Technische Universität Berlin
Fakultät III – Prozesswissenschaften
Institut für Prozess- und Verfahrenstechnik
Fachgebiet Regelungstechnik

Master Thesis

System Identification of a Solar Tower Power Plant for Model Based Control

Christoph Reichenbach
Energie- und Verfahrenstechnik
Matrikel-Nr. 372946

Berlin, 12.05.2023

Supervised by Prof. Dr. S. Knorn (Fachgebiet Regelungstechnik),
Terrance Wilms (Fachgebiet Regelungstechnik)
as well as Kevin Iding (Deutsches Zentrum für Luft- und Raumfahrt)

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig und eigenhändig sowie ohne unerlaubte fremde Hilfe und ausschließlich unter Verwendung der aufgeführten Quellen und Hilfsmittel angefertigt habe.

Berlin, den 12.05.2023

Ort, Datum

Christoph Reichenbach

Abstract

In this study, thermal models of solar tower power plant components for the purpose of model-based control are set up and fit to operational data of the solar tower in Jülich, Germany. Solar tower power plants provide renewable energy by concentrating solar radiance and converting the heat into electrical power in a power block. The technology inherently enables the use of thermal storages allowing the decoupling of power production and solar radiance. The volatile nature of the primary energy source, the sun, and the use of a thermal storage set challenges for the control of these systems compared to conventional power plants. Model-based control methods like model predictive control are promising for these power plants. For that a comprehensive model of the plant is needed. In future studies the performance of model based control is to be compared for physic's based models and for data driven models based on neural networks.

This study aims to provide physics-based models of the power plant's components for a modular design of a solar tower power plant. Model reduction efforts are conducted in order to keep computational expenses down for optimization processes and to mitigate numerical issues when solving the system of equations. The models are fit to operational data by adjusting a chosen set of parameters. Moving horizon estimation and sequential quadratic least square programming are used to identify these sets of parameters. The systems of equations describing the dynamic behavior of the components is simulated using the *do-mpc* framework based on *Casadi*. For the automatic differentiation performed in *Casadi*, differentiable functions describing the thermodynamic properties of the fluids are needed. The thermodynamic properties have been approximated using polynomial functions. Their accuracy is evaluated for different degrees and domains.

The results show that the methods for parameter identification are suitable and the simulations are able to reproduce the measured operational data. Sets of parameters for each simulated component are found and their accuracy is evaluated. The results show that the use of polynomials for steam property approximation as implemented in this study is very restricted for the simulation and optimization of large scale systems of differential algebraic equations. A comprehensive knowledge of the process and the accuracy of the polynomials in various domains is needed for their application. Simplifications made for model reduction efforts lower the model's accuracy but enhance their computational performance. The effects of simplifications are analyzed and evaluated.

Zusammenfassung

In dieser Arbeit werden thermische Modelle von Komponenten eines Solarturmkraftwerks zum Zweck der modellbasierte Regelung aufgestellt und an Betriebsdaten des Solarturmkraftwerks in Jülich angepasst. Solarturmkraftwerke liefern erneuerbaren Strom, indem sie Sonneneinstrahlung konzentrieren und die Wärme mittels eines Wärmekraftprozesses in elektrische Energie umwandeln. Konzentrierende Solarkrafttechnologien ermöglichen die Nutzung thermischer Speicher, was eine zeitliche Entkopplung der Energieerzeugung und der Sonneneinstrahlung ermöglicht. Die nur bednigt vorhersagebare Menge an Primärenergie, namentlich die Sonneneinstrahlung, und die Nutzung eines thermischen Speichers stellen Herausforderungen für die Regelung und Automatisierung von Solarturmkraftwerken im Vergleich zu konventionellen Kraftwerken dar. Modellbasierte Regelungsmethoden wie die modellprädiktive Regelung zeigen sich hier vielversprechend. Für die Verwendung modelbasierter Regelungsmethoden ist ein umfassendes Modell des Kraftwerks erforderlich. In zukünftigen Studien soll die Leistung modellbasierter Regelungen für Solarturmkraftwerke basierend auf physikalischen Modellen mit der von datengetriebene Modelle auf Basis von neuronalen Netzwerke verglichen werden.

Ziel dieser Arbeit ist die Erzeugung von physikalischen Modellen von Komponenten für ein modulares Design von Solarturmkraftwerken. Reduktionsmaßnahmen werden durchgeführt, um Rechenaufwand für Optimierungsprozesse zu senken und numerische Probleme beim Lösen des Gleichungssystems zu vermeiden. Anhand von Betriebsdaten werden die Modelle mittels Justierung ausgewählter Parameter an das dynamische Verhalten der realen Anlage angepasst. Zustandsschätzung mit bewegtem Horizont und sequentielles quadratisches Programmieren werden verwendet, um diese Parameter zu identifizieren. Das Gleichungssystem, welches das dynamische Verhalten der Komponenten beschreibt, wird mithilfe des *do-mpc*-Frameworks gelöst, welches auf *Casadi* basiert. *Casadi* verwendet automatische Differenzierung und macht damit differenzierbare Funktionen erforderlich, die die thermodynamischen Eigenschaften der Fluide beschreiben. Die thermodynamischen Eigenschaften von Wasser und Dampf werden mithilfe von Polynomen approximiert. Ihre Genauigkeit wird für verschiedene Polynomgrade und Gültigkeitsbereiche bewertet.

Die Ergebnisse zeigen, dass die Methoden zur Parameteridentifikation geeignet sind und die Simulationen die gemessenen Betriebsdaten wiedergeben. Für jede simulierte Komponente wurden Parameter gefunden und deren Genauigkeit wurde bewertet. Die Ergebnisse zeigen, dass die Verwendung von Polynomen zur Dampfeigenschaften-Approximation, wie in dieser Studie implementiert, sehr begrenzt ist für die Simulation und Optimierung großer Systeme von Differential-Algebraischen-Gleichungen. Ein umfassendes Verständnis des Prozesses und Kenntniss über die die Genauigkeit der Polynome in verschiedenen Domänen ist erforderlich für deren Anwendung. Vereinfachungen, die für Reduktionsmaßnahmen vorgenommen werden, verringern die Genauigkeit des Modells, verbessern jedoch die Rechenleistung und numerische Stabilität des Modells im Bezug auf Eingangsgrößen und die Parameterwahl. Die Auswirkungen der Vereinfachungen werden analysiert und bewertet.

Contents

1 Introduction

Resource scarcity, environmental pollution and political dependency set challenges for the ever-growing demand of electricity in the coming decades. Renewable energies provide opportunities for a reliable, clean supply. Concentrating solar power (CSP) production serves a unique role by storing heat over a time span and thereby decoupling solar irradiation and electricity production. While large scale batteries storing electricity from wind and photovoltaic remain unsuited and too expensive, solar thermal power inherently contains the possibility to store heat in isolated thermal storages and use it for energy production when needed.

The most prominent CSP technologies are parabolic throughs, solar tower power plants (STPP), linear Fresnel lenses and dish receivers. Most relevant for large scale use are the first two technologies. Depending on desired working temperatures, a technology can be chosen. The highest concentration factors within the different CSP technologies are realized in STPP, meaning the ratio between reflecting area of mirrors and the receivers surface is the highest. They achieve the highest temperatures leading to high efficiencies in power conversion and storage.

While providing that utility, STPP are systems of higher systemic complexity than e.g. a photovoltaic panel and raise challenges in operation, optimization and control. Important differences between conventional and STPP are the volatility in energy intake as well as a degree of uncertainty in parameters and input values. The amount of energy brought into the system of, e.g., a coal plant is known through the caloric value and mass of supplied coal. The position of the sun, bypassing clouds or state of the mirrors alter the energy intake of a solar plant. Flux of solar rays at the receiver or its surface temperature are difficult to measure accurately and show relevant uncertainties. In plants around the world and specifically at the Solar Tower Jülich (STJ), a crew of operational staff is necessary to steer the process during operation and adapt to environmental changes. With STPP being commonly located in desert regions, staff is a critical factor in production costs and an automation is desirable also in terms of efficiency.

Robust control methods need to be applied to ensure compliance with material limits and operation close to the optimal operational point. Model predictive control (MPC) is a modern control method for multi variable systems using dynamic models to predict future behaviors depending on input variables. It allows the manipulation of a controlled variable to obtain optimal output variables with respect to a given objective function, while boundaries for input and output variables are met. The method of model predictive control is ressource demanding in terms of computational power. Accurate yet efficient models in terms of computational expense are needed.

In recent studies, MPC concepts for STPP have been conducted and appear to be viable in the results. Prior authors however did not take into account the high uncertainty of variables in the power plant system such as flux density and temperature on the receiver surface being primarily based on simulations. In addition, an implementation of said concepts was not performed. A comprehensive model of the STPP is needed to investigate effects of uncertainty in parameters on predictive control methods. This study aims to provide modeled components suitable for predictive control. They can be the basis of a comprehensive model of a power plant by connecting the respective inlet and outlet streams of the components. The modular approach allows variations in the power plant configuration. A model of the components of a solar tower power plant located in Jülich is implemented as an application example. The physical mechanics governing the components behavior are simplified and in part replaced with empirical correlations. The systems of equations for the components is set up, and relevant parameters are chosen to fit the model to operational data using the *DoMPC-Toolbox* based on *Casadi* for *Python*.

1.1 Structure of the Work

This work aims to provide physics based models of solar tower power plant components suitable for optimization and MPC. In chapter 2, the fundamental principles of STPP and their simulation are explained. Relevant mathematical and thermodynamic concepts are introduced. Software and methods for model optimization and MPC, implemented in this work, are described. An introduction into class-oriented programming, used in this work to implement the model, is given. In chapter 3, the governing physical effects in the power plant components and their describing equations are laid out. This work focuses on the systems for energy storage and conversion, largely excluding the heliostat field. The differentiable approximation of water, steam and air properties is of particular interest as a trade-off between accuracy, calculation power and numerical stability in the simulation has to be made. The methods for approximation and assessing their accuracy are described in section 3.1.1. In chapter 4 the simulation results are presented validated against plant data obtained from the solar tower power plant STJ. Their accuracy is evaluated. The adjusted parameters are named and for the storage the quality of estimation is assessed.

2 Theoretical Basis

In this section, theoretical fundamentals and principles are introduced. Mathematical and thermodynamic fundamentals relevant for this work are listed and explained as well as the basic functionality of solar tower power plants (STPP) and their key components. Information on power plant simulation and important methods is given. The basics of model-based control as well as simulation frameworks to perform it are explained. Finally, the structure of the code and the concept of object-oriented programming are specified.

2.1 Mathematical Fundamentals

In this section, two important forms of equation systems, the ordinary differential equations (ODE) and the differential algebraic equations (DAE), are explained. The fundamental terminology relevant when dealing with ODEs and DAEs as well as forms of their solution are presented.

2.1.1 Ordinary Differential Equations

An ODE is a type of differential equation that involves one or more functions of one independent variable and its derivatives. ODEs describe the relationship between a dependent variable x and an independent variable t. They are used to model many different phenomena, such as the motion of a particle in time, the flow of a fluid or the changes in a chemical reaction. By solving an ODE, we can gain insight into the behavior of the system being modeled. The derivative of the order n of the state variable $x^{(n)}$ can be expressed by a function dependent on the independent variable t and the derivatives of x smaller than n.[1][2]

$$\mathbf{x}^{(n)} = f(\mathbf{x}, \dot{\mathbf{x}}(t), \ddot{\mathbf{x}}(t), ..., \mathbf{x}^{(n-1)}, t) \tag{2.1}$$

The bold notation $\mathbf{x}$ refers to the vector of dependent variables $\mathbf{x}(t) = (x_1(t), ..., x_n(t))$, while f refers to functions of the system in their explicit form. Systems of ODEs are relevant in power plants as they are used to model the dynamic interactions between and inside different components. In STPP, ODEs can be used to model the storage devices, the flow of the coolant or the operation of the turbines.

ODEs can be solved using analytical or numerical methods. Analytical methods involve finding closed-form solutions to the equations, and can be used to solve a group of ODEs including linear first-order equations, linear second-order equations, separable equations and homogeneous equations. Some higher order ODEs may be reduced to first order equations. To solve other ODEs they must be solved numerically. Numerical methods involve approximating the solutions to the equations using iterative algorithms. These methods include the Runge-Kutta method, the finite difference method, and the finite element method.[1][2] Each of

these methods has advantages and disadvantages, and the choice of method depends on the particular ODE being solved.

2.1.2 Differential Algebraic Equations

Systems of equations that include ODEs and algebraic equations which describe the state variables are called DAEs. The algebraic equations within these systems are sometimes referred to as constraints. DAEs are used to describe a range of physical phenomena. Oftentimes, DAEs are used to describe systems in which the variables are interdependent. The equations in these systems may contain derivatives of the unknown variable with respect to the independent variable, making it an ODE or may not include a derivative, making it an algebraic equation. A DAE system is a set of equations of the form

$$F(\mathbf{x}^{(n)}(t), \mathbf{x}^{(n-1)}(t), ..., \mathbf{x}(t), \mathbf{z}(t), t) = 0 \tag{2.2}$$

where F refers to functions of the system in its implicit form, the bold notation $\mathbf{x}$ is a vector of dependent variables $\mathbf{x}(t) = (x_1(t), ..., x_n(t))$ for which derivatives are present, $\mathbf{z}$ is a vector of dependent variables $\mathbf{z}(t) = (z_1(t), ..., z_n(t))$ for which no derivatives are present and t is the independent variable. In this work, DAEs often appear in their semi-explicit form with derivatives of the first order. They can be written as follows.

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{z}(t), t) \tag{2.3}$$

$$0 = g(\mathbf{x}(t), \mathbf{z}(t), t) \tag{2.4}$$

A useful concept to describe complexity of a DAE is the index of a system. The index of a differential algebraic equation is an integer that indicates the degree of implicitness of the equation. With the index DAEs can be classified and numerical methods for solution can be chosen accordingly. The number of derivatives of the unknowns in the system determines the index. For example, if a DAE contains two derivatives of the unknowns, then its index is two. The greater the index of a DAE, the more difficult it is to solve numerically.

Linear DAEs with an index of one and and linear DAEs with an index of two can be solved analytically. Additionally, some higher-order DAEs may be reduced to first-order equations by reformulation. Especially, when they are non linear an analytical solution might not be available. DAEs can be non-linear if the equation contains non-linear terms. To solve these DAEs, numerical methods such as the Runge-Kutta method, finite element method, and the finite difference method can be applied.[3][2] These methods are applied to approximate the solution of the DAE system and to iteratively improve the accuracy. Additionally, there are specific algorithms for solving DAEs, such as index reduction and symbolic manipulation.[2] Index reduction will not be a matter of interest in this study as derivatives of the second order have been avoided within the DAE subsystems. They can however arise through interconnection of subsystems.[4] Symbolic manipulation is a technique used to solve DAEs by transforming them into different forms. This can be done by introducing new variables, substituting certain terms, or rearranging the equations. Symbolic manipulation can be used to reduce the number of implicit terms in a DAE, or to eliminate the need for certain numerical

algorithms. Symbolic manipulation can therefore be used to reduce the complexity of the equations and simplify the numerical solution.[5][6]

Consistency of initial values in DAE systems requires that the initial conditions of a system must be coherent with the equations of the system. This means that all of the initial values of algebraic variables in the system must satisfy the equations of the DAE.[7] Particularly, it means that for the set of starting values the following is true:

$$F(\dot{\mathbf{x}}(t_0), \mathbf{x}(t_0), \mathbf{z}_0, t_0) = 0 \tag{2.5}$$

Consistency of initial values is crucial for the accuracy of the solution as well as for the ability of a number of numerical methods to converge.[7][8][9] Modern softwares for the solutions of DAE systems include specialized algorithms in order to determine consistent initial values for algebraic variables based on starting values provided by the user. Several authors propose methods to make the algorithms more robust to inaccurate starting values of algebraic variables. Biegler proposes the integration of Newton algorithm with the Armijo line search [9]. An additional solution to providing adequate starting values is to perform several steps of algorithms more robust to unprecise starting values of algebraic variables. Examples for more robust algorithms are the sequential-modular approach or the moving horizon estimation (MHE).[10] The process chosen in this study for initializing algebraic variables in DAE is explained in section 2.4.3.

2.2 Thermodynamic Fundamentals

In this section, some fundamental principles and vocabulary in the field of thermodynamics are presented. The definitions used in this work are based on definitions given by Tsatsaronis [11] as they are widely accepted and overlap with definitions of other relevant authors.[12][13]

2.2.1 Thermodynamic Systems

A thermodynamic system in general is the subject of analysis and is separated from its surroundings by a well defined interface. This interface is called system boundary. Everything outside of that boundary is the system's environment. Distinctions for systems can be made based on their permeability for energy, mass and information. Closed system's boundaries are permeable for energy but not for mass. Isolated systems are not permeable for mass nor energy. For power plant applications systems with boundaries permeable for mass and energy are relevant which are called open systems.[11][12][13]

2.2.2 States and Properties

A thermodynamic property is a physical characteristic of a substance which can be used to describe its state. In general, a combination of two properties can be used to unambiguously define its state. From that information, all other properties can be derived.[13] A system's properties are only dependent on the system's state but not on its history. They are usually a result of a physical operation.[11] Examples for these thermodynamic properties of a system in the narrow sense are temperature, enthalpy, pressure and density.[12]

2.2.3 Isentropic Efficiency

Isentropic efficiency η_s is a measure of how efficiently a device or system performs work in comparison to operating isentropically, meaning with constant entropy. It is typically expressed as the ratio of the actual work output of the system to the work output of an ideal isentropic system operating between the same inlet and outlet pressures.[11] The enthalpy difference of the process $h_{\text{in}} - h_{\text{out}}$ is set in relation to the enthalpy difference of the isentropic process $h_{\text{in}} - h_{\text{s,out}}$. The index s refers to the state after an isentropic process.

$$\eta_s = \frac{h_{\text{in}} - h_{\text{out}}}{h_{\text{in}} - h_{\text{s,out}}} \tag{2.6}$$

The higher the isentropic efficiency, the more efficient the system. Isentropic efficiency is useful as it indicates how much energy is lost due to friction or other energy-dissipating activities in the device or process. It can be used to determine the effectiveness of e.g. a compressor, turbine, or pump.

2.3 Solar Tower Power Plants

This section will go into the general operation of STPP and the functionality of their key components. A basic scheme of a STPP can be found in fig. 2.1. The part of the hot air cycle includes the heliostat field, the solar tower and the thermocline heat storage. The part of the steam power cycle includes the super heaters, the evaporator, the economizer, the preheater, the feed water tank, the condenser, the steam turbine system and the pumps. A comprehensive flow diagram of the STJ can be found in fig. 2.3

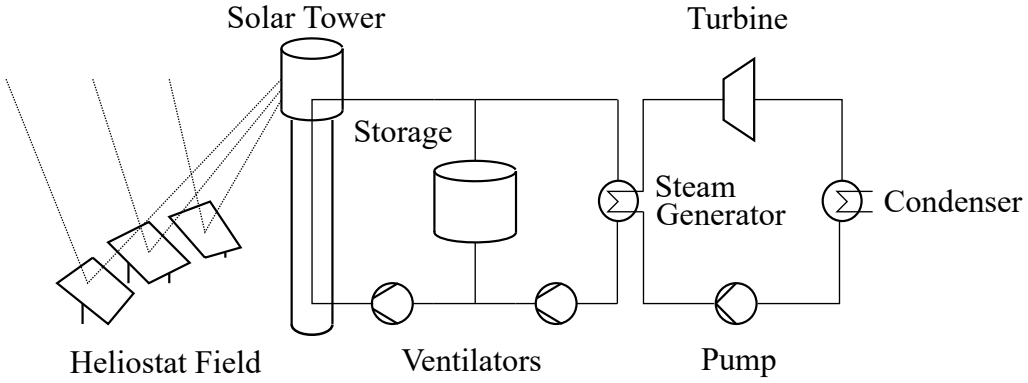


Figure 2.1: Scheme of solar tower power plants consisting of a heliostat field, a solar tower, a thermal storage and a power block.

STPP are a type of concentrating solar power system that use large numbers of movable mirrors, called heliostats, to focus and concentrate sunlight onto a central receiver located on the top of a tower. The concentrated sunlight is used to heat up a heat transfer fluid which flows through the receiver. This heat transfer fluid will then heat up a working fluid inside a power generation cycle, for example steam in a steam cycle. The steam is finally

used to drive a turbine generator to produce electricity. In addition, it is possible to use the generated heat to fill a thermal storage, thereby time-decoupling the electricity production and solar radiance. This inherent feature of concentrating solar power to store heat gives a key advantage towards photovoltaic panels which otherwise would be cheaper and easier to deploy [14]. To store electricity provided by photovoltaic panels, large-scale batteries would be necessary. Batteries however are expensive, often contain rare materials and their life span as well as the number of charging cycles are limited. Solar power towers are an efficient way of generating electricity from the sun, as the concentrated sunlight can reach temperatures up to 1000 °C, which is much higher than the temperatures achievable by other concentrating solar power technologies like parabolic troughs or linear Fresnel reflectors [15]. This is due to the fact that the sunlight is concentrated in a single point in stead of a long pipe like in the other two mentioned technologies. In dish collectors the sun light is also concentrated in a point but the surface area of the single mirror is limited. In STPP the sun light can be reflected from multiple mirrors to the solar receiver. With higher temperatures, the efficiency of the steam cycles rises. The efficiency of the thermal storage rises as well, as more energy can be stored in less material .

2.3.1 Heliostat Field

The heliostat field consists of a large number of mirrors that reflect sunlight to a fixed point on top of the solar tower. The sun is tracked by the heliostats as it moves across the sky, ensuring that the maximum amount of sunlight is always being collected and concentrated onto the receiver. Energy losses caused by overshadowing, the cosine effect or inaccurate reflection can occur depending on the mirrors' positions relative to the solar tower or to each other. To minimize losses, several algorithms for configuration of heliostat fields are available, among others proposed by Eddhibi [16]. To prevent losses through inaccurate reflection, regular calibration of the heliostat field is needed. State-of-the-art method here is the camera target method, most notably implemented in form of the Stein-method [17]. Recent developments point to the use of neural networks in that regard as indicated by Pargmann [18].

2.3.2 Solar Receiver

Solar receivers in STPP are used to capture and absorb the concentrated solar radiation from the sun and convert it into thermal energy. The thermal energy can then be used to power a water/steam circuit for electricity generation, or it can be stored in a thermal storage. Various models of receivers can be implemented depending on information of the heliostat field, the receiver's surface temperature, temperature of the heat transport fluid at the inlet and outlet of the receiver, hydrodynamic properties of the heat transfer fluid and constrictions through mechanical stress. A closer description of a design process can be found among others by Rodriguez-Sanchez [19].

At the STPP studied in this work (STJ) an open volumetric receiver is implemented. Atmospheric air is used as heat transfer fluid in these type of receivers. It consists of a large number of modular ceramic absorbers which are heated up to 1000 °C. Atmospheric air is sucked through these ceramic modules and heated up to almost the absorber's temperature. [20] The

modules, also called absorber cups, have a honeycomb-like structure ensuring high areas of heat transfer and beneficial flow conditions when air is flowing through them. The metal structure upholding the ceramic modules needs cooling in order to maintain its structural integrity. Cool air is being circulated between absorber cups to perform the cooling of the structure. The circulated air is led to the inlet of the receiver where it mixes with ambient air, thereby increasing the inlet temperature of the air flowing into the receiver. This means some of the heat from cooling the steel structure that would be otherwise lost to the environment is being used. The temperature difference between air inlet and outlet temperature at the receiver is smaller with higher inlet temperature thereby increasing the efficiency of the receiver.

Volumetric receivers are a trade-off between two contradicting ambitions of a receiver. The lay out is a trade off between reducing the size for radiation loss to the environment while simultaneously increasing the area available for heat transfer with the air used as heat transport fluid. The volumetric receiver limits the area of loss to the environment largely to the surface. The heat transfer to the heat transport fluid takes place inside the receiver, hence the name 'volumetric'. [21] A comprehensive model of a solar receiver can be found among others in Gall [22] and Schwarzboezl [20].

2.3.3 Thermal Storage

Thermal storages are a key component in STPPs as the capability of storing heat increases the efficiency and reliability of the plant significantly in both annual plant capacity and annual solar-to-electricity efficiency.[23] Through thermal storages, the solar irradiation and the production of electricity can be decoupled in time. The heat supply to the power block can be adjusted according to demand. Unlike battery components, these storages do not typically degrade with loading cycles and are made of readily available materials.[24] Several different types of storages are known with the most common ones being the two-tank direct storage, the two-tank indirect storage and the thermocline storage. In the two-tank direct storage, the hot heat transfer fluid, often a molten salt, is stored in a hot isolated tank and can be extracted when needed. After exiting the heat exchangers of the power block, the cool heat transport fluid is stored in a second, cool tank. From there, it can be lead into the solar receiver once the conditions are suited for reheating. The indirect storage works in a similar way but the heat transfer fluid and the storage media are of different material. The heat transfer is conducted through heat exchangers at each tank where the heat transfer fluid is cooled or heated up respectively.[24]

The technology used at the STJ and looked at closer in this study is the thermocline storage. The functionality of the thermocline storage at the STJ is explained in the following. The thermocline storage consists of a ceramic block with a large number of small air tunnels inside, to facilitate heat transfer between heat transport fluid and storage material. At any given time in operation, one part of the storage will be at a hot temperature T_H, while another part will be at a cold temperature T_C. To heat up the storage, the heat transport fluid, in this case hot air, is led from the top through the ceramic block, thereby heating up the storage material. The heat transfer coefficient here can be assumed as very high and the heat capacity of the air low in comparison to the heat capacity of the storage material. This causes the air to take

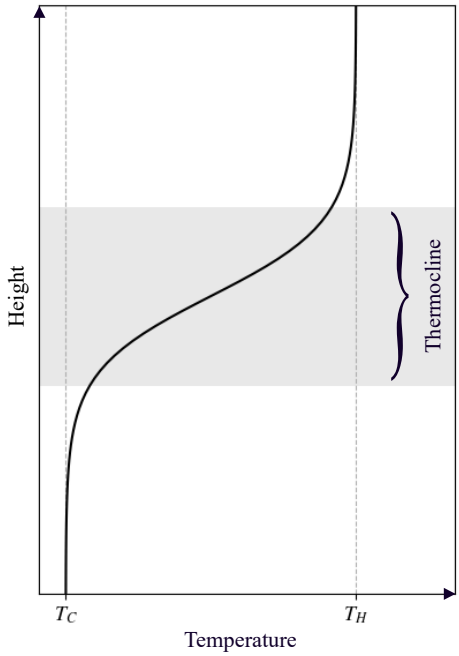


Figure 2.2: Scheme of the temperature curve inside a thermocline storage. Region with temperature gradient (thermocline) marked in gray.

on the storage material's temperature rapidly. The upper layers of the storage heat up to the inlet temperature of the air while the lower layers stay cool. Between these two regions, a temperature gradient, also called thermocline, can be observed, giving this technology its name. A schematic temperature curve inside the thermocline in correlation with its height can be found in fig. 2.2 to illustrate that dynamic. The region with the thermocline is marked in gray. During the operation it has to be avoided that the storage is fully discharged meaning, that the upper, hot side of the storage should not fall below T_H. This ensures the high outlet temperature when discharged. In analogy, the temperature of the cold side at the bottom of the storage should be below T_C to ensure a consistent supply of cold air to the solar receiver.[25]

The operating states, charge or discharge of the storage, are determined by the flow direction of air inside the storage. The direction in turn is a result of the difference of the two mass flows induced by two ventilators. These are namely the receiver ventilator $VREC$ and the steam generator ventilator VSG, see fig. 2.3. The ventilator $VREC$ responsible for the mass flow through the solar tower should be controlled on the basis of information about the incoming sunlight, expected energy intake and outlet air temperature. It should be adjusted in order to keep a uniform outlet air temperature at the solar tower during operation. This control problem can be addressed e.g. with a nonlinear model predictive control (NMPC) or an extended Kalman filter. The ventilator behind the steam generator VSG is responsible for the energy intake of the steam cycle. It should be adjusted on the basis of energy demand, storage filling level and ideal operating conditions of the steam cycle. Is the air mass flow through the

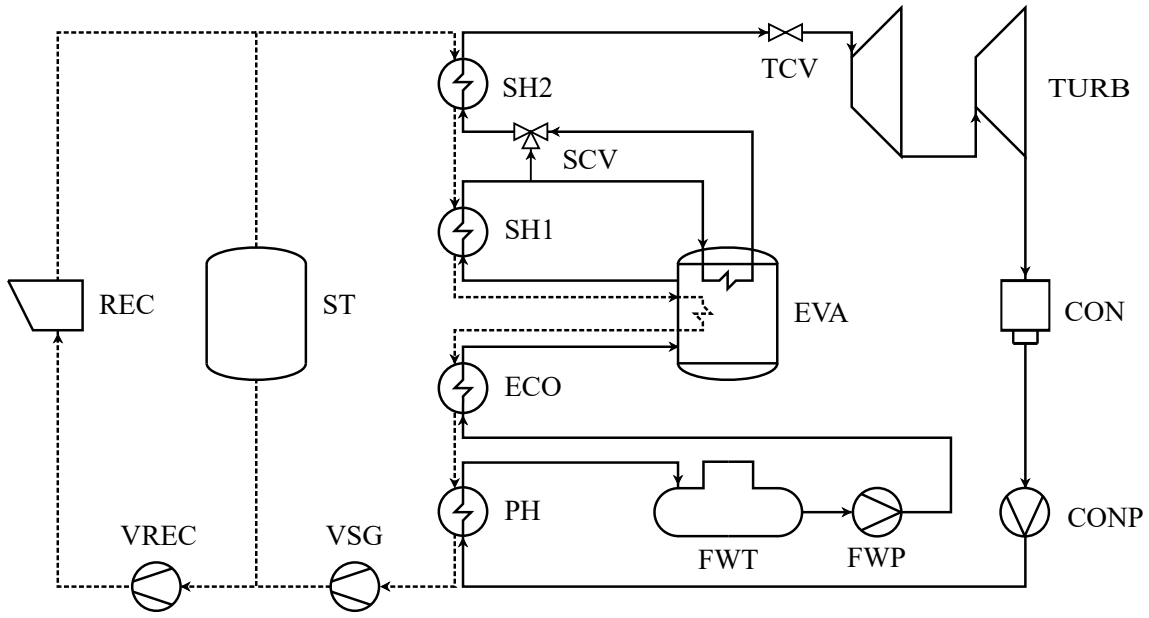
solar tower higher than through the steam generator, the thermal storage is charged. Vice versa, is the mass flow through the steam higher than through the solar tower, the storage is discharged.

2.3.4 Steam Cycle

The steam cycle is a process designed for the conversion of heat into mechanical work. In its idealized thermodynamic form, the Rankine cycle, it consists only of four components, namely of a boiler, a turbine, a condenser and a pump. Heat is supplied to a pressurized working fluid, usually water, which is then vaporized in a boiler, expanded in a turbine, and finally condensed back to its liquid state in a condenser. The working fluid is pressurized in pumps and is afterwards recirculated back to the boiler to complete the cycle. The Rankine cycle is an important model of thermodynamic cycles used in power plants and other systems that convert heat into mechanical work.[12] In more refined versions of the steam cycle, the steam generation is performed in a series of preheater, an economizer, an evaporator and a super heater. Lower grade energy from the cooler air is used to preheat the water before evaporation. The hot air directly from the receiver and the storage is used in the super heaters to achieve the life steam temperature. By using methods of regenerative feed water preheating, the efficiency is increased in modern power plants.[12] These methods of preheating and super heating have been implemented at the STJ. Several turbines instead of a single one can be used to exploit heat more efficiently. Steam turbine versions specialized for certain pressure levels can yield better results. Sequential heating can be implemented through the use of turbine tappings and multiple turbines. [12] This method of sequential heating, however, has not been implemented at the STJ. A schematic plan of the STJ is found in fig. 2.3 and is described closer in the subsequent paragraph.

The steam cycle at the STJ includes a feed water regenerative system consisting of a preheater, an economizer, an evaporator and two super heaters. The preheater is used to extract low-grade heat from the air at lower temperatures, meaning air that has already been cooled down in the process of evaporation. It is used to bring water which is exiting the condenser close to boiling temperature at atmospheric pressure before it is being pumped into the feed water tank. From the feed water tank, the water is brought up to a pressure above life steam condition. Afterwards, a control valve behind the feed water pump regulates mass flow and inlet pressure into the economizer, thereby also regulating the mass flow into the evaporator. The economizer is a heat exchanger that will use the lower grade heat from the air outlet stream of the evaporator to bring the water inlet steam into the evaporator just below evaporation temperature.

There are multiple operation modes for power cycles with the two major branches called floating pressure operation and turbine driven operation. The difference is the method on how the output of mechanical work is controlled. In floating pressure operations, the power output is controlled by the energy intake into the evaporator, in this case the air mass flow. The pressure level inside the evaporator can then be regulated by the water inlet mass flow. From an energetic standpoint, this is the preferred option as the turbine control valve can stay fully opened throughout the operation and no throttling loss is caused here.[26] On the



CON - Condenser	FWT - Feed Water Tank	ST - Thermal Storage
CONP - Condenser Pump	PH - Preheater	TCV - Turbine Control Valve
ECO - Economizer	REC - Receiver	TURB - Turbine Group
EVA - Evaporator	SCV - Steam Cooler Valve	VREC - Ventilator Receiver
FWP - Feed Water Pump	SH - Super Heater	VSG - Ventilator Steam Generator

Figure 2.3: Scheme of the Solar Tower Power Plant in Jülich.

downside, the amount of heat converted into mechanical work can be controlled only slowly using this method. To be able to adapt to changes in demand, a turbine driven operation can be implemented. The turbine control valve is used to determine mass flow and inlet pressure into the steam turbine and simultaneously control evaporator pressure. This enables fast response to load changes, but at the cost of throttling losses. At the STJ, a turbine-driven approach is chosen. The pressure in the evaporator is therefore controlled by the position of the turbine control valve.[26] Advanced approaches are available, one of which including the alteration of the evaporator's energy intake based on the turbine valve's position aiming to bring it back to a fully opened position.

After the pressure increase in the feed water pump, the hot water is heated up in the economizer to the now much higher saturation temperature. Subsequently, the water flows into the evaporator where the phase change takes place. The temperature and filling level inside the evaporator is determined by mass flow and by inlet and outlet temperatures of the air, water and steam. The steam temperature at the outlet of the evaporator is slightly over the saturation temperature. It is heated up close to life steam temperature inside the first super heater, super heater 1. The second super heater, super heater 2, in combination with the steam cooler is used to precisely regulate the life steam temperature, ensuring the compliance with temperature limits and operation close to the optimal operating state. The steam cooler is a heat exchanger between the super heated steam at the outlet of super heater 1 and the content of the evaporator. In fig. 2.3 the steam cooler is not explicitly depicted

as the component is constructed within the evaporator. It is indicated by the heat transfer symbolic on the upper part of the evaporator depiction in fig. 2.3. The mass flow through the steam cooler is controlled by a control valve SCV adjusting the inlet temperature into super heater 2 and hence it's outlet temperature. The life steam reaches a turbine control valve, regulating inlet pressure into the turbine in a final instance and adjusting the mass flow inside the turbine. The turbine group is found in the top right corner of fig. 2.3. The mass flow is determined through a correlation between inlet and outlet pressure and the valve position. The turbine is driven by the life steam, thereby providing mechanical work which powers the generator. The electrical generator then produces electricity which is in part used for the power plant's pumps and ventilators and in part fed into the electrical net. The turbine well and the generator are not depicted in fig. 2.3 and have not been modeled explicitly.

2.3.5 Control Problems

In STPPs, a number of control problems arise as the primary energy source, the sunlight, is only available at certain times and can be disturbed by events like weather changes, passing clouds or defect mirrors. Keeping temperatures and temperature gradients within material limits and simultaneously close to the optimal operating point is the aim of the control system. To mitigate sunlight fluctuations and to make the energy production more economical, a thermal storage is integrated into the system which requires additional control effort. In this section, control problems within the STPP are presented.

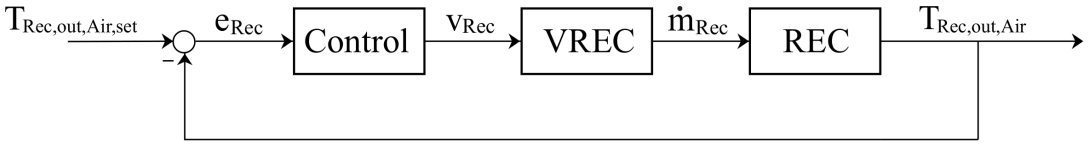


Figure 2.4: Scheme of control loop responsible for following the set air temperature at the outlet of the receiver.

The air mass flow through the solar tower $\dot{m}_{Rec}$ is controlled through the rotation speed v_{Rec} of the receiver ventilator $VREC$. The aim is to keep the air outlet temperature of the solar receiver $T_{Rec,out,Air}$ as high as possible. Doing that, the material's upper temperature limits must be respected while temperature gradients in the solar receiver that are too steep should be avoided. Abrupt changes in temperature may lead to damage in the receiver material. With a simple energy balance it becomes clear, that the amount of air lead through the receiver will impact the air outlet temperature, see eq. (2.7). Considering a constant energy intake at the receiver Q_{Rec}, a larger air mass flow thorough it would lead to a lower outlet temperature. Vice versa, a lower air mass flow would lead to a higher air outlet temperature. Through manipulation of the air mass flow through the receiver the outlet air temperature can be kept a set trajectory or set point even when the energy intake varies.

$$\dot{Q}_{Rec} = const.\ ,\ \dot{m}_{Rec,Air} \nearrow \Longrightarrow\ T_{Rec,out,Air} \searrow \tag{2.7}$$

The controlled variable is altered based on information on the outlet temperature of the receiver $T_{\text{Rec, out, Air}}$, the surface temperature of the receiver and a possible heat flux estimation

at the receiver. In a last consequence, where the alteration of the mass flow is not capable of preventing damage to the material, the defocus of mirrors can be initiated. The flux density at the solar receiver is of importance for operation and control of the STPP. Through a measurement of flux density, the energy intake and its gradient can be determined. At industrial scale, the reflection of the receiver's surface can be used to obtain information on the flux density. Digital cameras capture footage of the irradiated receiver and together with reflection properties of the receiver's surface, the flux density can be calculated.[27] Göhring presents a method of flux measurement using affordable camera equipment [28]. The effects of the absorber material's inhomogeneities on the reflection as well as the direction of incidence is taken into account by the 'Cloud and Gonioreflectometer Method' proposed by Göhring [28]. An estimation of this variable using state estimation techniques is despite that of interest as measurements and analytical deduction of the flux remain elaborate.

The air mass flow into the steam generator is controlled through the rotation speed of the steam generator ventilator. In the steam generator, the air passes its thermal energy successively onto super heater 2, super heater 1, the evaporator, the economizer, and the preheater. The goal is to provide air for the steam generation at a constant, high temperature while simultaneously preventing high fluctuations in the mass flow, using the thermal storage if necessary. The air mass flow through the steam generation controls the use of the thermal storage and is adjusted to provide the right amount of energy needed to follow power demands. When the mass flow through the receiver $\dot{m}_{Rec,Air}$ is higher than the mass flow through the steam generator $\dot{m}_{\mathrm{SG,in,Air}}$, the storage is charged. In analogy, the storage is discharged if $\dot{m}_{Rec,Air}$ is smaller than $\dot{m}_{\mathrm{SG,in,Air}}$. Comparing to fig. 3.1 it can be observed that the upper portion of the storage is at maximum temperature. Charging the storage results in a larger volume inside the storage being at maximum temperature. Consequently the volume with a vertical temperature gradient, the thermocline, is moving farther to the bottom of the storage. The thermal storage is explained in more detail in section 2.3.3.

$$\dot{m}_{Rec,Air} < \dot{m}_{\mathrm{SG,in,Air}} \Longrightarrow \text{Discharge} \tag{2.8}$$

Gradients when charging and discharging the storage have to be below a certain value, and upper and lower limits of the storage's filling level are regarded as constraints as well. Changes of temperature in to short time spans can results in damage to the storage material or the structure of the component. Keeping the top part of the storage at maximum temperature is necessary to reliably provide the steam generator with sufficiently hot air when discharging the storage. Similarly the bottom part is kept to be at minimum temperature in order to ensure a sufficient cooling of the receiver cups while charging the storage. The control of the air mass flow into the steam generator is based on information on the power demand curve, inlet temperature into the steam generator, and the filling level of the thermal storage.

The water mass flow from the feed water tank to the steam generator as well as its pressure is controlled through control valves behind the feed water pump. In the feed water pump, the water is brought up to pressure levels over life steam pressure plus pressure losses during steam generation. The control of pressure and mass flow is then performed in the valves behind the feed water pump. The valve position regulating pressure and mass flow into the

steam generator is based on life steam temperature, life steam pressure and steam cooler valve position. The pressure and temperature limits for the life steam have to be kept. As mentioned in section 2.3.4, the pressure level inside the evaporator is controlled through the turbine control valve leaving the inlet mass flow to mainly ensure the filling level of the evaporator to be kept from running dry or spilling.

The mass flow through the steam cooler $\dot{m}_{SC}$ is controlled through the position of the steam cooler control valve. By adjusting the steam cooler mass flow, the life steam temperature can be regulated more precisely into the steam generator than by controlling the outlet steam mass flow out of the evaporator. The steam cooler is a heat exchanger between the steam at the outlet of super heater 1 and the content of the evaporator. When steam is directed through the steam cooler, it exits the steam cooler at evaporator temperature plus a constant temperature difference passing its thermal energy to the content of the evaporator. By mixing hot and cooled steam in the three way steam cooler control valve, the inlet temperature into super heater 2, $T_{SH2,in,\,H2O}$, and thereby also life steam temperature, $T_{SH2,out,\,H2O}$, can be adjusted. Constraints are set through the upper temperature limit of the turbine equipment and piping. The control of the mass flow through the steam cooler is based on measurements of life steam temperature, lowering the steam cooler mass flow when life steam temperature is too low and vice versa.

$$\dot{m}_{SC} \nearrow \Longrightarrow T_{SH2,in,\,H2O} \searrow,\ T_{SH2,out,\,H2O} \searrow \tag{2.9}$$

The mass flow into the steam turbine group is controlled by the turbine control valve. The position of the valve in combination with inlet and outlet pressure of the turbine determines the mass flow directed through the turbine group. It regulates the pressure inside the evaporator consequently. The control of the mass flow through the turbine is relevant for secure power plant operation. It prevents too high turbine inlet pressures and outlet temperatures and ensures the compliance with filling and pressure limits in the evaporator. The alteration of the controlled variable is performed based on information on the filling level in the evaporator, temperature and pressure at inlet and outlet of the turbine as well as the evaporator pressure.

2.4 Power Plant Simulation

This section addresses the set-up and solution of the system of equations describing the STPP. Relevant types of equations are listed and an overview of solution methods for large sets of equations as they appear in power plant simulations is given. Basic thermodynamic terminology relevant in this regard is explained in section 2.2 and mathematical fundamentals are presented in section 2.1.

The model of a steam cycle power plant consists of a set of equations which describe the thermodynamic behavior of each component and their interaction [10]. Martelli describes in his review on power plant simulation a procedure followed by a number of authors to deduct a system of equations for a power plant model [10]. The procedure involves the set-up of the following equations:

- Thermodynamic properties
- Energy, impulse and mass balances
- Flow sheet topology
- Performance correlations

Thermodynamic properties such as enthalpy, entropy, density or thermal conductivity of a material are determined through property equations at each state. These property equations enable to calculate all properties of the system when the state is known as explained in section 2.2.2. For a wide range of materials, tables for thermodynamic properties or semi-empiric equations closely approximating the material properties are available.[13] The equations for each component are set up according to energy, impulse and mass balance.[11] The equations describing the flow chart topology connect the outlet stream of a component to the inlet stream of the down stream component.[11] In addition, performance correlations such as heat transfer coefficients, valve characteristics or mass flow dependent efficiency of the turbine need to be considered when solving the power plant's model.[10] The aim of the simulation is to find pressure, temperature, enthalpy, entropy and other relevant thermodynamic properties for each state in the plant and determine the energy intake and output for each component. The equations listed for each component are found in chapter 3.

2.4.1 Steady-State Simulation

Steady-state systems are assumed to remain constant over the observed time period and therefore the observed behavior will continue into the future [11]. Steady-state simulations in the context of power plant simulations are an essential tool to understand the process as well as to find opportunities for improvement [29]. The describing algebraic system of equations (AE) has an equal number of variables x_i and equations f_i. In steady-state systems, the equations can be formulated in their vectorial form:

$$f(\mathbf{x}) = 0 \tag{2.10}$$

Models deducted from steam cycles are non-linear as property equations or performance equations like heat transfer coefficients or logarithmic temperature differences are non-linear. The AE describing steam cycles are sparse, as only a part of all variables appear in each equation. They can involve hundreds of variables and equations as power plants involve numerous pipelines and components.[10] In many power plant simulation softwares, either an equation-oriented or a sequential modular approach is used.

Equation-oriented modeling relies on well-known and specifically developed numerical algorithms such as the Newton-Raphson algorithm or the Powell dogleg algorithm [9]. The Newton-Raphson algorithm is a root-finding algorithm which produces successively better solutions to find the roots of a problem. The Powell dogleg algorithm is an iterative optimization algorithm for least squares problems to find a local minimum of a function of several variables. These algorithms have quadratic rates of convergence and can efficiently solve the systems of equations.[30] One downside of these methods on the other hand is the necessity of providing

good initial values for the variables as they may fail to converge when they are too far off the actual solution. Additionally, they require differentiable equations $f(x)$ with non-singular Jacobians and second derivatives for the Newton-Raphson method respectively.[30] That means that the user has to provide reasonably good starting values for all properties for each stream and variables of components. Depending on the complexity of the model and the user's knowledge of the plant, this task may be very time consuming and prone to errors. Inadequate starting values may lead to a failure of convergence within the solution algorithm with very limited options of debugging.[10] The concept of consistency of initial values for algebraic variables in DAE systems is explained in more detail in section 2.1.2. The method used in this study to initialize the systems of equations with consistent values is presented in section 2.4.3.

The sequential modular approach orientates itself on the flow sheet order of components firstly solving a component upstream and using its output values as input for the downstream component [31]. Is an input dependent on a downstream component, the value for that state is estimated and updated iteratively until convergence.[10] The sequential modular method deals with a rising number of converging issues as the number of cycle loops increases and more variables need to be estimated. Power plants consist of at least one of these loops, the steam cycle, but can have an increased amount when multiple pressure levels in a heat recovery system or multiple temperature levels in heat regenerators are involved. Convergence issues may occur in power plant applications due to that reason. These algorithms, however, are more robust to the choice of starting values and debugging can be conducted more precisely as convergence issues may occur on the component level.[10]

2.4.2 Dynamic Simulation

In dynamic models the system of equations can be formulated in its vectorial form as in eq. (2.11).

$$f(\mathbf{x}, \dot{\mathbf{x}}, \mathbf{u}) = 0 \tag{2.11}$$

Where x denotes the time-dependent state variables (pressures, temperatures, mass flow rates, etc. of each stream and unit of the plant), $\dot{x}$ denotes the derivative over time, and u the time dependent control variables [10]. The bold notation here refers to vectors. The equations are build in the manner described in section 2.4 but contain instationary terms with time derivatives such as the mass or internal energy stored in a piece of equipment. Some of these equations are algebraic such as material property functions or heat transfer correlations and others are differential, containing time derivatives such as the temperature of a body of water or the temperature of casing material in a component. Differential equations with algebraic constraints lead to DAEs, which are mentioned and further explained in section 2.1.2. As implied by the name, these systems of equations contain ODEs as well as AE. The DAE resulting from models of power plants are non-linear as mentioned in section 2.4.1.

The differential equations in these DAEs can often be stiff which can cause numerical issues when solving them and will also limit the methods for resolving them. The term stiffness refers to a coupled system of equations in which certain components vary on a very different time scale [32]. In other words, the system contains very fast and very slow

components[33]. A more mathematical and in-depth definition of the term has been presented by Brugnano [32]. Curtis [34] gives an additional concept of stiffness linking the variation speed of a time-dependent variable to a chosen evaluated time step. In certain problems, this may lead to unreasonable small time steps to keep the differential equation solvable with traditional methods[32]. Several authors denote that stiff problems can be solved better via implicit methods than explicit methods.[32][34][2] Solvers with implicit methods have been chosen in this study to address these stiff sets of equations. Depending on specifications of the given DAE systems, methods for the solution have been selected. In case of index one DAEs, the backward Euler method is suitable for DAE systems with stiff ODE [3]. In case of higher index systems, reformulation is necessary in order to apply the backward Euler method, Runge-Kutta method or in fact any other multi-step method [3][2]. Ascher [3] shows that these methods fail in higher index problems.

Several softwares are available which aid in the solution of DAE systems. The *IDAS* code is part of a larger software package called SUNDIALS (SUite of Nonlinear and DIfferential/ALgebraic equation Solvers) and was used to solve the DAE systems in this study [35][36]. The *IDAS* package is designed specifically to solve DAEs. The integration method in *IDAS* is variable order, variable coefficient backwards differentiation formulas in fixed leading coefficient form. The method order varies between 1 and 5. The solution of the resulting non-linear system is accomplished with some form of Newton iteration. As mentioned in section 2.4.1, Newton iterations require a choice of relatively good starting values for algebraic variables in order to converge. This problem remains in the dynamic simulation causing numerical issues when solving the DAE which are hard to track. The developers of *IDAS* are addressing this issue augmenting a Newton iteration with a line search globalization strategy [37]. Line search algorithms are iterative methods to determine the minimum of a cost function. For problems not falling into a category being suitable for that method, the user is responsible for providing adequate starting values [37]. In this work line search augmented Newton algorithms as implemented in the *IDAS* have been used but have been shown to be insufficient on their own. The following section section 2.4.3 explains the method implemented in this work to find suiting initial conditions for the solution of DAE systems.

2.4.3 Initialization of algebraic variables.

While appropriate starting values for the state variables can often be derived from plant data, the initialization of the affiliated algebraic variables represents a hurdle as they differ greatly for different sets of input variables, parameters and state variable starting values. As mentioned in section 2.4.2, the choice of initialization of algebraic variables is crucial for the dynamic simulation to converge to a solution.[30] Finding these initial values is typically subject to a non-linear programming problem (NLP) and analytical solutions are therefore not available. As suggested by Martelli [10], several time steps of a more robust optimization algorithm are performed to find suitable initial algebraic variables consistent enough for the methods implemented in *IDAS*.

In the first time step of each simulation in this study, a non-linear optimization is performed to determine a suitable set of algebraic variables either consistent or close enough to consis-

tency for the line search algorithm augmented Newton method of the *IDAS* solver to converge to a solution. For this optimization problem, a moving horizon estimation (MHE) has been performed to determine the algebraic variables in the first time step. The MHE is an optimization method for state and parameter estimation. Based on a finite set of past measurements, often called the estimation horizon, the current state of the system is inferred.[38][39] The MHE is explained in more detail in section 2.5.2. In this work, *Casadi's* [40] interface to the Interior Point Optimizer *IPOPT*, an open source software package for large scale non-linear optimization, has been used to solve the optimization problem. It can be applied to solve general non-linear programming problems of the form

$$\begin{aligned} \min \quad & h(x)\ x\epsilon\mathbb{R}^n \\ \text{s.t.} \quad & g^L \leq g(x) \leq g^U, \\ & x^L \leq x \leq x^U \end{aligned} \tag{2.12}$$

Here, $x\epsilon\mathbb{R}^n$ are the optimization variables possibly with lower and upper bounds, $x^L\epsilon\ (\mathbb{R} \cup \{\infty\})^n$ and $x^U\epsilon\ (\mathbb{R} \cup \{+\infty\})^n$ with $x^L \leq x^U$, $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is the objective function, and $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ are the general non-linear constraints. Equality constraints of the form $g_i(x) = \overline{g}_i$ can be specified by setting $g_i^L = g_i^U = \overline{g}_i$.[41]

In order for this method to be applicable, a measurement of that time step of at least one output of the system has to be available. In this particular application of the MHE, the state variables of the ODE are given. The algebraic variables on the other hand are being manipulated in a way to fit the state variables of the system and simultaneously the measurement in that time step. Thereby, a set of algebraic variables that satisfy the set of equality constraints and the set state variables is found. In a strict sense, it cannot be spoken of a MHE as no parameter or state is estimated with this method. The framework for MHE in *do-mpc* however provides an interface for efficient optimization algorithms. The software framework *do-mpc* is a comprehensive open-source toolbox for robust model predictive control (MPC) and MHE based on *Casadi*. It is explained closer in section 2.6. Using the framework's interface for MHE, the objective function found in eq. (2.18) is reduced of all its terms but the first. Initial states are found which satisfy the starting values of the ODE and the algebraic constraints.

During the present study, this method has shown to be effective with less accurate starting values than the line search algorithms implemented by *IDAS* and the combination of the two enables converging simulations with broader ranges of starting values. This allows for simulation of different input variables without manually adjusting the initial values of the algebraic variables. The need for that manual adaptation is also being lifted when adjusting parameters for the purpose of data-fitting.

2.4.4 Model reduction

Model reduction is a concept in the field of process simulation where efforts are made to remove redundant or unnecessary terms from a mathematical model in order to simplify its computation. By reducing the number of terms a model has, it allows for faster and more efficient simulations and optimizations. It also increases the robustness of the systems in terms of

numerical stability. The set-up and solution of mathematical models of complex processes may require many resources and its accuracy needs to be coherent with its specific purposes.[42] A trade-off in accuracy and computational effort can be made in order to derive sufficiently accurate but easy-to-handle models. Especially for optimization purposes, simplified models are favorable.[43]

A model entirely based on physics and fundamental phenomena is reliable in a sense that the applicability of these phenomena has been validated in many instances. It can be used to extrapolate into the future as these fundamentals will hold true.[43] However, complex systems of equations which may include systems of partial differential equations, DAE systems of high order, and non-linearities arise from these physical interactions. As indicated in section 2.4.2, the systems of equations originating from power plants and their components can be of this nature and may be ressource-intensive to solve or optimize. The reduction of a model can yield a system of equations more suited for efficient numerical solution and optimization.[44]

The reduced models may include simplified physical equations, empirical correlations or possibly data driven subsystems. Here data driven models refer to models deducted from large sets of data using methods of machine learning, artificial intelligence and statistics. Just like other models they can be used to predict a systems behavior but they are not based on the underlying physical fundamentals. As the dynamic of the model is often not accessible for the user, these models are also referred to as black-box models. The reduced systems are interpolative in nature as they require the results of an accurate physics-based model or operational data from the system.[43] To bring back extrapolative qualities to these models, they have to be validated against physics-based models or data to thereby ensure their performance. Steps that can be taken regarding the set-up of of the system are:

- Simplifying physical equations by removing negligible or redundant terms [43]
- Time scale reduction by neglecting dynamic behavior too fast or too slow for the relevant time scale [45]
- Formulating data-driven input-output models [46]

Through empirical models, a correlation between input and output variables can be derived without relying on actual physical equations. It is possible for models to rely entirely on these correlations. Empirical models use measured input and output data of a system and manipulate the model with means of interpolation or regression to reproduce the observed behavior.[45][46] Coefficients of correlation functions for phenomena or parameters can be adjusted in these types of models in an optimization problem. Some strategies for finding hybrid models of that nature are explained by Marquardt [45]. In this work empirical correlations are used to increase the accuracy of the physics based models. In this context, the use of machine learning and artificial intelligence opens great possibilities of model reduction [47]. Often these models are also referred to as black-box models. If only one part of the phenomenon is described in that way, these systems are referred to as gray-box models. Data driven models however have not been implemented in this work.

A relevant tool in model reduction is the trust region method. The basic idea is that the approximation is done not for the entire domain, but for a subset of the domain in which the approximation is expected to be accurate. This results in a lower absolute error of the approximation and makes the functional form less crucial. Simpler functions for approximations are made possible while still maintaining a high accuracy. The approximation is however limited to the use within the trust region. In a first step, the optimization problem is formulated. Secondly, the approximation is constructed in the constricted region. Thirdly, said approximation is evaluated and then validated against the original model or data.[43] The trust region method finds use in this study in the approximation of steam properties which is explained in section 3.1.1.

2.5 Model Based Control

Processes in the physical world can be approximated by mathematical models. These mathematical models can then be used to design control mechanisms to influence the process with the variation of a set of manipulated variables to follow a desired behavior. The time constants of the proportional integral derivative (PID) controller often depend on local behavior of a process which can be approximated by a mathematical model.[48] Model-based control in this context however refers to control algorithms in which the model is explicitly embedded [48]. Meanwhile, the time constants of PID controllers can, but do not have to be identified using a system's model. They could also be derived from process data directly.[49] Model-based control techniques like MPC use the predictive nature of the model to derive the manipulated variables. They achieve high performance and often approach the optimal operating point better than PID controllers, but at the cost of higher computational expenses [48]. With the emergence of higher availability of computational power, online applications of model-based control became more relevant, especially in large-scale systems like power plants as these techniques have been applied in various studies [50][51]. The process model predicts values of the measurements. The measurements and the predicted values are being compared and the information of the comparison is conveyed to the controller or the optimizer respectively. The model as being shown in a general form in eq. (2.13) and eq. (2.14), is being formulated in this work in the continuous form.

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t), \mathbf{z}(t), \mathbf{p}, \mathbf{p}_{tv}(t)) + \mathbf{w}(t) \tag{2.13}$$

$$\mathbf{y}(t) = h(\mathbf{x}(t), \mathbf{u}(t), \mathbf{z}(t), \mathbf{p}, \mathbf{p}_{tv}(t)) + \mathbf{v}(t) \tag{2.14}$$

However, the software implemented in this work *do-mpc* transforms the model in its discrete form as shown in eq. (2.15) and eq. (2.16). The following explanations in this chapter are therefore using the discrete model formulation.

$$\mathbf{x}_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \mathbf{p}, \mathbf{p}_{tv,k}) + \mathbf{w}_k \tag{2.15}$$

$$\mathbf{y}_k = h(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \mathbf{p}, \mathbf{p}_{tv,k}) + \mathbf{v}_k \tag{2.16}$$

The bold notation here refers to vectors, e.g. $\mathbf{x}_{0:N+1} = (x_0, x_1, ..., x_{N+1})$, the states of the system are denoted as $x(t)$, x_k, the systems measurements as $y(t)$, y_k, the control inputs are denoted

as $u(t)$, u_k, the algebraic variables as $z(t)$, z_k, parameters as p, p and time varying parameters as $p_{tv}(t)$, $p_{tv,k}$. In this general form of the model, the system is disturbed by additive noise $w(t)$, w_k. The measurement noise is denoted as $v(t)$, v_k.[38] No measurements of potential disturbances have been included into this estimation process which makes the source of deviance from measured variables indistinguishable between measurement inaccuracy and disturbance. Consequently, only one term is needed for describing that deviance. The term v_K is kept in the estimation process and the term w_k is set to zero. This decision is taken for means of simplicity and the assumption would need to be assessed further in possible following studies as disturbances might in fact occur in the system and could be included through measurements such as the tracking of clouds. The parameters p differ from the time varying parameters p_{tv} in the form that parameters p stay constant over the span of the simulation. Time varying parameters p_{tv} change with every time step and have been used in this study to model input variables. The process model is of high importance for the success and accuracy of the control effort and therefore it is given special attention to in this work.

2.5.1 Model Predictive Control

Model predictive control is a control strategy based on the predictions of a model for the system's dynamic behavior. An optimization algorithm is used at each time step to determine the best possible control input. The optimization problem considers the model's predictions in combination with state measurements or state estimates within a finite time window. This time window is called the prediction horizon.[52] A figure illustrating the concept of prediction horizon, control horizon and the structure of MPC is found in fig. 2.6. MPC is a tool to ensure the satisfaction of constraints even in presence of uncertainties and disturbances. It is suited for applications with multiple input and output variables and complex non-linear dynamics. [52][53] When MPC is applied to non-linear systems it is by some authors also referred to as non-linear model predictive control (NMPC). The basic structure of predictive control is illustrated in fig. 2.5. For the controlled variable, it is desired to follow a set trajectory, labeled

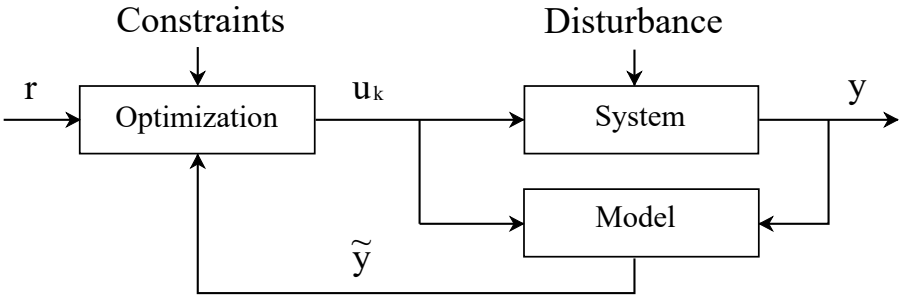


Figure 2.5: Basic structure of predictive control.

r in fig. 2.5. The quality of the control effort can then be quantified by deviance from that trajectory. The model provides a correlation between manipulated variable and controlled variable. Based on the trajectory r, the constraints and the prediction of the system's behavior $\tilde{y}$, the optimizer is generating a set of manipulated variables u. The first element of that series u_k is passed on to the system and is used as current input. The control loop is closed by integrating the predicted development of the system which is based on the predicted

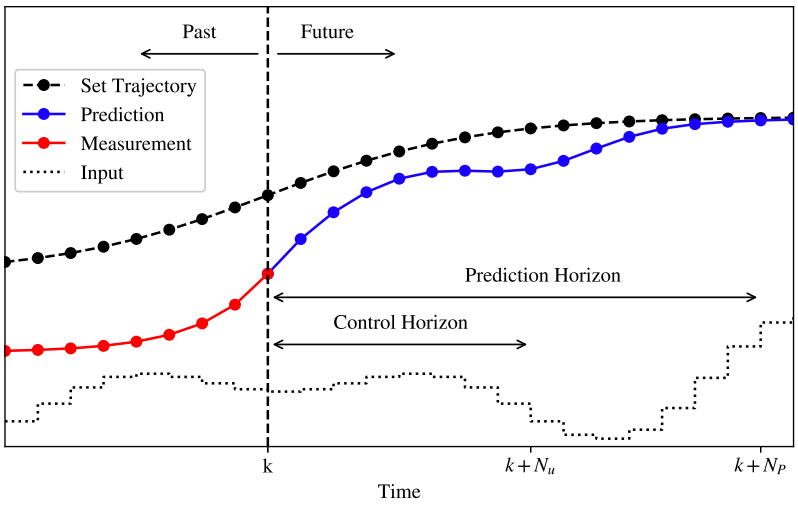


Figure 2.6: Scheme of Model Predictive Control.

future input variables and the current measurements y. A comparison between estimations and measurements is conducted to mitigate inaccuracy in the predictive model and possible disturbances.[52][54]

Additional objectives can be implemented with the help of the objective function. It consists of multiple objective terms penalizing for example deviance from the set trajectory or the intensity of changes in the manipulated variable.[52][53] The objective function can be found in the first line of eq. (2.17). The objective function can be split into a Meyer term (m), penalizing the final state of the system, and a Lagrange term (l), penalizing each time step. For the application of MPC, the current state of the system needs to be known. If comprehensive measurements y_k are not available, a state estimate $\hat{x}_k$ can be used alternatively. The estimate can be computed e.g. through MHE or a Kalman filter method.[38] The concept of MHE is mentioned earlier in this chapter and explained closer in section 2.5.2. The optimization of the manipulated variables with respect to the objective function and constraints, based on the current process state, is performed in each time step for the length of the control horizon. Just like the prediction horizon, the control horizon is moving with the the current time step k forward, see fig. 2.6. For time steps outside the control horizon, the manipulated variables remain constant.[52] Only the entry for the first time step in the control horizon is applied in the actual process. With the discrete model presented in section 2.5, the optimization problem can be formulated as illustrated in eq. (2.17). It is based on the formulation presented in the manual of *do-mpc* [38]. In fig. 2.6, an illustration of the principal function of MPC is depicted.

$$
\begin{aligned}
\min_{\mathbf{x}_{0:N+1},\mathbf{u}_{0:N_p},p,\mathbf{z}_{0:N_p}} \quad & \mathbf{m}(x_{N_p+1}) + \sum_{k=0}^{N_p} l(\mathbf{x}_k, \mathbf{z}_k, \mathbf{u}_k, \mathbf{p}, \mathbf{p}_{tv,k}) \\
\text{s.t.} \quad & \mathbf{x}_0 = \hat{\mathbf{x}}_0, \\
& x_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \mathbf{p}, \mathbf{p}_{tv,k}) \quad k = 0, \dots, N_p, \\
& 0 \geq g(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \mathbf{p}, \mathbf{p}_{tv,k}), \quad k = 0, \dots, N_p, \\
& \mathbf{x}_{lb} \leq \mathbf{x}_k \leq \mathbf{x}_{ub} \quad k = 0, \dots, N_p, \\
& \mathbf{u}_{lb} \leq \mathbf{u}_k \leq \mathbf{u}_{ub} \quad k = 0, \dots, N_u, \\
& \mathbf{z}_{lb} \leq \mathbf{z}_k \leq \mathbf{z}_{ub} \quad k = 0, \dots, N_p, \\
& 0 \geq g_{\text{terminal}}(\mathbf{x}_{N_p+1})
\end{aligned}
\tag{2.17}
$$

Here, N_p is the prediction horizon, N_U is the control horizon and $\hat{\mathbf{x}}_0$ is the current state of the process, either measured or estimated. Upper and lower bounds of states $\mathbf{x}_{lb}$, $\mathbf{x}_{ub}$, manipulated variables $\mathbf{u}_{lb}$, $\mathbf{u}_{ub}$ and algebraic variables $\mathbf{z}_{lb}$, $\mathbf{z}_{ub}$ are specified. Terminal constraints can be set with the term g_{terminal} and general, non-linear constraints can be implemented with g. The Meyer term m and the Lagrange term l together make up the objective function.

2.5.2 Moving Horizon Estimation

Moving horizon estimation is an efficient method for state estimation, especially suited for non-linear systems. With MHE it is also possible to include constraints for the estimated states.[43] The MHE outperforms traditional estimation methods like the extended Kalman filter in terms of accuracy, especially dealing with non linear systems. Reasons for that are the linearization of the non linear model within the extended Kalman filter algorithm. Additionally the extended Kalman filter falls into the category of unconstrained estimators allowing physically impossible solutions to arise from bad choices for initial conditions.[53] Using the MHE however results in an increase of computational complexity.[38] The basic idea of MHE is to estimate the state's trajectory based on a finite number of past measurements. This number of measurements is often called the estimation horizon. An optimization problem is formulated on each time step in the horizon where the state is estimated based on the measurements taken in that time step.[43][53] In this optimization problem, the finite sequence of states, algebraic states and control inputs are the optimization variables.[38] A schematic illustration of the principle of MHE can be found in fig. 2.7. The horizon of the length of N time steps is marked with the black arrow. Measurements from that time interval are taken into account for the estimation in step $k + 1$.

The manual of the *do-mpc* [38] software which was used to perform the MHE in this study formulates the determination of sequences as follows:

- The initial state has to be coherent with the last measurement of that state.
- The calculations resulting from the estimated system must match the measurements.
- The describing state equations are obeyed.

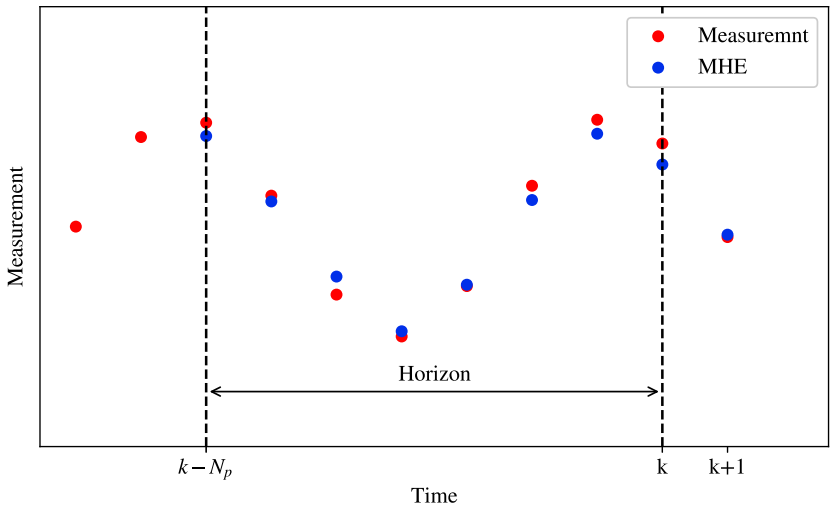


Figure 2.7: Scheme of Moving Horizon Estimation.

Following this concept, the problem can be formulated as:

$$\begin{aligned}
\min_{\substack{\mathbf{x}_{0:N},\mathbf{u}_{0:N},p,\\ \mathbf{w}_{0:N},\mathbf{v}_{0:N}}} \quad & \frac{1}{2}\|\mathbf{x}_0 - \overline{\mathbf{x}_0}\|^2_{P_x} + \frac{1}{2}\|\mathbf{p} - \overline{\mathbf{p}}\|^2_{P_p} + \sum_{k=0}^{N-1}\left(\frac{1}{2}\|\mathbf{v}_k\|^2_{P_v,k} + \frac{1}{2}\|\mathbf{w}_k\|^2_{P_w,k}\right) \\
\text{s.t.} \quad & x_{k+1} = f(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \mathbf{p}, \mathbf{p}_{tv,k}) + \mathbf{w}_k, \quad k = 0, \dots, N, \\
& y_k = h(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \mathbf{p}, \mathbf{p}_{tv,k}) + \mathbf{v}_k, \quad k = 0, \dots, N, \\
& 0 \geq g(\mathbf{x}_k, \mathbf{u}_k, \mathbf{z}_k, \mathbf{p}, \mathbf{p}_{tv,k}), \quad k = 0, \dots, N
\end{aligned} \tag{2.18}$$

The bold notation here refers to vectors, e.g. $\mathbf{x}_{0:N+1} = (x_0, x_1, ..., x_{N+1})$, the overline notation $\overline{x}_0$ refers to the measured initial state, the overline notation $\overline{p}$ refers to the last estimated parameter. The notation $\|x\|^2_P = x^T P x$ refers to the P weighted squared norm. Through weights in the cost function eq. (2.18), the importance of aspects in optimization can be acknowledged. Through a high P weighted norm P_x the importance of close estimated initial states with the last measurement can be emphasized. Through alteration of P_p the deviance of estimated parameters between time steps can be weighted. An increase of P_p would then lead to less fluctuation of an estimated parameter over the time span of the simulation. The weight P_v enables to weigh penalty for any deviance with measured values due to noise in the measurement. With the weight P_w, deviance between simulation and measurement caused by disturbances can be penalized.

In this study the estimation of constant parameters is the aim. These parameters can be used within empirical correlations to enhance the models ability to reproduce measured data. The time varying parameters are used to model time varying external influence. The ambient temperature could be an example for a time varying parameter. Also an input stream originating from a connected component which is not part of the simulation model can be modeled using a time varying parameter. The estimation of constant parameters can be conducted using the MHE method. In this study, parameters have been estimated using

relatively short time horizons of two to five percent of the data set. The final parameter is then determined through the mean over the time span of the simulation. When choosing data sets for fitting, it needs to be confirmed that the effects to be analyzed take place during that period. Constraints have been set for physical limitations, e.g. heat conductivity $\lambda > 0$. Alternatively, long estimation horizons that cover the whole data set could be set, finding parameters fit for the whole span of the simulation.

The implementation of the MHE in *do-mpc* has also been used to solve the non linear problem of finding consistent starting values for the algebraic variables in a DAE system. The procedure is mentioned and further explained in section 2.4.3.

2.5.3 Least Squares Method

Least squares is a mathematical method used to find the best fit line or curve for a set of data points. It is a type of regression analysis that minimizes the sum of the squares of the differences between the observed values and the values predicted by a given line or curve. The least squares line or curve is determined by minimizing the sum of the squares of the vertical deviation of each point from the line or curve. The sum of the errors S is to be minimized by altering a set of parameters $\mathbf{C}$.[55]

$$S = \sum_{0}^{N_D} (y_i - f(x_i, \mathbf{C}))^2 \tag{2.19}$$

Here, y_i and x_i are the data points, while N_D is the number of data points. If the fitted model consists of a polynomial the adjusted parameters are its coefficients $\mathbf{C}$, this method can also be called polynomial regression. Polynomial regression is a type of regression analysis that uses a polynomial function to model the relationship between a dependent variable and one or more independent variables. The polynomial can be of any degree, from linear to higher orders such as cubic or quartic. The curve is then fit to the data points by adjusting the coefficients of the polynomial to minimize the error between the observed points and the predicted values from the curve. With this method, multiple one dimensional polynomials can be combined to obtain approximations dependent on two variables. The minimum of the summed error S is found by setting the gradients of S in regard to the coefficients C_j to zero.

$$\frac{\partial S}{\partial C_j} = 0 \quad , j = 0, ...M \tag{2.20}$$

Here M is the number of coefficients and j is the corresponding index. In linear models, the parameters can be found analytically. In non-linear problems, initial parameters have to be chosen which are then refined in iteration. With an extension of the Newton algorithm for non-linear problems, minima can be identified. This method is called the Gauss-Newton method and is described in close detail by Nocedal [56].

2.5.4 Sequential Quadratic Programming

Sequential quadratic programming (SQP) is an iterative optimization method for solving non-linear optimization problems. It is used to find the minimum of an objective function

which can be subject to constraints. Through iteratively solving a sequence of subproblems, the solution is narrowed down with each calculation step. Step size and search direction are part of these subroutines. In this manner, information on the solution of the previous step is used to refine the solution of the next step and to find a local optimum. The method is introduced by Kraft [57] and also closely explained by Nocedal [56]. Among other problems, this method can be used to solve least square problems. In that case, the method is called sequential least square quadratic programming (SLSQP).

2.5.5 Bootstrap Method

The bootstrap method is a resampling technique used in statistics to estimate the accuracy of a model or statistic. It works by taking samples with replacement from the original data set, after which the model will be calculated on each sample. The bootstrap method can be used to better understand the uncertainty of the analyzed model and can help to obtain an estimation of its variance and other statistical properties.[58] Thereby, it helps assessing the accuracy of estimates. In this study, the bootstrap method is used to evaluate the quality of parameter estimation in the fitting process of chosen components. The steps followed to perform the bootstrap method in this study are listed below and have been recommended in this form by King [59].

1. Set index $i = 1$.

2. The model computes results using the set of estimated parameters at the specific points in time where measurements are available.

3. A noise is added onto the simulated results, imitating an uncertainty in the measurements.

4. The new data set, stemming from the disturbed, simulated results, is used to identify another set of parameters.

5. Set $i := i + 1$ and repeat step 2, 3 and 4 until $i = K$, where K is a sufficiently large number.

6. Numerous estimates are now available and statistical analysis of the results can be performed.

Bootstrapping has several advantages. It is an intuitive approach to evaluate the quality of estimates without relying on strong assumptions or any complex procedure. Confidence intervals for the evaluated parameters can be stated which can be helpful by judging the reliability of an estimate. On the downside, the bootstrap method can be expensive in terms of computing power as the model of interest is calculated multiple times with different samples of parameters. The samples need to be picked by random in order to avoid a bias in the results of the method. The accuracy of the estimates cannot be ultimately guaranteed as the bootstrap method is an approximation based on a limited number of samples.

2.6 Simulation and Optimization Frameworks

In this section, the relevant softwares and frameworks that have been used in this work are being listed and their fundamental principles are explained. The open-source tool for optimization *Casadi* and the open-source framework for MPC and MHE *do-mpc* are presented. In this context, optimization implies improvement rather than a global mathematical minimum [11].

Casadi [40] is an open-source tool for non-linear optimization and automatic differentiation. It facilitates rapid implementation of different methods for numerical optimal control, both in an offline context and for non-linear model predictive control. At the core of *Casadi* is a self-contained symbolic framework that allows the user to construct symbolic expressions. With *Casadi*, it is possible to generate derivative information efficiently using automatic differentiation to set up, solve and perform forward and adjoint sensitivity analyses for systems of ODEs or DAEs.[41] Moreover, it is possible to formulate and solve non-linear programs as well as optimal control problems.[41] It does not however provide solutions for optimal control problems entered by the user. It only provides building blocks for implementing optimal control solvers. Automatic differentiation is distinct from numerical simulation and symbolic differentiation as it uses exact formulas along with floating-point values to evaluate the derivative of a function. Expression strings as in symbolic differentiation lead to high calculation efforts and inefficient code while numerical differentiation involves a rounding error using discretization of the derivative with a differential quotient.[60] As computational programs are fundamentally made up of basic arithmetic operations and elementary functions, automatic differentiation applies the chain rule to these operations, thereby providing derivatives of arbitrary order. Computational efforts can be kept at only a small constant factor of the original operations while maintaining a high accuracy, especially in the higher orders.[61] Both of the two classical methods, symbolic and numerical differentiation, are slow at computing partial derivatives of a function with respect to many inputs, as is needed for gradient-based optimization algorithms like they are used in *Casadi*.

In order to perform parameter optimization with MHE, the software *do-mpc* [62] has been used which builds on the tools provided by *Casadi*. The *do-mpc* library is a comprehensive open-source toolbox for robust MPC and MHE. With *do-mpc*, it is possible to facilitate the implementation of non-linear control and estimation problems. Furthermore, it supports the handling of DAEs and uncertainty values. In this work, the embedded methods in this framework are used to perform the model simulations. The in the framework implemented MHE is being used to estimate unknown parameters in the power plant's components, fitting the simulation results to power plant data. Another option for data fitting is minimizing the squared error between simulation and measurement. *SciPy optimize* provides functions for minimizing or maximizing objective functions which can be subject to constrains. It includes solvers for nonlinear problems such as constrained and non-linear least squares methods. For the identification of local minima, the least squares method has been implemented to identify suitable parameter combinations in the fitting process of both polynomial approximation of water and steam properties as well as the parameter identification of power plant components.

2.7 Object-Orientated Programming

Object-oriented programming (OOP) is an approach to programming that focuses on the creation of objects containing data and methods which in turn operate on that data. The objects are organized into classes which contain the data and methods that all objects in that class share. This approach allows for code reuse and efficient data storage, making it an effective way to create scalable and maintainable programs. Its modular nature makes it possible to divide complex problems into smaller, more palpable problems. OOP helps create code that is easier to understand as objects can be thought of as tangible entities which contain data and methods that can be used to manipulate that data. It also makes code easier to maintain as changes to one object do not affect other objects in the same class. However, the performance of the program may suffer and may be slower than programs written with other programming methods. Debugging might also become more difficult as the programs become more intertwined. Should one object have a bug, it can affect other objects in the program, making it difficult to pinpoint the source of the issue.

2.7.1 Structure of the Code

The code is organized according to principals of OOP. For each component, two classes are set up. The *variables-class* contains all variables and parameters relevant for this component. In the *equation-class*, the governing equations of that component are formulated, using the symbolic representation of the variables defined in the *variables-class*. A schematic representation of the code's structure can be found in fig. 2.8. In the *variables-class*, values for parameters

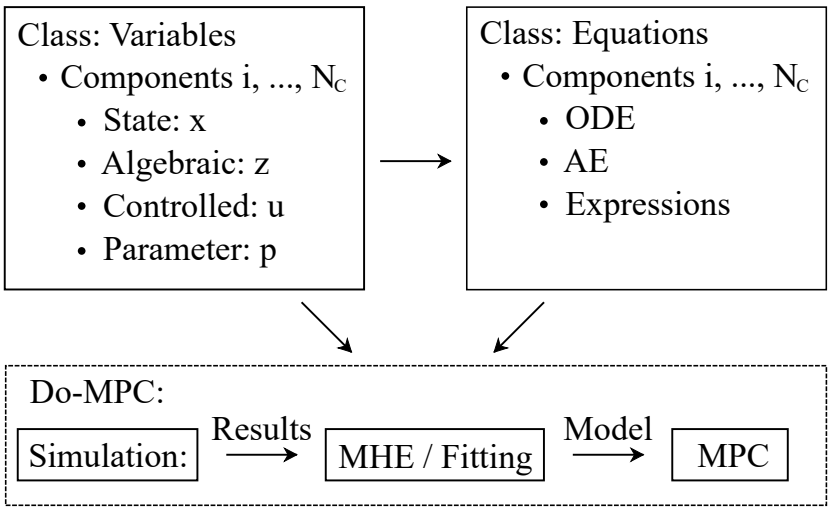


Figure 2.8: Scheme of the code structure.

and starting values for states and algebraic variables are stored. The *variables-class* contains subclasses which sort all variables by their function in the equations system, meaning into states, algebraic variables, controlled variables, and parameters. The *equation-class* uses the symbolic variables defined in the *variables class* to formulate the relationship between states,

algebraic variables, controlled variables and parameters. The *equation-class* contains two subclasses sorting the equations depending on the presence of a time derivative into algebraic and differential equations. In addition, the *equation-class* contains so called expressions. Expressions are used to express certain results which are not necessary for the calculation in other equations but which might be useful information in other contexts. The filling level of a tank as a function of its liquid volume and geometric parameters could be an example of such an expression. In general these equations could be formulated as algebraic variables. However, expressions are treated differently within the *do-mpc* framework and using them where possible, reduces computational expenses.

Through an interface, *do-mpc* communicates with these classes and builds the model. The simulation is performed which generates a set of results. In a comparison with measured plant data, the fitting process is performed using the MHE implemented in *do-mpc*. The improved model could then be used to conduct MPC, also within the *do-mpc* framework. The class-oriented structure of this code enables a more user readable and organized alternative to implementing all equations and variables directly into *do-mpc* using it's own syntax. In addition, it would be feasible to alter the interface in a manner that the same equations and variables can be used with another framework for simulation and optimization.

3 Modeling

In this section, the modeled components of the plant and their underlying physical equations are introduced. The material properties and their approximation used in the models are explained. The hot air cycle with the storage system are addressed followed by the heat recovery steam generator and the applied turbine model.

3.1 Thermodynamic Properties

In this section, the implementation of thermodynamic properties into the model is explained. The fluids used in the model are water, steam and air.

3.1.1 Water and Steam Properties

The property values for water and steam have been obtained from the *Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam* (IAPWS) [63]. *Casadi* requires smooth functions to be able to perform the automatic differentiation. To handle the calculation expense and to be able to pass *Casadi*-variables to the property functions, the results of the equation implemented by IAPWS, which is based on the Gibbs enthalpy, are approximated using polynomials. Through variable implementation of polynomial degrees, a trade-off between the calculation effort and the accuracy is made. The polynomials can be set according to the need of the application. Among others, Goemans [64] suggests the approximation of steam properties with Chebyshev polynomials and uses the least square method to obtain the coefficients. Åberg is implementing two-dimensional polynomials for optimization-friendly steam property functions [65]. In the relevant pressure and temperature intervals and for the regarded properties in this study such as enthalpy and entropy, an accurate description of properties is implemented using regular polynomials. This decision keeps the calculation expenses down and simplifies the procedure. Following suggestions made by Åberg, additional data points have been added along the saturation line in the approximation process to ensure accuracy in these regions [65]. The values for enthalpy and entropy have been modeled separately for the liquid and gaseous regions. The following equation shows the polynomial form with which the steam and water properties have been approximated.

$$F(x, y) = \sum_{i=0}^{m} \sum_{j=0}^{n} c(i, j) x^i y^j \tag{3.1}$$

The equation includes the independent variables x and y, the polynomial coefficients c as well as the polynomial degrees m and n. The polynomial coefficients c have been determined using the ordinary least square method minimizing the quantity.

$$\sum_{a} \sum_{b} \sqrt{(F(a, b) - G(a, b))^2} \tag{3.2}$$

Table 3.1: NRMSD between approximation and steam table values as a function of polynomial degree.

Property/Degree	1	2	3	4	5	6	7
$s(p,t)_{\text{steam}}$	0.0595	0.0304	0.0180	0.0248	0.0604	0.1198	0.2002
$h(p,t)_{\text{steam}}$	0.0069	0.0031	0.0038	0.0294	0.0782	0.1560	0.2622
$h(p,s)$	0.0390	0.0188	0.0124	0.0032	0.0017	0.0011	0.0008
$s(p,t)_{\text{water}}$	0.0178	0.0027	0.0004	0.0068	0.0131	0.0197	0.0252
$h(p,t)_{\text{water}}$	0.0031	0.0010	0.0002	0.0039	0.0083	0.0133	0.0178

In this equation, a and b are the variable properties and G is the discrete value of the IAPWS function at that value pair. To obtain a smooth function over the whole range of temperature and pressure including the wet steam region, the polynomial for the gaseous region and for the liquid region are summed together with Heaviside $\mathcal{H}$ functions as coefficients. The argument of the Heaviside function becomes positive in the gaseous region and negative in the liquid region. For the Heaviside function multiplied with the term of the liquid region, the argument is multiplied by −1.

$$\text{arg} = T - T_{sat}(p) \tag{3.3}$$

$$F(p,T) = \mathcal{H}(\text{arg})\, F_{gas}(p,T) + \mathcal{H}(-\text{arg})\, F_{liq}(p,T) \tag{3.4}$$

Through that, it is ensured that the property functions remain differentiable. For the property function at saturation line a similar approach has been chosen, with only one property variable to unambiguously describe the property state. This function needs to be differentiable as well. It needs to be considered, that the accuracy of these functions at saturation line needs to be accurate enough to follow the saturation line but multiply the polynomial degree when using the functions $F(a,b)$ in a region including the wet steam region. For a quantitative analysis of the polynomial's accuracy, the root-mean-square deviation ($RMSD$) is taken into account. For a comparison of the $RMSD$ between different properties, a normalized root-mean-square deviation $NRMSD$ is used.

$$RMSD = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(F_i - G_i)^2} \tag{3.5}$$

$$NRMSD = \frac{\sqrt{\frac{1}{N}\sum_{i=1}^{N}(F_i - G_i)^2}}{max(F) - min(F)} \tag{3.6}$$

with N referring to the number of sampled values. For the properties in dependency of two variables, the $RMSD$ shows minima for specific numbers of degrees. In section 3.1.1, the correlation between polynomial degree and relative $RMSD$ values is shown.

In section 3.1.1 the NRMSD between polynomial approximation and the IAPWS values for the relevant thermodynamic properties in dependency of the polynomial degree is found. For properties in dependency of pressure and temperature, flat minima on polynomials of third degree can be identified. For the enthalpy as a function of pressure and entropy, a sufficient trade-off between the accuracy and the polynomial degree can be chosen at the fourth degree. It comes to attention that for the values in dependency of pressure and temperature the accuracy decreases with polynomial degrees > 3. Expected would be an increase in accuracy

with larger polynomial degree. A possible explanation could be a discrepancy between number and position of the fitting points and evaluation points. An oscillation around the fitted values might be evaluated during high amplitudes of the oscillation. Another possibility could be inaccurate sorting of fitting points into liquid and gaseous region. Points close or on the saturation line might be sorted wrongly, causing the polynomial to approximate to the wrong points. The inaccuracy is not explained in conclusion and needs to be taken into account. A further discussion of the model is found in chapter 5.

For applications involving optimization and control, it can be desirable to further lower the degree of these polynomial approximations in order to decrease the computational expense and to increase the numerical robustness of the simulation or of the control problem. Using polynomials for steam property approximation, minima which are not found in reality, may occur, thus hindering the optimization process or leading to a failure of convergence during calculation. To mitigate this circumstance or to exclude this source of error, the degree of the polynomial can be further lowered by defining a smaller realm where the function is supposed to be accurate. This method is referred to as the trust region method and is further explained in section 2.4.4 on model reduction. Several components allow a much smaller realm for validity for the approximation functions as the result of the simulation can be estimated based on their functionality or on operational data.

The outlet of the steam turbine during regular operation for example, will always be close to the saturation line at pressures clearly below atmospheric pressure and close to nominal condenser pressure. This allows for a decrease of the pressure range relevant for approximating polynomials to pressures below atmospheric pressure or, even more precise, around condenser pressure. The range of temperature can be estimated for this component as it will be clearly below the saturation temperature of atmospheric pressure, namely 100 °C, but above ambient temperature. With knowledge of the temperature and pressure ranges, a corresponding entropy range can be calculated. Through information about the analyzed system, the trust region for the polynomials can be reduced. Hence, the accuracy of the approximation can be kept while reducing the degree of the polynomial. Values with this method can be found in section 3.1.1 where a polynomial degree of one has been chosen. A polynomial degree of one does not result in a linear correlation as the $x \cdot y$ term remains. Region 1 in section 3.1.1 describes a range of properties with little knowledge on the evaluated state. A range including maximum pressure and maximum temperature of the entire process was chosen. Region 2 in section 3.1.1 includes values with basic knowledge on the observed state. It should be below atmospheric pressure and below boiling temperature at atmospheric pressure. The lowest pressure should be below nominal condenser pressure. A margin was added. Region 3 in section 3.1.1 includes detailed knowledge on the expected values. These can be obtained from operational data.

It can be observed that the NRMSD values for region 1 are several times higher than the values obtained with the polynomials of a higher degree in section 3.1.1. They are in the range of several percent and the use of the polynomial functions of first degree are not suitable for that trust region. Region 2 has lower NRMSD values but still maintains an inaccuracy that is too significant for the application. The accuracy of region 3 is comparable with the results

Table 3.2: NRMSD between polynomial approximation of first degree and value derived from IAPWS for three trust regions.

	Region 1	Region 2	Region 3
Pressure-Range in MPa	[0.01 - 3]	[0.03 - 0.1]	[0.03 - 0.05]
Temperature-Range in K	[274 - 874]	[274 - 374]	[294 - 354]
Entropy-Range in $\mathrm{J\,g^{-1}\,K^{-1}}$	[0.013- 7.372]	[0.013 - 7.360]	[7 - 8]
NRMSD (s(p,t))	0.0861	0.039	0.0007
NRMSD (h(p,t))	0.0722	0.039	0.0005
NRMSD (h(p,s))	0.0242	0.009	0.0135

of polynomials of a higher degree and is valid for the entire region of relevant properties. The accuracy of the approximations of entropy and enthalpy in dependency on pressure and temperature, $s(p, t)$ and $h(p, t)$, are even lower. Polynomials of a low degree can only be used if the range of the expected results is narrow. In cases where the expected results are well known, this method can be used to mitigate convergence issues or to increase computational efficiency. The results in section 3.1.1 show that this method can only be applied with great caution. It is numerically favorable to exclude the wet steam domain of the trust region if possible to avoid the multiplication of polynomials, as can be seen in eq. (3.4).

3.1.2 Air properties

The air properties have been modeled using polynomials in analogy to the steam and water properties. Correlation with air pressure is not taken into account as the effects are minor for this application. Therefore, the air properties are only a function of the air temperature. Accuracy and numerical stability are no issue when approximating the airs thermodynamic properties using polynomials.

3.2 Thermal Storage

In this section the thermal behavior of the storage unit is described. Hot air flows through ceramic storage material and heats up the mass in an isolated tank. The storage unit consists of four separate, identical tanks. The model used in this work is based on the ”One-Temperature-Model” (”Ein-Temperatur-Modell”) by Gall [22] with enhancements proposed in Nolteernsting [66]. In Gall [22] the heat transfer is calculated with use of the number of transfer units, a dimensionless quantity used for heat exchanger characterization [67]. The number of transfer units puts heat transfer and heat capacity rate in proportion. The storage is modeled as a discrete series of heat exchangers in which the temperature of the storage mass is horizontally constant.

Under the assumptions that the heat capacity of the air is negligible in comparison to the heat capacity of the storage media, the fluid phase does not have to be explicitly modeled.[68] Due to the low flow rate of air in the storage media the number of transfer units becomes

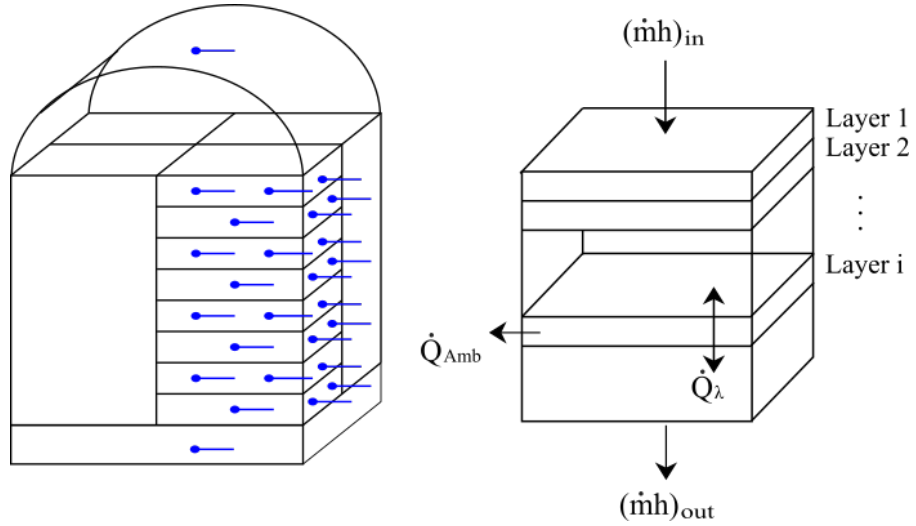


Figure 3.1: (Left) Scheme of the storage with it's measurement positions as constructed. (Right) Illustration of the discretization of the thermal storage model. Enthalpy flows are marked with arrows.

high and the heat exchange very efficient respectively. In other words, the heat transfer coefficient become infinite. Therefore it is assumed that at the exit of the discrete layer the air temperature is equal to the storage media temperature. From this reduction to the observation of one temperature the term "One-Temperature-Model" is deducted.

If the storage is neither charged nor discharged for an extended period of time, for example over night, the temperature profile becomes more homogeneous in axial direction, lowering the maximum temperature in the storage layer and thereby its utility. Therefore a term for axial heat conduction was added to the model in Nolteernsting [66]. Two further improvements proposed by Nolteernsting [66] and implemented in this work are the larger areas available for heat exchange in the top and bottom layers as well as a better resolution in form of a higher number of discrete layers going from eight layers in Gall [22] to one hundred in Nolteernsting[66]. The differential equation describing a discrete layer in the model is found in ??.

$$\begin{aligned} m_{St,i} \cdot c_{pSt} \frac{dT_{St,i}}{dt} &= \dot{m}_{Air,St}(h_{Air}(T_{St,i-1}) - h_{Air}(T_{St,i})) \\ &+ \alpha_{StAmb,i} A_{StAmb,i}(T_{Amb} - T_{St,i}) \\ &- \lambda_{St} \cdot \frac{H_{St}}{N_{St}}((T_{St,i} - T_{St,i+1}) + (T_{St,i} - T_{St,i-1})) \end{aligned} \tag{3.7}$$

The left hand side of that equation describes the temporal change of energy in the media of the storage layer. Layers of equal size *i* are used where *i-1* is referring to the layer before *i*. At the inlet to the storage the inlet temperature of the air is used. When switching from charging to discharging the order of discrete layers is reversed. Accordingly it is assumed, that the temperature at the exit upstream of the mass flow is equal to the temperature of the storage layer. The first term on the right hand side is describing the heat exchange between air and

storage media which is only dependent on the air mass flow $m_{Air,St}$ and the temperature of the storage T_{St}. It is based on the change of the specific enthalpy h_{Air} over the discrete layer. The following equation applies:

$$\frac{dh_{Air}}{dT_{AirSt}} = c_{pAir}(T_{AirSt}) \tag{3.8}$$

The change of the specific enthalpy represents the integral of $c_{p,Air} \cdot T_{St}$ over the temperature change. The entanglement of air and storage properties in $h_{Air}(T_{St})$ is due to the assumption of equal air and storage media temperature at the exit of the layer. The second term in eq. (3.7) describes the heat loss to the ambient environment. It depends on the heat transfer coefficient between storage and ambient environment $\alpha_{St,Amb,i}$ the area available to heat transfer in that layer $A_{St,Amb,i}$ as well as the driving temperature difference $(T_{Amb} - T_{St,i})$. The higher area available for heat transfer the top and bottom layer of the tank is considered.

The third term describes the axial heat conduction. It depends on the the storage media conductivity λ_{St} the heat conduction length $\frac{h_{St}}{N_{St}}$ and the temperature differences to the neighboring discrete layers $(i \pm 1)$. The heat transfer length is calculated from the height of the storage tank h_{St} and the number of discrete layers N_{St}. The temperature difference is calculated between the neighboring layers. At the top and bottom layer only conduction in one direction is calculated.

The storage at the STJ consists of four separate tanks which can be filled or discharged with independent mass flows $\dot{m}_{Air,St}$. Therefore the tanks can show different temperature profiles. The model is calculated with $4 \cdot 100 = 400$ states for the temperature T_{St} in four tanks with one hundred layers.

3.3 Power Block

In this section the components of the power block consisting out of heat recovery steam generator, steam turbine, feed water tank, condenser and pumps are introduced. The model of the power block is build on the basis of the model proposed in Gall [22] with refinements proposed by Vinnemeier [69]. Heat loss through outer shell of the components into the environment is considered as negligible. In general heat transfer through radiation becomes relevant at very high temperatures. In conventional power plants heat transfer due to radiation reaches the same order of magnitude as heat transfer through convection at temperatures above 1200 °C[70]. In this plant maximal temperatures of 700 °C are reached which is well below the mentioned value. As a consequence radiation is regarded as negligible in this study.

Heat transfer functions describe part-load characteristics of the heat exchangers and are modeled using transfer coefficient based on the applied correlations found in the VDI-Wärmeatlas (VDI) [71].The heat transfer coefficients k are calculated according to

$$\frac{1}{k} = \frac{1}{\alpha_{Air,W}} + \frac{s}{\lambda_W} + \frac{1}{\alpha_{W,H20}} \tag{3.9}$$

Table 3.3: Applied heat transfer correlations. VDI-Wärmeatlas (VDI)

Heat exchanger	Configuration	$\alpha_{Air,W}$		$\alpha W, H20$	
Preheater Economizer	Fin tube bundle	VDI section $M1$	[71]	VDI section $G1$	[71]
Evaporator	Plain tube bundle	VDI section $G1$	[71]	$\alpha = 1.96\dot{q}^{0.72}p^{0.24}$ α[W m^{-2} K], $\dot{q}$[W m^{-2}], p[bar] 0.2 < p 50 bar	[72]
Superheater1 Superheater2	Plain tube bundle	VDI section $G7$	[71]	VDI section $G1$	[71]

Comparison of simulation results with operational data using strictly heat transfer correlations derived from the sources listed in section 3.3 show significant discrepancy. This motivated the alteration of heat transfer coefficients to better fit the measurements. Constant coefficients or exponents to relevant values like the mass flow have been used to describe the heat transfer more accurately or avoid elaborate heat transfer correlations.

3.3.1 Preheater and Economizer

Dynamic modeling of the preheater and economizer leads to more accurate reproduction of the components behavior considering their thermal inertia. The amount of water inside the tube bundles as well as the mass of the steel piping is significant and therefore the accumulation of heat inside the pipe wall is considered. The wall temperature is assumed to be characteristic for heat transfer and contained heat in the wall. Therefore the temperature of the pipe is considered constant over its length, despite of a temperature gradient in flow direction. The following set of equations applies:

$$0 = (\dot{m}h)_{in,Air} - (\dot{m}h)_{out,Air} - \dot{Q}_{Air,W} \tag{3.10}$$

$$0 = (\dot{m}h)_{in,H20} - (\dot{m}h)_{out,H20} - \dot{Q}_{W,H20} \tag{3.11}$$

$$\frac{dT_W}{dt} = \frac{1}{m_W c_W}\dot{Q}_{Air,W} - \dot{Q}_{W,H20} \tag{3.12}$$

$$0 = \dot{Q}_{Air,W} - (\alpha A)_{Air,W}(T_{Air} - T_W) \tag{3.13}$$

$$0 = \dot{Q}_{W,H20} - (\alpha A)_{W,H20}(T_W - T_{H20}) \tag{3.14}$$

$$0 = \alpha^*_{W,H2O} - c_1 \dot{m}^c_{\mathrm{H2O}2} \tag{3.15}$$

$$0 = \alpha^*_{Air,W} - c_3 \alpha^*_{W,H2O} \tag{3.16}$$

For temperatures T_{H20} and T_{Air} relevant in the driving temperature difference the inlet temperatures of the respective fluid have been chosen. The residence time of water and air is in the range of several minuets and therefore considerable. Calculating both sides dynamically would lead to partial differential equations which are problematic to solve as they could cause instabilities and high calculation effort.

3.3.2 Evaporator

The evaporator is used to perform the phase transition of the working media within the water steam cycle. Heat from the hot air is introduced to evaporate the inlet flow of feed water. The high amounts of water inside the evaporator are cause to high thermal inertia in this component. The mass of water contained in this component is by magnitude of four higher then its inlet streams and therefore dynamic modeling is applied. The heat is introduced through a pipe bundle with hot air flowing through it and a pipe returning hot steam from the steam cooler. The mass of the piping as well as the shell of the evaporator are significant and are considered in the model. In order to represent the dynamics within the evaporator a two-phase equilibrium model is implemented as derived by Vinnemeier[69]. A summery of the differential algebraic system of equations is found in 5. The air side of the evaporator is modeled in a steady state as the mass of the fluid inside the pipe is relatively low and the flow rates substantially high. As mentioned above the mass of the pipe bundle is not negligible. Due to the high heat transfer coefficients on the water side the pipe walls temperature is assumed equal to the water temperature. The mass of the pipes is included in the model the same way as the casing of the evaporator. The differential algebraic system of equations used to describe the evaporator consists of

$$0 = (\dot{m}h)_{in,Air} - (\dot{m}h)_{out,Air} - \dot{Q}_{Eva} \tag{3.17}$$

$$0 = \dot{Q}_{Eva} - k^* A \Delta T \tag{3.18}$$

$$0 = k - f(\alpha_{\mathrm{Air,W}}, _W, \alpha_{\mathrm{W,H20}}, Geo) \tag{3.19}$$

$$0 = \Delta T - \left(\frac{T_{Eva,in,H2O} - T_{Eva,in,Air}}{2} - T_{Eva} \right) \tag{3.20}$$

$$0 = k^* - ck \tag{3.21}$$

$$0 = H - f(V_{\mathrm{liq}}, Geo) \tag{3.22}$$

In addition the two differential equations eq. (.6) and eq. (.7) and the associated algebraic equations found in chapter 5 are part of the full differential algebraic system. The preservation terms used in the model are

$$\sum_i \dot{m}_i = \dot{m}_{\mathrm{in,H20}} - \dot{m}_{\mathrm{out,H20}} \tag{3.23}$$

$$\sum_i \dot{m}_i h_i + \sum_j \dot{Q}_j = (\dot{m}h)_{in,H20} - (\dot{m}h)_{out,H20} + \dot{Q}_{\mathrm{Eva}} + \dot{Q}_{\mathrm{SC}} \tag{3.24}$$

$$\sum_k m_{\mathrm{steel,k}} c_{\mathrm{steel,k}} = m_S c_S + m_{\mathrm{Tube}} c_{\mathrm{Tube}} + m_{\mathrm{SC}} c_{\mathrm{SC}} \tag{3.25}$$

The DAE in chapter 5 is written in a compact form with emphasis on comprehension. In this form it would be required to solve for the gradients $\frac{dT}{dt}$ and $\frac{dV_{liq}}{dt}$ as algebraic variables. To avoid the implicit formulation and avoid the additional algebraic variables, the terms have been reformulated. The temperature gradient appears in the gradient of the liquid volume and vice versa. With simple mathematical operations, an explicit formulation can be derived. This formulation is longer and more difficult to read but numerically favorable as the interdependence within the model is avoided.

3.3.3 Steam cooler

The steam cooler enables the redirection of super heated steam back to the evaporator. The branch is located after super heater 1. Through the control of the mass flow of the redirected stream the outlet temperature of life steam can be regulated while recuperating its energy in the evaporator. The cooled steam is then mixed with the steam not redirected to the evaporator. The heat transfer, the mixing and the valve characteristics are modeled in a steady-state model. The following equations apply.

$$0 = (\dot{m}h)_{\text{in,H20}} - (\dot{m}h)_{\text{out,H20}} - \dot{Q}_{\text{SC}} \tag{3.26}$$

$$0 = \dot{Q}_{\text{SC}} - \dot{m}_{\text{SC}}\, c_{\text{p,SC}}\, \Delta T_{\text{SC}} \tag{3.27}$$

$$0 = \dot{m}_{\text{SC}} - f(S_{\text{SCV}}\, \dot{m}_{\text{in,H20}}) \tag{3.28}$$

$$0 = \Delta T_{\text{SC}} - (T_{\text{in,H20}} - (T_{\text{Eva,H20}} + \Delta T_{\text{SC,term}})) \tag{3.29}$$

The amount of redirected steam is controlled through the position of the three-way valve. Through analysis of operational data a terminal temperature difference between cooled steam and evaporator fluid consistently under 5 K can be observed. This implies high heat transfer under various working conditions which justifies the steady state modeling. For the definition of heat transfer efficiency a constant terminal temperature difference is set.

3.3.4 Super Heater

To bring steam temperature up to turbine inlet conditions the steam exiting the evaporator is entering a series of two super heaters. Super heater 1 will bring the temperature on a level below life steam temperature. Following the first super heater the steam enters the steam cooler where the temperature of the inlet stream in the second super heater can be controlled as described in section 3.3.3, thereby regulating life stream temperature. Low fluid content and thin-walled plain tube bundles allow the assumption of low thermal inertia within the component and therefore a steady-state model. The following set of equations is applied.

$$0 = (\dot{m}h)_{in,Air} - (\dot{m}h)_{out,Air} - \dot{Q}_{SH} \tag{3.30}$$

$$0 = (\dot{m}h)_{in,H20} - (\dot{m}h)_{out,H20} - \dot{Q}_{SH} \tag{3.31}$$

$$0 = \dot{Q}_{\text{SH}} - k^{*} A \Delta T \tag{3.32}$$

$$0 = k - f(\alpha_{\text{Air,W}}, \lambda_{W}, \alpha_{\text{W,H20}}, Geo) \tag{3.33}$$

$$0 = k^{*} - c_1 k \dot{m}_{\text{in,Air}}^{c_2} \tag{3.34}$$

The heat transfer coefficient k is calculated based on mean inlet conditions. It therefore does not correlate dynamically with inlet condition and simulation results. To keep the use of material property correlations within the model low, a simplified heat transfer based on the air mass flow is chosen and fit to operational data. An implicit formulation of temperature difference is chosen to avoid an additional interdependence of equations within the component. The temperature difference is chosen as follows:

$$\Delta T = T_{\text{in, Air}} - T_{\text{in, H2O}} \tag{3.35}$$

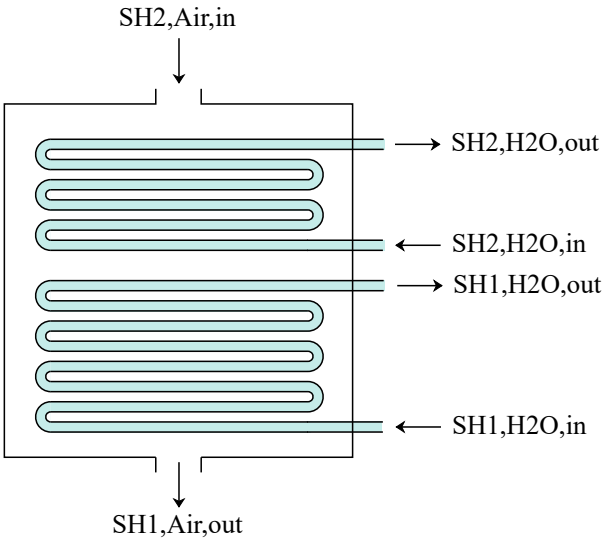


Figure 3.2: Schematic illustration of the super heater

The super heater component at the STJ consists of one continuous air section with two separate heat exchangers between air and steam. Consequently it has one air inlet stream, one air outlet stream two steam inlet streams and two steam outlet streams. A Schematic illustration of the component can be found in fig. 3.2. The super heater on the lower temperature level is called super heater 1 and on the higher temperature level is called super heater 2.

3.3.5 Steam Turbine

The steam turbine at the STJ is expanding the steam on two pressure levels. Turbine extractions are not conducted and no measurements at the outlet of the first pressure level are available. Only input and output values of the turbine group are considered. The steam turbine is connected with a turbine control valve (TCV) system. Objective of the model is to accurately determine electricity production and correlation between control valve position and turbine outputs. The following set of equations has been used to model the steam turbine.

$$\frac{dT_W}{dt} = \frac{1}{m_W cp_W} \dot{Q}_{\mathrm{H2O,W}} \tag{3.36}$$

$$0 = P - \eta_{\mathrm{mech}} \eta_{\mathrm{el}} \left((\dot{m}h)_{\mathrm{in,H2O}} - (\dot{m}h)_{\mathrm{out,H2O}} - \dot{Q}_{\mathrm{H2O,\,W}} \right) \tag{3.37}$$

$$0 = \dot{Q}_{\mathrm{H20,W}} - (\alpha A)_{\mathrm{H20,\,W}} \Delta T \tag{3.38}$$

$$0 = \Delta T - (T_{\mathrm{H2O}} - T_W) \tag{3.39}$$

$$0 = \frac{\dot{m}_{\mathrm{in,H20}}}{\dot{m}^{ref}_{\mathrm{in,H20}}} - f_{\mathrm{Stodola}} f_{\mathrm{TCV}} \tag{3.40}$$

$$0 = f_{\mathrm{Stodola}} - \frac{p_{\mathrm{in,H2O}}}{p^{ref}_{\mathrm{in,H2O}}} \sqrt{\frac{1 - \left(\frac{p_{\mathrm{out,H2O}}}{p_{\mathrm{in,H2O}}} \right)^2}{1 - \left(\frac{p^{ref}_{\mathrm{out,H2O}}}{p^{ref}_{\mathrm{in,H2O}}} \right)^2}} \sqrt{\frac{T^{ref}_{\mathrm{in,H2O}}}{T_{\mathrm{in,H2O}}}} \tag{3.41}$$

$$0 = f_{\mathrm{TCV}} - (3S_{\mathrm{TCV}}^2 - 2S_{\mathrm{TCV}}^2) \tag{3.42}$$

$$0 = \eta_s - \frac{h_{\mathrm{in,H2O}} - h_{\mathrm{out,H2O}}}{h_{\mathrm{in,H2O}} - h_{\mathrm{s,out,H2O}}} \tag{3.43}$$

To describe flow characteristics within the turbine the Stodola equation, f_{Stodola}, has been implemented. It correlates the mass flow through the turbine in dependency on inlet and outlet pressure, inlet temperature and the respective reference values during full operation. The Stodola equation was proposed by Stodola [73] and a derivation to the equation can be found in Kestin [74]. The Stodola equation is augmented with a term including the valve position of the TCV. A polynomial correlation for a single control valve with valve position S_{TCV} as proposed by Grote [75] has been chosen.

The driving temperature difference for the heat transfer between steam and the turbine casing wall has been formulated explicitly for this model to reduce computational complexity and reduce the number of interdependencies within the model, see eq. (3.39). A more realistic temperature difference would include the outlet temperature of the turbine, to more accurately describe the heat flow. It has shown, that with means chosen to simulate the components, convergence is often hindered by an implicit formulation of temperature difference while the impact on the solution remains small. For a more comprehensive model, different methods of solving the DAE system could be considered in order to include and implicit formulation.
In this model the isentropic efficiency of the turbine η_s is set constant while in reality it is a function of the steam mass flow. A more precise off design model can be formulated through the approximation of fundamental equations [76]. Purely empirical descriptions as well as a combination of the two have been proposed for similar applications.[69][77].

4 Results and Validation

In this section, the results of the modeled components are compared to real plant data. In order to keep discretion over actual values of the STJ, the results are displayed in a normalized form. Therefore all figures display values in a dimensionless form. The simulation results of the storage in different modes of operation are presented and discussed. The models for each component rely on the accuracy of certain parameters to achieve satisfying results. In order to increase model accuracy, chosen parameters have been adapted in comparison with a training data set, and the improved models have been validated against additional data sets obtained on different operational days. A simulation is carried out with initial values chosen on the bases of the first time step of the observed validation time span. For the storage, the evaporator, the steam turbine, and the heat exchangers, more than one parameter was fitted, leading to a non-linear optimization problem. These non-linear optimization problems lead by their nature to local minima, which can, but not necessarily must, be global minima. The results highly depend on starting conditions and the chosen optimization algorithm.

In this work, the sequential least squares programming (SLSQP) algorithm, implemented in *SciPy* [78], and the moving horizon estimation (MHE), implemented in *do-mpc*[62], have been used in order to determine a combination of parameters sufficiently capable of replicating the plants' measured data. In section 2.5.4, it is explained how the parameter estimation with SLSQP is conducted, and in section 2.5.2 the parameter estimation with MHE is explained. The MHE can find a minimum using a wide range for parameters while still maintaining low computational cost. Furthermore, it is able to include combinations of parameters which would lead to a failure in convergence of the simulation when implemented as constant parameters. Some parameter combinations can cause e.g. too high gradients or a division by zero when solving the system of equations. These combinations of parameters might not be clear when choosing possible ranges for parameter estimation, and so it is advantageous for MHE to be able to include these combinations, even though they will not be a solution. The accuracy of the parameters can be further improved by performing a minimization of the error between measurement and results using an SLSQP algorithm implemented by *SciPy*. Here, the computational expense increases drastically with high ranges of parameters and the problem of convergence failure translates into the optimization process, as the model is solved multiple times with different combinations of parameters. The results of the MHE have been chosen as starting values for optimization and small ranges of variation have been chosen around them.

4.1 Thermal Storage Validation

The thermal storage of the solar tower was modeled according to the equations and explanations given in section 3.2. The heat transfer coefficients between storage and ambient environment $\alpha_{\mathrm{St,Amb,i}}$, the heat capacity of the storage media cp_{St} and the heat conductivity

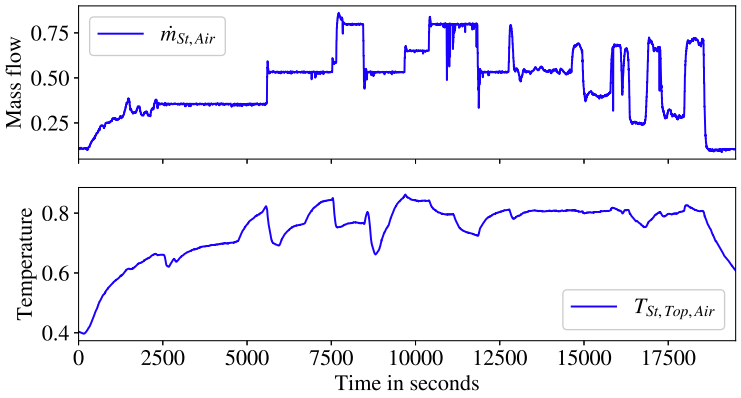


Figure 4.1: Input variables $\dot{m}_{\mathrm{St,Air}}$ and $T_{\mathrm{St,Top}}$ of the thermal storage during charge.

of the storage media λ_{St} have been selected for the fitting process. As starting values for the optimization problem, material properties of the ceramic filling material have been used. It is to be expected that the adapted values will differ due to the porous structure of the storage material.

The temperature in the storage is measured at eight different vertical positions with three thermal elements at each measured layer. The mean of the measurement of these thermal elements has been used in the fitting process and is compared to the simulated temperature at the respective position. The simulation treats the temperature horizontally as uniform. The input variables in the model are the air mass flow entering the storage $\dot{m}_{\mathrm{St,Air}}$ and the inlet air temperature at the top $T_{\mathrm{St,Top}}$ or bottom temperature $T_{\mathrm{St,Bottom}}$. In this context, a positive mass flow means the storage is being charged with hot air entering the storage from the top and cold air exiting it at the bottom. In analogy, a negative mass flow corresponds to the storage being discharged with cold air entering at the bottom and hot air exiting at the top. Starting values have been chosen to be coherent with the measured values.

4.1.1 Charge

In this section, simulation results of the thermal storage during charge are presented. Simulation results will be analyzed and validated against plant data. In this operational mode, the volumetric receiver is providing substantial amounts of hot air and it's majority is passed through the storage in order to heat up the ceramic material. In fig. 4.1, the input variables $\dot{m}_{\mathrm{St,Air}}$ and $T_{\mathrm{St,Top}}$ are depicted over time. In fig. 4.2, the measured and simulated temperature curves of four chosen layers over time are depicted.

In this data set, the storage starts out in a cooled down status. The temperature of the storage is low over the entire height of the storage and the temperatures are in a close range. At the beginning of the observed time period, the mass flow into the storage and the inlet temperature are low and start rising. With the inlet temperature being below storage temperature and a positive mass flow, a decline of storage temperature in the upper levels can be observed. Once

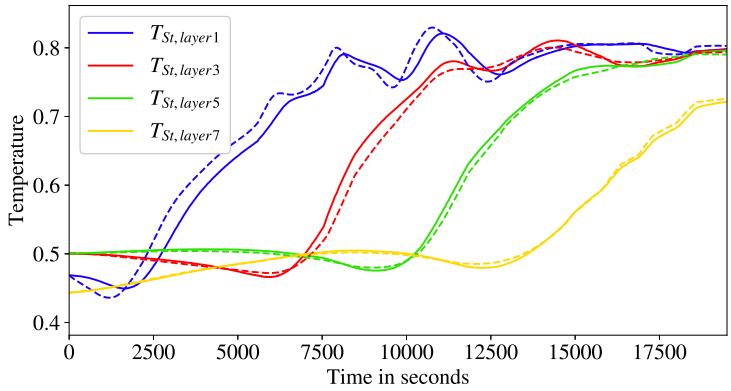


Figure 4.2: Comparison of measured and simulated temperature curves of selected layers in the thermal storage during charge. Measured data (——) and simulated data (- - -).

inlet temperature rises above the temperature of the first layer, it's temperature starts rising and with corresponding time differences the lower levels follow. The simulated temperature of the highest layer reacts slightly faster to changes in the input variables. The simulated temperature of layer 3 reacts slightly slower to changes in the input variables. Changes in mass flow and inlet temperature result in different temperature gradients. When drops in inlet temperature occur, like in the time interval between 8000 s and 10 000 s, see fig. 4.1, the upper levels of the storage follow the inlet temperature. At the lower levels however, the effect blurs out. The root mean square deviation (RMSD) between measured and simulated storage temperature during charge is 7,72 K and the normalized root mean square deviation (NRMSD) is about 2%

It can be observed that the simulation of the first layer's temperature corresponds faster to the inlet conditions than the measured temperature. The highest position of measurement is only ambiguously to be identified by construction drawings or from the outside of the thermal storage. A linear course between inlet air temperature and the measurement of the first measurement inside the thermal storage is assumed. Through comparison with measured data, the positioning has been chosen for which the measurement corresponds to the simulation. This approach can lead to an error in positioning as it is not clear that the temperature gradient between the first measurement and the air temperature measured at the inlet is constant. It could be a possibility that the first measurement is in reality constructed farther below than assumed in the simulation.

Another explanation can be the small number of simulated layers between inlet and the simulated layer paired up with the first measured layer. A possible way of improving that deviation can be the inclusion of a heat transfer coefficient between air and storage material in eq. (3.7). This would deviate from the One-Temperature-Model proposed in Gall and lead to the necessity of solving partial differential equations [22]. With that term implemented, a raise in number of layers would result in a convergence to a solution and the upper layers would not react as drastically to changes of inlet conditions. For this work, the simplicity

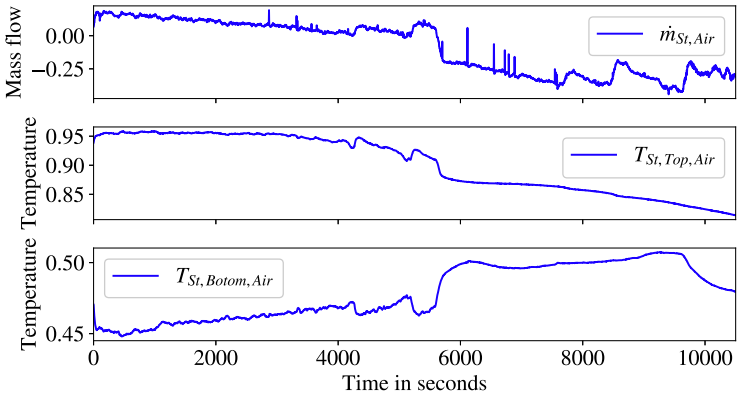


Figure 4.3: Input variables $\dot{m}_{\mathrm{St,Air}}$ and $T_{\mathrm{St,Top}}$ of the thermal storage during full operation with storage usage.

of the model is emphasized as the aim is to provide a model suitable for model predictive control and therefore online optimization. The inclusion of partial differential equations would lead to a coupled system of partial differential equations and ordinary differential equations. That would require further numerical effort like the inclusion of the Finite Element Method thereby increasing computation power requirements while decreasing numerical stability. The accuracy of the chosen model however is adequate for its use in this work. For improvement, this approach could be implemented with a careful selection of numerical solvers and their integration in the overall model.

4.1.2 Full Operation

During full operation, the thermal storage can be used to level the inlet temperature and mass flow into the water steam cycle. For efficient operation, it is desirable to achieve constant, high inlet temperature into the steam cycle steam generation. In case of a weather change like bypassing clouds, a mass flow of cold air can be directed from the bottom of the thermal storage to its top. The air is exiting at the top of the storage close to the temperature of the highest layer adding to the mass flow of hot air flowing towards the steam generation. The fig. 4.4 shows the temperature curve of selected layers in the thermal storage during full operation with storage usage while fig. 4.3 shows the corresponding input variables.

A rise in temperature in the upper layers of the storage can be observed until up to 5500 s with a decreasing gradient, see fig. 4.4, in both measurements and simulation results. The storage levels below layer 3 stay constant over the mentioned time interval. From that moment on, the temperature inside the storage decreases for both measurement data and simulation results in the upper layers. Close to the 5500 s mark, the mass flow passes from positive to negative values. A decrease of temperature in the upper storage levels as well as the decrease of temperature at the top of the storage to storage temperatures can be observed. The simulated values for the lowest layer, layer 8, start rising up to the temperature on the bottom of the storage at 6500. The measured values for layer 8, start rising up to the temperature on the bottom of the storage at about 7000. The RMSD between measured and simulated

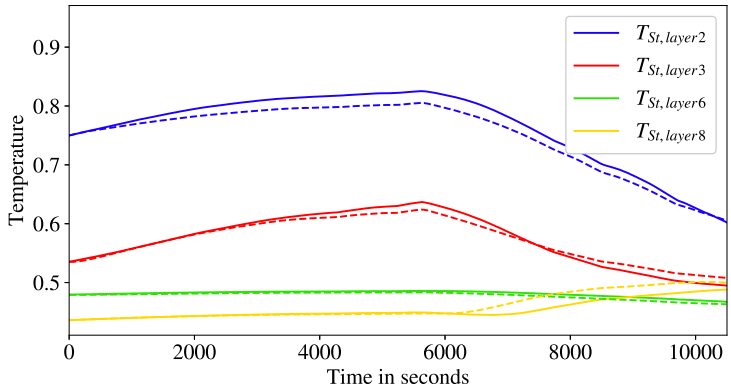


Figure 4.4: Comparison of measured and simulated temperature curves of selected layers in the thermal storage during full operation with storage usage. Measured data (——) and simulated data (-·-·).

temperatures inside the storage during full operation is 12,75 K and the NRMSD is about 3%.

The positive gradient up to 5500 s is the result of a positive air mass flow with high inlet temperature at the top. When the mass flow switches from positive to negative, the input variables switches from $T_{\mathrm{St,Top}}$ to $T_{\mathrm{St,Bottom}}$. Now the inlet to the storage is the temperature measured below the lowest storage level and the inlet to the storage is the lowest level. The temperature $T_{\mathrm{St,Top}}$ becomes a result. The lowest levels do not change in temperature over the entire interval, which is explained by two effects. Before 5500 s the thermocline has not reached the lower levels and the hot air charging the storage, has transferred its heat to the upper levels of the storage already. When the direction of the mass flow changes, the input variable becomes the temperature on the bottom of the storage. This temperature is now influenced by the outlet temperature of the steam generator, rather than the storage temperature of the lower levels. The increase of the measured temperature at the bottom is caused by the warm air from the outlet of the steam generator, which is now directed into the storage. This warm air causes the lower levels of the storage to slightly increase their temperature. This effect can be observed in the temperature curve for layer 8 in fig. 4.4. The faster reaction of the simulated storage temperatures to changes in input variables of the layer closest to the inlet than measured can be explained by the same effects mentioned in section 4.1.1. With slight deviation, the implemented model is able to replicate the real storage behavior during full operation when the storage is in use.

In fig. 4.6, we can observe the temperature curves in selected storage layers during full operation while the storage is not in use, meaning no mass flow is being directed through the storage either charging or discharging the thermal storage. The corresponding input variables can be found in fig. 4.5 showing a mass flow oscillating with small amplitudes around the value zero. These fluctuations can be caused by measurement dynamics or a controller circle resulting in a oscillation. During this specific operational state, large quantities of hot air are passing by the top end of the storage, however, none of it is directed through the storage. An

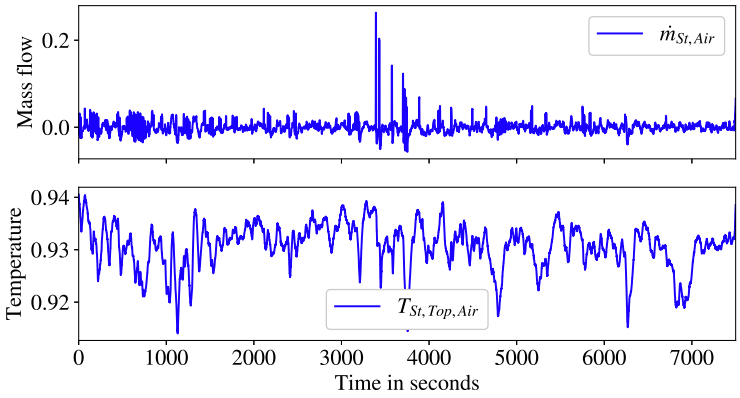


Figure 4.5: Input variables of the thermal storage during full operation without storage usage.

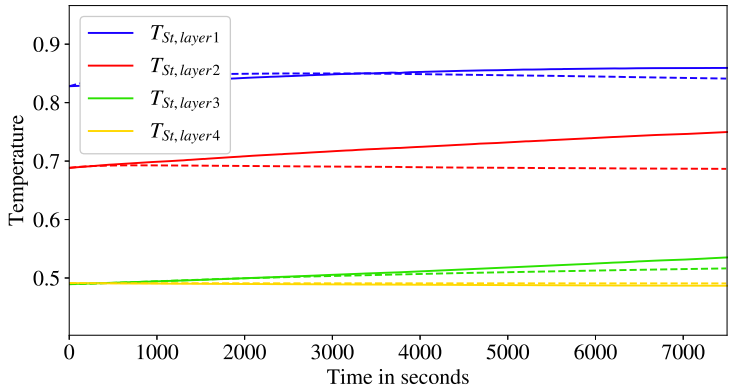


Figure 4.6: Comparison of measured and simulated temperature curves of selected layers in the thermal storage during full operation with storage usage. Measured data (——) and simulated data (-·-·-·).

increase in the upper layers temperature can be observed in the operational data. The simulated storage temperatures of the layer close to constant. The simulated temperature of the third layer rises slightly over time.The RMSD between measured and simulated temperatures inside the storage with no mass flow measured inside the thermal storage is 11,03 K and the NRMSD is about 3%.

A possible explanation for this effect could be an insufficient accuracy of the mass flow sensors. The mass flow through the thermal storage is computed by subtracting the mass flow through the steam generator from the mass flow through the solar receiver. In this scenario, a small mass flow of hot air would pass through into the storage, heating up the ceramics while no mass flow is being recorded. When this computed mass flow close to zero is considered, the model is not able to reproduce the behavior as no mass flow leads to no heat transfer through convection. For comparison, see eq. (3.7). The slight increase of the lower levels, represented by layer 3 in fig. 4.6, is caused by the thermal conduction within the storage. The upper levels are higher in the simulation as in the operational data, causing the conduction term to show its effect. The effect is stronger for the measured data, as the temperature of the upper levels rise.

4.1.3 Bootstrap Method for Estimated Storage Parameters

In this section, the bootstrap method for the estimated parameters within the model of the thermal storage is performed. The procedure for the implementation of the bootstrap method is explained in section 2.5.5. The bootstrap method in this component is applied exemplary for the other components. It can be conducted in analogy.

The parameters in this component that have been adjusted to fit the operational data are the heat transfer coefficients between storage and ambient environment $\alpha_{\mathrm{St,Amb,i}}$, the heat capacity of the storage media cp_{St} and the heat conductivity λ_{St}. The effects of an uncertainty in the temperature measurements inside the storage as well as an uncertainty of the mass flow through the storage and its effect of the estimated parameters is conducted. The starting points for the parameter estimation have been constant for each iteration and have been chosen according to realistic thermodynamic properties of the storage material.

A figure illustrating the estimated parameters $\alpha_{\mathrm{St,Amb}}$, cp_{St} and λ_{St} over measurement uncertainty of temperature sensors and the mass flow inside the storage can be found in fig. 4.7 and fig. 4.8 respectively. Confidence intervals for $\sigma \pm 2$ have been included into the figure. The slope of the line c_σ describing the confidence interval of the parameter θ with respect to the uncertainty δ has been derived by following the approach below. In the equation below, the original parameter estimation is denoted as μ.

$$2\sigma = f(\delta) = \pm c_\sigma \delta \tag{4.1}$$

$$\sigma^2 = \frac{1}{K}\sum_{0}^{K}\frac{1}{\delta_i}(\theta_i - \mu)^2 \iff c_\sigma = \pm\sqrt{2^2\frac{1}{K}\sum_{0}^{K}\frac{1}{\delta_i}(\theta_i - \mu)^2} \tag{4.2}$$

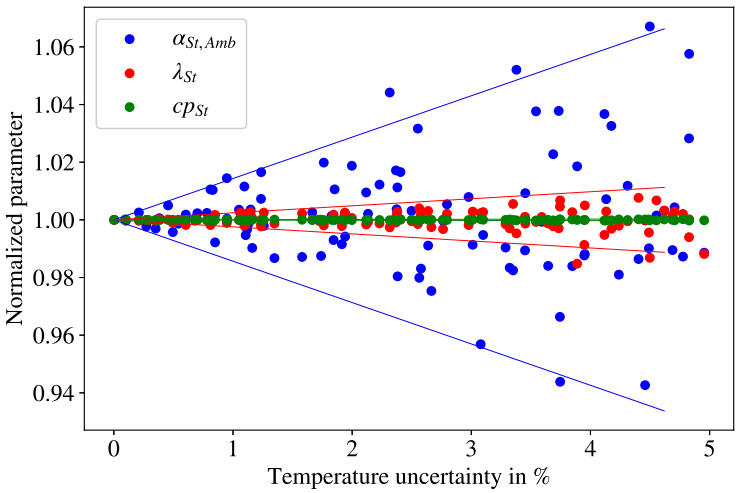


Figure 4.7: Estimated parameters $\alpha_{\mathrm{St,Amb}}$, λ_{St} and cp_{St} for $K = 100$ over uncertainties of thermo meters in the thermal storage. Straight lines marking the confidence interval of $\sigma = \pm 2$

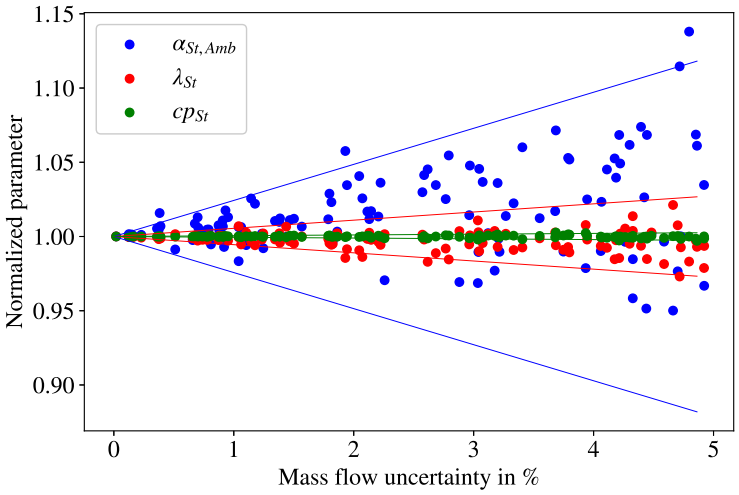


Figure 4.8: Estimated parameters $\alpha_{\mathrm{St,Amb}}$, λ_{St} and cp_{St} for $K = 100$ over uncertainties of the mass flow through the thermal storage. Straight lines marking the confidence interval of $\sigma = \pm 2$

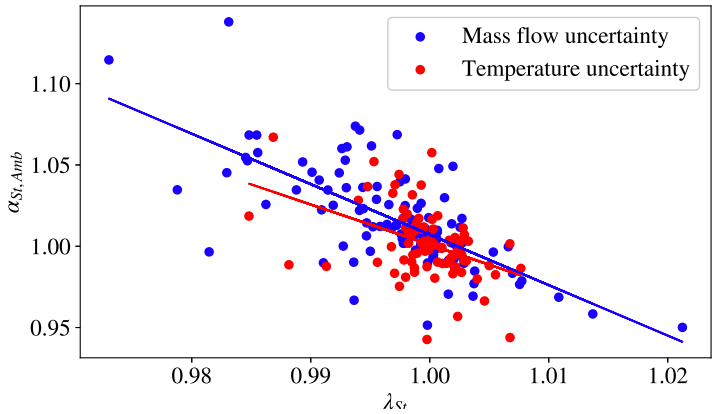


Figure 4.9: Correlation between heat transfer of the storage to the ambient $\alpha_{\mathrm{St,Amb}}$ and thermal conduction λ_{St} for temperature uncertainty (red) and mass flow uncertainty (blue).

For each of the three parameters, it can be observed that the confidence interval is becoming larger with higher uncertainty of the measured variables. The variation of $\alpha_{\mathrm{St,Amb,i}}$ is larger than the variation of λ_{St}, while λ_{St} is larger than cp_{St}. This is the case for both mass flow and temperature uncertainty. The confidence intervals dependent on the mass flow uncertainty are wider in comparison to the temperature uncertainty for all three parameters. In case of the mass flow uncertainty, found in fig. 4.8, it can be observed that the estimated parameters are not symmetrically distributed around the original value. The heat transfer coefficient is systematically over estimated. The thermal conduction parameter is estimated lower than the original value. A high estimated heat transfer corresponds with lower estimated heat conductivity and vice versa. This phenomenon can be seen more clear in fig. 4.9. In this graphic a straight line with the lowest squared error is depicted with the data points. The data in fig. 4.9 qualitatively indicates a correlation.

The larger confidence intervals for larger uncertainty is to be expected as the parameters are fitted to a broader range of measured variables. The confidence interval of heat conduction cp_{St} is close to insensitive to measurement uncertainty in both temperature sensors and mass flow. The highest inaccuracies caused by uncertainty are to be expected in the estimation of the heat transfer coefficient.

In the case of mass flow uncertainty, the majority of estimated heat transfer coefficients lies above the original value with the majority of heat conduction parameters lying below the original value correspondingly. For temperature uncertainty the parameters are estimated symmetrically around the original parameter. For temperature uncertainty this indicates, that the original parameter used in the boot strap method is close to the mean of the distribution. In case of the mass flow uncertainty there is a discrepancy between mean of the distribution of estimated parameters in the boot strap method and the original parameter. This might indicate that the original parameter describing the heat transfer, is in fact higher, than the original estimation, while the heat conductivity is in fact lower.

A possible explanation for the alpha-lambda correlation is given in this paragraph. The heat transfer between storage and its environment is stronger at the hotter, upper levels of the thermal storage and less energy needs to be transported within the storage to maintain the right temperature in this section of the storage. Due to the smaller temperature difference between wall and ambient at the lower levels, the higher heat transfer coefficient is less significant and the heat loss does not need to be balanced by the energy received through conduction in order to be coherent with measured data. Additional effects might be in play.

The accuracy of thermometers at the relevant temperature levels installed at the STJ is denoted with ±0.5% which accounts for the parameters λ_{St} and cp_{St} with a variance of less than 1% deviation from the original parameter. The confidence interval for heat transfer lies at the given uncertainty on about 1% deviation from the original parameter. For the mass flow uncertainty the operational data shows fluctuations of up to 3.5%. For cp_{St} the effects are insignificant which allows the conclusion that this parameter is not fazed by changes in mass flow nor temperature uncertainty. The confidence interval of λ_{St} lies at about 1% deviance from the original parameter at 3.5% mass flow uncertainty. The confidence interval of α_{St} lies at about 5% deviance from the original parameter at 3.5% mass flow uncertainty. This means that the measurement inaccuracy of the mass flow has a significant influence on the accuracy of the parameter estimation and this needs to be considered when assessing the accuracy of this model.

4.2 Preheater Validation

In this section, the simulation results of the preheater are presented and discussed. The model presented has been fit to operational data by adjusting the heat transfer coefficients $\alpha_{\mathrm{Air,\,W}}$ and $\alpha_{H2O,W}$. They have been assumed as independent of mass flow and temperature, which is why they remain constant throughout the simulation. This decision is taken as the temperature levels at the inlet and outlet remain in a narrow range during the operation. The input variables into the preheater are found in fig. 4.10 and the simulation results together with the corresponding operational data is found in fig. 4.11.

The outlet air temperature follows the data with a constant difference up to 7500 s. The measured outlet temperature then drops with the simulation result not reaching the minimum of that dip. It stays near to constant up to the change in mass flow at 13 500 s where it rises and the constant difference between measurement appears again. The simulated outlet water temperature fluctuates with a high frequency around the measured temperature. The mean of the simulation temperature rises in comparison to the measured value during the change in air mass flow. It then continues to correspond to operational data. The RMSD between measured and simulated temperatures inside the preheater is 5,33 K and the NRMSD is 0.8%.

As changes in mass flow are not considered in the heat transfer coefficient, high frequency fluctuations are translated into the outlet temperature. The air temperature is systematically higher than measured. When the deviance between simulated and measured air temperature decreases during the change in air mass flow, the deviance between simulated and measured

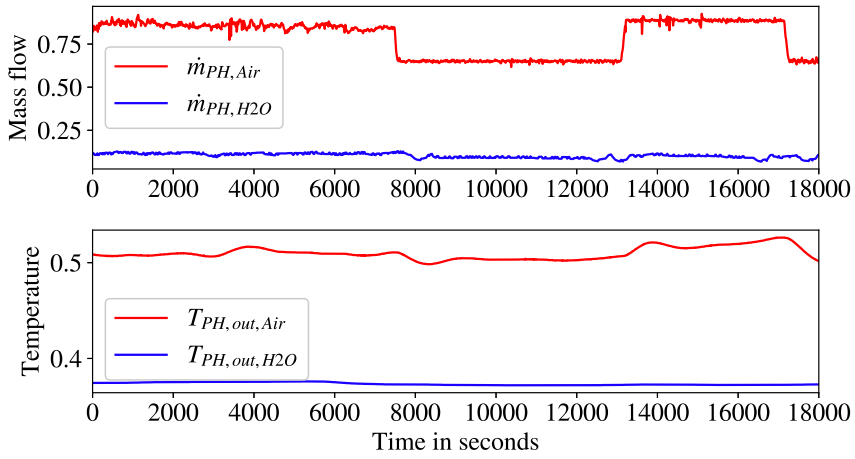


Figure 4.10: Input variables of the preheater model.

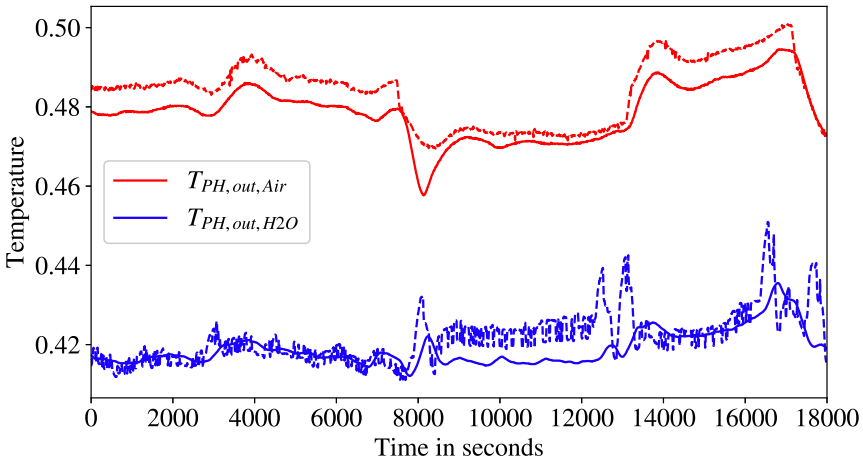


Figure 4.11: Simulation results of preheater plotted with operational data. Measured data (——) and simulated data (-·-·).

water temperature increases with the simulated outlet water temperature becoming higher. This indicates that heat loss to the environment is relevant in the energy balance. A corresponding term could be included in the components model to mitigate this effect.

The thermal inertia in this component is assumed to be in the wall between the two media. This simplification contributes to inaccuracies as a substantial amount of thermal inertia is caused by the water inside the preheater and the casing of the component as well. A more accurate description is the separate balancing of inner energy in each media. This would lead to a system of partial differential equations which is avoided in this study for reasons of model reduction efforts.

The assumption of a constant heat transfer coefficient has been tested with several data sets of different operational days. It has been shown that for all regular modes of operation the assumption is as accurate as for the chosen data set. Strong changes in the water mass flow during start or shut down or that caused by irregularities lead to large spikes in the simulated outlet water temperature which are however not observed in reality. The RMSD does not however, exceed 11 K for any observed data set.

4.3 Economizer Validation

In this section, the simulation results of the economizer are presented and discussed. The model presented has been fit to operational data by adjusting the heat transfer coefficients $\alpha_{\mathrm{Air,\,W}}$ and $\alpha_{H2O,W}$ in the following manner.

$$0 = \alpha^*_{W,H2O} - c_1 \dot{m}^{c_2}_{\mathrm{H2O}} \tag{4.3}$$

$$0 = \alpha^*_{Air,W} - c_3 \alpha^*_{W,H2O} \tag{4.4}$$

To fit to data in this component, the SLSQP method has been applied to address the optimization problem as the results have shown to be more accurate than the MHE in this particular case. This could be caused by a good initial guess for the estimated parameters derived from correlations listed in section 3.3. The input variables for the simulation can be found in fig. 4.12 and the results of the simulation together with the corresponding plant data can be found in fig. 4.13.

The water outlet temperature accurately follows the operational data. The calculated air outlet temperature is higher than the measured one and also experiences significant fluctuations. Especially during the change in air mass flow at around 7500 s, the changes in temperature are overemphasized in the simulation compared to the measurements. Additionally, the high frequency oscillation of the air mass flow is translated into the outlet air temperature of the simulation, which is not observed in the measurements. The air outlet temperature spike caused by the fluctuation in mass flow at 3500 s can be clearly observed in measured data and simulated data alike. It is pronounced more strongly in the simulation as the heat transfer is not dependent on the air mass flow which can be seen in eq. (4.3) and eq. (4.4). A similar event can be observed at 7500 s with the simulated temperature under shooting the measurements.

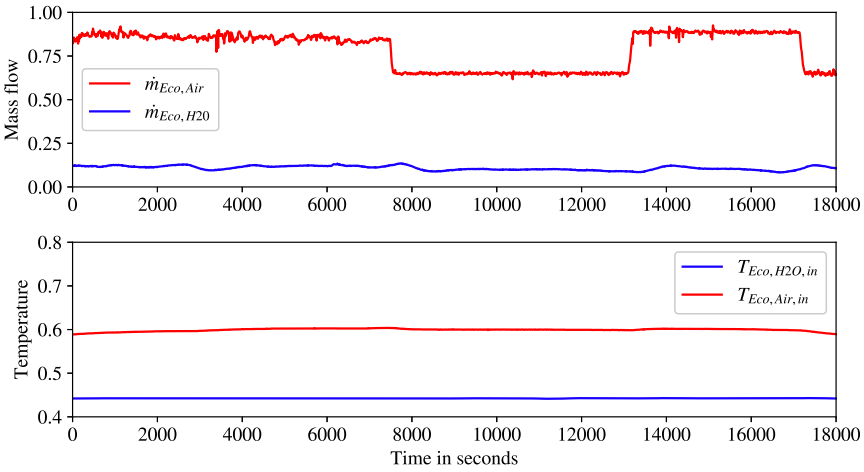


Figure 4.12: Input variables of economizer model.

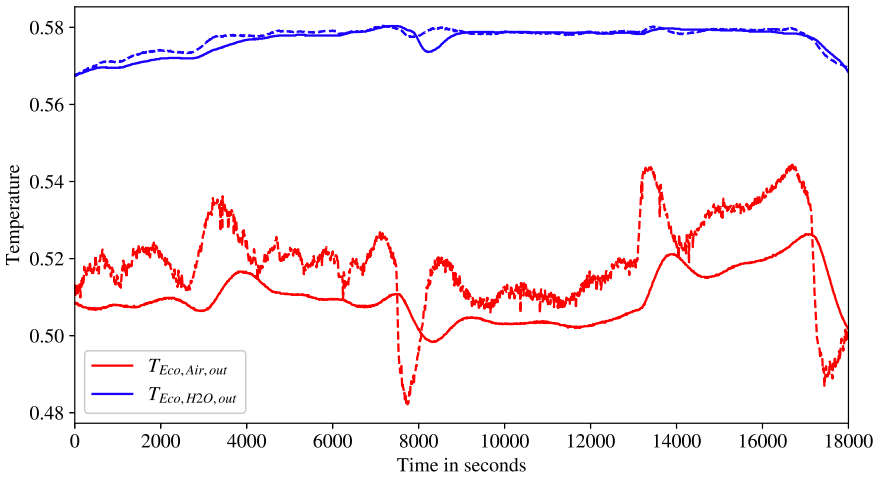


Figure 4.13: Simulation results of economizer plotted with operational data. Measured data (——) and simulated data (- - -).

It can be observed, that the simulated outlet air temperature is consistently higher than the measured temperature. The RMSD between measured and simulated temperatures inside the storage during full operation is 7,46 K and the NRMSD is 1.7%.

The inaccuracy of the model can be caused by the simplification of the driving temperature difference to an explicit form. In reality, a higher mass flow would result in a higher heat transfer during these events and the temperature would therefore be less sensitive to input changes. A similar effect can be observed at 7500 s. The decrease in mass flow results in an over proportional decrease in temperature. The heat transfer would reduce with the driving temperature difference becoming smaller, thereby slowing the rise in temperature. The temperature rises again when the water mass flow adapts to the controls of the power plant and the heat transfer is lowered, resulting in a decreased water outlet temperature and an increased outlet air temperature. The effect can be also observed by the air mass flow changes at 13 500 s and 16 500 s. A heat transfer correlation dependent on the air mass flow could mitigate these effects. That would however, require an additional parameter in the optimization problem leading to an additional non-linearity.

Just like in the preheater unit, the thermal inertia in this component is assumed to be in the wall between the two media. This simplification contributes to inaccuracies as a substantial amount of thermal inertia is caused by the water inside the economizer and the casing of the component as well. A more accurate description is the separate balancing of inner energy in each media. This would lead to a system of partial differential equations which is avoided in this study for reasons of model reduction efforts.

The fact that the simulated temperature is consistently higher than the measured results, excluding the mentioned effects at mass flow changes, points to a missing heat sink in the component's model. A heat loss to the environment can be the cause of this observation. To include this effect in the model, a corresponding term would need to be implemented. An additional approach to increase the model's accuracy would be the inclusion of T_{H20} and T_{Air} into the optimization process. A simple linear implementation could be of the following form.

$$T_{H20} = c_4 T_{in,H20} + (1 - c_4) T_{out,H20} \tag{4.5}$$

$$T_{Air} = c_4 T_{in,Air} + (1 - c_4) T_{out,Air} \tag{4.6}$$

This can help reduce the strong fluctuations of the outlet temperature as the driving temperature difference would influence the amount of heat that is being transferred. This however would include an additional interdependence within the DAE system to be aware of.

4.4 Evaporator Validation

In this section, the results of the evaporator simulation are presented and discussed. During the fitting process, a MHE has been used to identify a suitable combination of the evaporator's mass relevant for thermal inertia and a simplified equation for the heat transfer. The complex dependencies of water and steam properties within the heat transfer correlations presented in section 3.3 have been shown to increase numerical instability of the simulation. To circumnavigate these issues and to keep the calculation effort low, the rather complex equations for

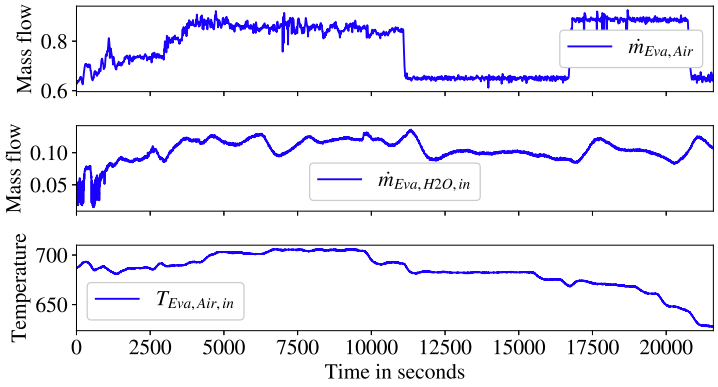


Figure 4.14: Input variables of the evaporator model.

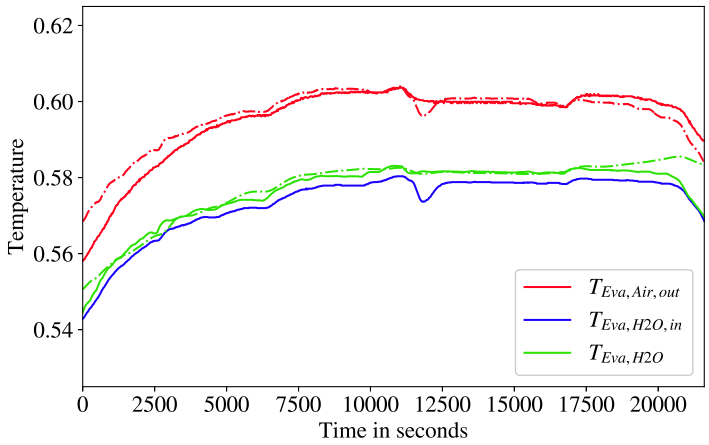


Figure 4.15: Temperature curves over time of outlet temperatures of air and steam and the temperature inside the evaporator. Measured data (——) and simulated data (-·-·).

the calculation of the heat transfer coefficients, see section 3.3, have been replaced by a more simple correlation only dependent on the air mass flow. For the heat transfer, an equation of the form

$$k = c_1 \dot{m}_{Eva,Air}^{c_2} \tag{4.7}$$

has been implemented with the two coefficients c_1 and c_2 being subject to parameter optimization. The correlation found in section 3.3 has been used to identify suitable starting values for parameter fitting. In supplemental fig. 4.15, the temperature curves of the outlet mass flows as well as the temperature inside the evaporator are depicted. The temperature value of the plant data has been derived from the pressure measurement inside the evaporator as phase equilibrium inside the evaporator is assumed. The accompanying input variables can be found in fig. 4.14. In fig. 4.15 the input variable $T_{Eva,in,H2O}$ is depicted together with the results for a better comparison.

The time frame for the validation shows variations in mass flow and inlet temperature of a usual operational day and has therefore been chosen to demonstrate the model's ability to follow the component's dynamics. The measured and the simulated temperature of the air at the outlet of the evaporator $T_{Eva,Air,out}$ rises with decreasing gradient up until 10 000 s. It experiences a slight drop for measured data and simulated data alike at around 12 000 s and a slight increase at 17 000 s it then decreases with increasing gradient. The simulated temperature follows the measured data closely with small deviations at the beginning and at the end of the interval as well as during the fluctuation in water inlet temperature around second 12000. The measured evaporator temperature T_{Eva} rises up until 10 000 s. It experiences a slight drop for measured data and simulated data alike at around 12 000 s and a slight increase at 17 000 s. It then decreases with increasing gradient. The simulated temperature rises sharp during the first 1000 s and continues to rise with a lower and further decreasing gradient. At around 19 000 s it experiences an increase followed by a temperature drop. The RMSD between measured and simulated temperatures inside the evaporator is 8,05 K and the NRMSD is 1.1%.

The evaporator temperature T_{Eva} never falls below the water inlet temperature and never rises above the air temperature outlet in both measured and simulated data. It thereby stays in a plausible temperature range throughout the given time span. The discrepancy of $T_{Eva,Air,out}$ at the beginning and at the end of the interval can be explained by the heat transfer being not only dependent on the air's mass flow but also on the water's mass flow as well. Flow characteristics in relation to the mass flows will have impact on the heat transfer inside the evaporator as well. Put simply, it can be assumed that with higher mass flows and constant diameters, higher velocities and thus increased heat transfers should be observed. With lower water mass flows $\dot{m}_{Eva,H2O,in}$ at the beginning of the interval and higher mass flows at the end, the heat transfer will also be lower at the beginning of the interval and higher at the end. Therefore, the simulated temperature difference at the beginning of the interval will be higher than the measured one. In analogy, at the end of the interval, it will be smaller.

The driving temperature in this model has been calculated in an explicit form as shown in eq. (3.20) to avoid a recursive loop in the solution of the DAE system. The logarithmic temperature difference has been avoided by determining a mean between air and water inlet temperature. This method differs from reality. A logarithmic temperature difference which is also dependent on the outlet temperature of the air $T_{Eva,out,Air}$ would describe the driving temperature difference more accurately. With the complex system of equations implemented in the two-phase model, an implementation of a logarithmic difference would not have been feasible as no convergence to a solution was achieved with the means of solving the DAE system that have been applied in this particular simulation. The chosen explicit formulation of the driving temperature difference amplifies the effect of the inlet temperature of the water $T_{Eva,in,H2O}$ on the heat transfer. This could explain the high sensitivity of the simulated outlet air temperature $T_{Eva,Air,out}$ to the water inlet temperature.

A possibility of improving the model's accuracy following that simplified approach and keep the explicit formulation can be the inclusion of a factor c_3 weighing the inlet temperatures into

the parameter optimization. The temperature difference can then be written in the following form.

$$\Delta T = (c_3 T_{Eva,in,air} - (1 - c_3) T_{Eva,in,H2o}) - T_{Eva} \tag{4.8}$$

The deviance between the simulated and measured value can be explained by the made assumptions laid out earlier in this section and the assumptions of a perfect equilibrium made to set up the system of equations of the two-phase model. The deviation is however acceptable and in the range of only a few Kelvin. The improvement in numerical robustness to different sets of input variables are valued over the loss in accuracy and the model is able to reproduce the component's dynamic sufficiently well.

4.5 Super Heater Validation

In this section, the results of the two super heater simulations are presented and discussed. To fit the simulation results to the operational data, a MHE has been performed to adjust the heat transfer coefficient. By altering the constants c_1 and c_2, the heat transfer and its dependency on the air mass flow is fit to operational data using the following equation.

$$0 = k^* - c_1 k \dot{m}_{\text{in,Air}}^{c_2} \tag{4.9}$$

At the STJ, the two heat exchangers between hot air and steam are constructed within one casing. A schematic illustration of the component can be found in fig. 3.2. Consequently, only one air inlet flow and one air outlet flow is available for measurement. The validation is therefore performed in combination of the two super heater models. The fitting for super heater 2 is performed only by comparing the steam outlet temperature simulation results with the measurements. The simulation results of the air outlet stream of super heater 2 have been used as input variables of the air stream into super heater 1. Its fitting has been performed penalizing the deviance of simulation results and measurements of both air outlet temperature and steam outlet temperature. The input variables relevant for this system of components are shown in fig. 4.16. In fig. 4.17, the simulation results of both super heaters are found. The starting values for the algebraic variables have been set equal to the measured values at the beginning of the simulated interval. For the analysis, a section with clear dynamics in inlet air temperature and mass flow has been chosen.

The steam outlet temperature of super heater 2 is shown in the upper graph of fig. 4.17. The simulated temperature rises with the rise of the air inlet temperature into the super heater while reacting faster to the temperature change than the measurements. The simulated data slightly overshoots the measured temperature. It follows a slow rise in inlet temperature between 1500 s and 6500 s for the measured temperature. The simulated outlet water temperature accurately reproduces the trend while oscillating around the measured values with a small amplitude. Afterwards, the measured and simulated temperatures drops and the simulation undershoots the measurements, reaching a temperature lower than the measurements. Between 7000 s and 13 000 s, the steam outlet temperature remains constant with the simulated temperature fluctuating around the measured values. From then on, the

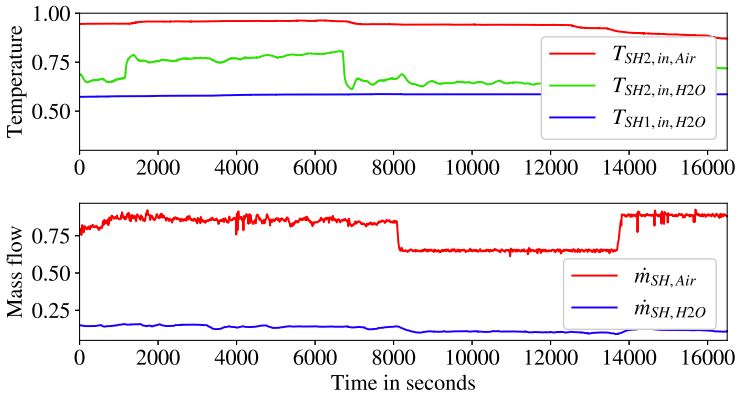


Figure 4.16: Input variables into the simulation of super heater 1 and super heater 2.

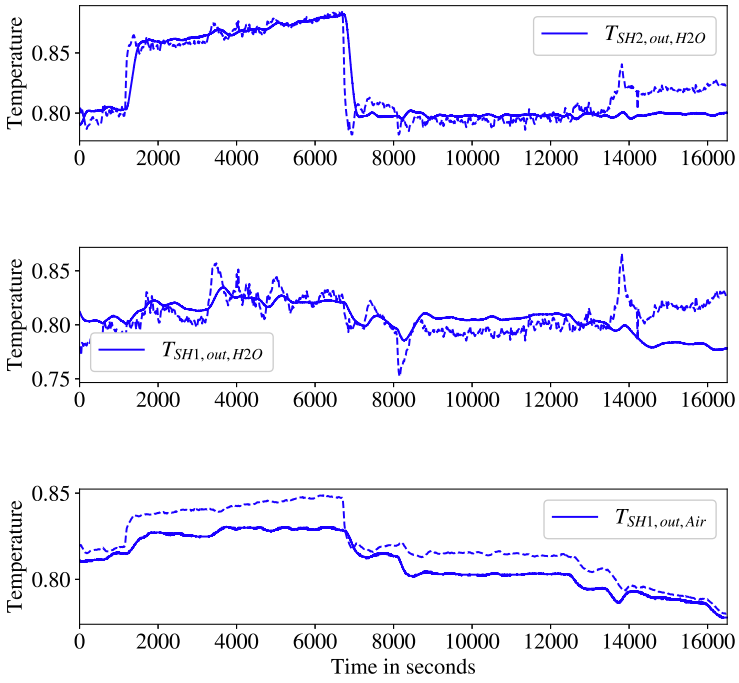


Figure 4.17: Simulation results over time of steam outlet temperature of super heaters. Measured data (——) and simulated data (- - -).

simulation results deviate noticeably from the measurements showing an increase in outlet temperature not recorded by the measurements. The RMSD of measured and simulated outlet water temperature of super heater 1 is 12,03 K and a NRMSD of 1%. The RMSD of measured and simulated outlet temperatures of water and air of super heater 1 is 15,08 K and a NRMSD of 1.9%

The simulated steam outlet temperature of super heater 1 can be found in the second graph of fig. 4.17. The simulation is able to follow measurements while reacting more strongly to changes in temperature then the real component. Fluctuations occurring in measurements are amplified. A deviance between simulation results and measurements can be observed in the interval starting at 13 500 s with the simulated outlet temperature being higher than the measurements.

The air outlet temperature of the super heaters can be found in the third graph of fig. 4.17. A rise in outlet temperature can be observed between 1500 s and 6500 s where the simulation shows a higher temperature than recorded by the measurements. A decrease in temperature between 6500 s and 8500 s can be observed. The simulation follows the measured behavior qualitatively, however, at higher temperatures.

The simulation is able to qualitatively react to changes in input variables as would be expected of a heat exchanger. The simulation results have been shown to be more sensitive to changes in input variables than the real component. This can be explained by the lack of thermal inertia within the model of the simulation. While the casing and the water inside the tube bundle has a relatively low mass, it levels out fluctuations in the input temperature. The steam outlet temperature of super heater 2 remains nearly constant while the air mass flow drops and the steam mass flow remains nearly constant as well. This shows that the approach of mass flow correlation of heat transfer is able to reproduce measured behavior of the component. With a constant heat transfer and its implicit formulation, as chosen in this model, a drop in the mass flow would result in an increase in steam outlet temperature and a drop in air outlet temperature. This effect is mitigated through the mass flow dependency of the heat transfer. The errors resulting from simplification can be seen in the interval starting at 13 500 s as the heat transfer is set too high. High air mass flow results in a high heat transfer and a too high steam outlet temperature in both super heaters. An appropriate inclusion of the water mass flow in the dynamic calculation of the heat transfer coefficients as suggested in section 3.3 would lead to a higher accuracy. Even an inclusion in data-based correlation could lead to better results. A fitting including both mass flows in that case is sensible only with an available model of the entire steam cycle including control of mass flows as operational data water mass flow and air mass flow are dependent. Alternatively, a wide range of possible inlet conditions would need to be considered in order to determine the behavior of the heat exchangers from data.

In the third graph of fig. 4.17, the air outlet temperature is consistently simulated to high up until 14 000 s. As the steam temperatures are closer to the measured values in this interval, a different heat sink needs to be considered. The heat transfer between heat exchanger casing and the ambient has not been considered in this study. The super heaters are the hottest component of the steam cycle resulting in highest heat loss to the environment in the real

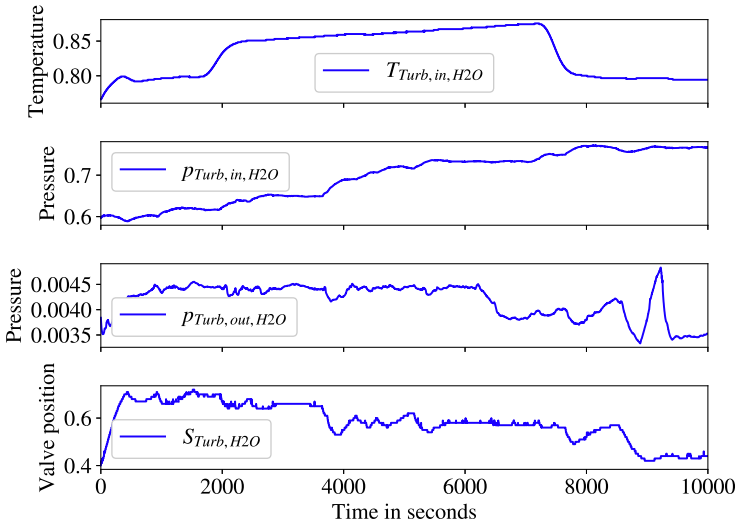


Figure 4.18: Input variables of steam turbine model.

plant. The data suggests considering these losses in a more comprehensive model if these deviations need to be mitigated. Accordingly, the simulated air temperature is closer to the measured values in the interval starting at 14 000 s while the steam temperatures deviates more strongly.

4.6 Steam Turbine Validation

In this section, simulation results of the steam turbine model are compared to operational plant data. The accuracy is discussed with respect to the ability to reproduce power production and to response to control inputs. The model used for the simulation of the steam turbine can be found in section 3.3.5. The control input for the steam turbine is the position of the turbine control valve S_{TCV}. A graphical illustration with the input variables into the steam turbine can be found in fig. 4.18 and the accompanying results of the simulation together with operational data are found in fig. 4.19. The validation data set has been chosen as it includes dynamics during typical operation excluding start up and shut down. The parameters which have been adjusted during the data fitting process are the isentropic efficiency of the turbine η_s, the heat transfer between steam and turbine casing and the part of the turbine casing mass that is involved in thermal inertia. While the turbine's as well as the casing's mass are known, it is unclear if all of it is heated fast enough in a relevant time scale to be part of thermal inertia. Starting values for the simulations have been chosen to be equal to the measurements in the beginning of the time interval.

The mass flow through the turbine is computed from all four input variables as shown in eq. (3.40). It rises to its operating point within the first 500 s and then slowly fluctuates in that region. It falls down to a lower value at 8500 s. In the first 2000 s and the last 2000 s, the mass flow is simulated slightly higher than it was measured. The simulated mass flow follows the

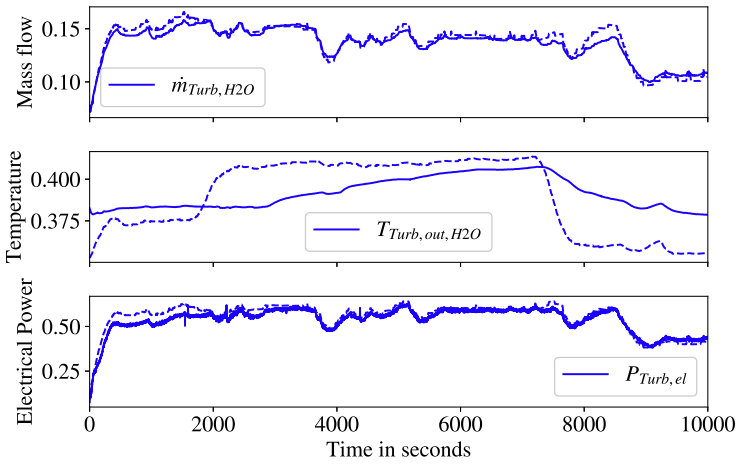


Figure 4.19: Simulation results over time of outlet temperature, produced electrical power and calculated mass flow in the steam turbine. Measured data (——) and simulated data (-·-·).

measurements accurately between 2000 s and 8000 s. The mass flow qualitatively resembles the course of the valve position. The RMSD of measured and simulated power output of the steam turbine is 44 kW and a NRMSD of 4.4%

The temperature at the outlet of the turbine over time can be found in the middle graph in fig. 4.19. The simulated outlet temperature drops under the measured values during the first 2000 s. It then rises quickly to a maximum temperature measured at the turbine outlet. It continues to rise slightly until 7000 s and then falls abruptly to the minimum of measured values while slightly undershooting below the minimal measured values. The measured temperature in turn rises slightly until 1000 s, remains constant until 3000 s and then rises almost continuously to its maximum at 7000 s. It then falls back onto its minimum within the time span between 8000 s and 10 000 s. The electricity output rises up to its operating point and remains on that level apart from slow fluctuation up to 8500 s. It then decreases to a lower level where it remains constant for the rest of the interval. The mass flow corresponds to the control valves behavior.

The inaccuracies at the beginning and the end of the interval overlap with the changes in inlet temperature into the turbine. A term including the inlet temperature is part of eq. (3.40) and could be the cause of this inaccuracy. To mitigate this effect, the term could be refined with a constant exponent. The outlet temperature of the turbine model is more sensible to input changes than the real component. While the outlet temperature of the real turbine rises slowly, the simulated output rises and falls significantly more abruptly. The choice of an explicit driving temperature difference in the model instead of a more accurate, implicit one can be part of the reason. The assumption of a constant heat transfer between the steam and the turbine wall does not correspond to real effects and could contribute to the inaccuracy. Consequently the casing of the model could be cause to more thermal inertia of the turbine than is displayed in the model. The combination of higher mass flow and lower outlet temperature results in

a higher electricity output than measured during the interval up to 2000 s. In the interval between 8000 s and 10 000 s the produced electricity is slightly under the measured values, which is caused by the small simulated mass flow. This effect is counteracted by the lower simulated outlet temperature.

5 Summary and Outlook

In this chapter, the procedure and contents of the work are summarized. Aspects of the study are analyzed critically and suggestions for future research is given. Firstly the use of polynomials for the water and steam properties is discussed. Secondly The use of the *do-mpc* frame work for solution and optimization of the system is evaluated. Suggestions for accuracy improvement of the system are given.

In this study thermal models of a solar tower power plant for the purpose of model based control have been set up and fit to operational data of the solar tower in Jülich. Relevant theoretical fundamentals and methods implemented in this study are presented in chapter 2. Model reduction efforts have been conducted in order to keep computational expenses down for optimization processes and to mitigate possible numerical issues when solving the system of equations. The sets of equation used to describe the components is found in chapter 3. The systems of equations describing the dynamic behavior of the components has been simulated using the *do-mpc* framework based on *Casadi*. For the automatic differentiation performed in *Casadi* differentiable functions describing the thermodynamic properties of the fluids are needed. The thermodynamic properties have been approximated using polynomial functions and their accuracy has been evaluated for different degrees and domains. The procedure and analysis is presented in section 3.1.1. Parameters of the models have been chosen and adjusted to fit the simulation to operational data. For that, a moving horizon estimation has been performed and implemented in the framework *do-mpc* to solve the non linear problem and determine a suited set of parameters. This set of parameters was refined by using sequential quadratic least square programming implemented in *SciPy-Minimize*. The simulation results are presented together with operational data and the performance of the model is evaluated in chapter 4. The effects of the made simplifications are explained.

The approximation of water and steam properties with polynomials, has to be evaluated critically in this work. In order to simplify the property functions and provide differentiable functions for the automatic differentiation, polynomials have been chosen. The implementation and evaluation of their accuracy is found in section 3.1.1. In order to keep accuracy in an acceptable range, the degree of these polynomials needs to be sufficiently high, see section 3.1.1. In the used models, e.g., the two phase model found in chapter 5, the property functions are part of larger functions, multiplying the results of polynomial approximations with each other. This leads to expressions with even higher degree causing numerical instability in components relying on the model. Oscillations in these functions, containing the multiplied terms, can cause local minima which hinder the optimization process. To mitigate these effects, a possibility to restrict the domain of validity of the polynomials was implemented. As can be seen in section 3.1.1, the degree of the polynomial can be reduced with extensive knowledge on the system. However, this method is elaborate as the expected state for each simulation

would need to be narrowly estimated before hand and the accuracy of each trust region would need to be evaluated individually. For the purpose of simulation and optimization a different approach should be chosen and great care should be taken for the choice of the property functions. As functions implemented by IAPWS have up to 48 degrees, they are not suitable for an implementation in *do-mpc*. A more complex formulation of polynomials as suggested by Åberg or Goemans could be a possibility in solving this problem [65][64]. An entirely data-driven surrogate model using neural networks or a comparable architecture can be a possibility for improvement in both accuracy and numerical robustness. The problem of accurate, differentiable water and steam properties needs to be addressed when joining the components for a comprehensive model of the power plant.

The *do-mpc* frame work is a multi functional tool which integrates various solvers for systems of equations, means for moving horizon estimation and model predictive control. The documentation is detailed and readily available. Additionally it is based on *Casadi* which allows an efficient calculation through use of automatic differentiation. However, the use of this frame work to solve the systems of equations resulting from the thermal models of the presented components lead often to error messages which are hard to interpret. The error would often point to the same line where the simulation time step was executed for a numerous number of causes. By using the implemented solvers, it is often difficult for the user to find the source of the error in a large system of equations. Whether the cause is to be found in a mistake in the formulation of an equation, starting conditions or the numerical structure of the system is often difficult to decide with information given by the frame work. A possible, yet elaborate option would be to implement the solvers manually or hand pick the solvers for the simulation, specifically with a focus on debugging. Especially for future students it can be helpful to find problems in the code and engage in the underlying mathematics of solving large systems of equations.

In case a more accurate model of the thermal storage is needed an improvement in accuracy can be the implementation of a heat transfer between air and the ceramic. As mentioned in section 4.1 the "One Temperautre Model" is used to avoid the occurrence of partial differential equations. With methods of solving these equations the partial differential equations are solved in a discrete manner which would thereby not increase numerical instability as only additional equations would need to be solved at each time step. The additional computational expense would need to be considered. In case a more precise model for the power block is wanted, the numerous simplifications performed in this study need to be reevaluated. The inaccuracy caused by these simplifications might be to significant for specific applications and improvement might be an aim for future studies. Implementing better suitable property functions for water and steam and simultaneously implement more transparent means of solving and optimizing the system, many simplifications might not be necessary to maintain high numerical stability and keep low computational expense.

For a comprehensive model of the power plant the connection of the components is necessary. The outlet streams of a component are then the input stream of the next component downstream in the flow chart. While the structure of the code and the modular nature of the the models allows a simple implementation of the connection, the complexity of the system of

equations will rise drastically when connecting components. Reducing the polynomial degree of the polynomials to one can be a possibility to reduce the numerical instability of the system and facilitate the process, compare section 3.1.1. However, an intense study of the simulated operational state would be necessary and the range of operational states would be limited with this method. It is recommended to find and test a more appropriate representation of the thermodynamic properties of water and steam. In addition, it can be helpful to investigate further methods for solving the system of equations, where the tracking of errors is more accessible. With larger systems of equations the possibilities for errors and their sources grow. An alternative frame work or a self programmed solver can help the set up of the complete dynamic model. A transfer of the simulation model to a frame work like *do-mpc* facilitating optimal control and parameter estimation would still be an option when the modular structure of equations, variables and solver is kept. It is to mention, that the parameter identification might need to be refined when the complete model of the system is set up. The applied methods for parameter identification have shown to be effective and can be used to identify the parameters.

Bibliography

1. KREYSZIG, E.: *Advanced engineering mathematics*. Wiley, 1972. ISBN 0471507288.
2. HAIRER, E.; WANNER, G.: *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems (Springer Series in Computational Mathematics)*. Springer, 2004. ISBN 9783540604525.
3. ASCHER, U. M.: *Computer methods for ordinary differential equations and differential-algebraic equations*. Society for Industrial and Applied Mathematics, 1998. ISBN 0898714125.
4. SOETJAHJO, A.: Mathematical analysis of dynamic process models; index, inputs and interconnectivity. In: 2006.
5. DHAMACHAROEN, A.: Efficient Numerical Methods for Solving Differential Algebraic Equations. *Journal of Applied Mathematics and Physics*. 2016, vol. 04, no. 01, pp. 39–47. Available from DOI: 10.4236/jamp.2016.41007.
6. WANNER, G.: *Solving Ordinary Differential Equations Ii*. Springer, 2010. ISBN 9783642052200.
7. BROWN, P. N. et al.: Consistent Initial Condition Calculation for Differential-Algebraic Systems. *SIAM Journal on Scientific Computing*. 1998, vol. 19, no. 5, pp. 1495–1512. Available from DOI: 10.1137/s1064827595289996.
8. NEDIALKOV, N. S.; PRYCE, J. D.: Solving Differential-Algebraic Equations by Taylor Series (I): Computing Taylor Coefficients. *BIT Numerical Mathematics*. 2005, vol. 45, no. 3, pp. 561–591. Available from DOI: 10.1007/s10543-005-0019-y.
9. BIEGLER, L. T. et al.: Systematic methods for chemical process design. 1997.
10. MARTELLI, E. et al.: Design Optimization and Dynamic Simulation of Steam Cycle Power Plants: A Review. *Frontiers in Energy Research*. 2021, vol. 9. Available from DOI: 10.3389/fenrg.2021.676969.
11. BEJAN, A.; TSATSARONIS, G.: *Thermal design and optimization*. Wiley, 1996. ISBN 0471584673.
12. BAEHR, H. D.; KABELAC, S.: *Thermodynamik*. Springer Berlin Heidelberg, 2016. Available from DOI: 10.1007/978-3-662-49568-1.
13. BORGNAKKE, C.: *Fundamentals of thermodynamics - 7. ed.* John Wiley, 2013. ISBN 9781118131992.
14. TURCHI, C. S. et al.: *CSP Systems Analysis - Final Project Report*. Office of Scientific and Technical Information (OSTI), 2019-05. Tech. rep. Available from DOI: 10.2172/1513197.
15. LOVEGROVE, K.; STEIN, W.: *Concentrating Solar Power Technology Principles, Developments and Applications: Principles, Developments and Applications*. Elsevier Science Technology, 2012. ISBN 9780857096173.

16. EDDHIBI, F. et al.: Design and analysis of a heliostat field layout with reduced shading effect in southern Tunisia. *International Journal of Hydrogen Energy*. 2017, vol. 42, no. 48, pp. 28973–28996. Available from DOI: 10.1016/j.ijhydene.2017.07.217.

17. STEIN, W.: Automatic heliostat track alignment method. 1986. Available also from: https://www.osti.gov/biblio/6030558.

18. PARGMANN, M. et al.: High accuracy data-driven heliostat calibration and state prediction with pretrained deep neural networks. *Solar Energy*. 2021, vol. 218, pp. 48–56. Available from DOI: 10.1016/j.solener.2021.01.046.

19. RODRIGUEZ-SÁNCHEZ, M. et al.: Thermodynamic and economic assessment of a new generation of subcritical and supercritical solar power towers. *Energy*. 2017, vol. 118, pp. 534–544. Available from DOI: 10.1016/j.energy.2016.10.079.

20. KOLL, G. et al.: The Solar Tower Jülich - a research and demonstration plant for central receiver systems. In: *SolarPACES 2009 : electricity, fuels and clean water powered by the sun ; 15 - 18 September 2009, Berlin, Germany ; the 15th SolarPACES conference ; proceedings*. 2013.

21. HENNECKE, K. et al.: The Solar Power Tower Jülich — A Solar Thermal Power Plant for Test and Demonstration of Air Receiver Technology. In: *Proceedings of ISES World Congress 2007 (Vol. I – Vol. V)*. Springer Berlin Heidelberg, 2008, pp. 1749–1753. Available from DOI: 10.1007/978-3-540-75997-3_358.

22. GALL, J.: *Betriebsführung und -optimierung eines solarthermischen Turmkraftwerkes*. Köln, 2012. PhD thesis.

23. FLUECKIGER, S. M. et al.: Thermomechanical Simulation of the Solar One Thermocline Storage Tank. *Journal of Solar Energy Engineering*. 2012, vol. 134, no. 4. Available from DOI: 10.1115/1.4007665.

24. ALVA, G. et al.: An overview of thermal energy storage systems. *Energy*. 2018, vol. 144, pp. 341–378. Available from DOI: 10.1016/j.energy.2017.12.037.

25. ODENTHAL, C. et al.: Modelling and operation strategies of DLR's large scale thermocline test facility (TESIS). In: *AIP Conference Proceedings*. 2017. Available from DOI: 10.1063/1.4984440.

26. ZOTICĂ, C. et al.: Optimal operation and control of heat to power cycles: A new perspective from a systematic plantwide control approach. *Computers &amp Chemical Engineering*. 2020, vol. 141, p. 106995. Available from DOI: 10.1016/j.compchemeng.2020.106995.

27. RÖGER, M. et al.: Techniques to Measure Solar Flux Density Distribution on Large-Scale Receivers. *Journal of Solar Energy Engineering*. 2014, vol. 136, no. 3. Available from DOI: 10.1115/1.4027261.

28. GÖHRING, F. et al.: Flux Density Measurement on Open Volumetric Receivers. In: 2011.

29. PATTANAYAK, L.; SAHU, J. N.: Steady state modeling on energy and exergy analysis of a pulverized coal fired thermal power plant. *Asia-Pacific Journal of Chemical Engineering*. 2015, vol. 10, no. 6, pp. 876–884. Available from DOI: 10.1002/apj.1924.

30. FLETCHER, R.; POWELL, M. J. D.: A Rapidly Convergent Descent Method for Minimization. *The Computer Journal*. 1963, vol. 6, no. 2, pp. 163–168. Available from DOI: 10.1093/comjnl/6.2.163.

31. HILLESTAD, M.; HERTZBERG, T.: Dynamic Simulation of Chemical Engineering Systems by the Sequential Modular Approach. *Modeling, Identification and Control: A Norwegian Research Bulletin*. 1986, vol. 7, no. 3, pp. 107–127. Available from DOI: 10.4173/mic.1986.3.1.

32. BRUGNANO, L. et al.: Fifty Years of Stiffness. 2009. Available from DOI: 10.48550/ARXIV.0910.3780.

33. ENRIGHT, W. H. et al.: Comparing numerical methods for stiff systems of O.D.E:s. *BIT*. 1975, vol. 15, no. 1, pp. 10–48. Available from DOI: 10.1007/bf01932994.

34. CURTISS, C. F.; HIRSCHFELDER, J. O.: Integration of Stiff Equations. *Proceedings of the National Academy of Sciences*. 1952, vol. 38, no. 3, pp. 235–243. Available from DOI: 10.1073/pnas.38.3.235.

35. GARDNER, D. J. et al.: Enabling new flexibility in the SUNDIALS suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)*. 2022. Available from DOI: 10.1145/3539801.

36. HINDMARSH, A. C. et al.: SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)*. 2005, vol. 31, no. 3, pp. 363–396. Available from DOI: 10.1145/1089014.1089020.

37. SERBAN, R. et al.: *User Documentation for IDAS* [urlhttps://sundials.readthedocs.io/en/latest/idas]. 2022. Available also from: https://sundials.readthedocs.io/en/latest/idas. v5.4.1.

38. LUCIA, S. et al.: *do-mpc Documentation Release 4.4.0*. 2022. Available also from: https://www.do-mpc.com/_/downloads/en/latest/pdf/.

39. RAWLINGS, J. B. et al.: *Model Predictive Control Theory, Computation, and Design, 2nd Edition: Theory, Computation, and Design, 2nd Edition*. Nob Hill Publishing, LLC, 2017. ISBN 9780975937730.

40. ANDERSSON, J. A. E. et al.: CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*. 2019, vol. 11, no. 1, pp. 1–36. Available from DOI: 10.1007/s12532-018-0139-4.

41. JOEL ANDERSSON Joris Gillis, M. D.: User Documentation for CasADi v3.4.4. 2018. Available from DOI: 10.1007/s12532-018-0139-4.

42. DORNEANU, B.: *Model Reduction in Chemical Engineering: Case studies applied to process analysis, design and operation*. 2011. PhD thesis.

43. EASON, J. P. et al.: Surrogate Equations of State for Equation-Oriented Optimization of Polymerization Processes. In: *13th International Symposium on Process Systems Engineering (PSE 2018)*. Elsevier, 2018, pp. 781–786. Available from DOI: 10.1016/b978-0-444-64241-7.50125-7.

44. CHAKRABARTY, A. et al.: Inherently Safer Design. In: *Multiscale Modeling for Process Safety Applications*. Elsevier, 2016, pp. 339–396. Available from DOI: 10.1016/b978-0-12-396975-0.00008-5.

45. MARQUARDT, W.: Nonlinear Model Reduction for Optimization Based Control of Transient Chemical Processes. In: 2002.

46. SIMPSON, T. W. et al.: Metamodels for Computer-based Engineering Design: Survey and recommendations. *Engineering with Computers*. 2001, vol. 17, pp. 129–150.

47. ADEL, A.; SALAH, K.: Model order reduction using artificial neural networks. In: *2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2016. Available from DOI: 10.1109/icecs.2016.7841139.

48. BROSILOW, C.; JOSEPH, B.: *Techniques of Model Based Control*. Prentice Hall PTR, 2002. ISBN 9780130280787.

49. WAKITANI, S. et al.: Design of a Data-Driven PID Controller using Operating Data. *IFAC Proceedings Volumes*. 2013, vol. 46, no. 11, pp. 587–592. Available from DOI: 10.3182/20130703-3-fr-4038.00077.

50. KATEBI, R.: MODELLING, SIMULATION AND CONTROL OF LARGE POWER PLANTS. *IFAC Proceedings Volumes*. 2007, vol. 40, no. 8, pp. 3–14. Available from DOI: 10.3182/20070709-3-ro-4910.00002.

51. THOMAS, J.: Model-Based Power Plant Master Control. *Automatika*. 2014, vol. 55, no. 3, pp. 228–238. Available from DOI: 10.7305/automatika.2014.12.434.

52. RAU, M.: *Nichtlineare modellbasierte prädiktive Regelung auf Basis lernfähiger Zustandsraummodelle*. München, 2003. Dissertation. Technische Universität München.

53. RAKOVIĆ, S. V.; LEVINE, W. S.: *Handbook of Model Predictive Control*. Birkhäuser, [n.d.]. ISBN 9783319774886.

54. SCHWENZER, M. et al.: Review on model predictive control: an engineering perspective. *The International Journal of Advanced Manufacturing Technology*. 2021, vol. 117, no. 5-6, pp. 1327–1349. Available from DOI: 10.1007/s00170-021-07682-3.

55. DEKKING, F. M. et al.: *A Modern Introduction to Probability and Statistics: Understanding Why and How (Springer Texts in Statistics)*. Springer, 2007. ISBN 9781852338961.

56. NOCEDAL, J.; WRIGHT, S.: *Numerical Optimization (Springer Series in Operations Research and Financial Engineering)*. Springer, 2006. ISBN 9780387303031.

57. KRAFT, D.: *A Software Package for Sequential Quadratic Programming*. Wiss. Berichtswesen d. DFVLR, 1988. Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht.

58. EFRON, B.; TIBSHIRANI, R. J.: *An Introduction to the Bootstrap*. Springer US, 1993. Available from DOI: 10.1007/978-1-4899-4541-9.

59. KING, R.: *Struktur- und Parameteridentifikation, Lecture script*. TU Berlin, 2022.

60. NEIDINGER, R. D.: Introduction to Automatic Differentiation and MATLAB Object-Oriented Programming. *SIAM Review*. 2010, vol. 52, no. 3, pp. 545–563. Available from DOI: 10.1137/080743627.

61. GRIEWANK, A.; WALTHER, A.: *Evaluating Derivatives*. Society for Industrial and Applied Mathematics, 2008. Available from DOI: 10.1137/1.9780898717761.

62. LUCIA, S. et al.: Rapid development of modular and sustainable nonlinear model predictive control solutions. *Control Engineering Practice*. 2017, vol. 60, pp. 51–62. Available from DOI: 10.1016/j.conengprac.2016.12.009.

63. IAPWS Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam. In: *International Steam Tables*. Springer Berlin Heidelberg, 2008, pp. 7–150. Available from DOI: 10.1007/978-3-540-74234-0_3.

64. GOEMANS, T.: Polynomial representation of the thermodynamic properties of saturated water and steam between 10 and 180 bar. *Nuclear Engineering and Design*. 1971, vol. 16, no. 2, pp. 179–192. Available from DOI: 10.1016/0029-5493(71)90033-1.

65. ÅBERG, M. et al.: Optimization-friendly thermodynamic properties of water and steam. In: *Linköping Electronic Conference Proceedings*. Linköping University Electronic Press, 2017. Available from DOI: 10.3384/ecp17132449.

66. NOLTEERNSTING, F.: *Betriebsassistenz für solarthermische Kraftwerke zur Effizienzsteigerung auf Basis regelungstechnischer Methoden: Operation Assistance for Concentrating Solar Power Plants to Increase Efficiency Based on Control Engineering Methods*. Aachen, 2020. PhD thesis.

67. LONDON, A.; SEBAN, R. A.: A generalization of the methods of heat exchanger analysis. *International Journal of Heat and Mass Transfer*. 1980, vol. 23, pp. 5–16.

68. VORTMEYER, D.; SCHAEFER, R. J.: Equivalence of one- and two-phase models for heat transfer processes in packed beds: one dimensional theory. *Chemical Engineering Science*. 1974, vol. 29, pp. 485–491.

69. VINNEMEIER, P.: *Model-Based Optimal Operation of Solar Thermal Power Cycles: Modellbasierter Optimalbetrieb von solarthermischen Kraftwerken*. Aachen, 2018. PhD thesis.

70. STRAUSS, K.: *Kraftwerkstechnik: zur Nutzung fossiler, nuklearer und regenerativer Energiequellen*. Springer, 2009. ISBN 9783642014307.

71. E.V., V. D. I.: *VDI-Wärmeatlas*. Springer Vieweg, [n.d.]. ISBN 9783642199806.

72. KNEER, R.: *Wärme- und Stoffübertragung, Lecture script*. RWTH Aachen, 2010.

73. STODOLA, A.: *Dampf- und Gasturbinen. Mit einem Anhang über die Aussichten der Wärmekraftmaschinen : Nachtrag zur 5. Auflage: Nachtrag zur 5. Auflage*. Springer, [n.d.]. ISBN 9783642505447.

74. KESTIN, J.: Ein Beitrag zu Stodolas Kegelgesetz. *Wärme- und Stoffübertragung*. 1982, vol. 16, no. 1, pp. 53–55. Available from DOI: 10.1007/bf01322806.

75. GROTE, W.: *Ein Beitrag zur modellbasierten Regelung von Entnahmedampfturbinen*. 2009. doctoralthesis. Ruhr-Universität Bochum, Universitätsbibliothek.

76. RAY, A.: Dynamic modelling of power plant turbines for controller design. *Applied Mathematical Modelling*. 1980, vol. 4, no. 2, pp. 109–112. Available from DOI: 10.1016/0307-904x(80)90114-6.

77. SAVOLA, T.; KEPPO, I.: Off-design simulation and mathematical modeling of small-scale CHP plants at part loads. *Applied Thermal Engineering*. 2005, vol. 25, no. 8-9, pp. 1219–1232. Available from DOI: 10.1016/j.applthermaleng.2004.08.009.

78. VIRTANEN, P. et al.: SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*. 2020, vol. 17, pp. 261–272. Available from DOI: 10.1038/s41592-019-0686-2.

Abbreviations

Abbreviations

AE	Algebraic system of equations
AI	Artificial intelligence
CON	Condenser
CONP	Condenser pump
CSP	Concentration solar power
DAE	Differential algebraic equations
ECO	Economizer
EVA	Evaporator
FWP	Feed water pump
FWT	Feed water tank
IAPWS	Industrial Formulation 1997 for the Thermodynamic Properties of Water and Steam
IPOPT	Interior point optimizer
max	Maximum
min	Minimum
MHE	Moving Horizon Estimation
MPC	Model predictive control
NLP	Non-linear programming problem
NRMSD	Normalized root-mean-square deviation
ODE	Ordinary differential equations
OOP	Object-oriented programming
PH	Preheater
PID	Proportional integral derivative
REC	Receiver
RMSD	Root-mean-square deviation
SC	Steam cooler
SCV	Steam cooler valve
SH1/2	Super heater 1/2
SLSQP	Sequential least square quadratic programming
SQP	Sequential quadratic programming
ST	Thermal storage
STJ	Solar tower Jülich
STPP	Solar tower power plants
TCV	Turbine control valve
Turb	Turbine group
VREC	Ventilator receiver
VSG	Ventilator steam generator

Latin symbols

A	Area
C	Polynomial coefficients
c	Fitting parameters
c_p	Isobaric heat capacity
e	Control error
H	Enthalpy, Heaviside function, height
h	Specific enthalpy
k	Combined heat transfer coefficient
$\dot{m}$	Mass flow
N	Natural number
P	Weighted norm
p	Parameter, pressure
u	Controlled variable
V	Volume
v	Velocity, measurement noise
w	Additive disturbance
$\dot{Q}$	Heat flow
$\dot{q}$	Heat flux
T	Temperature
t	Independent variable
S	Sum, Valve position
s	Entropy
x	State, dependent variable
y	System output, measurement
z	Algebraic variable

Greek symbols

α	Heat transfer coefficient
Δ	Difference
η_s	Isentropic efficiency
λ	Thermal conductivity

Superscript and subscript indices

0	Current state
Air	Air
Amb	Ambient
arg	Argument
gas	Gaseous
H2O	Water
in	Inlet
L	Lower
lb	Lower boundary
liq	Liquid
out	Outlet
p	Prediction
ref	Reference
s	Isentropic
sat	Saturation
set	Set trajectory
Stodola	Stodola function
term	Terminal
tv	Time variant
U	Upper
ub	Upper boundary
W	Wall

List of Figures

List of Tables

Appendix: Two phase equilibrium model for a single fluid system

Assumptions

$$T = T_{\text{liq}} = T_{\text{gas}} = T_{\text{steel}} \tag{.1}$$

$$p = p_{liq} = p_{\text{gas}} (= p_{\text{steel}}) \tag{.2}$$

Boundary conditions

$$\frac{dU}{dt} = \sum_i \dot{m}_i h_i + \sum_j \dot{Q}_j \tag{.3}$$

$$\frac{dm}{dt} = \sum_i \dot{m}_i \tag{.4}$$

Parameters

$$\{V, m_{\text{steel}}, c_{\text{steel}}\} = const. \tag{.5}$$

Differential equations

$$\frac{dT}{dt} = \frac{\left[\frac{dU}{dt}\right] + \frac{dV_{\text{liq}}}{dt}\left(\frac{u_{\text{gas}}}{v_{\text{gas}}} - \frac{u_{\text{liq}}}{v_{\text{liq}}}\right)}{\frac{V_{\text{liq}}}{v_{\text{liq}}}\left(B - \frac{u_{\text{liq}}}{v_{\text{liq}}}A\right) + \frac{V - V_{\text{liq}}}{v_{\text{liq}}}\left(D - \frac{u_{\text{gas}}}{v_{\text{gas}}}C\right) + \sum_k m_{\text{steel,k}} c_{\text{steel,k}}} \tag{.6}$$

$$\frac{dV_{\text{liq}}}{dt} = \frac{\left[\frac{dm}{dt}\right] + \left(\frac{V - V_{\text{liq}}}{v_{\text{gas}}^2}C + \frac{V_{\text{liq}}}{v_{\text{liq}}^2}A\right)\frac{dT}{dt}}{\frac{1}{v_{\text{liq}}} + \frac{1}{v_{\text{gas}}}} \tag{.7}$$

$$A = \left(\frac{\partial v}{\partial T}\right)_{\text{p,liq}} + \left(\frac{\partial v}{\partial p}\right)_{\text{T,liq}} \cdot \frac{\partial p_s}{\partial T} \quad B = \left(\frac{\partial u}{\partial T}\right)_{\text{v,liq}} + \left(\frac{\partial u}{\partial v}\right)_{\text{T,liq}} \cdot A \tag{.8}$$

$$C = \left(\frac{\partial v}{\partial T}\right)_{\text{p,gas}} + \left(\frac{\partial v}{\partial p}\right)_{\text{T,gas}} \cdot \frac{\partial p_s}{\partial T} \quad D = \left(\frac{\partial u}{\partial T}\right)_{\text{v,gas}} + \left(\frac{\partial u}{\partial v}\right)_{\text{T,gas}} \cdot A \tag{.9}$$

Algebraic equations

$$\{v_{\text{liq}}, v_{\text{gas}}, u_{\text{liq}}, u_{\text{gas}}\} = f(T) \tag{.10}$$

$$\left\{\left(\frac{\partial v}{T}\right)_{\text{p,liq}}, \left(\frac{\partial v}{T}\right)_{\text{p,gas}}, \left(\frac{\partial v}{p}\right)_{\text{T,liq}}, \left(\frac{\partial v}{p}\right)_{\text{T,gas}}\right\} = f(T) \tag{.11}$$

$$\left\{\left(\frac{\partial u}{T}\right)_{\text{v,liq}}, \left(\frac{\partial u}{T}\right)_{\text{v,gas}}, \left(\frac{\partial u}{v}\right)_{\text{T,liq}}, \left(\frac{\partial u}{v}\right)_{\text{T,gas}}\right\} = f(T) \tag{.12}$$