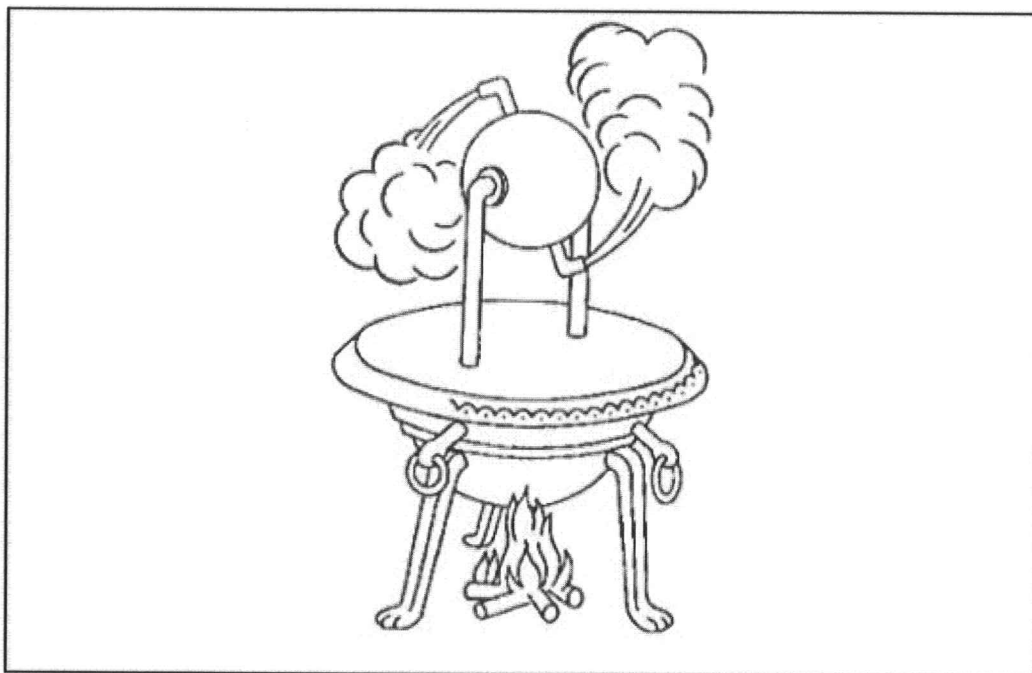


# CHALMERS



## Modelling and Control of a Steam Turbine

**Master of Science Thesis [in the Masters Programme Systems, Control and Mechatronics]**

**OLLE HÖGBLOM**

Department of Signals and Systems  
*Division of Automatic Control, Automation and Mechatronics*  
CHALMERS UNIVERSITY OF TECHNOLOGY  
Göteborg, Sweden, 2009  
Report No. EX063/2009

## **Abstract**

The goal of this master thesis project was to develop a model of a steam turbine for dynamic simulation and control purposes. The turbine was to be implemented in an extensive steam net model and the entire net was to be controlled with the method Split-Range. This means that the signal from one controller is used to control several actuators. The modeling was performed in Dymola, a graphical user interface for Modelica. The most important variables from the turbine model were the power output, the mass flow and the pressures. During the work it turned out that an existing model could be used but the numerical implementation in Dymola had to be improved.

The Split-Range application was built from scratch with several specifications considered, such as auto/hand mode transitions of the actuators and actuator limitations. The final results from this work are programs to be used for dynamic simulation in other real steam nets and the Split-Range function, which can, beside valves, also be used to control other approximately linear actuators.

## Sammanfattning

Målet med detta examensarbete var att utveckla en modell av en ångturbin för dynamisk simulering och reglering. Turbinen skulle användas i en komplex ångnätsmodell och hela nätet skulle regleras med en form av Split-Range, vilket innebär att signalen från en regulator används för att styra flera ställdon. Modelleringsarbetet utfördes i Dymola, ett grafiskt användargränssnitt för Modelica. De viktigaste variablerna som turbinmodellen skulle ge var effekt, massflöde och tryck. Under arbetet visade det sig att en befintlig modell kunde användas, men den numeriska implementeringen i Dymola var tvungen att förbättras.

Split-Rangemodulen byggdes upp från grunden och flera komplicerande specifikationer togs i beaktande såsom t.ex. auto-/handläge, omställningar av ställdon och ställdon begränsade till bara en del av sitt arbetsområde. Resultatet av arbetet är en noggrann turbinmodell som är mindre beräkningstekniskt komplicerad och en användarvänlig och generell Split-Rangefunktion.

Det slutliga resultatet av detta arbete är de program som tagits fram. De kan användas för dynamisk simulering av verkliga ångnät och Split-Range funktionen kan förutom till ventiler också användas för att styra approximativt linjära ställdon i andra applikationer.

## Aknowledgement

I would like to acknowledge and extend my heartfelt gratitude to the following persons who have made the completion of this master thesis possible:

My supervisor Veronica Olesen, for her vital encouragement and support.

Tosten Wik, my examiner, for his understanding and assistance.

Frida Ekström, Gabriel Ordéus and Mikael Eriksson, for the help and inspiration they provided.

Ulf Håll, prof. in turbo machinery who has straightened out several question marks regarding fluid dynamics inside turbines.

Everyone at Solvina who has endured my presence and answered all my questions.

Most especially to my family, friends and my loved dog Cheeta.

Last but not least I would like to thank god, evolution or whatever has now taken us this far and made this thesis possible.

# Table of Contents

1	Introduction .....	1
2	Theoretical Background .....	2
2.1	Fluid Dynamics.....	2
2.2	Steam Turbine Theory .....	2
2.3	Steam Nets .....	4
2.3.1	Turbines in a steam net.....	4
2.4	Split-Range Control .....	5
3	Turbine Model Development.....	7
3.1	Old Model, Problem Description.....	7
3.2	Method, Problem Verification .....	7
3.3	New Model .....	9
3.4	Simulation Results .....	9
4	Controller Development .....	10
4.1	Specifications.....	10
4.2	Method.....	10
4.2.1	General Implementation .....	11
4.2.2	Limitations.....	12
4.2.3	Auto to Hand Transition.....	13
4.2.4	Hand to Auto Transition.....	15
4.2.5	Valve characteristics.....	17
5	Simulation Model .....	18
5.1	Steam Net.....	18
5.2	Split-Range Controller Unit.....	19
5.2.1	Split-Range settings.....	19
5.2.2	Max selections .....	20
5.2.3	Accumulator control.....	21
5.2.4	Boiler control .....	21
5.3	PI tuning with the lambda method .....	21
6	Results .....	24
7	Discussion and Further Work.....	26
8	Conclusions .....	27
9	References .....	28

# 1 Introduction

Two important keys for successful design of control strategies are modelling and simulation. At the technical consulting company Solvina AB in Västra Frölunda, Sweden, Dymola is used as a tool for modelling and simulation. Dymola is a graphical user interface for the object oriented modelling language Modelica, in which both editing and simulations can be performed. Dymola also has the advantage of user defined model libraries. At Solvina a large model library called SteamPower, consisting of steam/water energy system components, is used. This and other model libraries are continuously improved and this master thesis is a part of that work.

The work has been carried out in two parts, first a modelling part and afterwards a control part. In the modelling part, improvements have been made to Solvina's steam turbine model. That work started with an extensive literature study to investigate different turbine models. In the control part the improved turbine model was used in a complex steam system with many control loops needing to work together.

## 2 Theoretical Background

### 2.1 Fluid Dynamics

In the study of all fluid flows, some basic equations, such as energy conservation and mass conservation, have to be considered. In a steam turbine, steam reaches high speeds and its impact on flow properties has to be carefully considered. Energy in a flowing fluid can be divided into two parts, one kinetic (motional) part and one static (pressure) part. At some points in the flowing fluid velocity is zero, for example in front of a static object in the stream line. Then all energy in the fluid is in its static form, and hence, at these points the pressure part is at its maximum. These points are called stagnation points and the pressure and temperature here are called stagnation pressure and temperature. This phenomenon can be described by the Bernoulli's equation, which states that the total pressure is the sum of the static pressure and the dynamic pressure.

$$\text{Total pressure} = \text{Dynamic pressure} + \text{Static pressure} \quad (1)$$

Note that dynamic pressure and total pressure are not pressures in the usual sense, since they are not measurable with ordinary pressure measurement equipment. Anyway, they are significant in the study of all fluid flows. A simplified form of Bernoulli's equation can be written as: This is just another form of the energy conservation principle since these pressures are highly related to the energies.

As a fluid flows through a hole, an orifice or a stage in a steam turbine, velocity increases in order to satisfy the equation of continuity. This increase results in a gain in kinetic energy and due to the principle of energy conservation, the static energy (pressure) has to decrease. This phenomenon is called the Venturi effect and is a very important issue to take into account when carrying out flow calculations. The limiting case of the Venturi effect is when the fluid reaches the speed of sound in the given media. Then the flow is said to be choked or sonic. In a choked flow, the mass flow is no longer related to the pressure drop, it is dependent only on upstream conditions. The Mach number is defined as the fluid velocity divided by the speed of sound in the given medium; hence the Mach number is one or more at choked conditions.

### 2.2 Steam Turbine Theory

In a steam turbine, high pressure steam is expanded in a number of steps of alternate stators and rotors. There are several different types of turbines, but they can all be divided into the two groups; axial and radial turbines. This thesis is only considering axial turbines where the flow is entering the stators and leaving the rotors in the same, axial direction (see Figure 1).

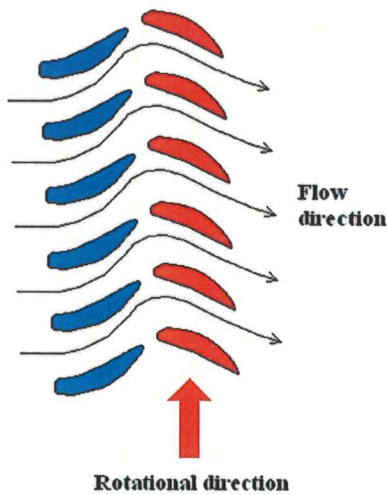


Figure 1 Cross section of an axial reaction turbine, the blue vanes are stator vanes and the red are rotor vanes

In a turbine, flow can be expanded and converted into rotational motion in two different ways, impulse or reaction, or a combination of them. In an impulse turbine the expansion is carried out in the stator, which works as a nozzle. The outgoing jets, containing high levels of kinetic energy are changing direction as they meet the rotor and convert their energy into rotational shaft motion. In a reaction turbine the expansion takes place in the rotor itself, which is constructed as nozzles and the reaction force is converted into rotational motion. The most common type of turbine is a combination of these two extreme cases i.e. the expansion takes place in both the stator and the rotor.

Several articles regarding steam turbine modelling were studied during this project and the conclusion from them was that almost all existing turbine models are based on the same old theory. For example (Asok, 1980), (Bolek, 2001) and (Chaibakhsh, 2008). In some other articles the authors were using CFD as a tool for detailed turbine modelling, which is not applicable here since it would result in a much more calculation intensive model to simulate. An example of this is (Anglart et al, 1992). From studying these articles, the conclusion could be drawn the existing model was correct in the sense of the equation used. The problem was the implementation in Dymola. The theory used in (Asok, 1980), (Bolek, 2001) and (Chaibakhsh, 2008) as well as in the existing Dymola model is based on an old relation stated by Prof A. Stodola in 1927, the Stodola parabola (Stodola, 1927):

$$\dot{m} = C_t \sqrt{\frac{p_1^2}{v_1} \left( 1 - \left( \frac{p_2}{p_1} \right)^2 \right)} \quad (2)$$

Where  $\dot{m}$  is the mass flow,  $p_1$  and  $p_2$  are the inlet and the outlet pressures respectively,  $v_1$  is the inlet specific volume and  $C_t$  is a constant which should be based upon experimental data. Since the steam is over heated during the whole passage through the turbine, it can be approximated with an ideal gas with satisfying accuracy. Hence, equation (2) can be rewritten in the following form:

$$\dot{m} = C_t \sqrt{\frac{p_1^2}{RT_1} \left( 1 - \left( \frac{p_2}{p_1} \right)^2 \right)} \quad (3)$$

where  $T_1$  is the inlet temperature and  $R$  is the gas constant. The existing theory describing steam turbines is almost 100 years old. However, development has not come much further and turbines look almost the same now as they did then. The most important difference is that the turbine working pressure is much higher today. Also there is often a much higher pressure drop through each step since turbines are built in fewer steps in order to reduce material costs. Another thing that has changed is the design of vanes; where performance has been improved using tools such as CFD. The rising pressure in new turbines often implies that the mass flow dependence on pressure disappears. The reason for this is that the flow becomes choked somewhere in the turbine, often at the inlet and then the mass flow remains constant independently of the pressure difference. This means that at choked conditions, mass flow only depends on upstream conditions and does not depend on anything going on downstream. Hence, mass flow should be calculated over the choked region and used downstream in the turbine model. The same equation, equation (2) can be used but the pressure ratio should be taken from where the flow became choked.

At choked flow conditions, it would of course be incorrect to try to calculate the mass flow using the Stodola's equation, without changing the pressure ratio. A lot of knowledge is required about the turbine to decide whether the flow is choked and where. This was never a problem with the old steam turbine operating with low pressure drops. However, by using the same models on today's choked turbines we know we introduce an error. A model solely based on Stodola's parabola does not take care of this fact and has to be corrected in some way to handle the issue of choked flow.

## 2.3 Steam Nets

In most industries, energy is an important raw material. Pulp and paper industries consume large amounts of energy in form of heat and electricity. Steam is very common as a heat carrier since it is cheap, environmentally friendly and has good heat transfer properties. The steam used is produced at high pressure in a boiler and used in many different facilities, such as paper machines, evaporators, drying equipment, etc. In a paper mill, steam is mainly produced in the recovery boiler, where the fuel is waste from the process and the boiler itself is an important step in the chemical recycling process. The steam is produced at high pressure and temperature while the steam demand is often at different, lower pressures. The steam net is therefore divided into subnets, one for each pressure and all connected through valves, steam turbines and accumulators.

### 2.3.1 Turbines in a steam net

When a turbine is used between several steam nets of different pressures, steam can be injected to or bled out from the turbine at the different nets depending on production and consumption on the nets. For a flow to be possible in the desirable direction, a driving force (pressure difference) is required. The pressure should be constant at the steam nets but the pressure profile inside a turbine is dependent on the mass flow through it if this flow is defined upstream. Hence, the driving force is not always met for the injection or bleed to be possible. If the mass flow through the turbine becomes much less than the design flow, pressure at an intermediate pressure bleed can decrease

below design pressure of that net and nothing can pass through the valve. In this case, it does not matter if the valve is open or closed and the phenomenon can sometimes be confusing for the operators as there is no flow through an open valve. Usually an in- or outlet valve at the turbine is only used for flow in one direction. In order to prevent flow in the wrong direction for example in case of decreased pressure inside the turbine the valves has to be check valves. Otherwise the steam can pass from the net through the low pressure part of the turbine and out to the back pressure net.

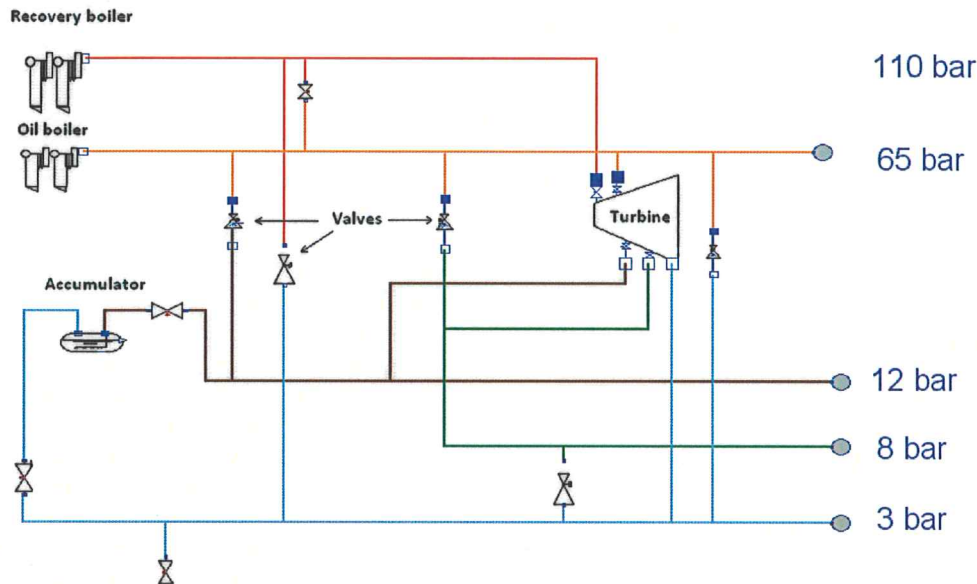


Figure 2. Example of a steam net

Steam turbines are the most energy efficient way to reduce pressure, since the energy can be converted to electricity or mechanical work. However, it is of great importance for the consumers that steam pressure is constant at all different levels. Hence, most commonly a steam accumulator, providing the different nets with steam and absorbing excess steam produced, is used. The accumulator is an important piece of equipment since dynamics of the boiler is much slower than dynamics of the consumers. Rapid disturbances in amount of consumed energy are common, for example at start up or shot down of a piece of machinery.

In order to keep pressure constant at all levels, a control strategy with built-in priorities of the nets is needed. The second part of this thesis was to develop one such strategy and use it to control a steam net model. The objective was to maximize the amount of steam passing through the turbine while fulfilling a number of constraints. This was performed using a form of split-range.

## 2.4 Split-Range Control

Split-range control is a common method used in control systems with two or more actuators, e.g. valves, being active in different parts of the control range 0. The benefit of split-range is that a single well tuned controller can be used for control of a number of actuators. The controller output is then fed through a split-range application and distributed as different control signals to the correct actuators.

For example, to reduce the pressure at a high pressure steam net, reduction through a steam turbine, reduction to accumulator and direct reduction to other subnets are possible, see Figure 2. It is of course preferable to reduce the pressure by first opening the valve to the turbine. Hence, priorities are needed. An example of different valve positions as functions of the controller output can be seen in Figure 3.

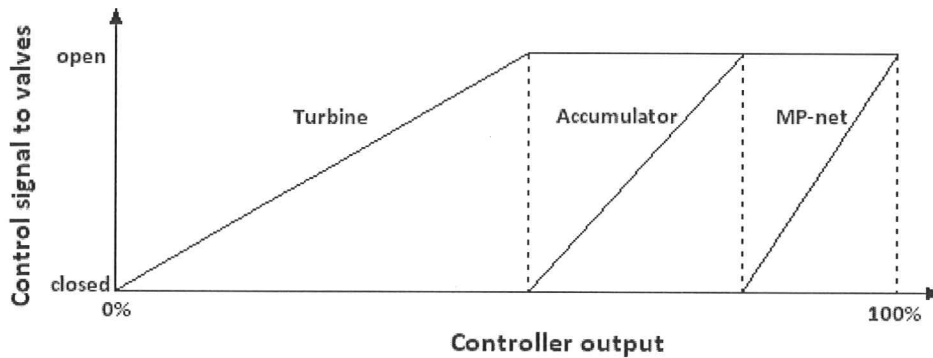


Figure 3. Example of split-range.

The slopes of the lines in Figure 3 are determined by the valve sizes, in order to achieve a constant gain in the whole working range. Hence, a large valve will have less slope than a smaller one. The orders in which the different valves are arranged in the above example are determined by the priorities of the valves.

A split-range controller can be used when one controller is controlling many valves. In this application the opposite is also valid, i.e. several measurements are used to control one actuator, since controllers associated to different nets use the same valves between the nets. Here, priorities of the nets are needed to choose the most important signal. This can be solved by implementing max functions.

## 3 Turbine Model Development

### 3.1 Old Model, Problem Description

The existing steam turbine model in the model library SteamPower is very simple in the sense that it does not take any internal phenomena into account. It calculates the mass flow and pressure drop using the Stodola equation (2), and power output is calculated by mass flow and enthalpy difference over the turbine together with a specific isentropic efficiency. The Stodola constant has to be specified by the user and should be based on data of the actual turbine.

All steam data needed for the calculations are achieved from a data library in Modelicas's built in database. The model works quite well in most cases and it is sufficient to simulate turbines with few in- and outputs operating at small pressure differences. When building large systems with multiple in- and outflows, unforeseen effects occur. Sometimes there are problems during initialization; Dymola is unable to find a good starting guess despite almost perfect input guesses. This phenomenon indicates that the simulation is very sensitive to starting conditions. Sometimes the power output from the model can also become unreasonably low or even negative.

Another problem with the model appears as the Stodola constant is to be determined. The model did only fit for a restricted range. If it should work far away from its working range a new Stodola constant was found. This indicated that something was wrong with the model, not only the implementation in Dymola.

### 3.2 Method, Problem Verification

To solve the mentioned problems above, they had to be investigated further. Several different steam net models, trivial as well as more complex, were built with the existing turbine in order to trigger the problems. The error messages from the different cases were studied in detail and some conclusions were drawn. Sometimes the calculations ended up at unreasonable values, for example negative pressures or pressures below the triple point. Sometimes the initial temperature or enthalpy guesses, calculated by Dymola, were far out of range, much higher than the maximum allowed temperature or enthalpy in the database.

The model works quite well at relatively small pressure drops but if the upstream pressure is increased and flow becomes choked, the model does not take the choking effect into account and still calculates the mass flow using the Stodola equation (2). This is not a problem during simulation of old turbines operating at non choked flows. However, when the actual flow becomes choked the model gives a larger mass flow than the real turbine, provided that the Stodola constant is based on data for the non choked flow. On the other hand, if the turbine is choked when the experimental data for the Stodola constant determination are achieved, the model gives lower mass flow than the real turbine for non choked flows.

As Dymola iterates, a number of simplifications are made and the equations finally solved are a lot easier with respect to numerical complexity. To be able to start a simulation initial guesses of all variables are needed. Some of these guesses are given as inputs by the user and some of them are calculated by Dymola using Newton based

methods. However, the users of Dymola do not have access to all details in the iteration process.

The Stodola equation, (2), does not contain any dynamics and hence the Dymola model is also totally static. If several static equations are combined, the expression to be solved by Newton's method becomes very complicated and sometimes too complicated to be solved with the Dymola solver. This is most likely to occur during the initialization process when no knowledge about the dynamics is available and the system should be solved only with the equations and the start guesses given. This can also happen later in the simulation for example at a relation triggered event. This is probably the reason for the initialization problem seen in the model when extremely low pressures and high enthalpies and entropies were found as input guesses by Dymola.

Sometimes when implemented in several steps, the old model failed during the initialization due to the way Dymola was rearranging the equations. The system was then solved by dividing an expression with the right hand side of equation (3) and somewhere during the initialization process, the pressure quota became one and hence the initialization failed due to division by zero.

Sometimes, when the back pressure conditions are close to the saturation line, steam data from the data library is completely wrong. This is a consequence of the way the model calculate the outflow conditions (see Moliere diagram in Figure 4). First it performs an isentropic pressure reduction, from point 1 to 2 on the red line, and gets the enthalpy at this "ideal" point. Afterwards the isentropic efficiency is used to determine the real outflow condition, point 3. This means that if the real conditions are close to the saturation line (blue 3), the ideal point is below it (blue 2). In the model it is specified that there is only vapour in the turbine and hence an ideal point below the saturation line is forbidden. A solution to this problem is either to allow water in the turbine model during iteration or perform the calculation in opposite order as shown by the green line in Figure 4; first move along the isobar and then make the pressure reduction. Using the later alternative, an assumption of parallel isobars must be made. This is no problem but unnecessary since the first method works satisfactory if a check is performed, that there is only vapour in the real turbine, i.e. in point 3.

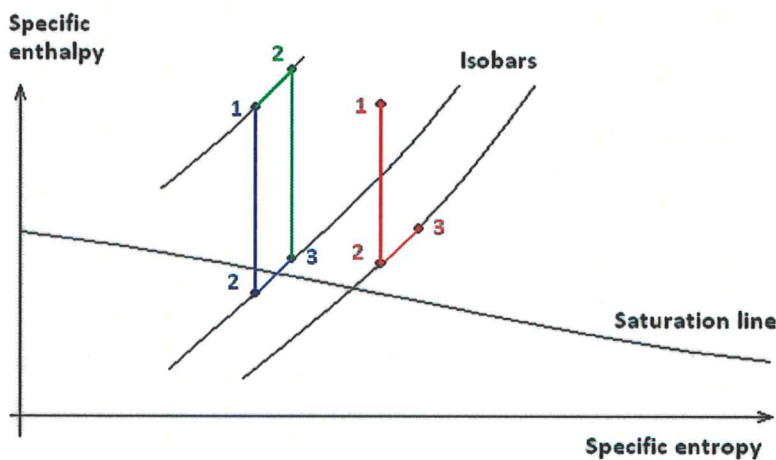


Figure 4 Moliere diagram with methods of turbine calculations

### 3.3 New Model

Development of a new improved steam turbine model began with a literature study in order to find new models easier to initiate and fit to measurements. However, after the literature study, the conclusion could be drawn that almost all turbine models are based on the old Stodola parabola, Equation (2). This was an indication that the old model in some way was correct and just had to be updated to include the choked flow condition and stabilized with respect to the numerical problems. As mentioned in the theoretical background, it is impossible to take the choked flow condition into account without knowledge about the pressure profile inside the turbine. When the model is used for a real turbine, the turbine manufacturer has to be contacted for turbine internal details in order to achieve a satisfying result. Possibility to include this kind of turbine data has been added to the model.

The initialization problem was solved by just excluding the impossible, negative pressures and pressure below the triple point by adding a max function at the inlet pressure. Extreme temperature was avoided in the same way but with a min function operating at the enthalpy instead.

Also singularities were avoided by this kind of function to exclude the pressure quota  $\frac{p_2}{p_1} = 1$  in equation (2) from the iterations. All together three max/min functions were added to the original model in order to avoid the initialization problems.

### 3.4 Simulation Results

The new model has been tested against the old model in different extensive models and the result of this is that the new model often converges where the old one fails, either in the initialization or further on in the iterations.

A steam net model was earlier developed for control purposes at a large paper mill. Because of the problems with the old turbine model, the net model did not initialize properly and the turbines in the net were modelled as valves with cooling to handle the problem. This is a satisfactory approximation to make when the power output is not needed in the simulation and that was true for this old control case. The valves in this steam net model were exchanged to the modified turbine model as a final test and the simulation converged.

## 4 Controller Development

### 4.1 Specifications

The split-range controller developed in this thesis should fulfil the following specifications:

- The total function should be general in its implementation, i.e. the same function can easily be used to control different steam nets. Data can be put in and removed easily.
- It should be easy to add and remove valves, change their priority etc.
- Consideration of whether a valve is in manual mode. If a valve is in manual mode it should be removed from the controller without any dead zone in the controller signal. The transfer should be bumpless and the signal should be split on the remaining valves automatically.
- Other restrictions should be accounted for. For example fixed valves or valves restricted to only a part of their working range, something taken out of service etc.
- Feed forward should be possible to achieve a faster control.
- The controller should easily be translated from Dymola to other programming languages to be implemented in a real plant. The function should be easy to understand and translate.

All these specifications had to be accounted for when the function block was created. The block was built in steps and each function was tested and evaluated carefully. In order to fulfil the specifications and make them work together they were also combined in several ways to detect combinatory problems.

### 4.2 Method

The model was first built in Matlab in order to derive a working structure which takes all the above mentioned specifications into account. Later the model was translated to Dymola and tested in a steam net model. The reason why the initial modelling was not performed in Dymola was the lack of competence the author had in this programming language. Although Dymola has a useful graphical interface, this was not used much, but the entire program was written in Modelica text. When using the function blocks in Dymola the structure becomes large and looks complicated even for easy calculations. Furthermore the Matlab code contains structured text and the translation from Matlab to Dymola becomes easier when not using the function blocks. The text based code is written in basic logical functions so that it can be translated into any programming language.

The modelling was performed in steps. First, a basic model was made, which only splits the signal into the desired number of control signals (see Figure 5)

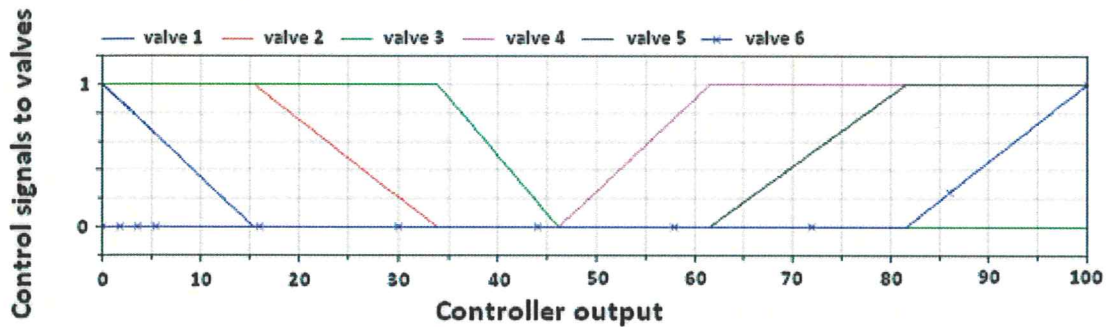


Figure 5. Control signal split into six control valves with no limitations

The controller output in percentage can be seen on the x-axis and the signals to the valves are on the y-axis. In this example the signals to three input and three output valves are visualized by different colors. To the left in the figure, the input valves can be seen and to the right the output valves. The priorities of the different valves increase from the middle of the figure, i.e. the point where all valves are closed. Thus, for the input valves the priority decreases and for the output valves the priority increases with an increasing controller signal.

The model was expanded to include the other specifications, one at a time. However, sometimes the steps were made in an unfavorable order and dead ends arose. This meant that the work was done from scratch a few times with greater and greater knowledge and understanding each time.

#### 4.2.1 General Implementation

The split-range function was made in one block separate from the controller used since it should not matter what kind of controller used. The split is not a controller itself; it is only the signal converter from the controllers output to several different valves. How it is used can be seen in Figure 6.

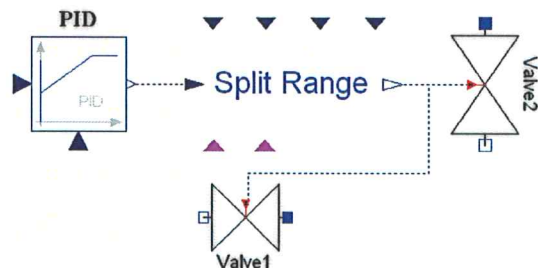


Figure 6. PID controller and split-range function

In this example, the function is used to control two valves but since it is built with vectors it can be used to almost infinitely many actuators of any kind. The other inputs to the split-range function, seen as blue and pink arrows in Figure 6, are inputs in order to be able to fulfill the different specifications.

In order to make the programme flexible and able to handle various numbers of valves, vectors of variable length are used. The vectors have one element for each valve to be controlled, thus valves can easily be removed or added just by changing the vector length. Often the valves connected to a steam net are of different sizes, and scaling is acquired in order to achieve a constant gain in the entire working range. The sizes, upper and lower limitations and the auto/hand mode are all introduced as vectors of the same length. The valve size vector is given in actual sizes in kg per second at a nominal

pressure drop, the lower and upper limitations are given as real decimal numbers between zero and one at which the limitation becomes active. The auto/hand vector is a Boolean vector where true means that the corresponding valve is in auto mode and false means manual mode.

Default values for all vectors except the size vector were introduced by using the cardinality function in Dymola. This means that the ports do not require a connection from the outside. The function is particularly useful when the split-range normally is in auto mode and no limitations are active. Then the only input vectors required for the programme to work are the ones with the valve sizes.

Since many commercial controllers provide output in percent, all valve sizes were scaled. The scaling was first made by a summation of all the capacities to achieve a total working range for the controller, from the case where all in-valves were fully open and all out-valves were closed to the opposite. Each valve was assigned a part of the total capacity depending on their sizes. This way the controller output working range was divided to the different valves. The entire working range of the controller output from the controller, given in percentage, was distributed to the scaled working range of the valves in priority order. The valves are assumed to be linear from valve position to mass flow at a constant pressure drop. This is not fully true but it is a satisfying approximation. If more valve characteristics is needed it has to be included in the valve model itself.

#### **4.2.2 Limitations**

If a valve is restricted by a lower limitation, the valve will have a leakage flow even if the controller tries to close it. A similar phenomenon is obtained when an upper limitation is active; the maximum flow specified for that valve may not be possible to achieve. This has to be accounted for so the curves are shifted in a way that gives the desired flow at a given controller output. When a desired flow in a valve is not possible because of a limitation, the next valve in priority should take over where the first one becomes limited. An upper limitation at an inlet valve works in a similar way as a lower limitation at an outlet valve since they both limit the pressure rise on the net. The same is valid for a lower limitation at an inlet valve and an upper limitation of an outlet valve since they limit the pressure reduction. The shifts are performed so that no dead zone appears in the control signal and so that two valves are never active at the same time.

The priorities of the valves were introduced to the programme by writing all the vectors in priority order for the inlet valves and reverse priority order for the outlet valves. Giving them in reverse order is a trick used to reduce the amount of code since Dymola, as well as control system languages are not developed to deal with vectors and a lot of code is required to do easy vector calculations such as reversing a vector.

The slopes of the curves were obtained by the straight line equation using the valve sizes and the fact that valves are working between zero (closed) and one (open). The slopes will remain constant for a given set of valves regardless what happens with limitations or auto/hand mode changes. To know where the curves start and end the priority order and limitations are important. Since the vectors were given in correct order, the only thing to take into account was the limitations. First a new vector with capacities were created where the limited parts of the valves were removed and then the start of the first valve was determined by taking the range of the upper limitations of the inlet valves and the lower limitations of the outlet valves, since these are the limitations limiting the pressure rise on the net. After this a cumulative summation of the new

capacity vector was performed where all the lower and upper limitations were taken into account. This gave the different intervals where the valves are active. Using the slopes of the lines and these points, the lines were found from the straight line equation. The domain of the curves is  $[0,1]$ , since a valve cannot be more open than fully open or more closed than fully closed, hence the domain is restricted by a maximum and a minimum function in Dymola. An example of a limited valve can be seen in Figure 7.

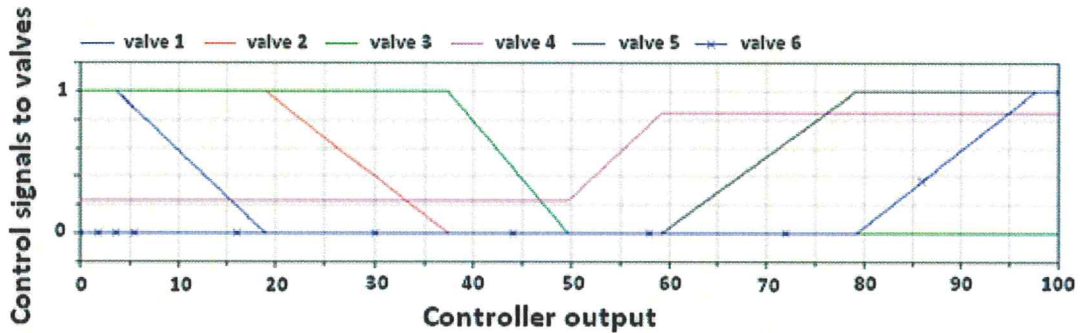


Figure 7. Both upper and lower limitations at the first outlet valve

In Figure 7, the valves are shifted to cover the gap left by the first outlet valve (purple) at its limitations, cf. Figure 5.

#### 4.2.3 Auto to Hand Transition

A similar shift is needed when a valve is changed from auto to hand mode. When this happens during operation, the valve becomes unavailable for the controller and it has to be removed from the split function. The other valves have to take over its part of the working range and leave the dead zone at the end of the range instead. The shift has to be made in three different ways depending on which valve is active when the transition is made, see Figure 8.

Either the valve is active, not active on the left side of the active valve or not active on the right side. These cases are shown in Figure 8, the red line describes the valve which is put in manual mode, and the blue dot describes the state of the system when the transition is made. If a not active valve is put in hand mode during simulation, the shift has to be made such that nothing is changed until that valve was supposed to be opened by the controller. If that valve is to the left of the active one, everything to the left of that valve should be shifted to take its place and leave a dead zone at low controller output, Case A in Figure 8. The opposite happens if a valve above the active one is taken out of auto mode, the curves to the right of that one should be shifted to the left and leave a dead zone at high controller output, Case B. If the active valve is put in hand mode, that one has to be removed and all other valves have to be shifted towards the point where the system was when the valve was put in hand mode, Case C.

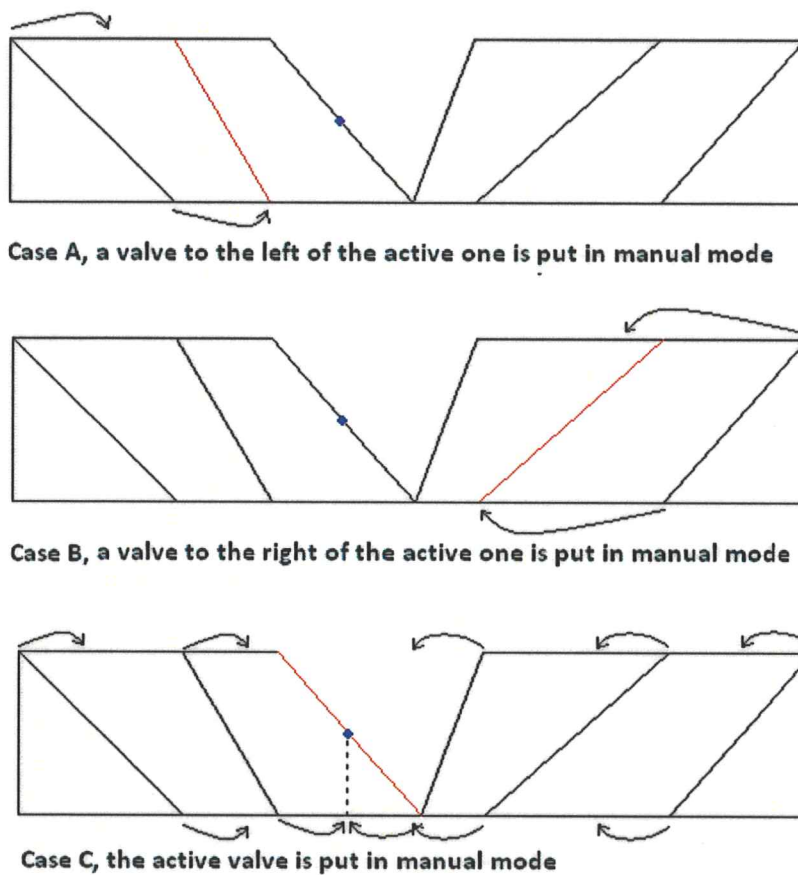


Figure 8. Three alternative cases of an auto/hand transition

When an auto/hand transition is performed, the valve is locked at its position, the shift is made and the controller signal is split on the remaining valves. After that, the locked hand position can be changed by the process operator and the resulting change in pressure will be taken care of by the controller through feedback. Another possibility is to use feed forward to reduce the effect by the operator acts. This would stabilize the pressure but reduce the possibility for the operator to achieve what he wants. The feed forward possibility is not used in the programme since knowledge about the reason for the hand mode transition is not known.

To perform the shifts described above, knowledge about the control signal in the transition moment is required, since all the curves should be shifted towards this point. The change is detected and the control signal coming into the block is then saved as a new variable and this value is compared to the vector created by the cumulative summation to detect which valve is active. After detecting the active valve, the valve number can be compared to the valve put in hand, and the shifts are possible to make by dividing the problem in the three different cases. An example of putting one valve in manual mode is shown in Figure 9

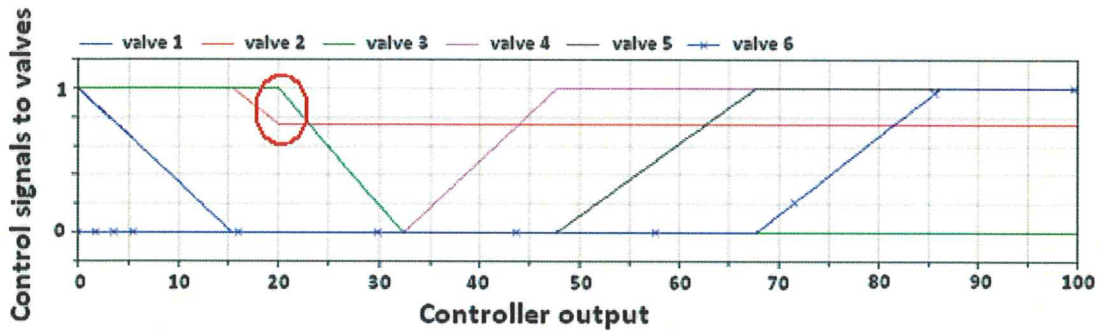


Figure 9. The second valve is put in hand mode when it is active after 20s of simulation

The figure shows case C described in Figure 8, valve number two is put in manual mode after 20 seconds of simulation and at a time when it is also the active valve. It is locked at its position and valve number three is taking over when controller output is continuing to increase.

If another valve also is put in hand the shift has to be made from the previous already “shifted” vector and not from the initial auto vector. This was at first leading to problems when implemented in the programme. The solution was to update the shifted vector each time a valve was put in hand in a new algorithm part of the program.

#### 4.2.4 Hand to Auto Transition

When the transition is made in the other direction, from hand to auto, things becomes a bit more complicated. Regardless of which valve is active, the shift should be performed to reset the earlier shift made when the valve was put in hand. If some other valve has been put in hand or auto during the time the valve has been in hand, another shift has been made and the consequences of this have to remain after the hand/auto transition. Since the valves cannot change position immediately and the hand position probably is different from the auto position some kind of smooth transition has to be used.

The current interval vector and the interval vector after the transition are known. To make the transition between them smooth, a ramp function is used. A number of different methods were tested to achieve this with various results. Common for the different methods was that a fixed time period to make the transition was predefined.

During the transition time, the system can change and the final valve positions are probably not exactly the same as they should have been if the transition could be made immediately. Since the final valve positions are not known during the ramping, it is not possible to make the ramping directly on the valve positions.

The first solution was to translate the Boolean auto/hand vector to a real vector containing ones and zeros. Afterwards the ramp could operate on the element corresponding to the valve that was put into auto. In this way more and more of the new interval vector was weighted in and the shift was performed smoothly without knowledge of the final valve positions. However, when ramping the auto/hand vector, the valves were ramped to their unsaturated values instead of their saturated values. The consequence of this was that the valves reached their final position at different times and a total deviation during the ramping time could not be avoided. Since the interval vectors are used to form the unsaturated valve positions and ramping the auto/hand vector will affect the intervals, it is impossible to achieve a satisfactory ramping of the saturated valve positions.

The problem with performing an adequate ramping of the valve positions is that the final positions are not known as the controller signal will change during the ramping time. It would be easy to ramp the positions correctly if future control signals were known. In the next method tried, the ramping time was divided in several small steps and in each step a prediction of the control signal was made in order to have something to ramp towards. The slope of the line if the transition was to be made in one time step was calculated and compared to a predefined maximum slope. If the slope exceeded the maximum slope, the valve should follow the maximum slope instead.

The fraction of the desired slope and the slope used was also calculated to decide all the other valves change in position during that time step. The objective was to have the sum of the valves position to correspond to the desired input signal from the controller. This method worked properly until the end of the ramping where the slope reached zero and the slope was used in the denominator during calculations. The solution to this was to calculate the fraction only if the slope was limited by the maximum slope, otherwise no slope had to be limited and the ramping could be made directly on all the curves. However, this method gave a little deviation during the entire time the ramping was active for heavy variations in the controller signal. For approximately constant control signal the method worked properly. Decreasing the step size was not effective, thus another method was investigated.

The new solution was to calculate only the slope of the transition valve in the previously described way and let feed forward take care of the other valves. That guarantees that no difference between the control signal and the sum of the valve positions can occur since the control signal is used directly to calculate the positions. Using this method no fraction needs to be calculated and the problem with division by zero was avoided without changing equations during the ramping. Figure 10 shows a simulation case when the same valve as in Figure 9 is put back in auto mode after 50 seconds further in to the simulation. The valve is ramped with its maximum slope and the other valves follow in order to achieve the desired total inflow outflow relation.

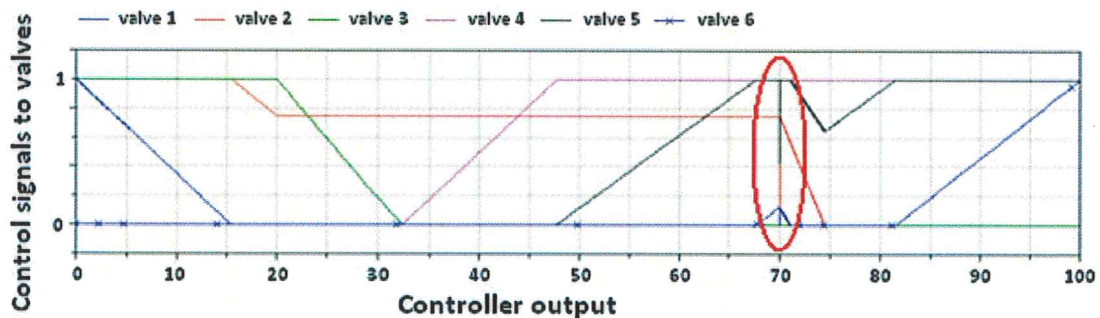


Figure 10. Valve two is put in manual mode after 20s and back in auto after 70s of simulation

#### 4.2.5 Valve characteristics

In SteamPower there valve characteristics can be included by using a handle which sets the maximum opening and closing time and a maximum acceleration rate. When trying to control a valve with the split-range function, it turned out that these valve characteristic handles could not be used. They use differentiation applied to the control signal in order to set the acceleration rate and for this it requires a differentiable signal. Since the split-range function uses different equations for the different cases, the requirement of differentiability is not met.

A solution to this was to apply the restrictions to the input of the split-range block instead by including it in the function block. This was possible since the output from a PI/PID controller always is differentiable. However, the same restrictions have to be applied to all the valves controlled by the same split-range function.

## 5 Simulation Model

When both the new turbine model and the split-range function block had been created, it was time to evaluate the components in an extensive steam net model including the turbine as well as the split-range function. The purpose of this was to find out advantages and disadvantages of the new models.

### 5.1 Steam Net

An extensive steam net was built according to Figure 11.

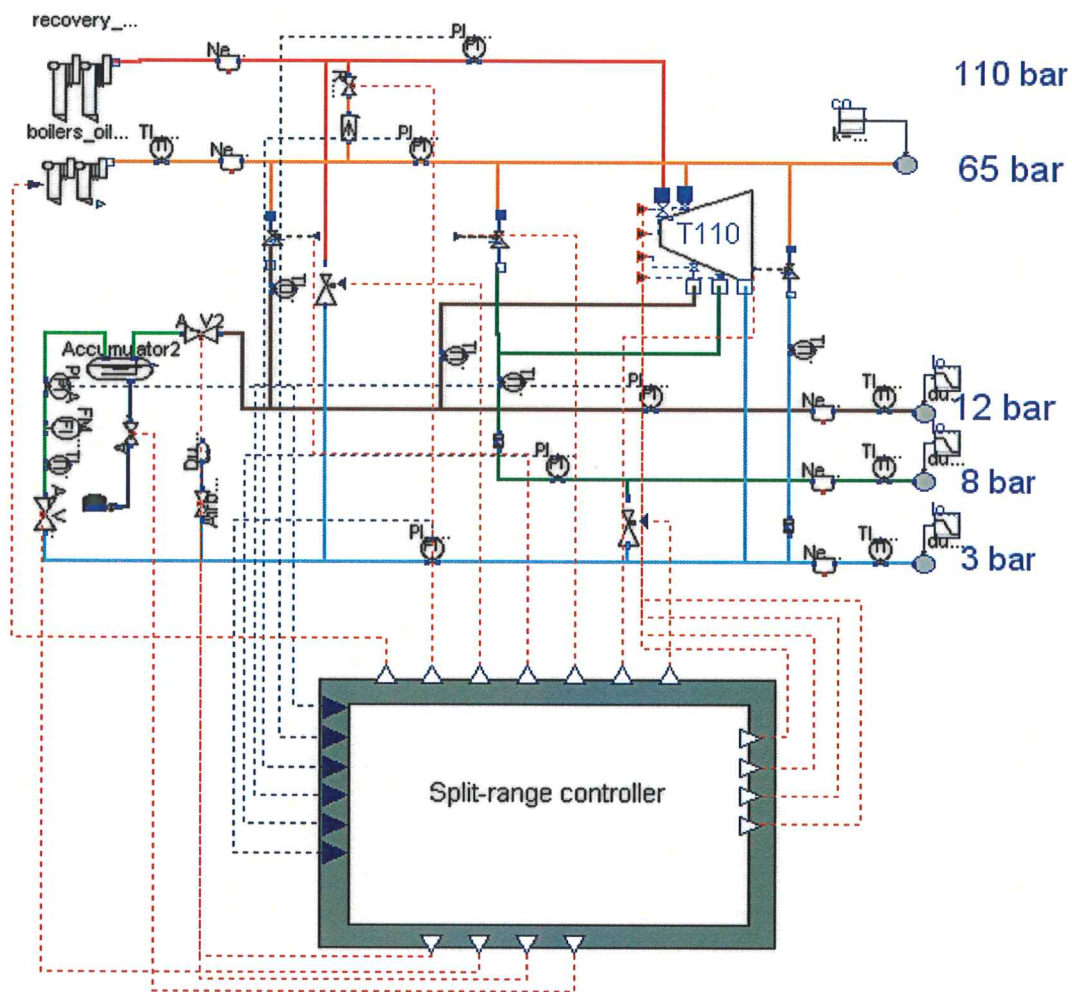


Figure 11. Steam net with controller

The procedure was to create one net at a time, simulate and gradually expand the model with new components and pressure levels. The first net built was the high pressure net at the top of Figure 11 (the 110 bar net). It contains a recovery boiler supplying the system with steam, but no steam consumer at this pressure level. The net was connected to a steam turbine made out of four turbine models connected with tanks in between to include some dynamics and avoid the numerical problem described in Section 3.2 (see Figure 12).

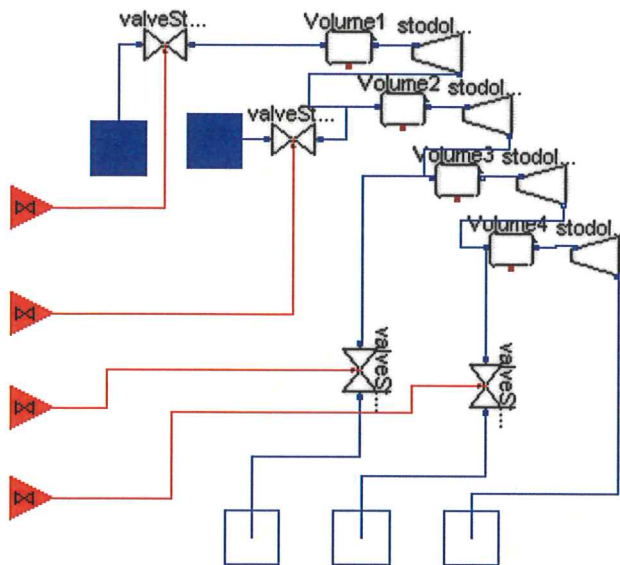


Figure 12. A steam turbine in four steps with injections and bleeds

The next high pressure net, at 65 bar, also contains a steam producer, an oil and bark boiler. There is a steam consumer at this net too but the production exceeds the consumption and hence there is a net flow from the steam net to the turbine and this excess steam is injected to the turbine at the current pressure level. This net is also connected to all the other nets through reduction valves.

Next steam net is at 12 bar and it is connected to some consumers, an accumulator and a bleed from the turbine. The 8 bar net is only connected to the turbine, consumers and two other nets. The back pressure connection from the turbine ends up at the low pressure net at 3 bar and this net is also connected to the accumulator, consumers and reduction valves from some other nets. It is also connected to the atmosphere through a relief valve taking care of excess steam.

Each net also includes a tank to describe the volume of the pipes etc. and they are equipped with pressure indicators. The signals from these indicators are used as input signals to the controller unit.

## 5.2 Split-Range Controller Unit

Each time a new unit was connected, a simulation was performed in order to keep stability and realistic flows and pressures. All valves in the turbine and the dump valve from the LP-net were connected to one PI controller each while building the net in order to keep the pressures constant. All the other valves were closed by default since the turbine should be able to reduce all steam and the reduction valves only should be opened in extreme cases.

### 5.2.1 Split-Range settings

Required for the split-range blocks developed in Chapter 4 are the sizes and priority order of the valves. The turbine and all the adjacent valves were dimensioned to take care of all steam produced in the boilers and supply all the consumers with their amount of steam at a nominal flow. The reduction valves were dimensioned in order to be able to reduce all steam produced at the high pressure nets to lower pressure nets without passing through the turbine in case of an error.

The priority order of the valves was determined in order to lead as much steam as possible through the turbine. Hence all the turbine valves were given the highest priority. At the 12 and 3 bar net the second choice was to use the accumulator, hence these valves were given priority order two. The priority of all the reduction valves were given in order to use as small pressure drop as possible primarily to save as much energy as possible. For example, the reduction from 65 bar to 12 was given higher priority than the reduction from 65 to 8 and 65 to 3 bar.

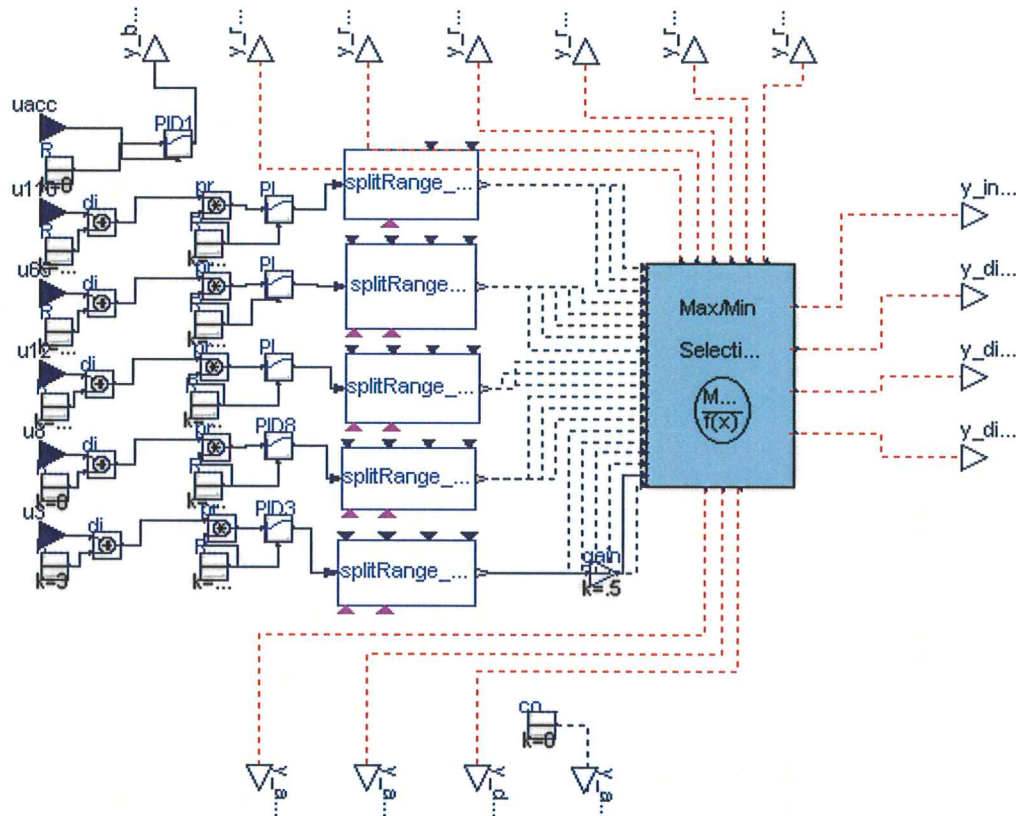


Figure 13. Controller with split-range functions

### 5.2.2 Max selections

One split-range function was required for each pressure level. All valves associated to a steam net were to be controlled by only one controller and one split-range function. This means that some valves can be controlled by several controllers. To solve this conflict some kind of supervisory controller was needed. A simple solution was to choose the signal wanting to open the valve the most, using a maximum function. This method was sufficient since, if a controller wants to open one of these valves, something must have went wrong at that net and it is more important to reduce or decrease its pressure than it is to keep the valve closed for energy optimizing purposes. If the valve is opened by one net, the controller associated with the other net takes care of the consequences at that net, by opening or closing other valves.

In the fictive steam net built in this thesis all six reduction valves needed this type of max selector since they were all connected to two split-range functions. A seventh max selector was implemented to control the input to the accumulator further described in Section 5.2.3.

### 5.2.3 Accumulator control

When the first simulation was performed, there was a problem with the accumulator. The 3 bar net was emptying the accumulator but it was never filled up again from the 12 bar net. The priority order had to be changed in order to solve this problem. If we let the accumulator inlet from the 12 bar net also be controlled by the controller and split-range function at the 3 bar net the problem could be avoided. If putting in the valve to the accumulator in the split-range function adjacent to the 3 bar net, the gain from the controller output to the pressure at the 3 bar steam net will not be constant anymore since the change in accumulator pressure will not affect the 3 bar net directly. The solution to this was to let the valve be working in parallel with the relief valve, i.e. letting them be controlled by the same split-range output.

### 5.2.4 Boiler control

The last thing that was implemented was the control of the boiler. If the steam production is not controlled there has to be an overproduction of steam in order to maintain constant pressures at all nets. Then there also has to be a dump of steam from the low pressure net. If there is no overproduction, the accumulator will first empty and then the pressures will start to fall to fulfil the steam mass balances. The pressure in the accumulator is an indication of how much steam there is in the system and can thus be used to control the fuel to the boiler with a slow controller. With this method there was no need to dump any steam from the low pressure net even if the loads were varying a lot. In order to include a minimum load at the boiler, the controller output was restricted to only the upper part of its working range.

## 5.3 PI tuning with the lambda method

Determination of the control parameters to be used in the PI-controller was made with the lambda method. This method was chosen since it is a common method to use for controller tuning in the industry and it is likely to be used together with this split-range function when it is implemented. The output values from the controllers connected directly to the turbine valves after steady state had been reached, were converted to input signals to the split-range and used there as constants. A diverging system was then achieved but after a while steady state was reached even here. At that time a step signal was added to the constants, one at a time, and the step response at each pressure was evaluated and used to calculate the controller parameters using the lambda method. Figure 14 shows the diverging pressure during the transient time period after start-up at a steam net without the controller and the stable pressure when the controller is used.

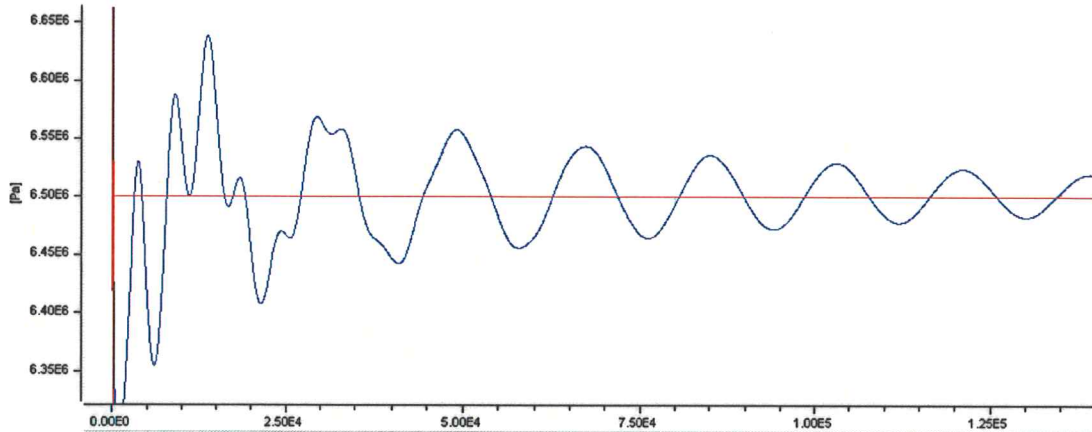


Figure 14 Pressure at the 65-bar net with (red line) and without (blue curve) controller.

The lambda method uses a step response from control signal to process output  $P_v$  to determine the optimal control parameters. The method is described in (INTAB, 2009) and can be summarized for a non-integrating process as follows. The time delay,  $L$ , the process time constant,  $T$  and the process gain is needed. The change in controller signal  $\Delta u$  and the change in the responding process value  $\Delta P_v$  are used to calculate the process gain  $K_p$  according to equation (4).

$$K_p = \frac{\Delta P_v}{\Delta u} \quad (4)$$

The process time constant is achieved as the time for the process to reach 63% of its final value after a step response 0. A step response with the time constant marked can be seen in Figure 15.

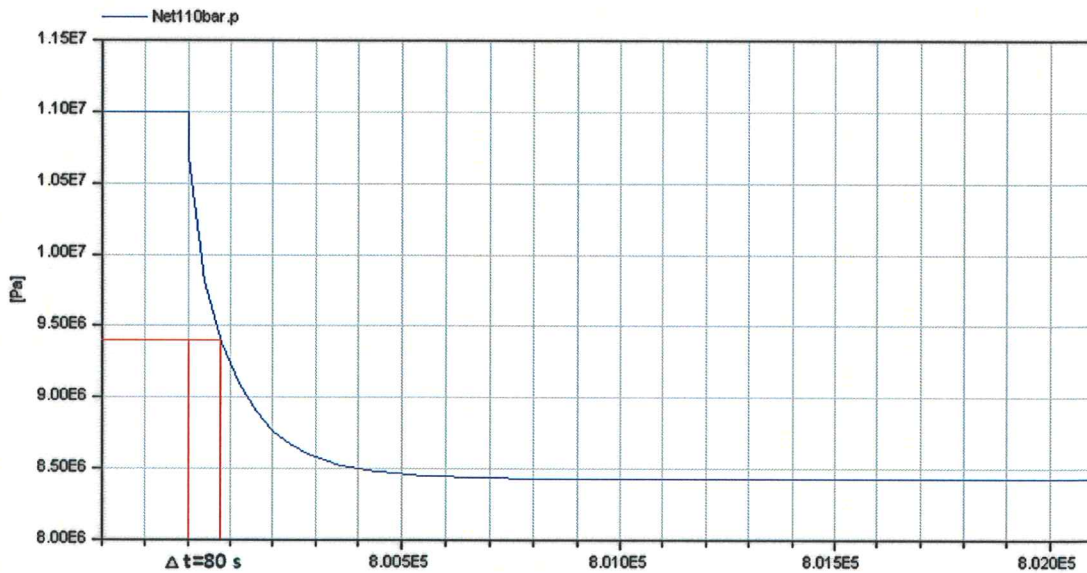


Figure 15 Pressure at a step response at the 110-bar net

As seen in Figure 15, the time constant is approximately 80 seconds. The change in controller output  $\Delta u$  is 10% and the change in the responding process value  $\Delta P_v$  can be calculated from the figure to -23,3%. This gives the process gain,  $K_p=2.33$ .

Parameters needed for the controller are the controller gain,  $K_c$  and the time constant for the integration,  $T_i$ . In the lambda method they are calculated with the following formulas:

$$T_i = T \quad (5)$$

$$K_c = \frac{T_i}{K_p(\lambda + L)} \quad (6)$$

$\lambda$  is the desired time constant for the controlled closed loop system and it is often determined to somewhere in the range  $\lambda \in [T \ 3T]$ , depending on how fast control is desired. In this case  $\lambda=2T$  was chosen which gave a satisfying result. The calculated parameters were modified a bit in order to further improve the stability and accuracy of the system.

## 6 Results

When all controllers, split-range functions and max-selectors had been installed in the steam net model, several tests were made in order to see if the split-range function behaved as expected. In Figure 16 the discharge valve from the turbine to the 8-bar net is locked by putting it in manual mode when steady state is almost reached after the transient initialization period. After that the load is decreased by a step, to simulate a shut down of a consumer. From the figure it can both be seen that the reduction to the 8-bar net from the 65-bar net becomes active when the discharge valve is fully open (green curve) and that the reduction to the 3-bar net takes care of the excess steam at the net when the load is decreased.

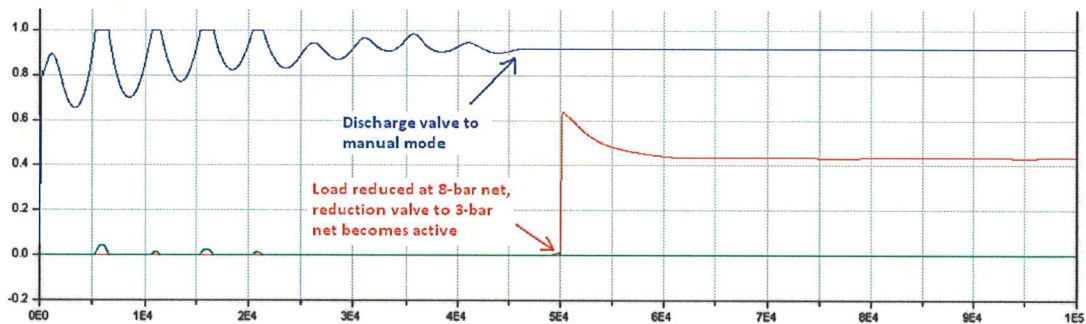


Figure 16 Valve positions for the 8-bar discharge valve (blue curve), the reduction from the 65-bar net (green curve) and the reduction to the 3-bar net (red curve).

The next figure, Figure 17, describes the same simulation as Figure 16 but the discharge valve is put back in auto after further 15000 seconds of simulation.

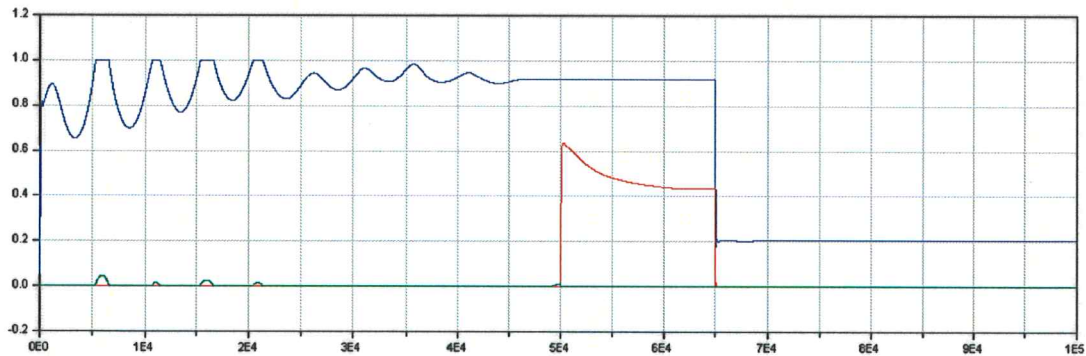


Figure 17. Same as Figure 14 but the discharge valve is put back in auto after further 15000s of simulation.

As can be seen in the figure the discharge valve becomes active again and the reduction valve closes at the same time. The simulation is totally 100000 seconds and hence the ramping looks like it is made immediately. This is not true and a close-up of the transition can be seen in Figure 18.

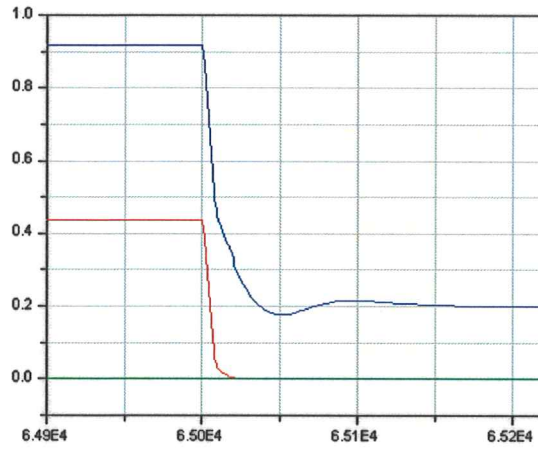


Figure 18. Closeup of hand/auto transition from figure 15.

Several other test simulations were also performed and the conclusion could be drawn that the split-range function works properly for control of steam nets and the requirements specified in Section 4.1 are all met.

## 7 Discussion and Further Work

Extra attention have to be taken to the max selectors in this project, where two controllers can be used to control one valve. If one valve is active in both split-range functions connected to the valve, problems may arise if one controller wants the valve to be open and the other wants it to be closed. One solution to this problem could be that the split-range function wanting the valve to be closed considers it as in hand mode since its signal cannot affect the valve position. However, this kind of function cannot be implemented directly in the split-range function. If the valve is opened by one controller and then removed from the range in the other, and the split-range with reduced range changes its signal and wants to open the valve, it is impossible since it is removed. Another way of dealing with this issue is to make the lower limitation of the valve in one split-range function follow the output signal from the other split-range function. If this tracking function is implemented in both split-range functions connected to the valve, theoretically there would be no problems arising from the max selectors. However, the function has not yet been implemented in the split-range function. This has to be done before the program can be implemented in a real plant or simulator.

When implementing these split-range function in a real plant, or a operator simulator, one has to remember that if max selectors are used for a valve and this valve is put in hand or auto mode, the transiton signal has to be connected to all split-range functions controlling that valve.

The split-range function created in this work is not specifically made for control of valves even if it is what it has been tested for. It could also be used to control several other kind of static actuators.

## 8 Conclusions

A steam turbine model for dynamic modelling and simulation has been improved in order to become less computationally demanding and to give more accurate results. Opportunity to consider choked flow has been included in the model but the complicated internal construction makes it impossible to generalize. The model has to be adapted to data for a specific turbine when being implemented in a steam net model.

A split-range device for control of steam nets has also been developed. It takes into account several requirements such as auto/hand mode transitions of the actuators and actuator limitations.

A steam net model was built in which both the turbine model and the split-range function were tested and verified. The modelling was performed in Dymola, a graphical user interface for Modelica. In this model boiler control was introduced as well, in order to match steam production and consumption. All controllers used were PI controllers tuned by the Lambda method, since those controllers are likely to be used in a real steam net.

In the test phase, consumer loads were decreased step wise and transitions between hand and auto modes were tested. The split-range function was able to function properly throughout all tests.

In order to use the same valves as controllable objects from more than one split-range function, max selectors were used. During the simulations this did not cause any problems. However theoretically they could cause windup phenomena or gaps in the split-range control. These issues can be avoided by using the valve minimum opening to follow the output from the max function. This tracking has not yet been implemented in the split-range function.

The final results from this work are programs to be used for dynamic simulation in steam nets and the Split-Range function, which can, beside valves, also be used to control other approximately linear actuators.

## 9 References

- Asok R, (1980). "Dynamic modelling of power plant turbines for controller design"  
*Applied Mathematical Modelling*, Volume 4, Issue 2, Pages 109-112
- Bolek, W. & Sasiadek, J. & Wisniewski, T. (2001). "Two-valve control of a large steam turbine", *Control Engineering Practice*, Volume 10, Issue 4, Pages 365-377
- Chaibakhsh, A & Ghaffari, A. (2008). "Steam Turbine Model", *Simulation Modelling Practice and Theory*, Volume 16, Issue 9, Pages 1145-1162
- Anglart, H., Andersson, S. & Jadry, R. (1992). "BWR steam line and turbine model with multiple piping capability", *Nuclear Engineering and Design*, vol. 137, Pages. 1-10
- Stodola A, Loewenstein L, (1927). *Steam and gas Turbines, with a supplement on the prospects of the thermal prime mover*, Vol.1, McGraw-Hill Book Company, Inc.,
- Forsman, K. (2005). *Reglerteknik för processindustrin*, Lund: Studentlitteratur AB
- Malmheden, M., Dynasim AB support, personal communication 090121-090901
- INTAB Interface Teknik AB, (2009) "LambdaTune"
- Glad T, Ljung L (2006). *Reglerteknik: Grundläggande teori*, Lund: Studentlitteratur AB
- Björk, G., (2005). *Dynamisk modellering av energisystem*. Institutionen för Energi och miljö, Värmeteknik och maskinlära. Examensarbete 4-2005
- Lo, K.L. Zeng, P.L. Marchand, E. Pinkerton, A. (2002), "Modelling and state estimation of power plant steam turbines", *Generation, Transmission and Distribution*, Volume: 137 Issue 2, Pages 80-94