel mass flow is burned, the flue gas temperature is rising again until the first soot blowing process is immediately stopped, which causes a temperature step up above the initial temperature. The power controller then reduces the coal mass flows until the second soot blowing, causing a comparable temperature characteristic which transitions into a swinging state when the soot blowing process has ended.

Figure 10 shows the same variables during the same time span for the soot blowing process with the improved control strategy. In comparison to Figure 9 it can be seen that the power controller raises the fuel mass flow already before the soot blowing process is started. This is caused by the disturbance signal of the pressure steam mass flow being activated 50 s prior to the soot blowing process. By the time the soot blowing process starts, the increased fuel mass flow already enters the burner levels resulting in a shorter and not so big initial flue gas temperature drop. In addition, the flue gas temperature only swings with a lower amplitude compared to the case without improvement and reaches a stable state in a significantly shorter time span which results in a higher quality of control. Because of the higher flue gas mass flows and a change of the heat transfer coefficients due to the fed in water, the super-
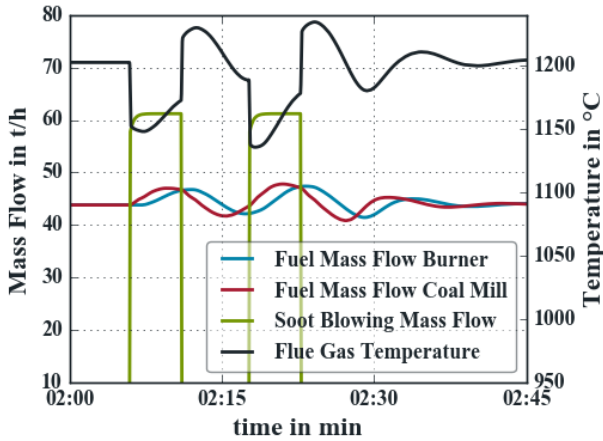
Figure 9. Fuel mass flows of mills and burners, soot blowing mass flow and superheater outlet temperature during wall soot blowing without control improvement

Figure 11. Impact of control optimisation on generator output during soot blowing at different loads

heater outlet temperature tunes in to a lower temperature during soot blowing.
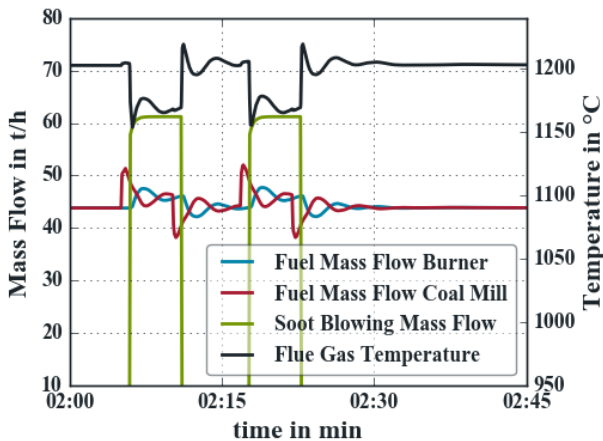

Figure 10. Fuel mass flows of mills and burners, soot blowing mass flow and superheater outlet temperature during wall soot blowing with control improvement

Figure 11 shows the impact of the alternative controller system on generator output at different power plant loads. In order to ensure, that the enhancement of control quality can be achieved at varying loads and not only the one discussed previously, additional simulations have been carried out for 100% and 65% load. As can be seen, the same improvements are obtained at all load points shown in the diagram where the best results are obtained for nominal load. The *ClaRa* library is an appropriate tool to carry out such additional comparisons very comfortably.

Power plants which participate at the secondary control power market, must guarantee the offered power at any time during operation. While control power is being called, the plant operators would like to avoid fluctuations
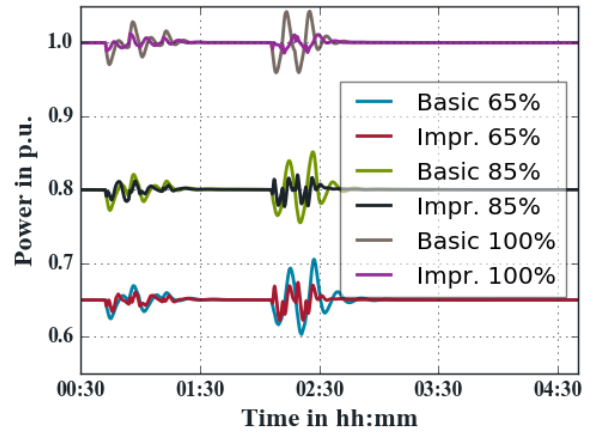
in power output, occurring for example during soot blowing processes which are a common routine, because underfulfillment would be charged with penalties and overfulfillment will not be compensated. This makes a higher quality of control beneficial. When the operators offers positive control power during full load operation, the plant only could be run with a certain margin to its maximum load under consideration of load fluctuations. If the deviations in power output during soot blowing cause the highest amplitudes in the current power plant operation schedule it can be seen, that an improved control strategy, like the one proposed in this paper, enables the operator to run the plant at a higher load and making more profit while maintaining the needed margin to maximum load for secondary control power.

4 Conclusion

The current state of the library *ClaRa* for the simulation of power plants has been presented to be one of the most complete and complex open source library for the simulation of Clausius Rankine cycles. Due to its deep insight at equally high transparency it addresses both new and experienced users. The library is open to be extended in the future, by both new component and physics models within the *ClaRa* (which is work in progress by the authors) as well as in terms of new libraries, the so-called *ClaRa_AddOns*. The latter mentioned path allows the scope to be widened to new fields like biomass or solar heated applications.

As a use case the optimisation of a power plant unit control has been outlined. The results prove the outstanding opportunities that are introduced by system simulation allowing to understand complex processes better by evaluating unmeasurable process variables and to test innovative control concepts without risks.

5 Outlook

The further development of the library will, amongst others, introduce so-called six-equation-models for extra precise calculation of two phase flow conditions in pipes. Furthermore, all models are under permanent review with respect to zero flow and other non-design conditions. The library is planned to be extended by special header components enabling stress evaluation.

References

L. Andresen, P. Dubucq, R. Peniche, G. Ackermann, A. Kather, and G. Schmitz. Status of the transient library: Transient simulation of coupled energy networks with high share of renewable energy. In *Proceedings of the 11th International Modelica Conference, Versailles, France*, 2015.

B. Bachmann, P. Aronsson, and P. Fritzson. Robust initialization of differential algebraic equations. In *Proceedings of the 4th International Modelica Conference, Vienna,Austria*, 2006.

J. Brunnemann, F. Gottelt, K. Wellner A. Renz, A. Thüring, V. Roeder, C. Hasenbein, C. Schulze, G. Schmitz, and J. Eiden. Status of ClaRaCCS: Modelling and Simulation of Coal-Fired Power Plants with CO_2 Capture. *Proceedings of the 9th International Modelica Conference, Munich, Germany*, pages 609 – 618, 2012.

F. Casella, M. Sielemann, and L. Savoldelli. Steady-state initialization of object-oriented thermo-fluid models by homotopy methods. In *Proceedings of the Modelica Conference 2011, Dresden, Germany*, 2011.

C. Gierow, M. Hübel, J. Nocke, and E. Hassel. Mathematical model of soot blowing influences in dynamic power plant modelling. In *Proceedings of the 11th Modelica Conference*, 2015.

F. Gottelt, K. Wellner, V. Roeder, J. Brunnemann, G. Schmitz, and A. Kather. A Unifieded Control Scheme for Coal-Fired Power Plants with Integrated Post Combustion CO_2 Capture. In *Proceedings of the In 8th IFAC Conference on Power Plant & Power System Control, Toulouse, France*, 2012.

International Energy Agency. Tracking Clean Energy Progress 2015. Technical report, 2015.

S.E. Mattsson, H Elmqvist, M. Otter, and H. Olsson. Initialization of Hybrid Differential-Algebraic Quations in Modelica 2.0. In *Proceeding of the 2nd International Modelica Conference, Oberpfaffenhofen, Germany*, 2002.

M. Najafi. Selection of variables in initialization of modelica models. In *Proceedings of the 2nd International Workshop on Equation-Based Object-Oriented Languages and Tools, Paphos, Cyprus*, 2008.

C. Schulze. Table based calculation of thermophysical properties for simulation of thermodynamic systems. In *Proceedings of the ITI Symposium*, 2013.

C. Schulze. *A Contribution to Numerically Efficient Modelling of Thermodynamic Systems*. PhD thesis, Technische Universität Brauschweig, 2014.

T. Vahlenkamp and S. Wischhusen. FluidDissipation - A Centralised Library for Modelling of Heat Transfer and Pressure Loss. In *International Modelica Conference, Bielefeld, Germany*, 2008.

T. Vahlenkamp and S. Wischhusen. FluidDissipation for Applications - A Library for Modelling of Heat Transfer and Pressure Loss in Energy Systems. In *Proceedings 7th Modelica Conference, Como, Italy*, September 2009.

VDI/VDE. VDI/VDE Guideline 3508: Unit control of thermal power stations. *Association of German Engineers (VDI) / German Association of Electrical Engineering and Information Technology (VDE)*, 2003.

S. Velut and H. Tummescheit. Implementation of a Transmission Line Model for Fast Simulation of Fuid Flow Dynamics. In Christoph Clauß, editor, *Proceedings of the 8th International Modelica Conference; March 20th-22nd; Technical Univeristy; Dresden; Germany*, Linköping Electronic Conference Proceedings, 2011. doi:http://dx.doi.org/10.3384/ecp11063446.