Chair of Automation

Master's Thesis

Sensorless Temperature Control in a Metal Forming Process

Patrick Löbl, BSc

September 2021

MONTANUNIVERSITÄT LEOBEN
www.unileoben.ac.at

EIDESSTATTLICHE ERKLÄRUNG

Ich erkläre an Eides statt, dass ich diese Arbeit selbständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt, und mich auch sonst keiner unerlaubten Hilfsmittel bedient habe.

Ich erkläre, dass ich die Richtlinien des Senats der Montanuniversität Leoben zu "Gute wissenschaftliche Praxis" gelesen, verstanden und befolgt habe.

Weiters erkläre ich, dass die elektronische und gedruckte Version der eingereichten wissenschaftlichen Abschlussarbeit formal und inhaltlich identisch sind.

Datum 27.09.2021

Unterschrift Verfasser/in
Patrick Löbl

Acknowledgements

First, I want to express my deep gratitude to Prof. Dr. Paul O'Leary for providing me with the opportunity to write this thesis at the Chair of Automation, his support and the warm and welcoming atmosphere I experienced at the institute.

Furthermore, I would like to convey my profound gratitude to Prof. Dr. Martin Stockinger for providing me with an immensely interesting problem statement and the possibility to conduct my thesis in cooperation with the Chair of Metal Forming.

I especially want to thank Dr. Gerhard Rath for supervising my thesis and his immense and continuous support. I could not thank him enough for all the work and time he invested in getting the system up and running with me.

My profound gratitude also extends to Dipl.-Ing. Andreas Schwarz-Gsaxner, for all his the time spent on creating and optimising countless FEA simulations and his exemplary time flexibility regarding all questions and experiments. In addition, I want to thank Dipl.-Ing. Maria Haider for her help in proofreading my thesis.

I am also immensely grateful for my very supportive friends, who made all the time spent in Leoben the extraordinarily enriching and fun experience that it was, even during bleak and challenging phases. I cannot imagine having become the person I am without them and will forever cherish our shared memories.

Finally, I want to thank my parents for always supporting me, not only financially, in the pursuit of my education and my dreams. Only through their continuous support, it was possible for me to conduct and finish my studies.

Kurzfassung

Bei der Umformung metallischer Proben hat deren Temperatur maßgeblichen Einfluss auf den Umformprozess und die sich einstellenden Eigenschaften. Die Probentemperatur im Werkzeug kann jedoch kaum direkt durch physische Sensoren gemessen werden, was auch deren Regelung erschwert.

Sensorlose Temperaturregelung mithilfe virtueller Sensoren ist eine Lösung für dieses Problem und zusätzlich aufgrund der geringen Kosten und einfachen Anpassung sehr attraktiv für Umformprozesse. Die zur Modellierung eines virtuellen Sensors benötigten Input-Output-Daten stehen zudem oft bereits aus Finite-Elemente Simulationen zur Verfügung. Diese weisen insbesondere mit variablen Zeitinkrementen große Kosten- und Zeitersparnisse gegenüber realen, mit Experimenten verbundenen Messungen auf.

Das Ziel dieser Masterarbeit ist es, eine sensorlose Temperaturregelung für einen speziellen Umformprozess zu entwickeln, welche auf einem virtuellen Sensor zur Abschätzung der Probentemperatur basieren soll und so die Notwendigkeit einer direkten, physischen Messung eliminiert. Der virtuelle Sensor soll durch ein dynamisches Modell verwirklicht werden. Zur Modellierung stehen Input-Output Temperaturdaten mit ungleichmäßigen Sampling-Intervallen aus Finite-Elemente Simulationen der Probe zur Verfügung, zur Validierung gleichmäßige abgetaste Messdaten.

Die zentralen Forschungsfragen sind die Modellierung von Systemen auf Basis von Input-Output-Daten mit variablen Sampling-Intervallen und die Entwicklung einer geeigneten sensorlosen Regelung der Probentemperatur für einen Umform-Prüfstand. Um diese zu erfüllen, wurde eine Methodik zur Modellbildung aus der Literatur abgeleitet und eine Methode zur Systemidentifikation mit Global Least Squares numerisch realisiert und angewandt. Als Regler wurden PID und Model Predictive Control evaluiert, die Umsetzung erfolgte mittels PID-Regelung. Die Ergebnisse wurden simuliert und experimentell validiert.

Eine sensorlose Temperatursteuerung wurde erfolgreich an dem Equal Channel Angular Pressing (ECAP) Prüfstand am Lehrstuhl für Umformtechnik der Montanuniversität Leoben zur Anwendung gebracht. Der Einsatz der Systemidentifikation mit Global Least Squares hat sich bei Modellen in Zustandsraumdarstellung mit definierter Struktur sowohl für Input-Output-Daten mit gleichmäßigen als auch ungleichmäßigen Abtastintervallen bewährt. Mithilfe der entwickelten Steuerung konnte die Aufheizzeit am Prüfstand auf 30% der im Betrieb ohne Steuerung benötigten Zeit reduziert werden. Eine finale, konstant bleibende Sollwertabweichung von kleiner 1% wurde ebenfalls durch die Steuerung ermöglicht, wogegen im Betrieb ohne Steuerung die Probentemperatur nur für 100 - 150 s konstant gehalten werden konnte und mit weitaus größeren Abweichungen zu rechnen war.

Abstract

In metal forming, the specimen temperature has a decisive influence on the forming process and the resulting material properties. However, the specimen temperature within the tool can scarcely be measured directly by physical sensors, which also makes temperature control difficult.

Sensorless temperature control by means of virtual sensors is a solution to this problem and also very attractive for forming processes due to its low cost and possibilities of easy adaptation. The input-output data required for modelling a virtual sensor are also often already available from simulations in Finite Element Analysis software. These provide great cost and time savings compared to measurements conducted on real experiments, especially if variable time increments are used in the simulation.

The goal of this thesis is to develop a sensorless temperature control system for a special forming process, which is to be based on a virtual sensor to estimate the specimen temperature and thus eliminate the need for a direct, physical measurement. The virtual sensor is to be implemented using a dynamic model. Input-output temperature data with uneven sampling intervals from finite element simulations of the specimen are available for modelling, and uniformly sampled measurement data for validation.

The central research questions include the modelling of systems on the basis of input-output data with variable sampling intervals and the development of a suitable sensorless control system of the specimen temperature for the test bed. In order to fulfil these, a methodology for model construction was derived from literature and the global least squares system identification method was numerically implemented and applied. PID and model predictive control were evaluated for the control system and the implementation was carried out using a PID controller. The results were simulated and validated experimentally.

A sensorless temperature open-loop control system was successfully implemented at the Equal Channel Angular Pressing (ECAP) test bed at the Chair of Metal Forming at the Montanuniversität Leoben. The use of the global least squares system identification method for parameter estimation has proven suitable for models in state-space representation with a defined structure by utilising input-output data with uniform as well as non-uniform sampling intervals. As a result of the developed control system, the heating-up time on the test bed could be reduced to 30% of the time required in operation without the control system. A final, constant set point deviation of less than 1% was also made possible by the control system, whereas in operation without the control system the specimen temperature could only be kept constant for 100 - 150 s and much larger set point deviations were to be expected.

Contents

Chapter 1

Introduction

This chapter highlights the motivation for this thesis and gives an overview of the topics covered as well as their state of the art. The goals of this work are defined and its structure briefly explained.

1.1 Motivation

The control of specimen temperature plays a major role in the Equal Channel Angular Pressing (ECAP) test bed. High specimen temperatures generated by an array of heating elements enable forming experiments with high-tensile alloys by increasing material ductility, as at room temperature, the specimen would tear or the required forming forces would exceed the limits of the test bed. In addition, the availability of precise temperature control allows the investigation into optimal process parameters. However, the placement of a physical sensor within the specimen for in-process temperature measurement necessary for control is not feasible.

A sensorless control approach requires a virtual sensor, which is based on suitable dynamic models derived from observed system behaviour. As the installation of an internal temperature sensor damages the specimen and prevents the conduction of the process, conventional measurement is a unsustainable approach of system observation. Equivalent information to measurement can be gained by means of Finite Element Analysis (FEA), in which a specific simulation is conducted for each experimental setup in regards to specimen material, temperature and geometry. However, the generation of simulation data for one experiment configuration takes several hours if fixed time increments are used in the FEA simulation. In contrast, utilizing variable time intervals allows a full simulation of the Equal Channel Angular Pressing process well within sixty minutes.

The motivation of this thesis is to create a specimen temperature control system for the ECAP test bed which avoids the unfeasible use of physical sensors and is highly adaptable to changing experiment configurations. This sensorless control is based on a dynamic model acting as a virtual sensor. Non-uniformly spaced FEA simulation data is used for modelling to allow model adaptation to a wide range of experiment configurations with minimal time and cost of data collection.

1.2 Literature Review

Virtual Sensors and Sensorless Control

Technological progress, decreasing sensor unit costs and increased connectivity increase the spread and complexity of the Internet of Things (IoT) and cyber-physical systems (CPS) [1, p. 315]. The data captured and fed to the connected information system is either directly observed by a physical sensor or indirectly derived by virtual sensors which fuse data from one or more physical sensors [1, p. 315].

The use of physical sensors is usually subject to a number of limitations, which was studied in-depth in [2]. First of all, applying sensors is cost-intensive, especially as their numbers increase [1, p. 317]. Purchasing the sensors and maintaining them creates costs as well as their physical process integration, which at the same time increases the complexity and the operating cost of the system [3, p. 1]. Another limitation of physical sensors is the fact that due to spatial or environmental conditions, their use is not feasible in some situations and in many industrial processes, some process outputs required for monitoring and control cannot be measured by a sensor at all [4, p. 1714, 1, p. 317]. Reasons are access requirements for measurement and maintenance, disturbances on the process by the sensor or harshness of the environment [5, p. 2801, 3, p. 2]. Physical sensors also require frequent, offline calibration to ensure accurate measurements, which is often unpractical [3, pp. 1–2, 6, p. 3560]. They can be prone to failure during calibration due to many factors as well as be subject to miscalibration, in both of which they need to be repaired or replaced [3, pp. 1–2, 6, p. 3560]. At last, sensor signals can be noisy and interfere with each other as well as lose accuracy over time due to a multitude of reasons [1, p. 317].

Due to those limitations and especially because some process parameters cannot be measured at all, soft sensors became widely popular over the last three decades [5, p. 2801]. A soft sensor, or virtual sensor, is a means to estimate process parameters in miscellaneous applications if a physical sensor is either unavailable or unsuitable to directly measure the sought parameters [5, p. 2801]. These sensors are often used in real-time process monitoring and diagnostics, modelling and control, as well as for detecting or estimating faults [5, p. 2801]. According to [7, p. 2], they can be divided into two categories. First, model-based virtual sensors are often implemented through system modelling and data validation and use available measurements for calculations. Second, data-driven virtual sensors use historical measurement data or detailed estimation for their calculations [7, p. 2]. In either case, dynamic models often play an integral part in virtual sensing as they enable the indirect determination of quantities that are not directly measurable [8, p. 3]. For the purpose of creating a virtual sensor, model and modelling method should be as simple as possible to achieve the desired performance, robustness and realise the model maintenance with minimum time and cost [2, p. 17].

The advantages of virtual sensors include the lower cost of software compared to hardware for initial investment and maintenance [1, p. 317]. They are a valid alternative if spatial conditions or a hostile environment do not allow for a physical sensor and may compensate the resulting delay or inaccuracy of a physical sensor in a sub-optimal spot as well as reduce signal noise [1, p. 317]. Drifts due to wear or pollution in physical sensors leading to inaccuracy over time can be recognised or compensated and finally, virtual sensors can be redesigned as required and are thus more flexible in comparison to mechanically installed physical sensors [1, p. 317].

Virtual or soft sensors and corresponding sensorless control techniques have found applications in many fields and industries [5, p. 2801]. The need for virtual sensing of temperature inspired many solutions: A sensorless temperature measurement technique based on an artificial neural network for the purpose of predicting temperature otherwise unfeasible to measure in battery cells is proposed in [6]. As the measurement of the temperature of slabs in reheating furnaces is equally difficult, but important for process parameter monitoring, reference [9] addresses a soft sensor as a means to slab temperature estimation. The soft sensing of die melt temperature in polymer extrusion processes, in which physical sensors cannot be placed, with data-based dynamic models is the focus of [5], while reference [10] addresses sensorless temperature estimation to realise a thermal management system for lithium-ion battery cells in contrast to an external temperature sensor placed on the battery case surface. A similar approach for monitoring greenhouse temperature is found in [7]. Finally, reference [11] introduces a method of estimating the amount of heat inflow without using a heat flow sensor. Special interest in sensorless control can be found in electric drive technologies. An overview of the expanding application of sensorless AC motor drives and control strategies is given in [12]. For wind technologies, sensorless control approaches were developed in [13] and [14], which utilise sensorless control to eliminate the direct measurement of wind speed.

System Identification in Non-Uniformly Sampled Systems

Uniform and single-rate sampling of variables has been a standard assumption in system identification, which implies that sensors are delivering measurements synchronously and at the same constant rate [15, p. 314]. However, it is often difficult to obtain equidistant sampled data for system identification purposes, especially in areas where measurement can not be controlled or uniform sampling is practically impossible [15, p. 13]. This is further elaborated in [15, p. 314]: Data sampled in non-uniform intervals occurs often if it is recorded by means of manual sampling or if the process involves time-consuming, unprojectable laboratory analysis. Another reason can be missing elements in measurement data, which are commonly caused by sensor breakdowns, off-range measurements, data-acquisition failures, blackouts or transmission interruption. In robust data analysis, outliers are often discarded or treated as missing, leading to missing values even when a uniform sampling rate is utilised [15, p. 314]. According to [16, p. 1], non-uniform sam-

pling becomes more interesting and beneficial with low activity signals like temperature, pressure or speech, which are characterised by evolving smoothly or sporadically. These signals remain constant most of the time and then vary significantly during a short period of time. As uniform sampling takes samples even without signal changes, many redundant and useless samples are recorded, especially if the Shannon theorem is utilised, resulting in an artificial increase in the computational load [16, p. 1].

Per definition, non-uniform data consists of data sets in which input and output signals are sampled at irregular intervals, while the time instants of the sampling are the same, meaning that the data is available at synchronous points [15, p. 324]. For these non-uniformly sampled data systems, a discrete-time linear, time-invariant model is not applicable due to the violation of the assumption of a uniformly sampled environment required for the existence of a corresponding discrete-time model [15, p. 13]. In contrast, continuous-time identification methods can handle the uneven sampling problem provided that appropriate numerical methods are applied [15, pp. 314–315]. This is because the coefficients of continuous-time models are assumed to be independent of the sampling rate [15, p. 13]. Although interpolation between available sampled data is a commonly used approach for identification purposes, the utilised interpolations are generally univariate and ignore the multivariate dynamic behaviour of the process [17, p. 2]. This can result in sub-optimal models as well as untrustworthy predictions [17, p. 2].

In the field of continuous-time system identification, miscellaneous methods utilising non-uniformly sampled data have been proposed. Based on autoregressive models, estimating continuous time models from irregularly observed discrete-time data was the focus of [18] and [19]. Maximum likelihood and expectation maximisation approaches were proposed in [17] for the identification of chemical processes with irregularly sampled outputs as well as in [20], where an expectation maximisation algorithm to identify continuous-time state-space models from non-uniformly fast-sampled data is presented. The maximum likelihood method was also used to identify continuous time state-space models from non-uniform, but small sampling periods in [20]. There is also a selection of specialised approaches available. In [21], the unique reconstruction of a continuous-time system from its non-uniformly sampled discrete-time model is discussed. An algorithm for identifying low-order models based noisy, non-uniformly sampled data and a-priori information is presented in reference [22]. The development of a lifted state-space model and its transformation into a discrete transfer function is presented in [23], in which the parameter estimation is conducted using an auxiliary-model-based recursive least-squares algorithm. Finally, reference [24] made use of the simplified refined instrumental variable method, which allows the numerical simulation and solution in the *MATLAB™Continuous-Time System Identification toolbox* by iteratively using an auxiliary model.

1.3 Goals

The main goal of this thesis is to develop a sensorless control system to regulate the temperature of the test specimen within the Equal Channel Angular Pressing test bed at the Chair of Metal Forming at the Montanuniversität Leoben. To that end, the following objectives are to be achieved:

1. Research and utilisation of procedures and methods to derive dynamic models of the systems required for the sensorless specimen temperature control system and application of the global least squares system identification method for model parameter estimation based on non-uniformly sampled input-output data.
2. Investigation into different controller types as well as the design and simulation of a control system able to regulate the specimen temperature in a way that satisfies the requirements of the Chair of Metal Forming.
3. Implementation of the proposed sensorless control system at the Equal Channel Angular Pressing test bed and experimental validation of the simulation results.

1.4 Thesis Overview

Chapter 2 provides a detailed description of the Equal Channel Angular Pressing test bed and its elements as well as experiment requirements and procedure, for which the sensorless control system is developed. In Chapter 3, dynamic systems, models for dynamic systems as well as their characteristics, representations and mathematical descriptions are introduced. The construction of models of dynamic systems as well as procedures of model parameter estimation are the focus of Chapter 4. In Chapter 5, PID and model predictive control are introduced as control strategies for the specimen temperature control. Chapter 6 includes the derivation of suitable models and control design for the test bed as well as the implementation and experimental validation of the control system. Finally, Chapter 7 provides a summary of the results, the scientific contributions of this thesis and an outlook on future work.

Chapter 2

Process Description

This chapter gives an overview of the test bed concerned in this thesis as well as experiment specifications and requirements for the sensorless control approach.

2.1 The Equal Channel Angular Pressing (ECAP) Test Bed

The Equal Channel Angular Pressing (ECAP) test bed, shown in Figure 2.1, is used to manufacture ultra fine grained (UFG) material using a modified equal channel angular pressing process. The test specimen is pressed around the corner in a tool consisting of two intersecting channels (Figure 2.2). In this setup the specimen is mainly subject to simple shear, not to be confused with pure shear, during the forming process. The shape of the test specimen does not change when pressed through

Fig. 2.1: Equal channel angular pressing test bed ©2021 Chair of Metal Forming

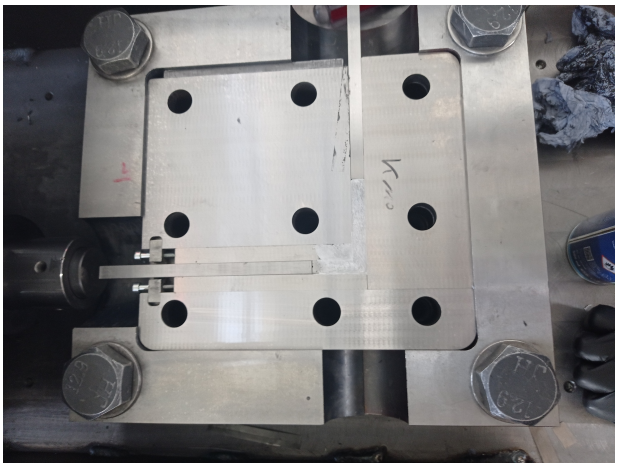


Fig. 2.2: Specimen in forming tool after angular pressing ©2021 Chair of Metal Forming

the tool due to its geometry. This means that individual samples can pass several times through the tool. There is an accumulation of strains, and due to the simple shear, cell blocks are formed from dislocations. Because of the different slip planes and the predetermined shear direction, the

difference in orientation between the cell blocks increases. Given a sufficiently high deformation, i.e. a high enough number of passes of a specimen through the tool, an ultra fine grained (UFG) material structure is formed. The resulting ultra fine grained structures have average grain sizes d in the range of $d = 100\,\text{nm}$ to $d = 1000\,\text{nm}$, resulting in a major influence on the mechanical, physical and chemical properties of the material. The greatest benefit of the ultra fine grained material however is the increase of strength of the material. Pure metals achieve an improvement in strength of up to 800 %, while alloys achieve values of up to 50 % in improvement when compared to their coarse-grained counterparts. As the flow stress is heavily dependent on the temperature, this process is only possible for difficult to form alloys by utilising the test bed's heating system. By heating the specimen, the necessary force can be reduced. The increased temperature also improves the the ductility of the material. Some alloys that the Chair of Metal Forming wants to investigate cannot even be processed at the test bed at room temperature, because they begin to crack during the forming process, making a heating system necessary. In addition to those two major factors, the heating system allows for the investigation of ideal process parameters.

2.2 Overview of the Specimen Heating System

The specimen heating system consists of six heating elements surrounding the specimen (Figure 2.3, Pos. 1) and can be separated into two halves. Three heating elements are placed on the upper side of the specimen receiver, the remaining elements on the lower side, as shown in Figure 2.3. The top half of the heating circuit is noticeably closer to the specimen than the bottom half. The temperatures of the top middle heating element (Figure 2.3, Pos. 2) and the bottom middle heating element (Figure 2.3, Position 3) can be accessed from the internal programmable logic controller (PLC) of the test bed. The temperature of the four outer elements surrounding the middle elements on the upper and lower half (Figure 2.3, Position 4) can only be modified manually.

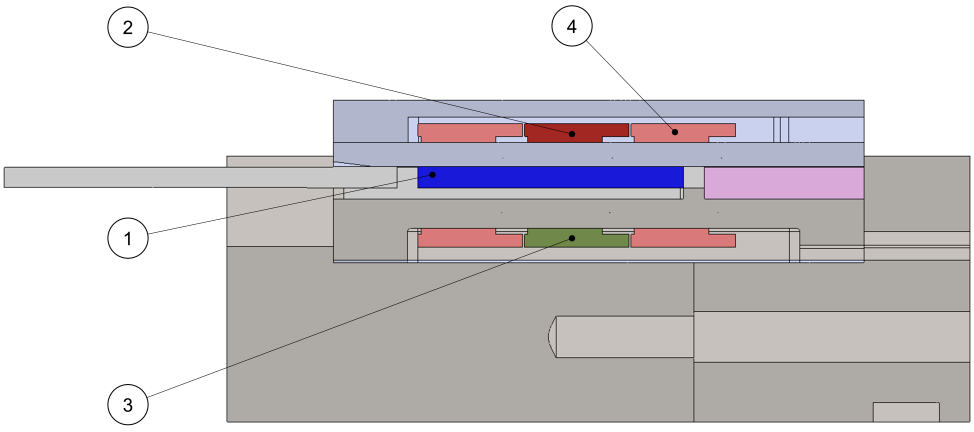


Fig. 2.3: Half-section view of the metal forming system ©2021 Chair of Metal Forming

2.3 Requirement Analysis

The goal of the temperature control system is to reach a homogeneous temperature throughout the specimen. A different set temperature is required from the upper and lower half of the heating system, due to the different distance to the specimen. The main elements to be controlled by the internal controller are the middle heating elements and the upper and lower half, whereas the top middle heating element should follow the set temperature of the bottom middle upper element with a specified fixed temperature difference to keep a homogenous temperature throughout the specimen. The four remaining heating elements are modified manually during the experiment by the staff of the Chair of Metal Forming. The specimen should thus be seen as a thermal system with a single input (temperature of one of the controllable heating elements) and a single output (specimen temperature). Table 2.1 shows all test bed experiment requirements and specifications.

Table 2.1: Experiment requirements and specifications ©2021 Chair of Metal Forming

Requirement	Value	Comment
Specimen temperature range for pressing	100 °C … 200 °C	A maximum of up to 250 °C could be the case in future experiments.
Specimen steady-state temperature tolerance	±2.5 %	The main requirement for the modelling and control is staying within the tolerance of the set temperature in steady-state.
Specimen temperature transient response	As quick as possible	The specimen temperature should be reached as quick as possible for maximum productivity as well as the suppression of temperature induced effects in the alloys.
Specimen material	Aluminum alloys	Different aluminum alloys are used, but all of them have insignificant differences in their thermal behaviour.
Maximum specimen heating time	800 s	The maximum available time for heating up the specimen in the test bed.
Specimen pressing time	120 s	Approximate length of the pressing process.
Specimen cooling	No cooling available	As there is no cooling system, attention should be paid to a time and energy efficient heating curve.
Disturbances	None	There are no external disturbances influencing specimen temperature.
Maximum heating element temperature	500 °C	Heating element temperature should not exceed 400 °C to prevent any damage.

Chapter 3

Dynamic Systems and Models

This chapter gives an introduction to dynamic systems, the methods and possibilities to represent or describe them mathematically, their characteristics and the concept of models.

3.1 Introduction to Dynamic Systems

A system is defined as any collection of interacting elements, which exhibit cause-and-effect relationships between their variables [25, p. 1]. In dynamic systems, the variables are time-dependent, which means that they depend either or both of the initial conditions of the system and its previous inputs [25, p. 2, 26, p. 1]. Many already studied relationships as well as newly discovered phenomenons in technical and non-technical areas of scientific significance can be characterised as dynamic systems [8, p. V]. According to Isermann et al. [27], the mutually effected entities in a dynamic system can also be called processes. The further definition of a process is then a conversion, transport or both, of material, energy or information as well as combinations of those. Also, it can consist of multiple, independent sub-processes. The behaviour of a system is hence defined by that of its processes [27, p. 1]. A system is often represented by a so-called black box, showing the input and output variables, but no relationship between them yet [25, p. 4]. An example is shown in Figure 3.1. Technical systems can be usually characterised by three different variables [28, p. 34]:

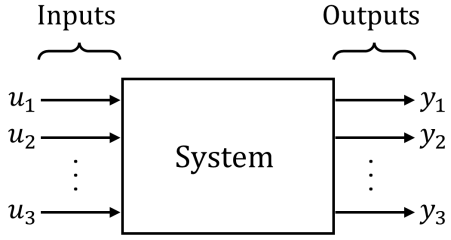


Fig. 3.1: Black box representation of a system [25, p. 4]

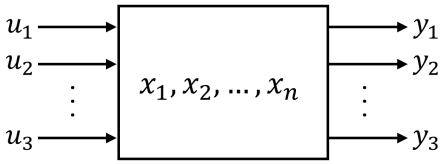


Fig. 3.2: General system representation with inputs, state variables and outputs [25, p. 5]

- Input variables u

  A system features one or several inputs, each representing a function of time [25, p. 5]. Some typical examples for system inputs include a force applied to a mass, a voltage source powering an electrical circuit and a heat source applied to a tank containing liquid [25, p. 5]. Inputs are denoted by $u_1, u_2, \ldots, u_m$ for a total number of m inputs.

- Output variables y

  Outputs are system variables which are to be calculated or measured [25, p. 5]. Typical outputs are the velocity of a mass, voltage across a resistor or the temperature of a liquid inside a tank [25, p. 5]. Outputs are designated by $y_1, y_2, \ldots, y_p$ for a total number of p inputs. In a system, there is always a cause-and-effect relationship between the outputs and the inputs [25, p. 5].

- State variables x

  According to Close et al. [25], a system can also be described by a set of state variables that may include one or more of the outputs (Figure 3.2). With knowledge of the state variables' values at any reference time t_0 and knowledge of the inputs for all $t \geq t_0$, it is possible to determine the outputs and state variables for all $t \geq t_0$. However, the state variables must be chosen in a way that fulfils this condition. The state variables must also be independent, meaning that one state variable cannot be represented as an algebraic function of the others [25, p. 5]. State variables are denoted by $x_1(t), x_2(t), \ldots, x_n(t)$ for a total number of n state variables.

Systems with only one input variable and one output variable are called single-variable systems or single-input, single-output (SISO) systems [29, p. 3]. Systems with several input variables and one or more output variables are referred to as multi-variable systems or multiple-input, multiple-output (MIMO) systems, if at least one output is dependent on multiple inputs [29, p. 3, 30, pp. 31–32]. A system is also a multi-variable system, if it possesses only one input, but several outputs [30, pp. 31–32].

3.2 System Identification and Modelling of Dynamic Systems

When interacting with a system, an observer usually needs an idea on how the output signals relate to the input signals. This assumed relationship between the variables is called a model of the system [31, p. 6]. Although there are mental and graphical models available for easily comprehensible relationships like driving a car, where the actions of the driver ensue a response of the car, models that describe the relationships between the system variables as mathematical expressions like differential or difference equations are usually required for more advanced applications [31, p. 6]. These mathematical models, or simply models for short, represent a description of a system

in terms of equations [25, p. 2]. They can be further characterised by the properties signifying the type of differential or difference equation used [31, p. 6].

3.2.1 Construction of Models

There are two general routes available for creating a model of an observed dynamic system [31, p. 7, 8, p. 5]. Firstly, there is the theoretical procedure, also known as theoretical modelling [27, p. 3]. Here, the observed system is split up into subsystems with properties that are well understood physically and can be described easily mathematically [8, p. 5]. The whole system is then created by joining all the subsystems [31, p. 7]. The second route is the experimental procedure, which is known as experimental analysis or identification [27, p. 3]. Here, input and output signals from the system are recorded and analysed to derive a fitting model [31, p. 7].

3.2.2 Verification of Models

After creation, the process of making a model reliable, valid and accurate often proves difficult, which is further elaborated by Ljung et al. [32]: Confidence in the model can only be obtained by the process of verification or validation of the model in comparison to the real system, in which the difference between model and system behaviour has to be analysed. A model is also characterised by a specific domain of validity, corresponding to a predefined demanded accuracy and the limits of system variables and parameters. As an example, the typical linearised model of a pendulum is only valid for small angles [32, p. 17].

3.2.3 Purpose of Models

The construction a model of a dynamic system usually aims to provide a better understanding of the cause-and-effect principle observed in the system [8, p. 4]. For this purpose, the best possible correspondence in relation to the actual system behaviour is sought, meaning that the model should contain all essential properties of the observed system [8, p. 1]. However, the applications of dynamic models are very diverse, as can be shown by examples in two relevant fields presented in [8, pp. 2–3]:

- Models in control engineering

  In control engineering, models of the controlled system are an integral part of all model-based controller design methods (e.g. methods for designing optimal, predictive and adaptive con-

trollers). The determination of state variables by means of an observer or a Kalman filter is also based on the use of system models.

- Models in measurement technology

  Dynamic models are often an integral part of intelligent or virtual sensors as they can enable the indirect determination of quantities that are otherwise not directly measurable.

3.3 Classification of Models and Systems

Mathematical models derived for specific systems usually have varying characteristics, which depend on the observed systems as well as on the used methods [32, p. 19]. In this section, the definitions in respect to system also apply to models.

3.3.1 Linear and Nonlinear

The most general separation is between linear and nonlinear systems [8, p. 7]. In a dynamic system

$$\boldsymbol{y}(t) = T\left[\boldsymbol{u}(t)\right], \tag{3.1}$$

the link between the input $\boldsymbol{u}(t)$ and the output $\boldsymbol{y}(t)$ is represented by an operator T, so that for every real input $\boldsymbol{u}(t)$ exist a real output $\boldsymbol{y}(t)$ [30, p. 22]. A system is then linear if and only if the superposition principle

$$\sum_{i=1}^{p} c_i \boldsymbol{y}_i(t) = T\left[\sum_{i=1}^{m} c_i \boldsymbol{u}_i(t)\right] \tag{3.2}$$

applies to any linear combination of inputs $\boldsymbol{u}_i(t)$ with real constants c_i while using the operator T. If the superposition principle is not fulfilled, then the system is nonlinear [30, pp. 23–24].

3.3.2 Time-Variant and Time-Invariant

Systems can be separated into the categories time-variant and time-invariant. If the system parameters are time-dependent, the model is called time-variant [28, p. 18]. If the system parameters are constant regardless of the point in time, the system is called time-invariant [28, p. 18]. One important case occurs if the system is time-invariant as well as linear: These types of system are then called linear time-invariant (LTI) systems [28, p. 18]. Utilising the operator T from Equation 3.1, a time-invariant system can be expressed by

$$\boldsymbol{y}(t-t_0) = T\left[\boldsymbol{u}(t-t_0)\right], \tag{3.3}$$

which shows that a time shift t_0 of the input results in the same time shift of the output without distorting it [30, p. 29].

3.3.3 Lumped and Distributed Parameters

Systems can be classified as lumped or distributed parameter systems, with definitions provided by Ljung et al. [32] and Unbehauen [30]. A system can be thought of as composed by a finite number of idealised individual elements. In that case, the events are described by a finite number of changing variables and the corresponding systems are called systems with lumped parameters. They are expressed by ordinary differential equations. If a system possesses an infinite number of infinitely small individual elements, then it represents a system with distributed parameters. Such systems are expressed mathematically by partial differential equations [30, p. 28, 32, p. 21].

3.3.4 Continuous Time and Discrete Time

In continuous time systems, every occurring signal is a continuous function of time, meaning that every signal is assigned a unique value at each time value in an observed time interval [33, p. 101]. A system is called a discrete time system, if at least one discrete time signal occurs in the system, a discrete time signal being produced when a signal is only measured (sampled) at certain time points [33, p. 101]. Continuous time systems are often described by differential equations, while discrete time systems are typically expressed by difference equations [32, p. 20]. The choice of representing a model in either continuous time or discrete time domain usually depends on the context and the use of the model, as described in [32]: If the model is created on basis of the laws of nature, a time continuous representation is often easier due to the availability of suitable mathematical expressions. For a model created based on measurement values, the discrete time representation is the obvious choice as the data is available in discrete time form [32, p. 71].

Sampling

When a digital computer or controller is used for processing data, continuous measurements are converted into digital form by an analogue-to-digital (A/D) converter which samples the signal, i.e. measures it at specified time instances [34, p. 300]. The time interval between successive samples is called the sampling period Δt, which corresponds to the sampling rate $f_s = 1/\Delta t$ [34, p. 300]. In Figure 3.3, the value of $\boldsymbol{y}(t)$ is sampled at equal intervals of time Δt, so that $\boldsymbol{y}(t)$ is described by a sampled signal $\boldsymbol{y}^*(t)$ shown in Figure 3.4 consisting of a series of impulses: [35, p. 2]

$$\boldsymbol{y}(0), \boldsymbol{y}(\Delta t), \boldsymbol{y}(2\Delta t), \boldsymbol{y}(3\Delta t), \ldots, \boldsymbol{y}(k\Delta t), \ldots \tag{3.4}$$

It can be seen from Equation 3.4 that by sampling only a limited description of the continuous time function $y(t)$ is retained [35, p. 2]. For instance, the value of $y(t)$ between the sampling times is not available, so a certain amount of information hast been lost in the process [35, p. 2]. While the intermediate values of $y(t)$ can be interpolated with reasonable accuracy for well-behaved functions, it holds in general true that the shorter the sampling interval, the better correspondence between continuous time and discrete time model is achieved [32, p. 75, 35, p. 2].

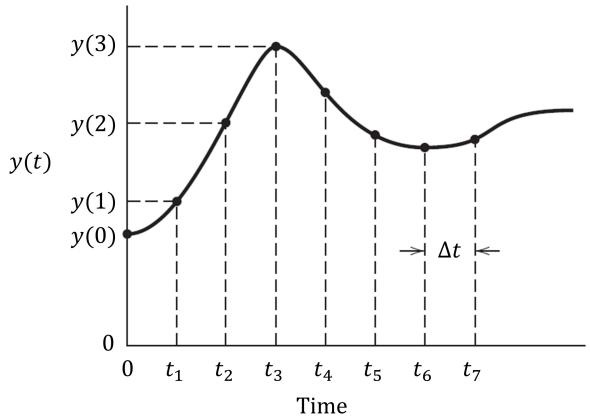


Fig. 3.3: Continuous signal $y(t)$ and sampling intervals Δt [34, p. 301]

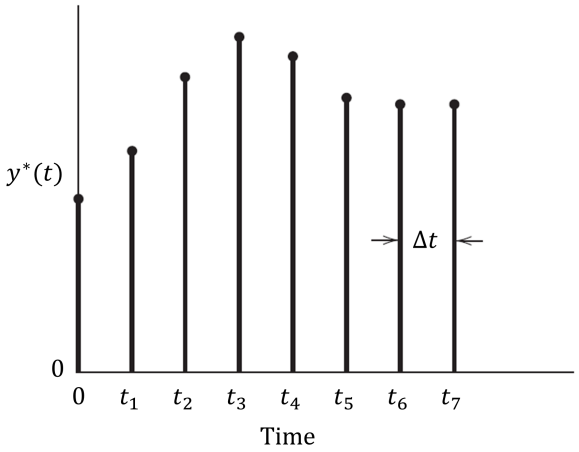


Fig. 3.4: Resulting sampled signal $y^*(t)$ [34, p. 301]

After processing, a digital-to-analogue (D/A) converter is used for transforming a series of pulses from the digital computer or controller back into a continuous signal [34, p. 300]. The D/A converter usually also contains a zero-order hold (ZOH) to convert the digital signal into a staircase function, i.e. the output signal from the zero-order hold is held constant for one sampling period until the next sampling instant [34, p. 300].

Aliasing

The sampling rate must be fast enough to not lose significant process information. As an example provided by [34], if a sinusoidal signal is sampled at a rate of $4/3$ samples per period, the reconstructed signal appears as a sinusoid with a much longer period than the original signal. This is known as aliasing [34, p. 302]. According to the Shannon theorem, the oscillation of the highest frequency must be sampled at least twice per period to recover the original signal, meaning the sampling frequency must be at least twice the frequency of the sine wave [36, p. 30, 34, p. 302].

Analogue and Digital Systems

It is important to note the difference between the concept of continuous time and analogue signals as well as the difference between discrete time and digital signals, as they are often confused. Definitions are provided by Lathi [37]: The terms analogue and digital qualify the nature of the signal amplitude. While the amplitude of an analogue signal can take on any value in a continuous range, the amplitude of a digital signal can only assume a finite number of values. In contrast, the concepts of continuous time and discrete time qualify the nature of a signal along the time axis [37, p. 78].

3.3.5 Order

A system described by an ordinary differential equation of order n is called a system or process of order n and can be represented by

$$f(t) = a_0 x + a_1 \frac{\mathrm{d}x}{\mathrm{d}t} + a_2 \frac{\mathrm{d}^2 x}{\mathrm{d}t^2} + \ldots + a_n \frac{\mathrm{d}^n x}{\mathrm{d}t^n} \tag{3.5}$$

[38, p. 29]. It is important to note that an ordinary differential equation of order n is equivalent to a set of n first order ordinary differential equations [38, p. 29].

3.4 Description and Analysis in the Frequency Domain

In this section, the Laplace transform and its use in obtaining the response of dynamic systems expressed by linear, time-invariant differential equations is presented [26, p. 247]. In addition, the z-transform is introduced as a tool of discrete time system analysis [34, p. 307].

3.4.1 Laplace Transform

The Laplace transform is an elegant mathematical procedure for solving linear or linearised differential equations and serves as a definite reversible correlation of an original function in time domain and a corresponding function in in the domain of a complex variable [30, p. 51, 26, p. 247, 38, p. 29]. Its importance especially comes from the fact that it converts an ordinary differential equation into an algebraic equation, which is much easier to handle and manipulate [26, p. 247]. In control engineering, the Laplace transform is used for the creation of simple continuous input–output models, thus enabling the investigation of the influence of external variables on the observed process [38, p. 29].

Due to the superposition property of linear time-invariant systems, their response can be found by breaking the input into several components and then summing up the system response from all the separate parts [37, p. 330]. In frequency domain analysis, the input is broken up into exponentials of the form e^{st}, where the parameter s is the complex frequency of the signal e^{st} [37, p. 330]. The Laplace transform of a function $f(t)$ to the complex function $F(s)$ is then

$$\mathscr{L}\{f(t)\} = F(s) = \int_0^\infty f(t)e^{-st}\mathrm{d}t, \tag{3.6}$$

while the inverse operation is called inverse Laplace transform and converts the complex function $F(s)$ into the time domain function $f(t)$ [26, pp. 247–248],

$$\mathscr{L}^{-1}\{F(s)\} = f(t). \tag{3.7}$$

To compute the inverse Laplace transform of $F(s)$, it is usually helpful to expand $F(s)$ to a sum of simple rational fractions and subject each of them separately to the inverse transform [38, p. 31].

3.4.2 z-Transform

The z-transform method of analysis of discrete time systems parallels the Laplace transform method of analysis of continuous time systems, with some minor differences [37, p. 488]. Just like the Laplace transform converts differential into algebraic equations, the z-transforms changes difference equations into algebraic ones, thus simplifying their analysis [37, p. 488].

An ideal, periodic sampler converts a continuous signal $f(t)$ into a discrete signal $f^*(t)$ at k equally spaced intervals of time T_s. It is mathematically convenient to consider impulse sampling, where $f^*(t)$ is the sampled signal formed by a series of impulses or Dirac delta functions based on the value of $f(t)$ at each sampling instant: [34, pp. 308–309, 33, p. 107]

$$f^*(t) = f(t)\sum_{k=0}^{\infty}\delta(t-kT_s) = \sum_{k=0}^{\infty}f(kT_s)\delta(t-kT_s) \tag{3.8}$$

The application of the Laplace transform on Equation 3.8 yields [33, p. 107]

$$F^*(s) = \mathscr{L}\{f^*(t)\} = \sum_{k=0}^{\infty}f(kT_s)e^{-kT_s s}. \tag{3.9}$$

In the next step, defining the z-transform variable as $z := e^{Ts}$ and substituting it into Equation 3.9 results in the z-transform

$$\mathscr{Z}\{f(kT_s)\} = F_z(z) = \sum_{k=0}^{\infty}f(kT_s)z^{-k} \tag{3.10}$$

of $f(k)$ [33, p. 109]. The inverse transformation is denoted by $\mathscr{Z}^{-}1$,

$$f^*(t) = \{f(kT_s)\} = \mathscr{Z}^{-}1\{F(z)\}. \tag{3.11}$$

However, the conversion from z-transform $F(z)$ to the continuous function $f(t)$ is not unique, as explained in [38]. As $F(z)$ is the z-transform of the continuous function $f^*(t)$ obtained by a sampler with period T_s and the zero-order holder of the function $f(t)$. The resulting information loss between two sampling instants prohibits the reconstruction of $f(t)$. Nevertheless, there are several methods available for obtaining $\{f(kT_s)\}$. In addition to numerical computation, the residual method, polynomial division or the expansion as a sum of rational fractions are valid options [38, p. 383].

Approximated z-Transform

Although the determination of an exact z-transfer function for the discrete representation of a continuous system is only possible for a few forms of the input signal, a discrete system representation also exists for an arbitrary input [33, p. 126]. These representations however can only apply approximately, which is why it is referred to as an approximated z-transfer function from as the result of an approximated z- transformation [33, p. 126].

1. Euler approximation

   By making use of the Euler method [33, p. 104], a difference equation describing the observed continuous model is derived. This difference equation is then be easily subjected to z-transform and results in [33, p. 127]

   $$s \approx \frac{z-1}{T_s z}. \tag{3.12}$$

2. Tustin approximation

   A more precise approximation is gained by series expansion of $\ln(z)$ in $s = \frac{1}{T}\ln(z)$:

   $$s = \frac{2}{T_s}\left[\frac{z-1}{z+1} + \frac{1}{3}\left(\frac{z-1}{z+1}\right)^3 + \frac{1}{5}\left(\frac{z-1}{z+1}\right)^5 + \ldots\right] \tag{3.13}$$

   By stopping after the first term of Equation 3.13, the Tustin-Formula

   $$s \approx \frac{2}{T_s}\frac{z-1}{z+1} \tag{3.14}$$

   is derived, which works well in calculating $G(z)$ from $G(s)$ for small values of T_s [33, p. 127].

Forward and Backward Shift

As shown in [39], the z-transform creates a power series in z from a given sequence of numbers $f(k)$. For example, the z-transform of the series $f(k) = (1,-1,1,-1,\ldots)$ is

$$F(z) = 1 - z^{-1} + z^{-2} + z^{-3} + \ldots, \tag{3.15}$$

which shows that z^{-1} signifies a time shift by one sampling interval. Thus, z^{-1} is denoted as backward shift-operator while z is called forward shift-operator [39, pp. 501, 502].

3.5 Representation of Dynamic Systems and Models

Although system analysis may involve analytical or numerical methods, the entire system in question must be available in some sort of convenient standard format [26, p. 115]. This section introduces some of the most common formats.

3.5.1 Block Diagrams

One of the simplest depictions of a system are block diagrams, in which a system is represented by one or several different block diagram models, where arrows between the blocks signify interaction [32, p. 33]. They are logical decompositions of the functions within a system and showcase the relationship between them [32, p. 33]. While they excel at the logical description of a system, they usually neglect the physical aspects [32, p. 36]. As an example, Figure 3.5 showcases a simple block diagram for a system in which the outflow of the tank depends on the inflow. Block diagram

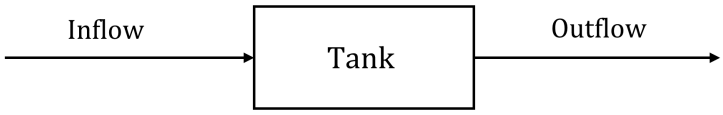


Fig. 3.5: Simple block diagram of a tank [32, p.34]

models usually provide an important starting point when trying to build a mathematical model of a system and they are particularly useful for structuring systems, especially if it contains many or more complicated relationships [32, p. 36].

3.5.2 Transfer Functions

With the transfer function of a system it is possible to directly calculate the system's output with a known input value [30, p. 71]. Figure 3.6 shows a dynamic model with a single input u and single output y. If $u(t)$ and $y(t)$ are representing a linear continuous time-invariant system with lumped

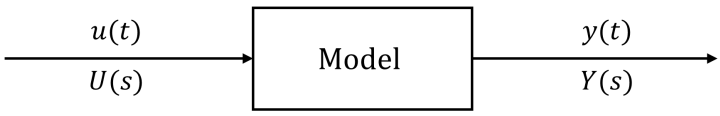


Fig. 3.6: Input-output model [34, p. 54]

parameters, the system can be described by an ordinary differential equation

$$\sum_{i=0}^{n} a_i \frac{\mathrm{d}^i y(t)}{\mathrm{d}t^i} = \sum_{j=0}^{m} b_j \frac{\mathrm{d}^j u(t)}{\mathrm{d}t^j}, \tag{3.16}$$

as long as dead-time is not considered [30, p. 71]. By setting all initial conditions to zero and applying the Laplace transform on both sides, Equation 3.16 becomes

$$Y(s) \sum_{i=0}^{n} a_i s^i = U(s) \sum_{j=0}^{m} b_j s^j, \tag{3.17}$$

which can be put as the ratio of the system output to its input, resulting in the transfer function

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_0 + b_1 s + \ldots + b_m s^m}{a_0 + a_1 s + \ldots + a_n s^n}. \tag{3.18}$$

of the system. Numerator and denominator of $G(s)$ are rational functions in s, whose coefficients only depend on the structure and parameters of the system [30, p. 71]. In analogy to the transfer function $G(s)$ of continuous systems, the discrete transfer function

$$G(z) = \frac{Y(z)}{U(z)} = \frac{b_0 + b_1 z^{-1} + \ldots + b_m z^{-m}}{a_0 + a_1 z^{-1} + \ldots + a_n z^{-n}} \tag{3.19}$$

is defined as the quotient of the z-transform of the output and input [39, p. 507, 34, p. 309].

A transfer function is called strictly proper if $m < n$ [40, p. 1]. The system described by such a transfer function is causal, meaning that the effect does not anticipate the cause [40, p. 1]. As all physical systems are causal, systems with improper transfer functions are physically unattainable [30, p. 71, 40, p. 1].

3.5.3 State-Variable Equations

State variables are one of the standard methods of representing a system and a set of dynamic variables of a system, which completely define all of its characteristics [26, p. 115]. The state of a system is the vector of variables necessary for the description of the future behaviour of the system, if the initial conditions of those variables are known [41, p. 12]. Thus, the state variables are dynamic variables that add up to the state of the system [26, p. 115]. State variables are usually denoted by $x_1, x_2, x_3, ..., x_n$ where n is the total number of state variables directly corresponding to the order of the system. The state variable equations are then a set of n differential equations

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, x_2, ..., x_n, u_1, u_2, ..., u_m) \\ \dot{x}_2 &= f_2(x_1, x_2, ..., x_n, u_1, u_2, ..., u_m) \\ &\vdots \\ \dot{x}_n &= f_n(x_1, x_2, ..., x_n, u_1, u_2, ..., u_m) \end{aligned} \tag{3.20}$$

in which each is the first-order derivative $\dot{x}_i$ of the corresponding state variable x_i and m inputs [26, p. 115]. The right-side functions f_i can be linear or nonlinear, but they only depend on the input variables or the state variables [26, p. 116].

3.5.4 State-Space Representation

In case that the mathematical equations describing the system model are linear, all right-side functions f_i in Equation 3.20 are linear. That means, that the resulting linear state-variable equations can be rearranged into a convenient matrix-vector format, which is known as state-space representation [26, pp. 116, 119]. In this form, the relationship between input and output signals and also, potentially disturbance, is specified as a system of first order differential or difference equations [31, p. 93]. The fact that state-space representation allows for a more easily incorporation of physical insights into a model makes it especially useful [31, p. 93].

The state-space representation describes a system of n-th order by n differential equations of first order [28, p. 39]. For arbitrary systems with multiple outputs and inputs the universally valid state space representation is given by the state equation

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\,\boldsymbol{x}(t) + \boldsymbol{B}\,\boldsymbol{u}(t) \tag{3.21}$$

and the output equation

$$\boldsymbol{y}(t) = \boldsymbol{C}\,\boldsymbol{x}(t) + \boldsymbol{D}\,\boldsymbol{u}(t) \tag{3.22}$$

[28, p. 41, 26, p. 119]. The state equation represents the system dynamics, meaning that $\boldsymbol{A}$ and $\boldsymbol{B}$ contain the linear coefficients from the differential equations composing the mathematical model, while the output equation is an algebraic linear mapping between state and input variables as well as the outputs [26, p. 121]. The state and output equation consist of the following vector and matrix definitions that are necessary to enable their complete and compact matrix-vector representation: [26, p. 121, 28, p. 41]

- $\boldsymbol{x}(t)$ is the $n \times 1$ state vector composed of the n state variables x_i,
- $\boldsymbol{u}(t)$ the $r \times 1$ input vector containing r input variables,
- $\boldsymbol{y}(t)$ the $m \times 1$ output vector consisting of m output variables,
- $\boldsymbol{A}$ the state or system matrix with dimensions of $n \times n$,
- $\boldsymbol{B}$ the $n \times r$ input matrix,
- $\boldsymbol{C}$ the $m \times n$ output matrix and
- $\boldsymbol{D}$ the direct-link matrix.

The state space representation can be applied to discrete systems in the same way as to continuous systems. Similar to how a linear differential equation of n-th order can be converted into a system of n first order linear differential equations, a n-th order linear difference equation can be represented as a system of n first order linear difference equations [33, p. 164]. In matrix notation, the state space representation for a linear, time-invariant, discrete multivariate system therefore is given by [33, p. 164]

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\,\boldsymbol{x}(k) + \boldsymbol{B}\,\boldsymbol{u}(k) \tag{3.23}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\,\boldsymbol{x}(k) + \boldsymbol{D}\,\boldsymbol{u}(k). \tag{3.24}$$

From Transfer Function to State-Space Representation and Controllable Canonical Form

For an n-th order system, n states have to be defined. This choice is ambiguous [42, p. 16]. The deduction of the state representation from a transfer function of a linear, time-invariant system is therefore of interest, as there exists a multitude of equivalent solutions [33, pp. 34–35]. Given an arbitrary n-th order transfer function

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_0 + b_1 s + \ldots + b_n - 1 s^{n-1} + b_n s^n}{a_0 + a_1 s + \ldots + a_n - 1 s^{n-1} + a_n s^n} \tag{3.25}$$

of a linear single-input single-output system, which can be standardised in a form that the coefficient of the highest power $a_n = 1$, the procedure of transforming it into a the controllable canonical form is shown by [33, pp. 22–25]. The transfer function shown in Equation 3.25 is the result of the Laplace transform of a differential equation of the form

$$\frac{\mathrm{d}^n y}{\mathrm{d}t^n} + a_{n-1}\frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n-1}} + \ldots + a_1 \dot{y} + a_0 y = b_0 u + b_1 \dot{u} + \ldots b_n \frac{\mathrm{d}^n u}{\mathrm{d}t^n} \tag{3.26}$$

and in the special case that there are no derivatives of the input u

$$\frac{\mathrm{d}^n y}{\mathrm{d}t^n} + a_{n-1}\frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n_1}} + \ldots + a_y \dot{y} + a_0 y = b_0 u. \tag{3.27}$$

By rearranging Equation 3.27 according to the highest derivative

$$\frac{\mathrm{d}^n y}{\mathrm{d}t^n} = b_0 u - \left(a_{n-1}\frac{\mathrm{d}^{n-1} y}{\mathrm{d}t^{n_1}} + \ldots + a_y \dot{y} + a_0 y \right), \tag{3.28}$$

a corresponding block diagram shown in Figure 3.7 can be created. If the outputs of the integrators

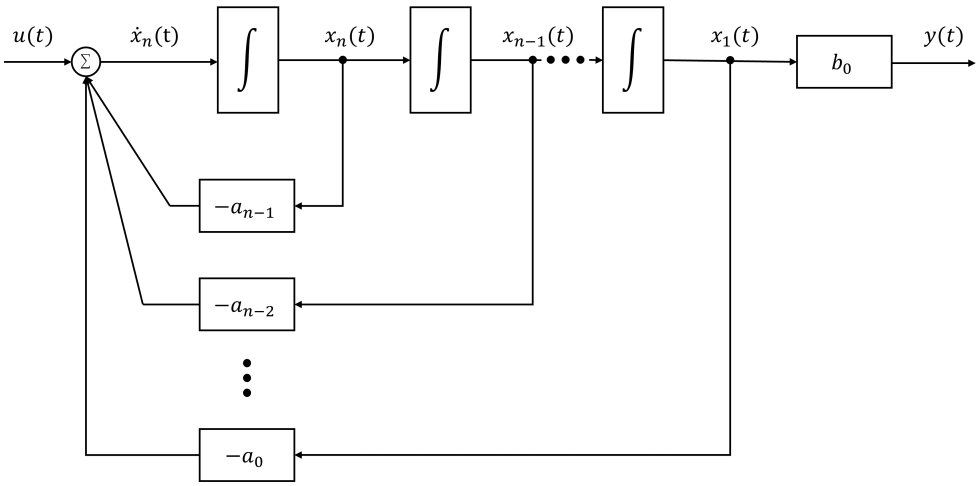


Fig. 3.7: Block diagram of the controllable canonical form [42, p. 18, 33, p. 23]

shown in Figure 3.7 are chosen as the states, the state equations directly follow as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= x_3 \\ &\vdots \\ \dot{x}_n &= u - a_0 x_1 - a_1 x_2 - \ldots - a_{n-1} x_n \end{aligned} \tag{3.29}$$

and the output of the block diagram is

$$y = b_0 x_1. \tag{3.30}$$

Summing up all state variables x_i to the state vector $\boldsymbol{x}$, the state-space representation is given by

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \dots & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 & \dots & -a_{n-1}, \end{bmatrix} \quad (3.31) \qquad \boldsymbol{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1, \end{bmatrix} \quad (3.32)$$

$$\boldsymbol{C} = \begin{bmatrix} b_0 & 0 & 0 & \dots & 0, \end{bmatrix} \quad (3.33) \qquad \boldsymbol{D} = 0. \quad (3.34)$$

The structure $\boldsymbol{A}$ is known as controllable canonical form, which offers the following advantages: [33, p. 23, 42, p. 21]

- The numerator coefficients of the transfer function appear directly in $\boldsymbol{C}$
- The denominator coefficients of the transfer function appear directly in $\boldsymbol{A}$
- In $\boldsymbol{A}$, only the n elements in the last line are dependent on the system
- The input matrix $\boldsymbol{B}$ is completely independent of the system properties

If this procedure is extended to an arbitrary case

$$y = b_0 x_1 + b_1 \dot{x}_1 + \dots b_n \frac{\mathrm{d}^n x_1}{\mathrm{d}t^n}, \quad (3.35)$$

in which there are derivatives of the input variable u and the first state variable x_1 is chosen in a way that results in Equation 3.35 for y, then the same structure of matrix $\boldsymbol{A}$ is achieved as in Equation 3.31 [33, p. 24]. This is shown by applying the Laplace transform to Equation 3.35, resulting in

$$Y(s) = X_1(s) \left[b_0 + b_1 s + \dots + b_n s^n \right], \quad (3.36)$$

which can in turn be substituted into Equation 3.25, yielding

$$\frac{X_1(s)}{U(s)} = \frac{1}{a_0 + a_1 s + \dots + a_{n-1} s^{n-1} + s^n}. \quad (3.37)$$

This corresponds exactly to the special case with $b_0 = 1$. The definition of state variables according to Figure 3.7 is therefore directly applicable and the equations 3.28 are universally valid, i.e. $\boldsymbol{A}$ and $\boldsymbol{B}$ are unchanged [33, p. 24]. Adding equation 3.35 to Figure 3.7, y for the arbitrary case is

$$\begin{aligned} y &= b_0x_1 + b_1x_2 + \ldots + b_{n-1}x_n + b_n\dot{x}_n \\ &= b_0x_1 + b_1x_2 + \ldots + b_{n-1}x_n + b_n[u - a_0x_1 - \ldots - a_{n-1}x_n] \\ &= (b_0 - b_na_0)x_1 + (b_1 - b_na_1)x_2 + \ldots + (b_{n-1} - b_na_{n-1}x_n + b_nu) \end{aligned} \tag{3.38}$$

and in conclusion, for a single-input single-output system of n-th order

$$G(s) = \frac{Y(s)}{U(s)} = \frac{b_0 + b_1s + \ldots + b_{n-1}s^{n-1} + b_ns^n}{a_0 + a_1s + \ldots + a_{n-1}s^{n-1} + s^n}, \tag{3.39}$$

the state-space representation in controllable canonical form is given by: [33, p. 25]

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 1 & 0 & \ldots & 0 \\ 0 & 0 & 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & \ldots & 1 \\ -a_0 & -a_1 & -a_2 & -a_3 & \ldots & -a_{n-1} \end{bmatrix} \tag{3.40}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix} \tag{3.41}$$

$$\boldsymbol{C} = \begin{bmatrix} (b_0 - b_na_0) & (b_1 - b_na_1) & \ldots & (b_{n-1} - b_na_{n-1}) \end{bmatrix} \tag{3.42}$$

$$\boldsymbol{D} = b_n \tag{3.43}$$

Chapter 4

Model Construction and Parameter Estimation

This chapter introduces background and procedures to derive suitable models for existing systems. Methods to identify parameters for determined model structures are also discussed.

4.1 Preliminaries

4.1.1 Modelling and System Identification Procedure

There are two paths available to derive mathematical models of dynamic systems, theoretical modelling and experimental modelling, also termed system identification [27, pp. 1–2, 15, p. 1]. In theoretical modelling, the model is obtained by applying methods from calculus to equations derived from physical laws and in many cases, simplifying assumptions is required to make the mathematical treatment of a system feasible [27, p. 3]. The characteristics of this approach are given by [43] as follows. Major advantages of a full theoretical model are reliability and flexibility. However, a lack of precise knowledge and required assumptions often undermine this approach. In addition, pure theoretical modelling of systems is in many cases impossible or unattractive, due to their often extremely complex nature [43, p. 361]. If sufficient experimental or operational data is available, data-based system identification is an option [15, p. 1]. This approach is suited for any system and typically results in relatively simple models that describe the system behaviour with acceptable quality [15, p. 1]. According to [43], a data based model is guided by usefulness rather than truthfulness. This means it is characterised by a reasonably good input–output relationship within a certain range, without any exact or meaningful connection. Disadvantages include misleading conclusions from missing or faulty data and the fact that extensive validations are necessary for model viability confirmation [43, p. 361]. Theoretical and experimental models often complement each other, with knowledge passing from theoretical insights to the experimental results and the other way around [27, p. 4]. The general system identification procedure for the derivation of data-based models while incorporating available system knowledge is summed up in Table 4.1.

Table 4.1: General system identification procedure [43, p. 361, 8, p. 56]

Procedure step	Description
1. Determine modelling objectives	Define the model purpose and its requirements.
2. Gather prior knowledge	Collect theoretical understanding and information on operating conditions, variable ranges and limits and potential noise or disturbances.
3. Design experiment and gather data	Specify any required input signals, measurement settings and excite the system for the purpose of measuring sufficiently informative data.
4. Determine a model set	Define a general model structure in form of a arbitrary transfer function, a set of differential equations etc. and corresponding parameters.
5. Estimate model parameters	After input and output measurement data is gathered, identification takes place via parameter estimation.
5.1. Define a model structure	Choose a specific model structure out of the defined model set to estimate its parameters.
5.2. Define a quality criterion	Specify an error or quality functional with which the quality of the model output can be compared with the actual system output.
5.3. Choose an estimation method	Select an estimation method to determine the model parameters in terms of the specified quality functional and calculate the parameters.
5.4. Validate the model structure	Test the model structure using verification procedures and execute the parameter estimation again if necessary.
6. Validate the model	Test if the final obtained model meets all requirements on validation data not used in the model construction.

4.1.2 Continuous Time and Discrete Time System Identification

A known continuous time transfer function is required for many modelling and control design methods [19, p. 1]. A situation encountered especially often is the identification of such a continuous time model from discrete-time data, a problem for which two approaches exist [18, p. 709]:

1. Direct approach

   Direct methods identify continuous time models directly from discrete time data [17, p. 2]. Discrete time approximations are used for signals and operators in the continuous time model, resulting in an approximate continuous time model with corresponding parameters [18, p. 709].

2. Indirect approach

   Indirect methods first identify discrete time models and then convert them to equivalent continuous models [17, p. 2]. First, a continuous time model is transformed into an exact discrete-time model used for identification and then, the desired parameters of the original model are obtained by transferring the identified discrete time model back to continuous time [18, p. 709].

One disadvantage of the indirect approach is that parameter translation between continuous time description and discrete time representation as well as vice versa is not trivial [18, p. 709]. There is no undisputed algorithm for parameter transfer, as problems arise in the system zeros translation from discrete time to continuous time [19, p. 1]. In addition, indirect approach methods often include enormous computational complexity and costly minimisation algorithms [18, p. 710]. Thus, direct approach methods have attracted a lot of interest as of today [18, p. 709]. However, compared to discrete time model identification, continuous model identification requires a differentiation of input and output signals, which often enhances the signal noise due to the methods used [17, p. 2].

4.1.3 Well-Known Transfer Elements

A wide range of methods exists for determining the characteristics of a rational transfer function with or without dead-time directly from the measured step response of the system [30, p. 307]. For example, one of those is the method of Thal-Larsen [44], which utilises a third order delay model with dead time [30, p. 307]. The main idea of all procedures lies in approximating a given transfer function through combining simple and well-known transfer elements, of which a selected amount are shown in Table 4.2 [8, p. 15].

Table 4.2: Important well-known transfer elements [30, pp. 107–108]

Transfer element	Time domain $y(t) =$	Transfer function $H(s) =$
Proportional (P)	$K_P\, u(t)$	K_P
Integrating (I)	$\frac{t}{T_I} u(t)$	$\frac{1}{sT_I}$
Differentiating (D)	$T_D \delta(t)$	sT_D
First order delay (PT_1)	$K_R \left(1 - e^{-\frac{t}{T}}\right) u(t)$	$\frac{K_R}{1+sT}$
Second order delay (PT_2)	$K_R \left(1 - \frac{T_1}{T_1 - T_2} e^{-\frac{t}{T_1}} + \frac{T_2}{T_1 - T_2} e^{-\frac{t}{T_2}}\right) u(t)$	$\frac{K_R}{(1+sT_1)(1+sT_2)}$

4.1.4 Experiment Design for Time Domain Analysis

Bohn and Unbehauen [8] suggest that due to the multitude of models and methods, there exists no standard experimental procedure for system identification that provides guaranteed results. In any case however, the sampling rate and measurement duration must be specified before performing the measurements used for the identification. Also, appropriate input signals must be selected and generated [8, p. 495].

Input Signal

A series of input signal considerations is presented by Isermann et al. [27]. Signals from normal operation or special test signals can be utilised. Normal operation signals are suitable, if they excite the process sufficiently in the range of the interesting dynamics. However, it is usually advantageous to use artificial test signals, whose properties are exactly known and which can be specifically tuned. In any case, the limitations of the process have to be considered, such as maximum allowable change rate and amplitude of input and output, as well as the maximum measurement time. The shape of the test signal is limited by the actuator, as it is the bottleneck for the maximum speed of change and frequency of the input signal. Suitable test signals are defined by exciting the interesting process eigenvalues continuously and strongly in relation to noise [27, pp. 565–566]. Regarding the shape of the signal, it is important to consider [27, p. 567]:

- The test signal height should be as large as possible.
- High frequencies are excited stronger, as the edge steepness increase.
- Medium to high frequency excitation increases when the pulse width decreases.
- The excitation of low frequencies increases with the broadness of the pulses.

Based on this considerations, pseudo-random binary signals and generalised random binary signals are well suited for identification and parameter estimation purposes [27, p. 567].

Sampling Rate

For acquiring output data by measuring the values at a series of discrete time points, a suitable sampling rate has to be defined. The maximum acceptable time interval between the measurement points of the sampling frequency is given by the Shannon theorem [30, p. 300]. Other than that, the choice of the sampling rate should mainly depend on: [27, p. 569]

- Application of the discrete time model

  The sampling rate should be chosen according to that of the control algorithm, if the model is subsequently used for digital controller design [27, p. 568].

- Model fidelity

  According to [27], the absolute values of estimated parameters change with the sampling time, which is critical, as even small absolute errors in the parameters can have a significant impact on the model. In addition, the resulting model order is reduced if the sample time is chosen too large [27, pp. 568–569].

- Numerical problems

  The result of a sample time chosen too small is a badly conditioned system of equations, as the difference equations become nearly linearly depended for sampled data values [27, p. 569].

Measurement Time

The general upper limit of the measurement time is the moment in which drift phenomena occur due to the limited state of stationariness of real systems [30, p. 300]. In addition, more considerations are given in [8]. The quality of the identification results typically increases with the duration of the measurement. This is because the influence of noise averages out more strongly as the amount of recorded data increases and any systematic estimation error decreases with the amount of measurement data used. The absolute lower limit of collected data should not fall below ten measured data points per estimated parameter [8, p. 508].

4.2 Model Structure Determination

The start of every estimation problem is selecting an eligible model structure, i.e. a parameterised set of candidate models, in which a suitable model is searched [45, p. 1]. In the case of linear models

$$G(s) = \frac{b_0 + b_1 s + \ldots + b_n s^n}{a_0 + a_1 s + \ldots + a_n s^n} e^{-sT_t}, \tag{4.1}$$

only the order of the model n and, if applicable, the existing dead time T_t have to be determined to fully define its mathematical structure [8, pp. 60, 141, 554, 27, pp. 570, 571].

4.2.1 Incorporation of Prior Knowledge

Prior knowledge and physical insight about the system should be utilised when selecting the model structure, as not to estimate already known information [45, p. 1]. The model structure denomination is usually colour-coded to indicate the usage of prior knowledge and shown in Table 4.3.

Table 4.3: Model structures and prior knowledge [45, p. 1]

Term	Description
White-box models	The creation of a model can be entirely done from physical insight and prior knowledge.
Grey-box models	A limited amount of physical insight is available, but some parameters have yet to be identified from observed data.
Black-box models	Physical insight is neither available or cannot be utilised, but the chosen model structure is characterised by good flexibility and has been successfully used in the past.

4.2.2 A Priori Model Order Estimation

Finding or assuming the system order is usually the first step in the identification of linear system parameters [46, p. 1]. In a priori model order estimation methods, a special data matrix is formed with the measured input and output signals of the system to be identified for different model orders, which offers the advantage of enabling an model order approximation before the actual parameter estimation [8, p. 142].

Determinant Ratio Test (DTR)

The determinant ratio test (DTR) presented in [46] and [47] is based on using a data matrix to determine the statistical dependency of the system input and output. It is assumed that the signal noise is equal to zero. First, a $2\hat{n} \times 1$ vector

$$\boldsymbol{h}(k,\hat{n}) = [u(k-1), y(k-1), \ldots, u(k-\hat{n}), y(k-\hat{n})]^T \tag{4.2}$$

is created, in which $\hat{n}$ denotes the estimated system order, is created from the sampled system input and output signals [47, p. 91]. In the next steps, following [8], a $2\hat{n} \times 2\hat{n}$ data matrix

$$\boldsymbol{H}(\hat{n}) = \frac{1}{N} \sum_{k=\hat{n}+1}^{\hat{n}+N} \boldsymbol{h}(k,\hat{n}) \boldsymbol{h}^T(k,\hat{n}) \tag{4.3}$$

is computed based on $\boldsymbol{h}$ and evaluated for different orders $\hat{n} = 1, 2, \ldots, \hat{n}_{max}$ successively. If the estimated order $\hat{n}$ is greater than the actual order n of the system, then $\hat{n} - n$ columns of $\boldsymbol{H}(\hat{n})$ turn linear combinations of the remaining columns and the matrix becomes singular. However, due to noise, this is not exactly the case and the determinant is therefore not exactly zero [8, p. 142]. If the determinant ratio

$$DR(\hat{n}) = \frac{\det \boldsymbol{H}(\hat{n})}{\det \boldsymbol{H}(\hat{n}+1)} \tag{4.4}$$

is computed for the model orders $\hat{n}$ and $\hat{n}+1$, then the $\hat{n}$ for which $DR(\hat{n})$ shows a significant increase compared to $DR(\hat{n}-1)$ corresponds most precisely to the actual system order n [47, p. 92]. This behaviour can be observed easily in a logarithmic plot of the determinant ratios [46, p. 728]. However, the results of the DTR should be viewed as a rough preliminary assessment of the order and a model structure verification should be executed regardless [8, p. 142].

4.2.3 Assessment of Dead Time in the System

According to the description of dead time systems in Lunze [48], the value of the output $\boldsymbol{y}$ at the time t depends on the value of the input$\boldsymbol{u}$ that was a fixed time difference back, which is called dead-time T_t. The rest of the system relationship applies, i.e. a pure dead time element shifts the input variable without deforming it. Dead times typically occur when signals are transmitted with a finite speed of propagation, as in the case of conveyor belts or pipelines [48, p. 118]. Dead time can be observed directly in the response of a system to a given input, as shown in Figure 4.1.

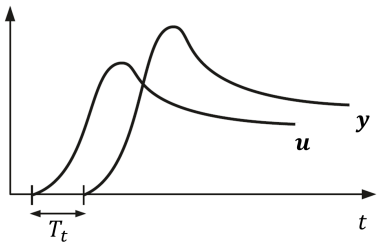


Fig. 4.1: Response of a dead time system with proportional gain [48, p. 209]

4.2.4 Theoretical Derivation of Thermal Models

Usually, the physical properties of the system are responsible for the system order, namely its total number of independent energy reservoirs [8, p. 141]. It is therefore often possible to at least approximately determine the model order by means of a theoretical system identification [47, p. 90]. Based on the remarks in [25], thermal systems are based on the laws of thermodynamics in which the storage and flow of heat are involved. They are further generally distributed parameter systems and therefore described by partial rather than ordinary differential equations. However, by focusing on lumped mathematical models resulting from making approximations where necessary, linear ordinary differential equations capable of describing the dynamic behaviour sufficiently can be obtained. The variables used for describing the behaviour of thermal systems are the temperature T and the heat flow rate q. In addition, for modelling and analysis purposes, the assumption that all points in a body have the same temperature is sensible [25, p. 369]. The energy balance

equation for an arbitrary open thermal system with a net rate of heat energy $\dot{\xi}$ within its boundaries is

$$\dot{\xi} = \sum \dot{h}_{in} - \sum \dot{h}_{out} + \sum q_{in} - \sum q_{out}, \tag{4.5}$$

in which $\dot{h}$ is the rate of enthalpy, caused by mass transfer across the system boundaries [26, p. 99]. Following the definitions of [26], in a closed system without mass transfer the rate of enthalpy change over time is zero. Generally, heat transfer q can occur by conduction, convection and radiation. The rate of heat transfer

$$q = \frac{1}{R}\Delta T \tag{4.6}$$

for conduction and convection is approximately a linear function of the temperature difference ΔT, where R is the thermal resistance. If convection and radiation are neglected, the thermal resistance

$$R = \frac{x}{kA} \tag{4.7}$$

for conduction depends only on the thickness of the material x, its thermal conductivity coefficient k and the area normal to the heat flow A [26, p. 100]. The relation between the temperature change in a physical body and the heat stored within it can be described by

$$\dot{\xi} = C\dot{T}_S = Mc\dot{T}_S, \tag{4.8}$$

in which C is the thermal capacitance for a body with mass M and specific heat c [25, p. 370].

4.3 Model Parameter Estimation

After the model structure or set describing a specific system is chosen, the next step is to find a particular estimate for the model parameters, which best fit the given data [41, p. 498]. The parameters represent specific details or subsystems in the observed system and without accurate knowledge, the numerical values usually have to be determined by statistical methods [32, p. 227].

4.3.1 Error-Based Approach

For an exemplary linear model represented by the transfer function

$$G(s,\boldsymbol{\theta}) = \frac{b_0 + b_1 s + b_2 s^2}{a_0 + a_1 s + a_2 s^2}, \tag{4.9}$$

a parameter vector $\boldsymbol{\theta}$ containing all parameters of the exemplary linear model can be given by

$$\boldsymbol{\theta} = \begin{bmatrix} a_0 & a_1 & a_2 & b_0 & b_1 & b_2 \end{bmatrix}^T \tag{4.10}$$

[41, p. 496]. Reference [41] then states that assuming that there exists a vector of parameters $\boldsymbol{\theta_0}$ for which the error $\boldsymbol{e}$ between model and system is zero, then $\boldsymbol{\theta_0}$ denotes the true value of the system parameters. If a function of $\boldsymbol{e}$ is defined as the cost function

$$J(\boldsymbol{\theta}) = \sum_{k=1}^{N} \boldsymbol{e}^T(k,\boldsymbol{\theta})\boldsymbol{e}(k,\boldsymbol{\theta}) \tag{4.11}$$

of $\boldsymbol{\theta}$, then the parameter vector $\hat{\boldsymbol{\theta}}$ found in the minimum of $J(\boldsymbol{\theta})$ is the best estimate of $\boldsymbol{\theta}_0$. However, to determine a suitable approximation of parameters for a specific model, an idea of goodness of fit is required. Therefore, as a measure for the error between model and process, there are three methods available that have been studied extensively [41, pp. 498–499].

Equation Error

In the equation error approach as described in [41], the complete equations of motion are required. The equation error is then defined as the model deviation from the actual data, as shown in Figure 4.2. A cost function based on the equation error $\boldsymbol{e}_E$ is defined, which is then minimised to find the best fit of parameters. For this method it is necessary to be able to measure all state variables as well as state derivatives, which is often not realistic due to insufficient sensor systems [41, p. 499].

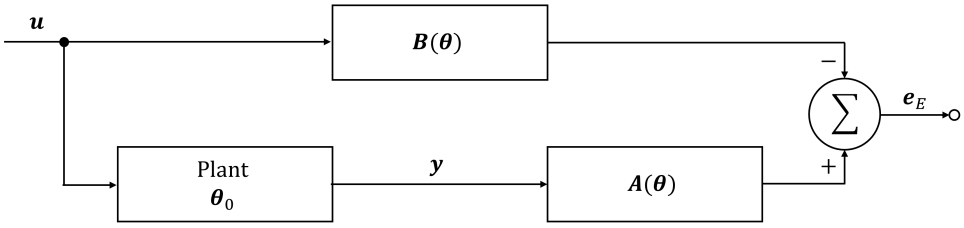


Fig. 4.2: Equation error block diagram [38, p. 421]

Output Error

In order to avoid knowing all the states and state derivatives, an output error based criterion can be used [41, p. 500]. The principle of the procedure is shown in Figure 4.3 and is described as follows in [41]. The estimated parameters are used in model to calculate the model output $\boldsymbol{y}_m$ as a function of θ, while the measured output is a function of the true parameter vector $\boldsymbol{\theta_0}$. The error $\boldsymbol{e}_O$ is defined as the difference between the two output values and subject to minimisation

to achieve the best fitting parameters $\hat{\boldsymbol{\theta}}$ [41, p. 501]. As the output usually is a sensor-measured quantity, the output error model allows the determination of an unbiased estimate of the system parameters [49, p. 1]. Output errors also can be quantified with their physical quantity alongside their corresponding quantified uncertainty [49, p. 1].

Prediction Error

The key idea in prediction error methods is to compute a model such that its prediction error sequence is as small as possible [50, p. 2]. An output predictor is used instead of a simple model in comparison to the output error approach, which will then predict the best possible output prediction based on previous observations within the confines of known structures [41, pp. 501–502]. The calculation of the prediction error $\boldsymbol{e_P}$ is shown in Figure 4.4.

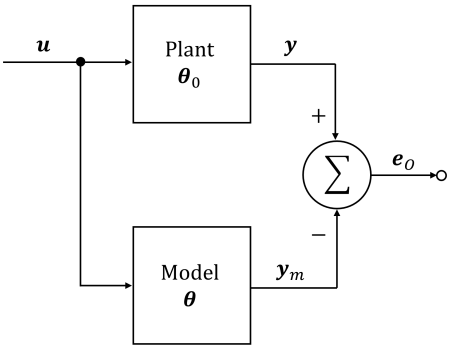


Fig. 4.3: Output error block diagram [41, p. 501]

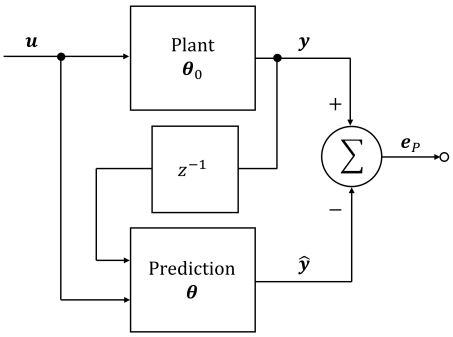


Fig. 4.4: Prediction error block diagram [41, p. 502]

4.3.2 Estimation Theory for Model Parameters

After defining an error dependent on $\boldsymbol{\theta}$, a performance criterion is specified and used to find the best estimate $\hat{\boldsymbol{\theta}}$ of the true values $\boldsymbol{\theta}_0$ [41, p. 502]. The performance criterion is typically chosen based on computational difficulties, how hard it is to find $\hat{\boldsymbol{\theta}}$ and how good of a fit the result is [41, p. 502]. There are many different methods available and for the comparison of performance, a reasonable criterion is necessary [51, p. 135].

Overview of Methods

Among the criteria most widely used are the least-squares estimate, the best linear unbiased estimate and the maximum likelihood estimate [41, p. 502]. In sampled-data systems with uniformly

spaced sampling intervals, methods traditionally used also include instrumental variable techniques and subspace identification [17, p. 1]. Of these methods, the least squares method will be explained in more detail because it represents the basis for the method proposed in Chapter 4.3.3.

Least Squares Estimation

This section follows the remarks of Åstrom [52]. For a model given by

$$y_i = \theta_1\phi_1(x_i) + \theta_2\phi_2(x_i) + \ldots + \theta_n\phi_n(x_i) + e_i, \tag{4.12}$$

in which x_i are known values, ϕ_i are known functions and θ_i are the unknown model parameters, the parameter vector $\boldsymbol{\theta}$ containing $\theta_1, \theta_2, \ldots, \theta_i$ is to be determined in a way that the error criterion

$$J(\boldsymbol{\theta}) = \sum_{i=1}^{N} e_i^2 \tag{4.13}$$

is minimised. For the purpose of minimising $J(\boldsymbol{\theta})$, Equation 4.12 is written by means of a vector formalism:

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \end{bmatrix}, \quad \boldsymbol{\Phi} = \begin{bmatrix} \phi_1(x_1) & \ldots & \phi_n(x_1) \\ \phi_1(x_2) & \ldots & \phi_n(x_2) \\ \vdots & \ddots & \vdots \\ \phi_1(x_N) & \ldots & \phi_n(x_N) \end{bmatrix}, \quad \boldsymbol{e} = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_i \end{bmatrix}, \quad \boldsymbol{\theta} = \begin{bmatrix} \theta_1 \\ \theta_1 \\ \vdots \\ \theta_1 \end{bmatrix} \tag{4.14}$$

By utilising this formalism, the model description and cost function can be written as

$$\boldsymbol{y} = \boldsymbol{\Phi\theta} + \boldsymbol{e} \tag{4.15}$$

and

$$J(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i=1}^{N} e_i^2 = \boldsymbol{e}^T\boldsymbol{e} \tag{4.16}$$

respectively [52, p. 3]. To minimise the cost function, $J(\boldsymbol{\theta})$ is first expanded to

$$\begin{aligned} J(\boldsymbol{\theta}) &= \boldsymbol{e}^T\boldsymbol{e} \\ &= (\boldsymbol{y} - \boldsymbol{\Phi\theta})^T(\boldsymbol{y} - \boldsymbol{\Phi\theta}) \\ &= \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{\Phi\theta} - \boldsymbol{\theta}^T\boldsymbol{\Phi}^T\boldsymbol{y} + \boldsymbol{\theta}^T\boldsymbol{\Phi}^T\boldsymbol{\Phi}^T\boldsymbol{\theta} \end{aligned} \tag{4.17}$$

and second, if $\boldsymbol{\Phi}^T\boldsymbol{\Phi}$ is regular, then it holds true that

$$\begin{aligned} \boldsymbol{e}^T\boldsymbol{e} &= \left(\boldsymbol{\theta} - \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{y}\right)^T \boldsymbol{\Phi}^T\boldsymbol{\Phi}\left(-\left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{y}\right) \\ &\quad + \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{\Phi}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{y} \\ &\geq \boldsymbol{y}^T\boldsymbol{y} - \boldsymbol{y}^T\boldsymbol{\Phi}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T y, \end{aligned} \tag{4.18}$$

in which case the equality occurs for

$$\boldsymbol{\theta} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{y}, \tag{4.19}$$

meaning that the vector of estimates $\hat{\boldsymbol{\theta}}$ for the least squares method is [52, pp. 3–4, 27, p. 381]

$$\hat{\boldsymbol{\theta}} = \left(\boldsymbol{\Phi}^T\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{y}. \tag{4.20}$$

Weighted Least Squares

A more general method of least squares can be obtained if there is a different weighting for the individual values, in which case the vector of estimates is

$$\hat{\boldsymbol{\theta}} = \left(\boldsymbol{\Phi}^T\boldsymbol{W}\boldsymbol{\Phi}\right)^{-1}\boldsymbol{\Phi}^T\boldsymbol{W}\boldsymbol{y}, \tag{4.21}$$

where

$$\boldsymbol{W} = \begin{bmatrix} w_{11} & 0 & \dots & 0 \\ 0 & w_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & w_{NN} \end{bmatrix} \tag{4.22}$$

represents a diagonal weighting matrix [27, p. 279].

4.3.3 Global Least Squares System Identification for State-Space Models

For the purpose of model parameter estimation in this thesis, a variable projection method for system identification first proposed in [53] and later adapted in [49] for the identification of models in state-space-representation called global least squares system identification is utilised. It provides a best-fit of model parameters based on output error and noisy measurement data, without solving the system of differential equations [49, p. 1]. The approach is characterized by the following points:

- The method utilises models in state-space representation, which have the advantage of being nearer to the physical process compared to transfer functions, as the parameter vector corresponds to the physical properties that characterise the system [38, p. 431].
- The estimated parameters will be unbiased if the real system is linear and the measurement errors follow Gaussian distributions, meaning that the expected value will be identical to the parameter being estimated [51, p. 136, 49, p. 1].
- If the measurement noise in the input-output data is assumed to be independent and identically distributed, it can be handled conveniently by the proposed method [49, p. 4].
- It is possible to utilise non-uniformly sampled input-output data as well as uniformly sampled data. In both cases, the result is the direct identification of the best fitting continuous time model parameters from discrete time data.

Procedure

The method of global least squares system identification for states-space models proposed in [49] is summed up in this section. A linear system of differential equations is given in state-space form as

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) \tag{4.23}$$

and measurements $\boldsymbol{\xi}$ providing information about the system behaviour are provided by

$$\boldsymbol{\xi}(t) = \boldsymbol{H}\boldsymbol{x}(t) + \boldsymbol{G}\boldsymbol{u}(t). \tag{4.24}$$

Equation 4.23 can be discretised by transposition of the state matrix $\boldsymbol{A}$ and the input matrix $\boldsymbol{B}$ as well as making use of an appropriate differentiation matrix $\boldsymbol{D}$ described in Appendix A.2, resulting in

$$\dot{\boldsymbol{X}} = \boldsymbol{D}\boldsymbol{X} = \boldsymbol{X}\boldsymbol{A}^T + \boldsymbol{U}\boldsymbol{B}^T, \tag{4.25}$$

in which the matrices $\dot{\boldsymbol{X}}, \boldsymbol{X}$ and $\dot{\boldsymbol{U}}$ are defined as

$$\dot{\boldsymbol{X}} = \begin{bmatrix} \dot{x}_1(t_0) & \dots & \dot{x}_p(t_0) \\ \dot{x}_1(t_1) & \dots & \dot{x}_p(t_1) \\ \vdots & \ddots & \vdots \\ \dot{x}_1(t_f) & \dots & \dot{x}_p(t_f) \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} x_1(t_0) & \dots & x_p(t_0) \\ x_1(t_1) & \dots & x_p(t_1) \\ \vdots & \ddots & \vdots \\ x_1(t_f) & \dots & x_p(t_f) \end{bmatrix}, \quad \boldsymbol{U} = \begin{bmatrix} u_1(t_0) & \dots & u_p(t_0) \\ u_1(t_1) & \dots & u_p(t_1) \\ \vdots & \ddots & \vdots \\ u_1(t_f) & \dots & u_p(t_f) \end{bmatrix}. \tag{4.26}$$

The system of measurements $\boldsymbol{\xi}$ is discretised correspondingly. Equations 4.23 and 4.24 can be written in vector form as

$$(\boldsymbol{I} \otimes \boldsymbol{D})\boldsymbol{x} = (\boldsymbol{A} \otimes \boldsymbol{I_n})\boldsymbol{x} + (\boldsymbol{B} \otimes \boldsymbol{I_n})\boldsymbol{u} \tag{4.27}$$

and

$$\boldsymbol{\xi} = (\boldsymbol{H} \otimes \boldsymbol{I_n})\boldsymbol{x} + (\boldsymbol{G} \otimes \boldsymbol{I_n})\boldsymbol{u} \tag{4.28}$$

respectively, in which $\otimes$ denotes the Kronecker product (Appendix A.1). The definitions of $\boldsymbol{x}$ and $\boldsymbol{u}$ for Equations 4.27 and 4.28 are given by

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_p \end{bmatrix} \quad \text{with} \quad \boldsymbol{x}_k = \begin{bmatrix} \boldsymbol{x}_k(t_0) \\ \boldsymbol{x}_k(t_1) \\ \vdots \\ \boldsymbol{x}_k(t_f) \end{bmatrix} \text{and} \quad \boldsymbol{u} = \begin{bmatrix} \boldsymbol{u}_1 \\ \boldsymbol{u}_2 \\ \vdots \\ \boldsymbol{u}_p \end{bmatrix} \quad \text{with} \quad \boldsymbol{u}_k = \begin{bmatrix} \boldsymbol{u}_k(t_0) \\ \boldsymbol{u}_k(t_1) \\ \vdots \\ \boldsymbol{u}_k(t_f) \end{bmatrix}. \tag{4.29}$$

The matrix I is a $p \times p$ identity matrix, where p represents the order of the system, while I_n is a $n \times n$ identity matrix for n data points provided by the input-output data:

$$\boldsymbol{I} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix}, \quad \boldsymbol{I_n} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 0 & 1 \end{bmatrix} \tag{4.30}$$

The matrices $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{G}$ and $\boldsymbol{H}$ all depend on the system parameters given as a vector $\boldsymbol{\Psi}$, i.e.:

$$\boldsymbol{A} = \boldsymbol{A}(\psi), \;\; \boldsymbol{B} = \boldsymbol{B}(\psi), \;\; \boldsymbol{H} = \boldsymbol{H}(\psi), \;\; \boldsymbol{G} = \boldsymbol{G}(\psi) \tag{4.31}$$

Considering a corruption of the measurements by covariant Gaussian nose, denoted by the covariance matrix $\boldsymbol{\Lambda}$ and a vector $\boldsymbol{\delta}$ containing independent and identically distributed Gaussian random variables with a unit standard deviation leads to

$$\boldsymbol{\xi} = (\boldsymbol{H} \otimes \boldsymbol{I}_n)\,\boldsymbol{x} + (\boldsymbol{G} \otimes \boldsymbol{I}_n)\,\boldsymbol{u} + \boldsymbol{\Lambda}^{\frac{1}{2}}\boldsymbol{\delta} \tag{4.32}$$

as the vector of measured states. Any estimate for the state of the system should minimise the cost function

$$\varepsilon\,(\boldsymbol{x}, \boldsymbol{\Psi}) = \|\boldsymbol{\Lambda}^{-\frac{1}{2}}\,((\boldsymbol{H} \otimes \boldsymbol{I}_n)\,\boldsymbol{x} + (\boldsymbol{G} \otimes \boldsymbol{I}_n)\,\boldsymbol{u} - \boldsymbol{\xi})\,\|_2^2, \tag{4.33}$$

which is a nonlinear weighted least squares problem. The system in question is constrained by the system of discretised differential equations

$$((\boldsymbol{I} \otimes \boldsymbol{D}) - (\boldsymbol{A} \otimes \boldsymbol{I_n}))\,\boldsymbol{x} = (\boldsymbol{B} \otimes \boldsymbol{I_n})\,\boldsymbol{u}. \tag{4.34}$$

By defining a discretised linear differential operator

$$\boldsymbol{L}(\psi) = (\boldsymbol{I} \otimes \boldsymbol{D}) - (\boldsymbol{A}(\psi) \otimes \boldsymbol{I_n}) \tag{4.35}$$

dependent on the system variables, Equation 4.34 can be written as

$$\boldsymbol{Lx} = (\boldsymbol{B} \otimes \boldsymbol{I_n})\boldsymbol{u}. \tag{4.36}$$

The singular value decomposition (A.3) of $\boldsymbol{L}$ is

$$\boldsymbol{L} = \sum_{j=1}^{pn} \sigma_j \boldsymbol{u}_j \boldsymbol{v}_j^T \tag{4.37}$$

and therefore, the particular generalised inverse of $\boldsymbol{L}$, is given by

$$\boldsymbol{L}^- = \sum_{j=1}^{pn-p} \frac{1}{\sigma_j} \boldsymbol{v}_j \boldsymbol{u}_j^T. \tag{4.38}$$

Thus, the the parametric solution for $\boldsymbol{x}$ is

$$\boldsymbol{x} = \boldsymbol{L}^-(\boldsymbol{B} \otimes \boldsymbol{I_n})\boldsymbol{u} + \tilde{\boldsymbol{V}}\boldsymbol{\beta}, \tag{4.39}$$

where

$$\tilde{\boldsymbol{V}} = \left[\boldsymbol{v}_{pn-p+1} \ldots \boldsymbol{v}_{pn-1} \boldsymbol{v}_{pn}\right] \tag{4.40}$$

is a basis for the approximate null space for $\boldsymbol{L}$. The vector $\boldsymbol{\beta}$ contains a set of p arbitrary parameters and substituting $\boldsymbol{\beta}$ into Equation 4.33 results in

$$\varepsilon(\boldsymbol{\beta}, \psi) = \| \Lambda^{-\frac{1}{2}} \left((\boldsymbol{H} \otimes \boldsymbol{I_n}) \left(\boldsymbol{L}^- (\boldsymbol{B} \otimes \boldsymbol{I_n}) \boldsymbol{u} + \tilde{\boldsymbol{V}} \boldsymbol{\beta} \right) + (\boldsymbol{G} \otimes \boldsymbol{I}_n) \boldsymbol{u} - \boldsymbol{\xi} \right) \|_2^2, \tag{4.41}$$

which means that the cost function is now a function of the system parameters $\boldsymbol{\Psi}$ and the arbitrary parameters $\boldsymbol{\beta}$ [49, p. 3]. The arbitrary parameters should be determined in a way that the cost function is minimised, which the definition

$$\boldsymbol{\beta} = \left(\Lambda^{-\frac{1}{2}} (\boldsymbol{H} \otimes \boldsymbol{I}_n) \tilde{\boldsymbol{V}} \right)^+ \boldsymbol{\Lambda}^{-\frac{1}{2}} \left(\boldsymbol{\xi} - (\boldsymbol{H} \otimes \boldsymbol{I_n}) \boldsymbol{L}^- (\boldsymbol{B} \otimes \boldsymbol{I_n}) \boldsymbol{u} - (\boldsymbol{G} \otimes \boldsymbol{I_n}) \boldsymbol{u} \right) \tag{4.42}$$

fulfils. By substituting Equation 4.42 back into the cost function in Equation 4.41, the resulting cost function

$$\varepsilon(\psi) = \| \boldsymbol{P} \boldsymbol{\Lambda}^{-1/2} \left(\left((\boldsymbol{H} \otimes \boldsymbol{I_n}) \boldsymbol{L}^- (\boldsymbol{B} \otimes \boldsymbol{I_n}) - (\boldsymbol{G} \otimes \boldsymbol{I_n}) \right) \boldsymbol{u} - \boldsymbol{\xi} \right) \|_2^2 \tag{4.43}$$

is only dependent on the system parameters ψ. The projection matrix

$$\boldsymbol{P} = \boldsymbol{I} - \boldsymbol{\Lambda}^{1/2}(\boldsymbol{H} \otimes \boldsymbol{I_n})\tilde{\boldsymbol{V}} \left(\boldsymbol{\Lambda}^{1/2}(\boldsymbol{H} \otimes \boldsymbol{I_n})\tilde{\boldsymbol{V}} \right)^{+} \tag{4.44}$$

is a function of the unknown system parameters, which is why this procedure is known as the variable projection method [49, pp. 2–3]. Summed up, the cost function for the identification procedure can be reduced to a function of the system parameters by taking advantage of the linearity of the system of differential equations [49, p. 3]. Any nonlinear optimisation method is suited to minimise the resulting cost function [49, p. 3].

4.4 Model Verification

4.4.1 Structure

Methods for model structure verification are used to determine the smallest possible model order in a way that the essential system properties can be described with sufficient accuracy by the corresponding mathematical model [8, p. 141].

Error Function Test

The error function test presented in [8] and [47] is based on the deviation between the calculated output $\hat{y}$ and the measured output y

$$\varepsilon(k,n) = y(k) - \hat{y}(k,n), \tag{4.45}$$

for which a quadratic cost function

$$I(n) = \frac{1}{2} \sum_{k=n}^{N} \varepsilon^2(k,n) \tag{4.46}$$

is calculated for different model orders n. The final and best estimated order of the model has been reached, when the value of the cost function does not continue to decrease significantly after raising the model order [8, pp. 144–145, 47, p. 94].

4.4.2 Accuracy

The assess the model accuracy, the correspondence of the calculated output from the identified model with the measured output is observed [8, p. 556]. This can be done graphically and visually as well as by utilising suitable error measures like the sum of the squared residuals and the maximum and the minimum value of the residual [8, p. 556]. A basic statistical assessment for the model can be done by computing the maximum error

$$S_1 = \max|\varepsilon(k)| \tag{4.47}$$

and the average error

$$S_2 = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k)} \tag{4.48}$$

for N data points for the model and comparing it to a chosen threshold for both [31, p. 511].

4.4.3 Validity

After a suitable model structure and its parameters are identified, the last step is validating the model. In the validation process, the model is confirmed to describe the system behaviour by testing it on measurement data not used in the model construction [41, p. 481]. In addition, Ljung [31] states that there is always a certain purpose for the model construction, for example, regulator design, prediction, or simulation. Thus, the most important question of model validation in practice is that if the best model found within the specified model structure is sufficient to serve its purpose. The ultimate model validation therefore is the ability of the constructed model to resolve the original problem statement. If utilising the obtained model solves the intended task, then the model is valid, regardless of any other formal aspects [31, p. 509].

Chapter 5

Control Design

This chapter gives an introduction to control engineering as well as two suitable control strategies for the sensorless control approach, PID control and model predictive control.

5.1 Introduction to Control Systems

A control system consists of subsystems and processes, congregated to obtain a desired output within desired performance boundaries in reaction to a specified input [54, p. 2]. Thus, the simplest form of a control system can be symbolised as in Figure 5.1. Control engineering is defined as the science of automatically influencing dynamic processes in a specified and targeted manner [55, p. 844]. Figure 5.2 shows the typical structure of a control circuit and the upcoming funda-

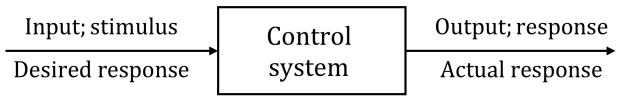


Fig. 5.1: Simple control system [54, p. 2]

mental explanation corresponds to the description found in [55]. The controlled variable $y(t)$ is supposed to exactly follow a specified trajectory and is measured and compared with the set value $w(t)$. The difference $e(t) = w(t) - y(t)$ is called the control deviation. The controller computes the manipulated variable $u(t)$ from the control deviation $e(t)$ in a way that the controlled variable $y(t)$ follows the set value $w(t)$ in the best possible manner. The disturbance variable $z(t)$ represents disturbances on the process. The feedback of process variables is characteristic of a control loop and in the case of the typical control circuit shown in Figure 5.2, the feedback variable is $y(t)$. If the disturbance $z(t)$ causes a deviation of the controlled variable $y(t)$ from the reference $w(t)$, the control deviation is unequal zero. This error can be regulated provided a suitable control element is utilised. Even if the dynamic properties of the process are only known approximately, the control deviation will be sufficiently small if a suitable control element is used [55, p. 844].

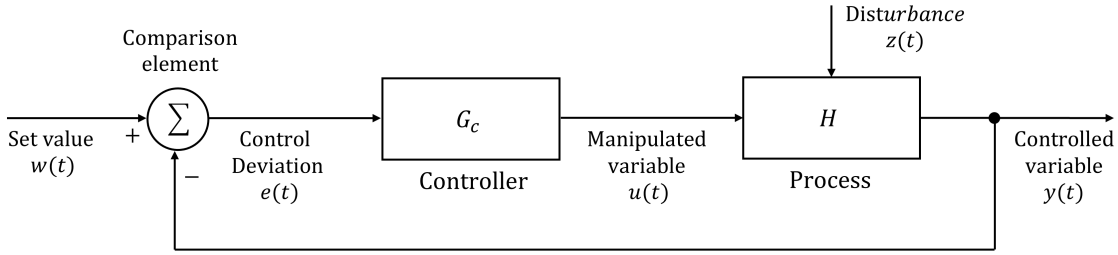


Fig. 5.2: Typical structure of a control circuit [55, p. 844]

5.1.1 Control System Configurations and Components

Control systems can exist in different configurations. These can be considered to be the internal architecture of the control system in Figure 5.1 [54, p. 6]. Two important configurations are:

- Closed-loop

  Closed-loop control systems utilise a constant comparison of the actual and the required value of the manipulated variable to maintain the value at the required or specified level [56, p. 83]. This is called feedback, which means that information from the output of the control system is fed back to the input to the system, so that the input can be adapted to the actual system requirements [56, p. 83]. A closed loop system is shown in Figure 5.2.

- Open-loop

  An open-loop control system is characterized by its inability to counterbalance disturbances adding to the controller output [54, p. 7]. If a manipulated variable is controlled by an initial set value and no further adjustments to that value are made, the system is called an open-loop control system [56, p. 83]. No information is fed back to allow a change in the set point for the manipulated variable, no matter if the operating conditions change and appropriate adjustment allowing for the adaptation to the disturbance would be required [56, p. 83]. An open-loop control system is depicted in Figure 5.3.

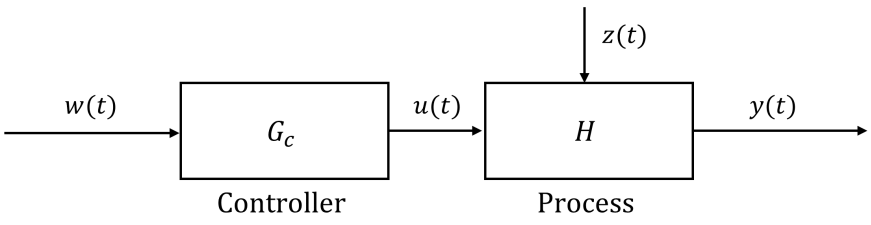


Fig. 5.3: Open-loop control system [55, p. 844]

Control System Components

It can be seen from the block schematic of the closed-loop system in Figure 5.2 that three basic components are required in a closed-loop control circuit:

1. Process

   The process is the system in which a variable is being controlled or influenced [56, p. 86]. It is an arrangement of structures that are related to one another [57, p. 1620]. Part of the process is usually also an actuator, which converts the manipulated variable to the power level required in the output of the control system [55, p. 844, 57, p. 1623].

2. Controller

   The controller in the circuit carries out the control function and consists of a control unit and an actuator [57, p. 1622]. The controller provides an output signal to influence the system output while receiving the error signal as input [56, p. 99]. How the controller reacts to changes in the error signal is specified by the so-called control laws or control modes [56, p. 99].

3. Comparison element

   The controlled variable is recorded by a measuring element, which feeds the data to the comparison element [57, p. 1622]. The comparison element is then used to compare the required value of the controlled variable with the currently measured output value to compute an error signal [56, p. 85]. If the output equals the required value, there is no error and no control action is initiated; If not, the error signals is unequal zero and control action commences [56, p. 86].

In addition, there are two paths shown in Figure 5.2, which can be described according to [56]. The forward path leads from the error signal to the output. In the simplest case, the elements on the forward path are the controller and the process. On the feedback path, a signal related to the actual condition being achieved is fed back to modify the input signal to a process. Per definition, feedback is negative when the signal fed back is subtracted from the input, thus negative feedback is required to control a system [56, p. 87].

5.1.2 Characteristics and Design of Control Systems

The process by which the performance of a control system is first created or later adjusted is called design [54, p. 9]. For example, if characteristics of the control system like its transient response and steady-state error are found not to satisfy the required specifications, parameters can be adapted or additional components added to improve said characteristics in the control design process [54, p. 9]. For a controlled system showcased in Figure 5.2, the closed-loop transfer function for the set

value in relation to the output value is defined as

$$S(s) = \frac{Y(s)}{W(s)} = \frac{G_c(s)H(s)}{1 + G_c(s)H(s)}, \tag{5.1}$$

if there are no disturbances [48, p. 360]. The response of this system $S(s)$ to a set input highlighting important control characteristics is then shown in Figure 5.4. According to [30], the rise time T_r signifies the moment in which 90 % of the set value has been reached. Further, the maximum overshoot e_{max} occurring at the overshoot time T_{max} is defined as the maximum control deviation after the set value has been reached for the first time. $e(t = \infty)$ is the lasting steady-state error. Finally, the settling time T_ε is the point in time from which on the control deviation is smaller than a specified limit ε [30, p. 188].

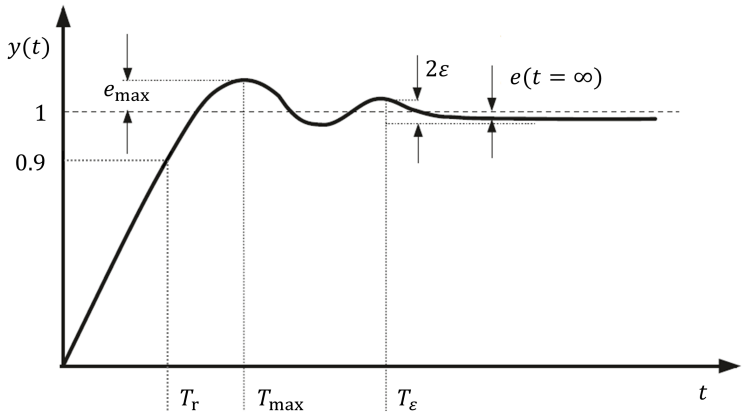


Fig. 5.4: Control system response characteristics [48, p. 355]

The purpose of control systems is to ensure that the closed-loop circuit has desirable dynamic and steady-state response characteristics [34, p. 200]. Ideally, the following criteria should be satisfied:

1. Stability

   The closed-loop control system must be stable [34, p. 200]. That means, it has to react to finite excitations from the set value or disturbances with a finite output signal [48, p. 354]. If the control circuit is not excited, it has to return to its idle state as free movement y_{free} and transitional agitation y_{trans} are required to subside [48, p. 354]:

   $$\lim_{t\to\infty} y_{free} = 0, \quad \lim_{t\to\infty} y_{trans} = 0 \tag{5.2}$$

2. Steady-State Error

   Any steady-state error for specified types of set values and disturbance signals should be eliminated, meaning that the controlled variable should asymptotically follow the set value [48,

p. 354, 34, p. 200]:

$$\lim_{t\to\infty}(w(t)-y(t))=0 \tag{5.3}$$

If the control circuit is stable and Equation 5.3 is fulfilled, the resulting control deviation vanishes when observed in terms of long periods of time [48, p. 355]:

$$e(t=\infty)=0 \tag{5.4}$$

3. Set Value Tracking

   Good set value tracking includes rapid and smooth responses to input changes while excessive control action is avoided [34, p. 200]. Typical characteristics for the dynamic behaviour are the steepness, amount and time of the first overshoot or the settling time, which are shown in Figure 5.4 [48, pp. 355–356].

4. Robustness

   The control system should be insensitive to changes in process conditions and inaccuracies in the process model, a property which is known as robustness [34, p. 200, 48, p. 358]. Often, the controller is designed under consideration of stability, transient response and steady-state characteristics, while a robustness assessment is performed afterwards [48, p. 358].

5.1.3 Digital Control Systems

In many modern control systems, the controller is a digital computer [54, p. 8]. The advantage of using a computer is that many circuits can be controlled by the same computer and any adjustments of the control parameters can be conducted by simple changes in software instead of hardware [54, p. 8]. Compared to analogue control, information storage is easier, accuracy can be greater and digital control circuits are less susceptible to noise [56, p. 95]. Following the remarks of [54] for

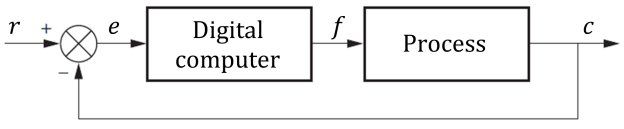


Fig. 5.5: Digital control [54, p. 709]

the purpose of explaining digital control systems, loops containing computers provide and process digital signals consisting of a sequence of binary numbers. The signals r, e, f, and c shown in Figure 5.5 can be either digital or analogue and when both signal types occur, a conversion from one form to the other is necessary. If plant output c and system input r are analogue signals, then an

analogue-to-digital converter is required before the digital computer. In case the plant requires an analogue input f, a digital-to-analogue converter is compulsory [54, pp. 709–710]. The program executed by a digital controller follows the listed steps below [56, p. 95]:

1. Import the set value and the current measured controlled variable value at the A/D port
2. Compute the error
3. Quantify the necessary controller output
4. Dispatch the controller output signal to the D/A port
5. Await the subsequent sampling point
6. Repeat the listed steps

5.2 PID Control

5.2.1 Introduction to PID Controllers

The PID controller is the most frequently implemented controller because of its simple configuration and easy implementation [58, p. 1]. It provides three principal control actions: [41, p. 66]

- Proportional action (P) provides a change in the manipulated variable relative to the error signal and is utilised to counter a large amount of error:

$$u(t) = K_p\, e(t) \tag{5.5}$$

- Integral action (I) creates a signal proportional to the time integral of error and is used to reduce the steady-state or offset error:

$$u(t) = \frac{K_p}{T_i} \int_0^t e(\tau)\mathrm{d}\tau \tag{5.6}$$

- Derivative action (D) contributes a signal proportional to the derivative of the error with the purpose of reducing maximum overshoot:

$$u(t) = K_p\, T_d\, \dot{e}(t) \tag{5.7}$$

All of these control actions are summed up to obtain a single control effort

$$G_c(s) = \frac{U(s)}{E(s)} = K_p\,(1 + \frac{1}{T_i\, s} + T_d\, s) = K_p + \frac{K_i}{s} + T_d s, \tag{5.8}$$

in which K_p is the proportional gain, $T_i = \frac{K_p}{K_i}$ the integral time and $T_d = \frac{K_d}{K_p}$ the derivative time [30, p. 124, 41, p. 66].

5.2.2 PID Design and Tuning

Ziegler Nichols Tuning Method based on Critical Gain

The most popular tuning procedure is known as the Ziegler-Nichols rules [59] for tuning PID controllers [58, p. 413, 30, p. 206]. These rules were empirically derived from extensive studies of controller settings [30, p. 206]. Although there are more sophisticated methods available, the results of Ziegler-Nicholas tuning still prove competitive [58, p. 414].

The process of tuning is documented in [30]. First, the controller present in the control loop is initially switched to pure proportional control. The proportional gain K_p is then increased incrementally from zero to a critical gain K_{cr}, at which the output first oscillates in a sustained manner. Next, the corresponding critical period P_{cr} of the continuous oscillation is measured [30, p. 207]. Finally, the values of the parameters K_p, T_i and T_d are set according to the Ziegler-Nichols rules shown in Table 5.1. This method can not be applied if the output does not exhibit sustained oscillations for any K_p [60, p. 570].

Table 5.1: Ziegler-Nichols rules for controller tuning based on critical gain [60, p. 571]

Controller	K_p	T_i	T_d
P	$0.5\,K_{cr}$	∞	0
PI	$0.45\,K_{cr}$	$\frac{1}{1.2}\,P_{cr}$	0
PID	$0.6\,K_{cr}$	$0.5\,P_{cr}$	$0.125\,P_{cr}$

Direct Synthesis Method

In the direct synthesis method, the PID controller in a closed-loop control system is designed by utilising an available process model and a desired closed-loop transfer function [34, p. 201]. The calculation of the unknown PID controller G_c is done with

$$G_c = \frac{1}{\tilde{G}} \left(\frac{(Y/Y_{sp})_d}{1-(Y/Y_{sp})_d} \right), \tag{5.9}$$

in which $(Y/Y_{sp})_d$ resembles the desired closed-loop transfer function and $\tilde{G}$ the known process model [34, p. 201]. The tuning parameters are then determined by comparing G_c with the standard form of the PID controller given by Equation 5.8.

5.2.3 Cascade Control Systems

Cascade control is based on the utilisation of two controllers in one control system, in which the first controller output signal is the set value of the second one [56, p. 289]. To achieve this configuration, the feedback loop of one controller is placed inside the feedback loop of the other one, as showcased in Figure 5.6. Multiple measurement variables can be used in contrast to a single loop control circuit, but still, only one control variable is utilised [38, p. 266, 61, p. 392]. Utilising

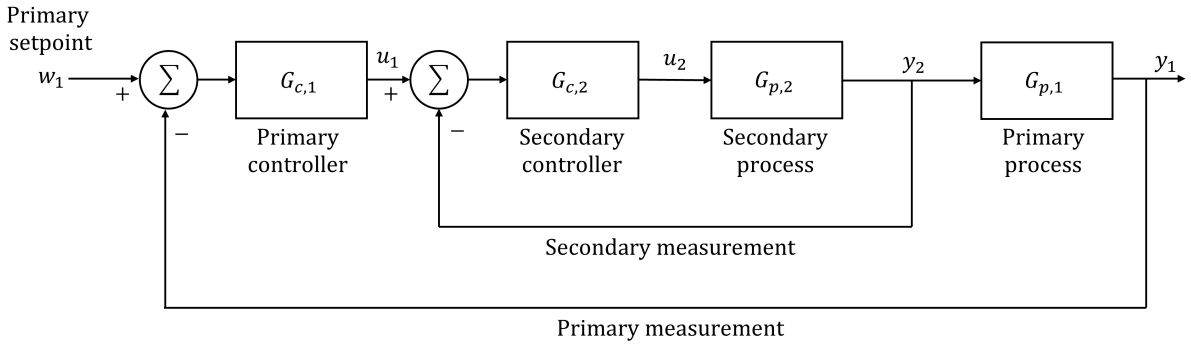


Fig. 5.6: Cascade control system [61, p. 394]

the definitions of [61], the first controller $G_{c,1}$ is called primary or outer-loop controller, while the second one is the secondary or inner-loop controller. If the inner loop dynamics are very fast, the secondary controller is easy to tune. If the outer control loop reacts much slower in proportion, the primary loop can be effectively tuned as if the response of the inner controller is instantaneous [61, p. 394]. Due to this, cascade control is generally recommended for control of rather slow processes, which are in turn controlled by a comparatively fast process [62, p. 606]. However, cascade control results in increased complexity and the requirement of an additional measurement device and controller [62, p. 607]. As a rule of thumb, the improved performance achieved by using cascade control justifies the increased investment if the inner loop is at least three times faster than the outer loop [62, p. 607].

Cascade control systems provide better control performance in comparison to single loop control due to following reasons: [63, pp. 49, 50]

1. Disturbances acting on the input section of the process are already subject to control by the inner control loop before they can influence the main controlled variable.

2. Due to the secondary feedback system, the parameter sensitivity in the inner process part is reduced. As a result, in the initial design of the main controller only parameter changes in the outer process need to be considered.

3. The behaviour of the main control variable becomes less inert and therefore swifter, if the inner control loop enables faster modes than those of the inner process.

Cascade Tuning

Tuning a cascade control system is done in two steps. First, the inner-loop controller $G_{c,2}$ is tuned based on the secondary process $G_{p,2}$ [61, p. 464]. Then, the outer-loop controller $G_{c,1}$ is tuned based on the resulting inner-loop closed-loop transfer function [61, p. 464]. Typical tuning methods for PID cascade control systems include the Ziegler-Nichols method, relay auto-tuning, internal model control and pole placement, while more sophisticated procedures make use of fuzzy logic, artificial intelligence or genetic algorithms to identify the optimal PID settings [58, p. 1].

5.3 Model Predictive Control (MPC)

5.3.1 Introduction to Model Predictive Control

References [64, 38, 34] define model predictive control (MPC) as an umbrella term for algorithms, which compute a sequence of adjustments of a manipulated variable over a future time horizon by using a reasonably accurate dynamic linear or nonlinear model of the process to be controlled. The future behaviour of the process is optimised by utilising a quadratic criterion subject to linear or nonlinear constraints [64, p. 1, 38, p. 632, 34, p. 368]. In model predictive control, an output variable is usually referred to as controlled variable (CV), an input variable as manipulated variable (MV) and a measured disturbance as measured disturbance variable (DV) [34, p. 368].

The basic principle of MPC is shown in Figure 5.7 and explained in [61]. An optimisation problem is solved at each time step k, in which a cost function based on the output predictions over a prediction horizon of P time steps is minimised by a series of manipulated variables moves over a control horizon of M moves. Only the first move is implemented. After the implementation of u_k, the outputs are measured at next time step. Since the measured output y_{k+1} will in general be different from the model prediction, a correction of the model error is executed. Then again, a new optimisation problem is solved over a prediction horizon of P steps by adjusting M control moves [61, p. 681].

The underlying algorithmic principle can also be explained by the block diagram in Figure 5.8 and by utilising the remarks provided by [38]. At each time step k, the residual $r(k)$ of the process output $y(k)$ and the model output prediction $y^*(k)$ is computed based on past inputs. The trajectories $\boldsymbol{y}^*(k)$ and $\boldsymbol{r}(k)$ are the predicted values of output and residual over the prediction horizon and are calculated for $y^*(k)$ and $r(k)$. Finally, the corrected trajectory of the future output $\hat{\boldsymbol{y}}(\boldsymbol{k})$

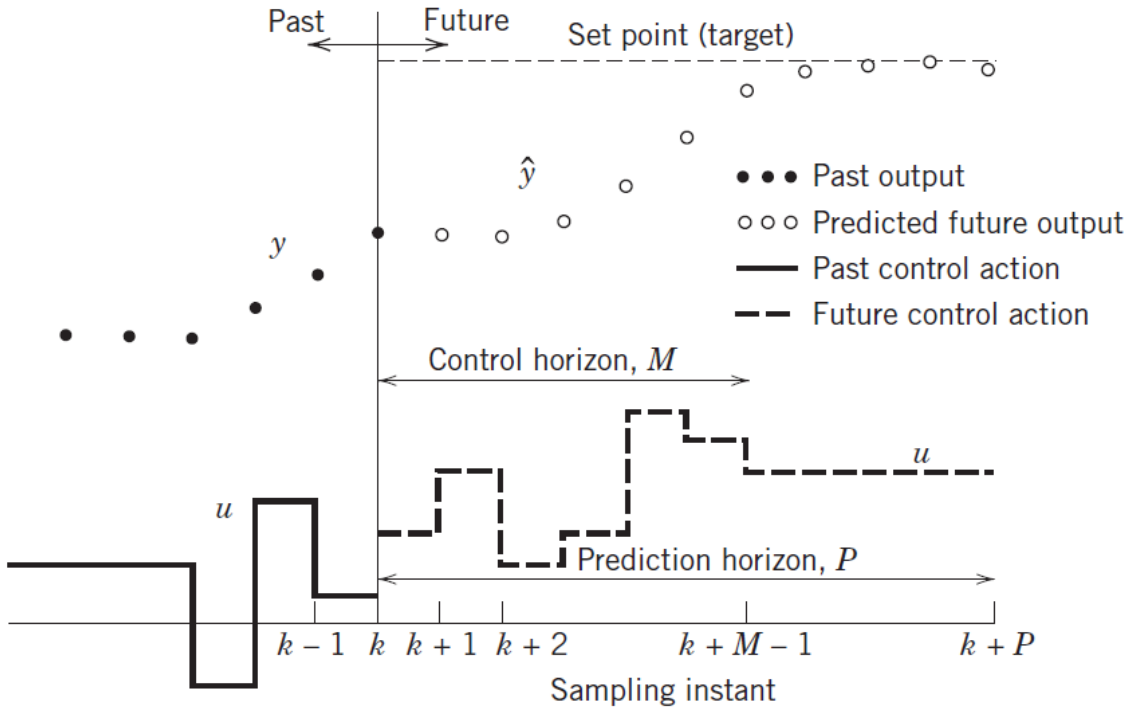


Fig. 5.7: Principle of model predictive control [34, p. 370]

is calculated and compared with the reference $\boldsymbol{y}^{ref}(k)$ for the calculation of the control deviation $\boldsymbol{e}(k)$, which is then subjected to the chosen control law [38, p. 638].

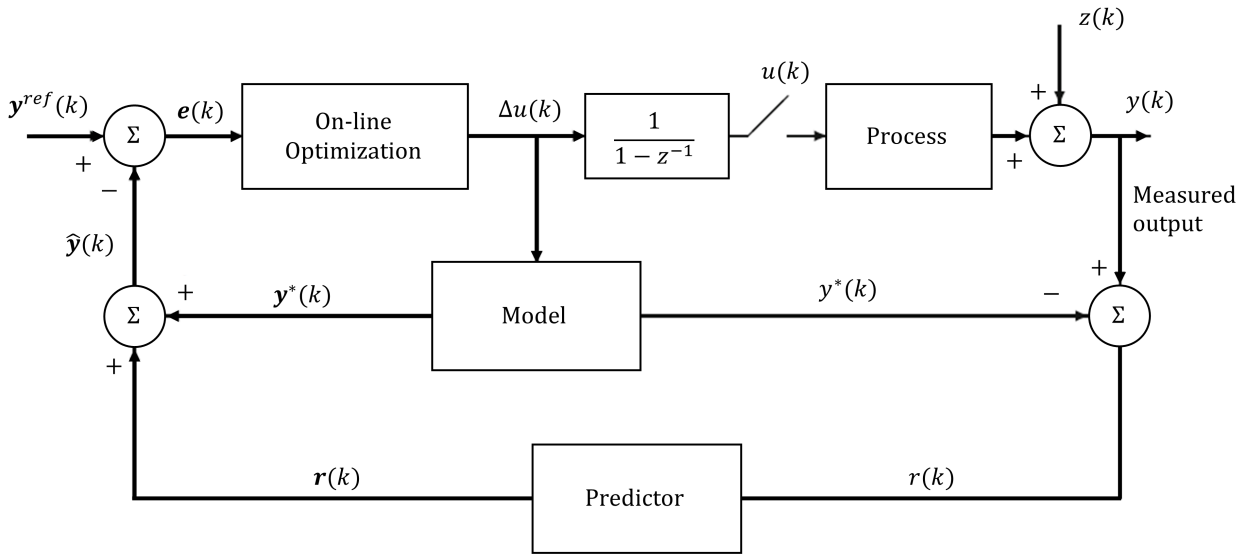


Fig. 5.8: Block diagram of model predictive control [38, p. 638]

Model predictive control is an advanced control technique and the most popular method of choice in process industries, especially for problems of multivariable, constrained or nonlinear nature, in

which case the PID controller and its proven tuning methods lose their superb efficiency [61, pp. 680–681, 65, p. 1252]. Also, if a model of the process to be controlled is available, then advanced control strategies such as model predictive control which utilise said model should be preferred to simple methods like PID control [65, p. 1252]. MPC offers several advantages: [34, p. 368, 66, pp. 1–2]

- Constraints on inputs and outputs are systematically considered.
- The calculation of optimal set points coordinates the overall control calculations.
- Multivariable systems can easily be dealt with.
- MPC has intrinsic consideration of dead time.
- Feed forward control is included to compensate for measurable disturbances.

However, the benefits of model predictive control directly depend on the discrepancies between the real process and the used model and therefore on the model quality [66, p. 2].

5.3.2 Optimisation Problem

The set points for the control calculations are computed from an economic optimisation based on a linear steady-state model of the process [34, p. 369]. Optimisation targets usually include minimising a cost function or maximising a profit function [34, p. 369]. The considerable number of available MPC algorithms propose different cost functions for obtaining the control law and they also differ in the type of approach for the chosen model as well as the numerical solution path [30, p. 294, 66, pp. 18–19]. An overview across commercially used model predictive control algorithms is given in [64]. However, as stated in [66], the common goal of all MPC approaches is that the future output y on the considered horizon follows a determined reference signal w, while the control effort Δu necessary for doing so is penalised simultaneously. The general expression for such an objective function is

$$J(N_1,N_2,M) = \sum_{j=N_1}^{N_2} \delta(j)[\hat{y}(t+j\,|\,t) - w(t+j)]^2 + \sum_{j=1}^{M} \lambda(j)[\Delta u(t+j-1)]^2, \tag{5.10}$$

in which N_1 and N_2 are the minimum and maximum prediction horizons, M is the control horizon and the coefficients $\delta(j)$ and $\lambda(j)$ are sequences that consider the future behaviour, usually represented by constant values or exponential sequences. The term $\hat{y}(t+k\,|\,t)$ is the expected value of $y(t+k)$ with available information at instant time t [66, pp. 18–19]. To obtain the corresponding values $u(t+k\,|\,t)$, it is required to minimise the functional J of Equation 5.10 [66, p. 21]. For that goal, the predicted output values $\hat{y}(t+k\,|\,t)$ are computed as a function of past values of inputs and outputs and future control signals by substituting the chosen model into the cost function, which leads to an expression whose minimisation results in the desired values [66, p. 21].

5.3.3 MPC Design and Tuning

A number of parameters must be specified in the design of a MPC system:

1. Sampling period T_S and model horizon N

   It is recommended in [34] to choose the sampling period T_S and the model horizon N in a manner that $NT_S = t_{set}$, where t_{set} is the settling time for the open-loop response. This way, the model reflects the full effect of an input variation over the time required to reach a steady state. The model horizon N is typically chosen in a range of $30 \leqslant N \leqslant 120$ [34, p. 384].

2. Control horizon M and prediction horizon P

   Recommendations for the control and prediction horizon tuning are given by [34]. With an increasing control horizon M, the controller tends to perform more aggressive actions and the computational effort is increasing. Typical rules of thumb for the control horizon selection are $5 \leq M \leq 20$ and $N/3 < M < N/2$. If required, a different value of M can be specified for each input. The prediction horizon P is often chosen as $P = N + M$ for the simple reason of taking the full effect of the last manipulated variable move into account. A decreasing value of P tends to result in a more aggressive controller. If the settling times of different outputs are rather dissimilar, it is also possible to choose a different value of P for each output [34, p. 384].

3. Weighting matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$

   In multivariable systems, the optimisation problem can be adapted through prioritising input and output values with the help of weighting matrices [34, p. 384]. This results in

   $$J(N_1, N_2, M) = (\hat{\boldsymbol{y}} - \boldsymbol{w})^T \boldsymbol{R}(\hat{\boldsymbol{y}} - \boldsymbol{w}) + \Delta \boldsymbol{u}^T \boldsymbol{Q} \Delta \boldsymbol{u}, \tag{5.11}$$

   in which the output weighting matrices Q and R allow the output and input variables to be weighted according to their relative importance[34, p. 384]. The variables are weighted individually, whereas the most important variables are assigned the largest weights [34, p. 384].

4. Reference Trajectory $\boldsymbol{y}^{ref}$

   The desired future output behaviour in a model predictive control application can be specified as a set point, high and low limits, a reference trajectory or a funnel [34, p. 380]. The point of using a reference trajectory is to enable a gradual transition to the desired set point [34, p. 385]. The reference trajectory can be specified in several different ways, for example, with a tuning factor or by utilising the performance ratio concept, which makes use of the ratio of desired closed-loop settling time to the open-loop settling time [34, p. 385].

Chapter 6

Implementation at the Equal Channel Angular Pressing (ECAP) test bed

This chapter guides through the modelling process and control design involved in the creation of a sensorless control system and presents the simulated and experimental results.

6.1 Model Construction

6.1.1 Approach

Models of the specimen and the heating elements are necessary for a sensorless specimen temperature control strategy. The specimen model is required to accurately compute the specimen temperature based on the heating element output. The heating element models are required for control design, as they are characterised by a limited rate in output change that should be considered. It is assumed that as the six heating element utilised in the test bed are identically constructed components of the same type and manufacturer, a model derived for one heating element will also represent all others sufficiently. It is also assumed that the specimen model is a single-input single-output (SISO) model where one of the two controllable heating elements serves as the input. This assumption holds true, as the other controllable heating element is simply required to follow the trajectory of the first with a specified fixed heating temperature difference for temperature homogeneity instead of having an independent influence on the specimen temperature, which would make it a multiple-input single-output (MISO) model. The non-controllable heating elements are to be treated as non-existent in the modelling process.

The modelling procedure for both components is adapted from the strategy outlined in Section 4.1.1. The resulting process is depicted in Figure 6.1. The method used for parameter estimation in every case is the global least squares system identification approach described in Section 4.3.3, which is why the choice of estimation method is not shown in Figure 6.1. Also, heating element model and specimen model are subject to different quality criteria as a result of their separate purpose; While the specimen model accuracy is important for the steady-state set temperature cri-

terion, the heating element model influences the optimal controller settings. However, the heating element model should still be sufficiently accurate, as it would be cumbersome to deal with simulated temperature curves that the real heating array at the test bed cannot follow.

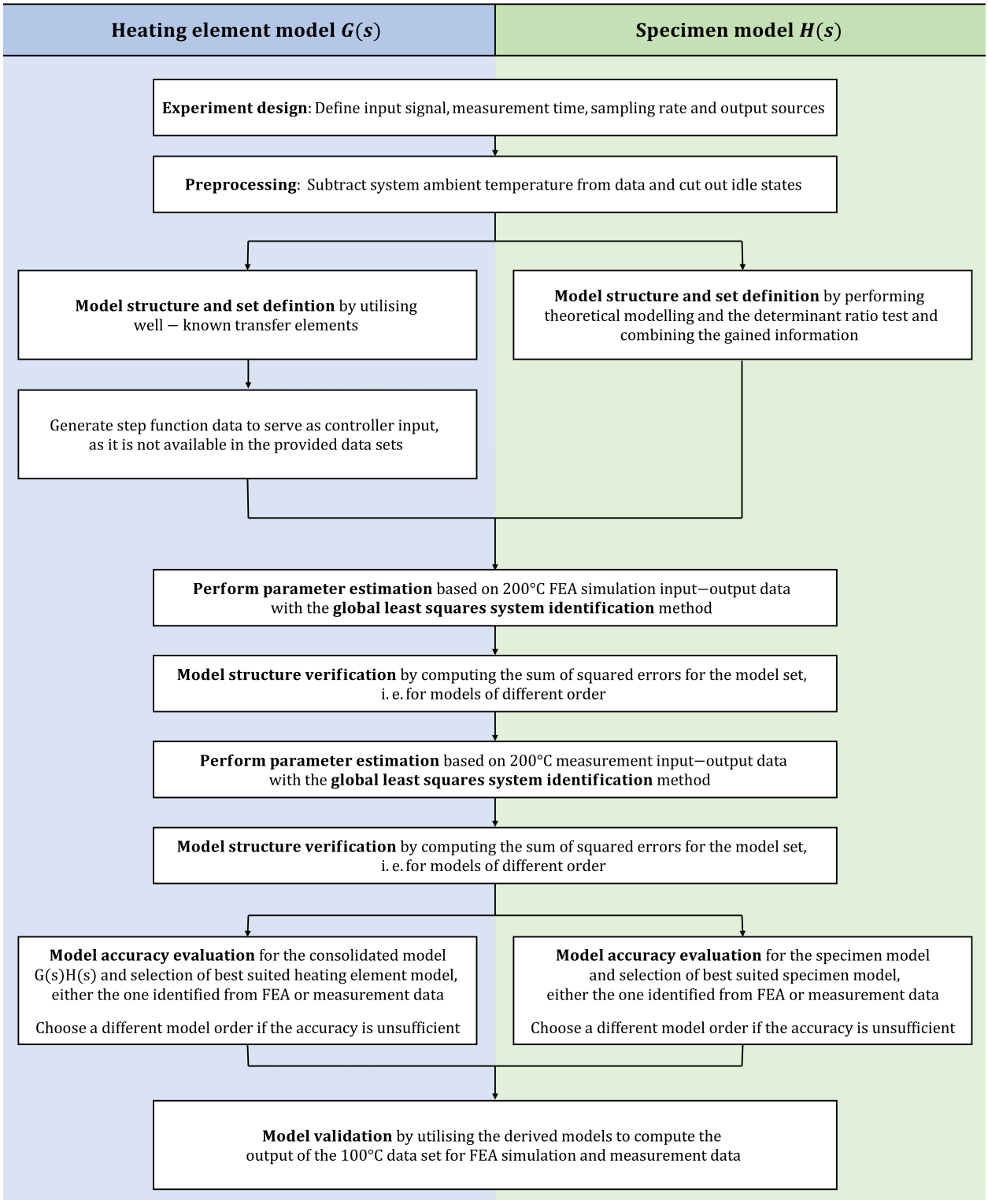


Fig. 6.1: Model construction approach

6.1.2 Experiment Design

Input Signal

The choice of the input signal for the purpose of generating input-output data for the identification of models for specimen and heating elements is based on the following considerations:

- A stationary temperature response after a long time is sought, to be able to determine the proportional gain as accurately as possible and observe occurrences of drift phenomena.
- A signal input as steep and high as possible should be utilized, however a heating element temperature of 400 °C should not be exceeded.
- The excitation of the system with a step response is advantageous, as this type of feedback allows the easy assessment of time delay and dead time behaviour.
- Heating up and cooling down behaviour of the specimen should be seen in the data, to be able to prioritize one or the other should different models be required.
- Provoking a well-behaved system response plays a major role, as non-uniform sampling with potentially high time intervals between recorded data points is utilised.

A response starting with an input step as steep as sensible and a cool down phase at the end of the experiment was chosen to fulfil the aforementioned requirements. There are also no oscillations of temperature expected by provoking the system in this way, therefore a well-behaved function between the sampling points can be expected. As the range for the desired specimen temperature in experiments is 100 °C - 200 °C, the experiment to generate the data for modelling was chosen at the upper boundary of 200 °C. This requires a steep heating element input and expectantly excites all possible thermal effects as the whole temperature range is covered. The bottom middle heating element was selected as the input actuator, as it has to provide the higher temperature out of the two controllable heating elements. For the purpose of model validation, data of experiments at the lower boundary of 100 °C were chosen to cover the whole experiment temperature range in the modelling procedure.

Measurement Time and Sampling Rate

The measurement time was defined as 1000 s, as the maximum duration of the heating up process is 800 s. In addition, the temperature of the specimen did not reach a stationary value even after a FEA simulation over a time range of 3000 s was conducted. This can be credited to the occurrence of temperature drift phenomena. The sampling intervals could not be selected or altered and are specified in Table 6.1 and 6.2.

Output Data

Two sources of input-output data are provided for an experiment at a set specimen temperature of 100 °C and 200 °C respectively, i.e. a total amount of four data sets is available for the modelling. In the first source, the input-output data is given by a FEA simulation of the test bed experiments. This simulation is conducted with adaptive time intervals, which results in non-uniformly spaced data in regard to the time intervals between the values, but enables a very time efficient FEA computation. The characteristics of this data source are shown in Table 6.1. In the second source, measurement data is generated by embedding a temperature sensor in the middle of the specimen. The measurements were originally recorded for validation purposes of the FEA model and are sampled uniformly. The characteristics of this source are defined in Table 6.2. As can be seen in the plots of the two data sources in Figure 6.2 and 6.3, there are differences in the temperature curves between the FEA simulation and the measurement data of the real process. This is especially obvious in Figure 6.3. However, the FEA data is vouched for by the Chair of Metal Forming as their main priority is achieving a homogenous and stationary constant temperature in the specimen. Both FEA models fulfil this condition, so the deviations in the transient response play a negligible role and the data is valid.

Table 6.1: Non-uniformly sampled FEA simulation data specification

FEA simulation data characteristics	Experiment at 100 °C	Experiment at 200 °C
Total data points	120	123
Total time range	0 … 962.6 s	0 … 986.0 s
Average time interval between data points	8 s	8 s
Minimum time interval between data points	1 s	1 s
Maximum time interval between data points	51.3 s	58.0 s
Initial system ambient temperature	35.25 °C	32.35 °C

Table 6.2: Uniformly sampled measurement data specifications

Measurement data characteristics	Experiment at 100 °C	Experiment at 200 °C
Total data points	2551	2625
Total time range	0 … 1020 s	0 … 1050 s
Time interval between data points	0.4 s	0.4 s
Initial system ambient temperature	35.20 °C	32.65 °C

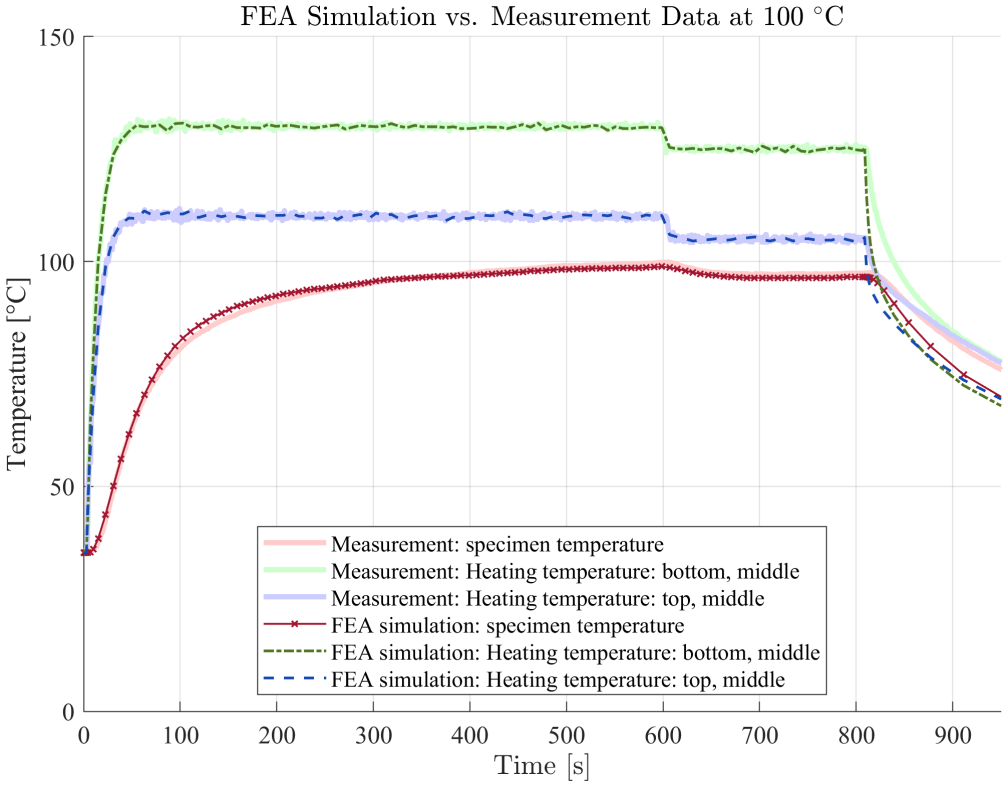


Fig. 6.2: Available data of the experiment at 100 °C

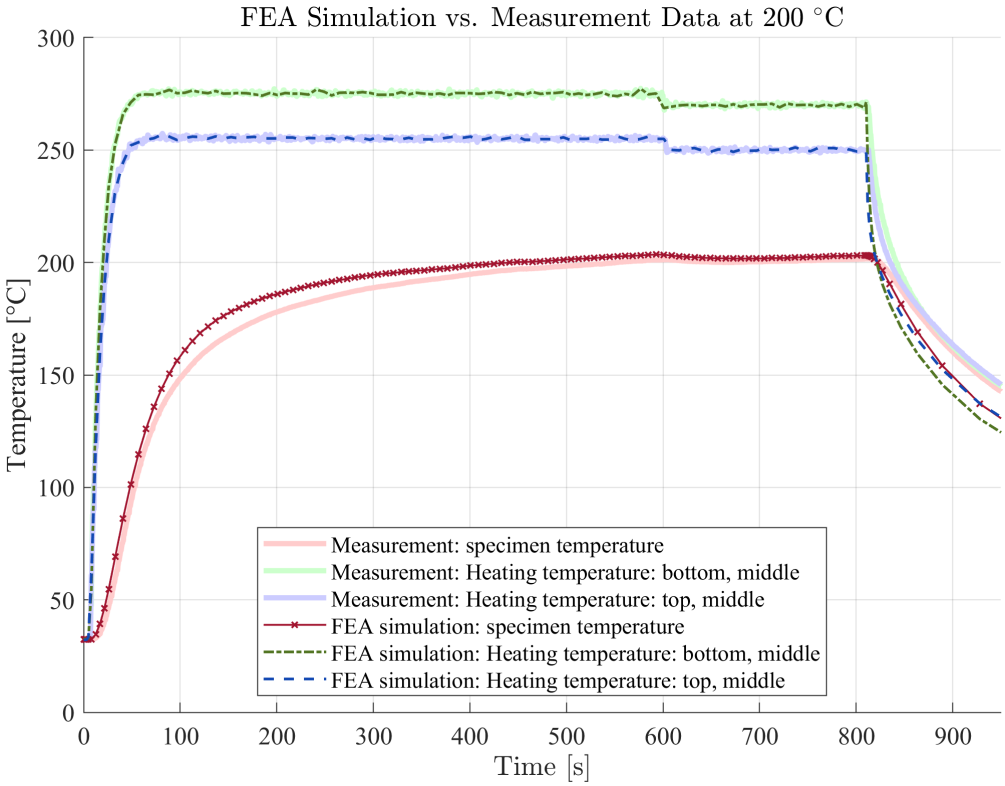


Fig. 6.3: Available data of the experiment at 200 °C

6.1.3 Numerical Implementation of the Global Least Squares System Identification

If it is assumed that the noise in the data provided by the temperature measurement sensors of the test bed as well as the FEA simulation is independent and identically distributed, then the covariance is defined by

$$\boldsymbol{\Lambda} = \sigma^2 \boldsymbol{I} \tag{6.1}$$

in a way that it multiplies the cost function by a constant and uniform factor (cf. [49, p. 4]) It can therefore be omitted, which simplifies Equation 4.43 and Equation 4.44 to

$$\varepsilon(\psi) = \| \boldsymbol{P} \left(\left((\boldsymbol{H} \otimes \boldsymbol{I_n}) \boldsymbol{L}^{-} (\boldsymbol{B} \otimes \boldsymbol{I_n}) - (\boldsymbol{G} \otimes \boldsymbol{I_n}) \right) \boldsymbol{u} - \boldsymbol{\xi} \right) \|_2^2 \tag{6.2}$$

and

$$\boldsymbol{P} = \boldsymbol{I} - (\boldsymbol{H} \otimes \boldsymbol{I_n}) \tilde{\boldsymbol{V}} \left((\boldsymbol{H} \otimes \boldsymbol{I_n}) \tilde{\boldsymbol{V}} \right)^{+}. \tag{6.3}$$

respectively. Utilising the Equations 4.38, 4.35 and 6.3, the optimal model parameters are identified by minimising the cost function specified in Equation 6.2. The numerical procedure to provide the solution in the sense of the method of global least squares system identification was realised in *MATLAB*™. A function `model_ss()` was created that calculates the squared error as defined in Equation 6.2. This function is shown in Listing 6.1. In it, the utilised state-space model is specified in controllable canonical form, meaning that $a_n = 1$ for any model of order n. As strictly proper transfer functions to describe causal, physical systems are utilised, the requirement $b_n = 0$ holds true for any model of order n. `model_ss()` includes state-space models of order $n = 1$ to $n = 3$, although higher order models can be added easily.

Listing 6.1: Function `model_ss.m` to compute the squared error in the global least squares system identification method in *MATLAB*™

```
1 function sqErr = model_ss(psi, specimen_temp, heating_temp, t, ...
     D_matrix)
2 % Syntax:
3 %   sqErr = modelss(psi, specimen_temp, heating_temp, t, D_matrix)
4 %
5 % Description:
6 %   This function calculates the squared error for the parameters of
7 %   a state-space model given by the psi compared to observed ...
     system data based on global least squares system identification
8 %
9 % Inputs:
```

```
%   psi              : Vector of parameter estimates
%   specimen_temp    : n x 1 array of the specimen temperature
%   heating_temp     : n x 1 array of the heating element temperature
%   t                : n x 1 array containing corresponding time ...
   values to the specimen
%                     and heating element temperatures
%   D_matrix         : Numerical differentiation matrix
%
% Outputs:
%   sqErr            : Squared error based on the calculation of the
%                      parameter estimates in comparison to the ...
   measurement data

order = 2;       % Predefine the model order
n = length(t);   % Calculate the number of data points
I = eye(order); % Create identity matrix of size order x order
In = eye(n);     % Create identity matrix of size n x n

% Define the model parameters based on the given psi
% Define the state-space model in controllable canonical form
if order == 1
    a0 = psi(1);
    b0 = psi(2);
    b1 = 0;

    A = -a0;
    B = 1;
    C = b0 - b1*a0;
    D = b1;
    rankDef = 1;
elseif order == 2
    a0 = psi(1);
    a1 = psi(2);
    b0 = psi(3);
    b1 = psi(4);
    b2 = 0;

    A = [[0 1]; [-a0 -a1]];
    B = [0 1]';
    C = [b0 - b2*a0, b1 - b2*a1];
    D = b2;
    rankDef = 2;
elseif order == 3
    a0 = psi(1);
    a1 = psi(2);
```

```
53      a2 = psi(3);
54      b0 = psi(4);
55      b1 = psi(5);
56      b2 = psi(6);
57      b3 = 0;
58
59      A = [[0 1 0]; [0 0 1]; [-a0 -a1 -a2]];
60      B = [0 0 1]';
61      C = [b0 - b3*a0, b1 - b3*a1, b2 - b3*a2];
62      D = b3;
63      rankDef = 3;
64  end
65
66  L = kron( I, D_matrix ) - kron( A, In ); % Calculate the linear ...
        operator
67  [U,S,V] = svd( L ); % Compute the full SVD
68
69  % Set up the preliminaries for the parametric solution
70  Sp = diag( 1./diag( S( 1:order*n - rankDef, 1:order*n - rankDef ) ) );
71  L_minus = V( :, 1:order*n - rankDef ) * Sp * U( : ,1:order*n - ...
        rankDef )'; % Calculate the particular generalized inverse of L
72  V_tilde = V( :, order*n - rankDef + 1:order*n ); % Calculate basis ...
        for the null space in L
73
74  % H and G possess the corresponding structure to C and D
75  H = C;
76  G = D;
77
78  % Calculate the projection matrix
79  P = In - (kron(H, In) * V_tilde) * pinv(kron(H, In) * V_tilde) ;
80
81  % Evaluate the cost function
82  r = P * (kron(H, In) * L_minus * kron(B, In) - kron(G, In)) * ...
        heating_temp - P * specimen_temp;
83
84  % Calculate the squared error
85  sqErr = norm( r, 2 )^2;
```

To solve for the optimal model parameters, the function `model_ss()` is then called by the non-linear solver `fminsearch()`, which utilises the Nelder-Mead simplex algorithm. This is shown in Listing 6.2. The function inputs are a vector of initial model parameters `phi_0` for the search as well as the temperature of the bottom middle heating element `heating_temperature`, the specimen `specimen_temperature` and their common sampling instants `t` on an absolute time axis. Both temperature data arrays were shifted to an initial system ambient temperature of 0 °C by

subtracting the original initial system ambient temperature from every data point. in the course of data preprocessing. The required numerical differentiation matrix `D_matrix` based on `t` is created by the function `diffColloc()`, which is provided by the toolbox [67]. `D_matrix` is a matrix of first order derivates and is calculated based on the polynomial interpolation as described in Appendix A.2. The computation of `D_matrix` is conducted outside of `model_ss.m` to minimise the time needed for each iteration of the function.

Listing 6.2: Exemplary code to perform the minimum search by utilising `fminsearch()` and `model_ss()` for a second order model in *MATLAB*™

```
% Define starting parameters for the minimum search
a0 = 9.091e-5;
a1 = 0.02776;
b0 = 6.558e-5;
b1 = 0.01278;

% Create a parameter vector
psi0 = [a0, a1, b0, b1];

% Calculate the numerical differentiation matrix
D_matrix = diffColloc(t);

% Set options for the solver
options = optimset('TolFun',1e-6, 'TolX', 1e-6, ...
   'MaxFunEvals',800*length(psi0), 'MaxIter', 800*length(psi0));

% Search optimal parameters psiOpt and function value fVal by ...
   utilising the Simplex method
[psiOpt, fVal] = fminsearch( @model_ss, psi0, options, ...
   specimen_temperature, heating_temperature, t, D_matrix);
```

Several strategies were applied if a problem with the identified cost function minimum occurred, e.g. if the found parameters led to a inappropriate model, or if no minimum was found by the nonlinear solver in an acceptable time frame:

- Plot the cost function for models consisting of up to two parameters and to visually assess the function minimum.
- Validate the physical parameter feasibility, e.g. if a parameter has to be positive or negative.
- Specify different starting values and magnitudes in the initial parameter vector and verifying if the same solution is reached.
- Keeping some parameters in the model fixed and change others as opposed to try and find all at the same time, to explore their influence on the cost function minimum.

- Keep the parameters of a correctly identified lower model order fixed when raising the model order and identify the additional higher order parameters, then optimize the lower order ones.
- Identify a model with uniformly sampled measurement data with standard methods and use its parameters as starting values for the identification based on non-uniformly sampled data of the corresponding experiment.
- Change the solver tolerances and number of maximum iterations.
- Conduct a cross-validation with the *MATLAB™ System Identification Toolbox* if applicable.

6.1.4 Specimen Model Structure and Candidates Definition

Specimen Model Dead Time Assessment

For the purpose of assessing whether a dead time term is required in the specimen model, the FEA simulation data of both experiments at respectively 100 °C and 200 °C was evaluated. A slight ascending slope of the specimen temperature can be recognised between the first rise of the heating element temperature and the next sampling point, i.e. the smallest possible distance in time available in the data in both cases. The temperature rise between those two sampling points has been calculated and visualised in Figure 6.4 for the experiment at 100 °C and 200 °C respectively. No time shift between input and output is observed. As a result, the dead time is assumed to be zero and a dead time term is not required for the specimen model structure.

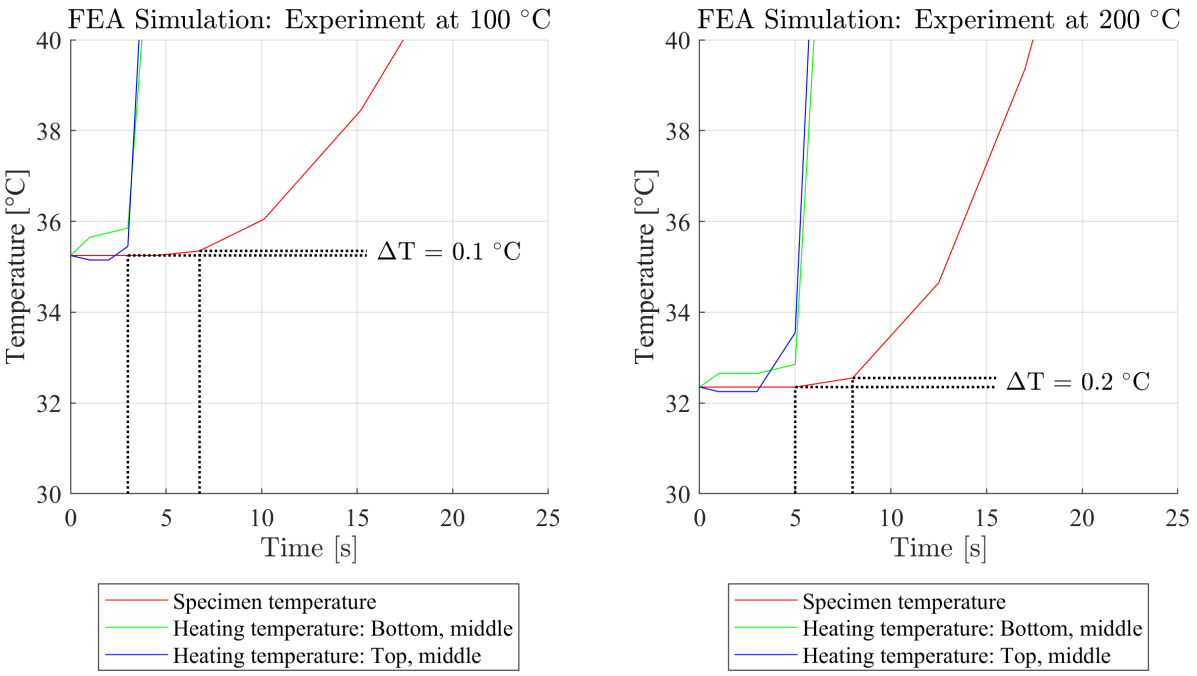


Fig. 6.4: Dead time assessment

Model Order Estimation by Determinant Ratio Test

The two sets of available FEA simulation data are expected to be relatively free of noise and therefore suited for the determinant ratio test, if the model to be identified is assumed to be linear. The determinant ratio was consequently computed with the specimen temperature as output and the temperature of the bottom middle heating element as input. Figure 6.5 shows that for both simulations, the most significant change in the logarithmic plot of the determinant ratio is from system order $n = 1$ to $n = 2$. From this result, the specimen model can be expected to be of order $n = 2$.

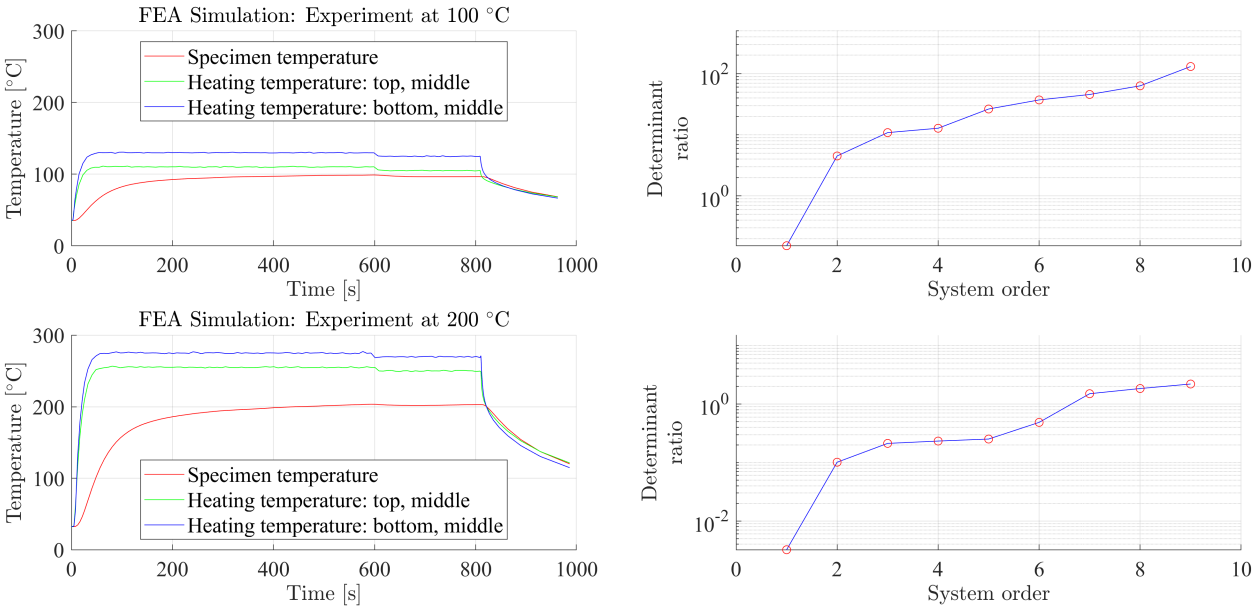


Fig. 6.5: Determinant ratio test for simulation data

Model Order and Structure Estimation by Theoretical Modelling

Theoretical modelling of the specimen was performed by creating a simplified lumped parameter model for the purpose of gaining further information about the required model order and structure. In the test bed, the specimen can be assumed to be in direct contact with the receiver, and the receiver in direct contact with the heating elements. This means that in both cases the transfer of heat takes place between two bodies with direct physical contact. Thus, the main mechanism of heat transfer from the heating elements to the specimen is heat conduction and the other mechanisms can be omitted. Based on this assumptions, a simplified model of the specimen containing all heat transfers q_i and thermal resistances R_i can be created. This is depicted in Figure 6.6, where q_{OUT} denotes the heat transfer out of the system, which is normal to all areas which are not subject to heat transfer from the heating elements normal to their surface. On the other hand, q_{IN} denotes the heat transfer into the system stemming from the upper and lower heating circuits. The thermal

resistances on the front and back are the same and thus denoted by R_F, the side resistances by R_S. Although the area for the upper and lower thermal resistances of the specimen are the same, they are separated here into R_U and R_L, due to the respective heating circuits having not the same distance to the specimen which will probably result in unequal heat transfer from above and below. To build a simple single-input single-output model by utilizing the temperature of the bottom

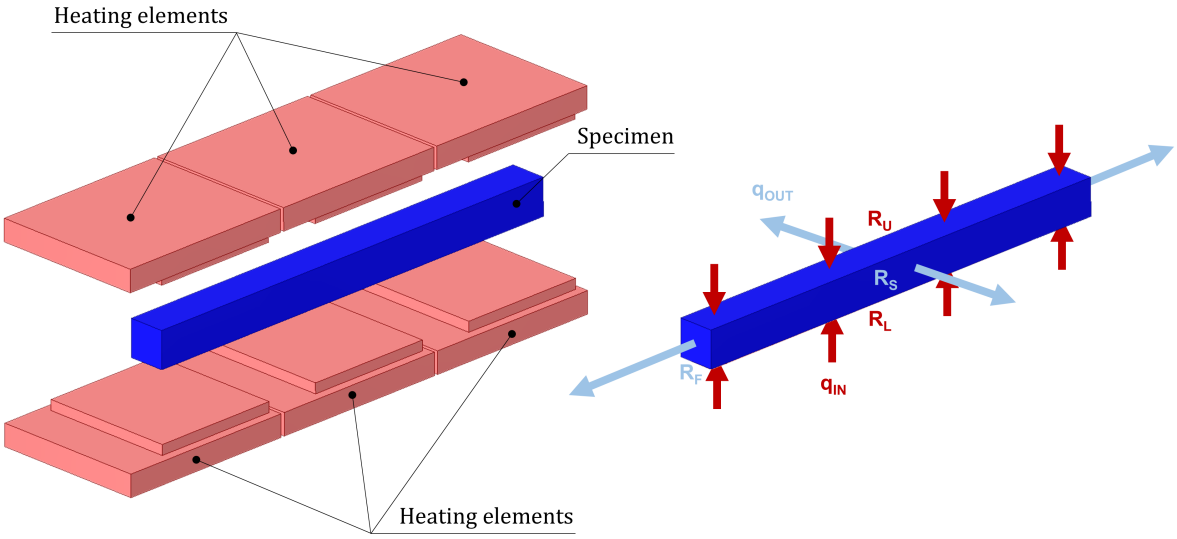


Fig. 6.6: Overview of the heat flows in the system

middle heating element as input and the homogenous, average specimen temperature in its middle as output, only the lower half of the system is required for the modelling. This principle is visualized in Figure 6.7. It can be seen that the heat flow emitted by the heating element passes through

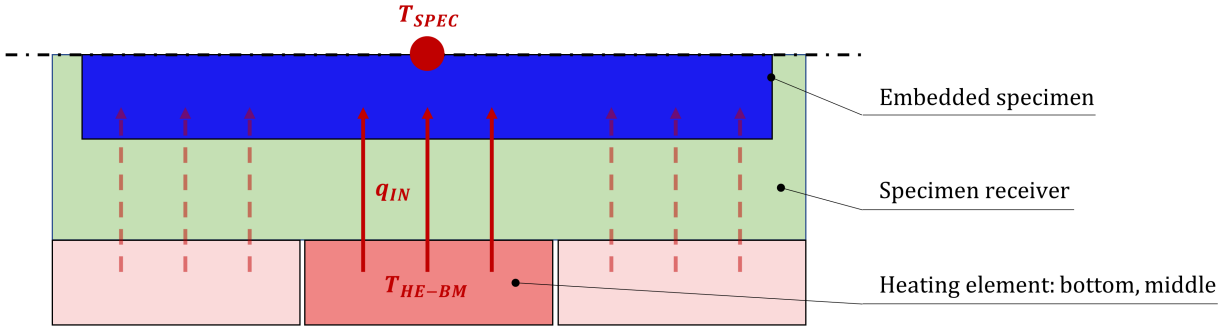


Fig. 6.7: Lower half of the heating system

two systems on its way to the specimen temperature measuring point T_{SPEC}: The receiver and the specimen embedded within it. Although each of them represent a separate transfer function $H_1(s)$ and $H_2(s)$, a summary transfer function can be formulated as $H(s)$, which is depicted in Figure 6.8. The first step of deriving $H(s)$ is taken by creating the thermal balance equation

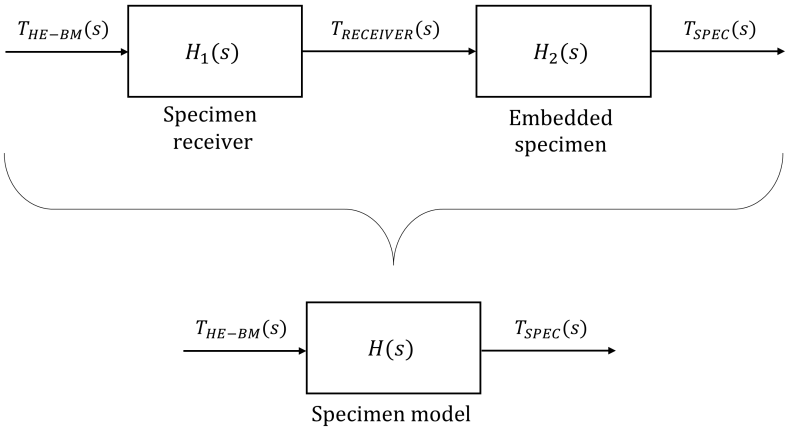


Fig. 6.8: Consolidated specimen model

$$C_R\dot{T}_R = \frac{1}{R_{IN,R}}(T_{HE,BM} - T_A) - \frac{1}{R_{OUT,R}}(T_R - T_A), \tag{6.4}$$

for the system H_1 and separating T_A as a constant term, leading to

$$C_R\dot{T}_R = \frac{1}{R_{IN,R}}T_{HE,BM} - \frac{1}{R_{OUT,R}}T_R + T_A\left(\frac{1}{R_{OUT,R}} - \frac{1}{R_{IN,R}}\right), \tag{6.5}$$

in which C_R is the thermal capacitance of the receiver, T_R and $\dot{T}_R$ its temperature and corresponding derivative, T_A the initial system ambient temperature and $T_{HE,BM}$ in heating temperature of the bottom middle heating element. The variables $R_{IN,R}$ and $R_{OUT,R}$ represent the total thermal resistances in the receiver for leaving or entering heat flows. A similar equation can be formulated for the system H_2 with

$$C_{SPEC}\dot{T}_{SPEC} = \frac{1}{R_{IN,SPEC}}(T_R - T_A) - \frac{1}{R_{OUT,SPEC}}(T_{SPEC} - T_A). \tag{6.6}$$

and by correspondingly separating T_A,

$$C_{SPEC}\dot{T}_{SPEC} = \frac{1}{R_{IN,SPEC}}T_R - \frac{1}{R_{OUT,SPEC}}T_{SPEC} + T_A\left(\frac{1}{R_{OUT,SPEC}} - \frac{1}{R_{IN,SPEC}}\right). \tag{6.7}$$

In this case, the index *SPEC* signifies the thermal properties of the specimen. The initial system ambient temperature T_A can be eliminated from both equations by setting it to zero. The model should oscillate around an idle position of zero, and T_A can be added later by means of superposition. The thermal resistances and capacitances are constants which are subject to parameter estimation later. Thus, they can simply be denoted by R_i and C_i from here on. Applying the Laplace transform on Equation 6.5 and 6.7 yields

$$C_1\, s\, T_R = \frac{1}{R_1}T_{HE,BM} - \frac{1}{R_2}T_R \tag{6.8}$$

and

$$C_2\, s\, T_{SPEC} = \frac{1}{R_3} T_R - \frac{1}{R_4} T_{SPEC}. \tag{6.9}$$

The substitution of Equation 6.8 into 6.9 to eliminate T_R then yields

$$C_2\, s\, T_{SPEC} = \frac{\dfrac{1}{R_1 R_3}}{C_1 s + \dfrac{1}{R_2}} T_{HE,BM} - \frac{1}{R_4} T_{SPEC}. \tag{6.10}$$

The transfer function $H(s)$ is then defined as the ratio of overall system output to input:

$$H(s) = \frac{T_{SPEC}}{T_{HE,BM}} = \frac{\dfrac{1}{R_1 R_3}}{\left(C_1 s + \dfrac{1}{R_2}\right)\left(C_2 s + \dfrac{1}{R_4}\right)} \tag{6.11}$$

Simplifying all the constant terms in Equation 6.11 to arbitrary constants d_i results in

$$H(s) = \frac{T_{SPEC}}{T_{HE,BM}} = \frac{d_1}{d_2 s^2 + d_3 s + d_4} = \frac{b_0}{a_0 + a_1 s + a_2 s^2}, \tag{6.12}$$

which is a second order system. The expected model structure therefore is a linear model of second order, which corresponds well with the result of determinant ratio test. Although the simplified thermal differential equations suggest that $b_1 = b_2 = 0$, we will assume that $b_1 \neq 0$ and $b_2 = 0$ for a strictly proper transfer function until confirmed otherwise in the model structure validation.

Specimen Model Structure and Set

The a priori model order determination based on the determinant ratio test and the theoretical modelling lead to the same result, which is a second order linear model. The requirement that the model must be time-invariant holds true and as there is no dead time expected,

$$H(s) = \frac{b_0 + b_1 s + b_2 s^2}{a_0 + a_1 s + a_2 s^2} \tag{6.13}$$

is the suggested model structure of the specimen. However, the adjacent model orders $n = 1$ and $n = 3$ should also be investigated in the parameter estimation to verify the structure. The model candidates for the specimen therefore are

$$H(s) = \left\{ \frac{b_0 + b_1 s}{a_0 + a_1 s}, \quad \frac{b_0 + b_1 s + b_2 s^2}{a_0 + a_1 s + a_2 s^2}, \quad \frac{b_0 + b_1 s + b_2 s^2 + b_3 s^3}{a_0 + a_1 s + a_2 s^2 + a_3 s^3} \right\}. \tag{6.14}$$

6.1.5 Heating Element Model Structure and Candidates Definition

The model for the heating element should emulate the combined behaviour of the internal controller of the test bed and the heating actuator controlled by it as depicted in Figure 6.9.

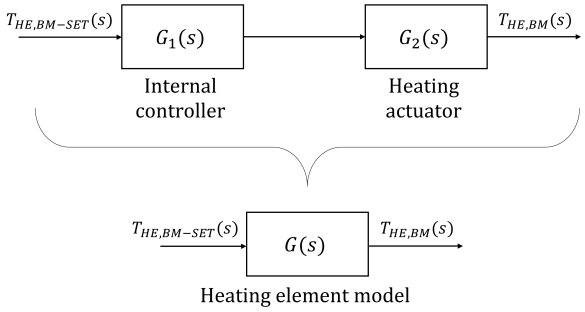


Fig. 6.9: Consolidated heating element model

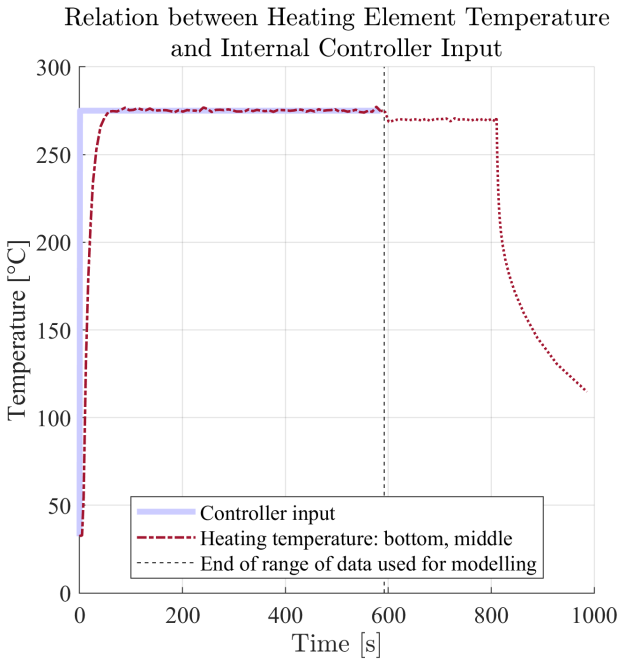


Fig. 6.10: Assumed relationship between internal controller and heating element temperature

The relationship between the internal controller input and the heating element temperature is assumed to possess a proportional gain of $V = 1$, as more information about the controller output is not required. The temperature response is presumed to be the result of a step input by the internal controller as shown in Figure 6.10. Using the approach of approximation by well-known transfer elements due to the typical low-order delay form of the step response seen in Figure 6.10, a corresponding delay structure was chosen for modelling the heating element. Thus, the models to be considered are a first order delay and the adjacent second order delay for verification purposes, resulting in

$$G(s) = \left\{ \frac{V}{(1+\tau_1 s)}, \quad \frac{V}{(1+\tau_1 s)(1+\tau_2 s)} \right\} \tag{6.15}$$

as the candidates for the heating element model. The range of input-output data for the parameter identification of the heating element model is the first constant temperature step marked in Figure 6.9. While the heating element temperature representing the output is available in the provided data sets, the controller step input is not, so it is simply generated.

6.1.6 Model Parameter Estimation and Structure Verification

Parameter Estimation based on Non-Uniformly Sampled Data

First, the specimen model was identified by utilising the FEA simulation input-output data. The transfer functions of first, second and third order of the model set were subjected to the global least squares system identification process. The results are documented in Table 6.3. A visual comparison of the computed model output and the FEA simulation data is given in Figure 6.11. While there is a significant drop in the sum of squared error from $n = 1$ to $n = 2$, this is not the case for $n = 3$. Therefore, $n = 2$ should represent the lowest sensible model order and suffice to create a reasonably accurate specimen model. For the heating element, first order and second order delay

Table 6.3: Specimen models identified from FEA Simulation Data with 122 data points

	$n=1$	$n=2$	$n=3$
$H(s)=$	$\frac{b_0+b_1 s}{a_0+a_1 s}$	$\frac{b_0+b_1 s+b_2 s^2}{a_0+a_1 s+a_2 s^2}$	$\frac{b_0+b_1 s+b_2 s^2+b_3 s^2}{a_0+a_1 s+a_2 s^2+a_3 s^2}$
a_0	0.0146	$7.0375 \cdot 10^{-6}$	$7.0367 \cdot 10^{-6}$
a_1	1	0.0196	0.0199
a_2	-	1	0.9994
a_3	-	-	1
b_0	0.0102	$7.0056 \cdot 10^{-6}$	$7.0058 \cdot 10^{-6}$
b_1	0	0.0124	0.0125
b_2	-	0	-0.0116
b_3	-	-	0
$\frac{1}{2}\sum_{k=\hat{n}}^{N} \varepsilon_y^2(k,\hat{n})$	1278	187	212

models were subjected to parameter estimation on basis of the FEA simulation data. The outcome is shown in Table 6.4 and in Figure 6.12. Although there is still a notable squared error drop from $n = 1$ to $n = 2$, the small value of τ_2 compared to τ_1 suggests that this parameter can be omitted. Thus, $n = 1$ is chosen until proven otherwise in the model accuracy evaluation.

Table 6.4: Heating element models identified from FEA simulation data with 74 data points

	$n=1$	$n=2$
$G(s)=$	$\frac{1}{1+\tau_1 s}$	$\frac{1}{(1+\tau_1 s)(1+\tau_2 s)}$
τ_1	12.56	11.53
τ_2	-	1.03
$\frac{1}{2}\sum_{k=\hat{n}}^{N} \varepsilon_y^2(k,\hat{n})$	528	166

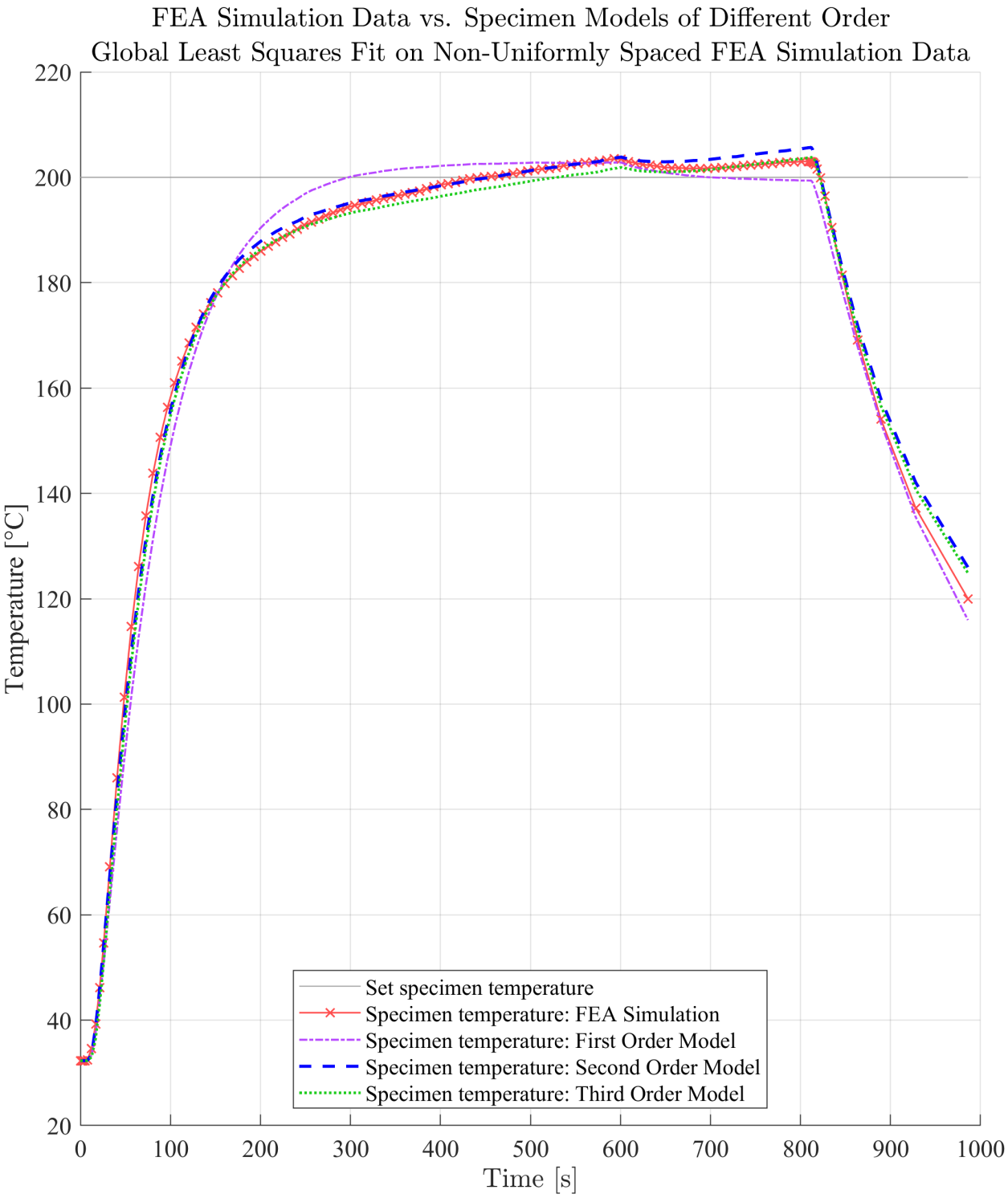


Fig. 6.11: Specimen models of different order fitted to non-uniformly spaced FEA simulation data at 200 °C experiment

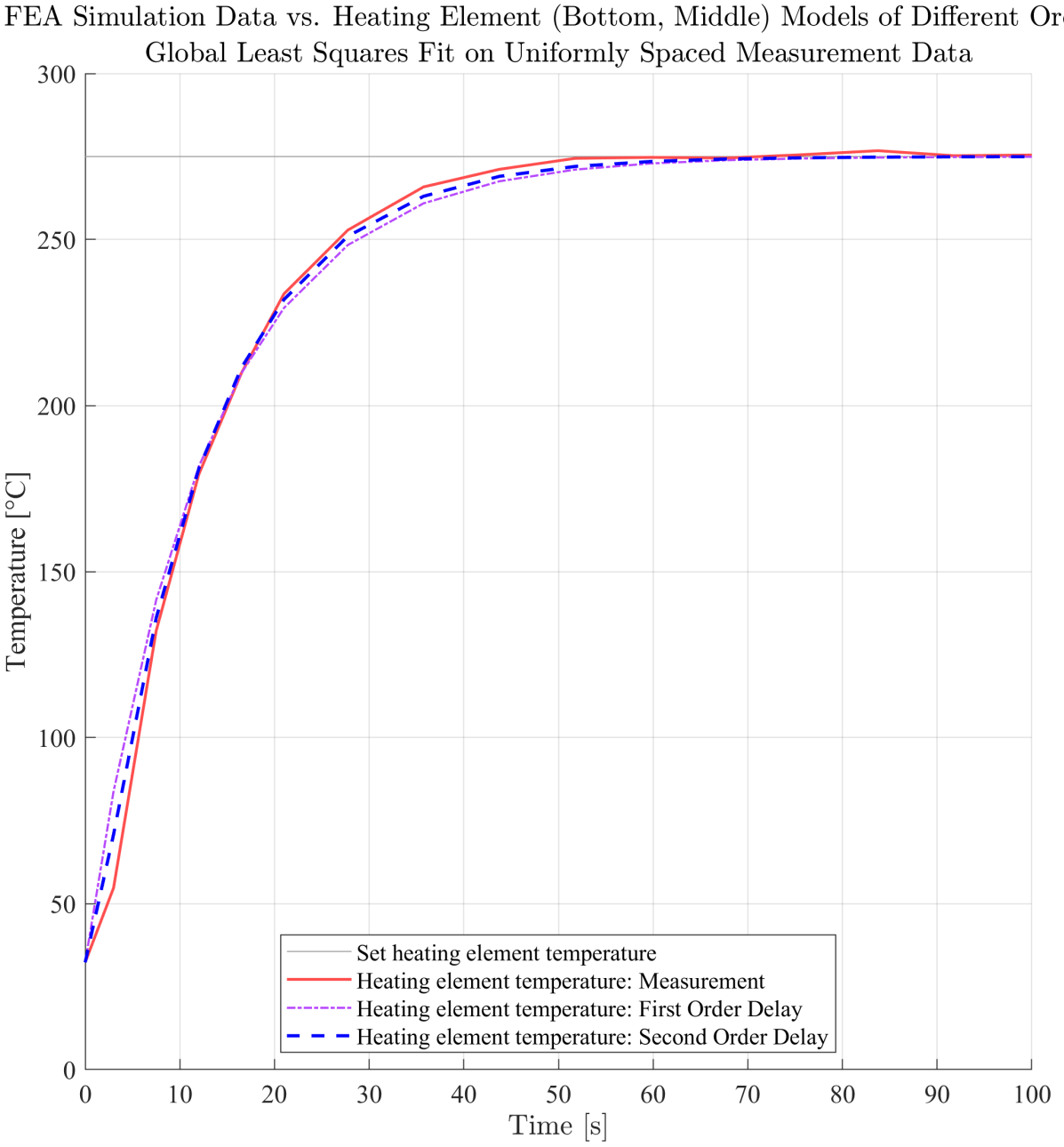


Fig. 6.12: Heating element models of different order fitted to non-uniformly spaced FEA simulation data at 200 °C experiment

Parameter Estimation based on Uniformly Sampled Data

For the purpose of comparing the results of the global least squares system identification performed on non-uniformly and uniformly sampled data, the procedure was executed again for the same model candidates, but this time the measurement data was utilised. For the specimen model, the results are shown in Table 6.5. The computed model output in comparison to the measurement data is depicted in Figure 6.13. There is a significant drop of the squared error sum from $n = 1$ to $n = 2$ and also from $n = 2$ to $n = 3$. However, as can be seen in Figure 6.3, the specimen temperature curve from measurement and FEA simulation data differs, which might explain the suggestion of a third order model as a better fit. As the squared error is summed over 2399 data points, the error per individual data point is reasonably small already and the $n = 2$ curve shows an excellent steady-state fit. Therefore, $n = 2$ is chosen for the specimen model and as the lowest sensible order until proven otherwise by the resulting model accuracy. For the heating element model, the results

Table 6.5: Specimen models identified from measurement data with 2399 data points

	$n = 1$	$n = 2$	$n = 3$
$H(s) =$	$\frac{b_0 + b_1 s}{a_0 + a_1 s}$	$\frac{b_0 + b_1 s + b_2 s^2}{a_0 + a_1 s + a_2 s^2}$	$\frac{b_0 + b_1 s + b_2 s^2 + b_3 s^2}{a_0 + a_1 s + a_2 s^2 + a_3 s^2}$
a_0	0.01326	$5.931 \cdot 10^{-5}$	$2.825 \cdot 10^{-5}$
a_1	1	0.02265	0.00709
a_2	-	1	0.23460
a_3	-	-	1
b_0	0.00907	$4.318 \cdot 10^{-5}$	$2.015 \cdot 10^{-5}$
b_1	0	0.01120	0.00293
b_2	-	0	-0.00812
b_3	-	-	0
$\frac{1}{2}\sum_{k=\hat{n}}^{N} \varepsilon_y^2(k,\hat{n})$	$2.7269 \cdot 10^4$	3137	346

are shown in Table 6.6. There is no significant drop in the squared error sum between $n = 1$ and $n = 2$. In addition, τ_2 is negligibly small. Therefore, a model order of $n = 1$ should suffice.

Table 6.6: Heating element models identified from measurement data with 1439 data points

	$n = 1$	$n = 2$
$G(s) =$	$\frac{1}{1 + \tau_1 s}$	$\frac{1}{(1 + \tau_1 s)(1 + \tau_2 s)}$
τ_1	11.06	10.64
τ_2	-	$2.92 \cdot 10^{-10}$
$\frac{1}{2}\sum_{k=\hat{n}}^{N} \varepsilon_y^2(k,\hat{n})$	1124	783

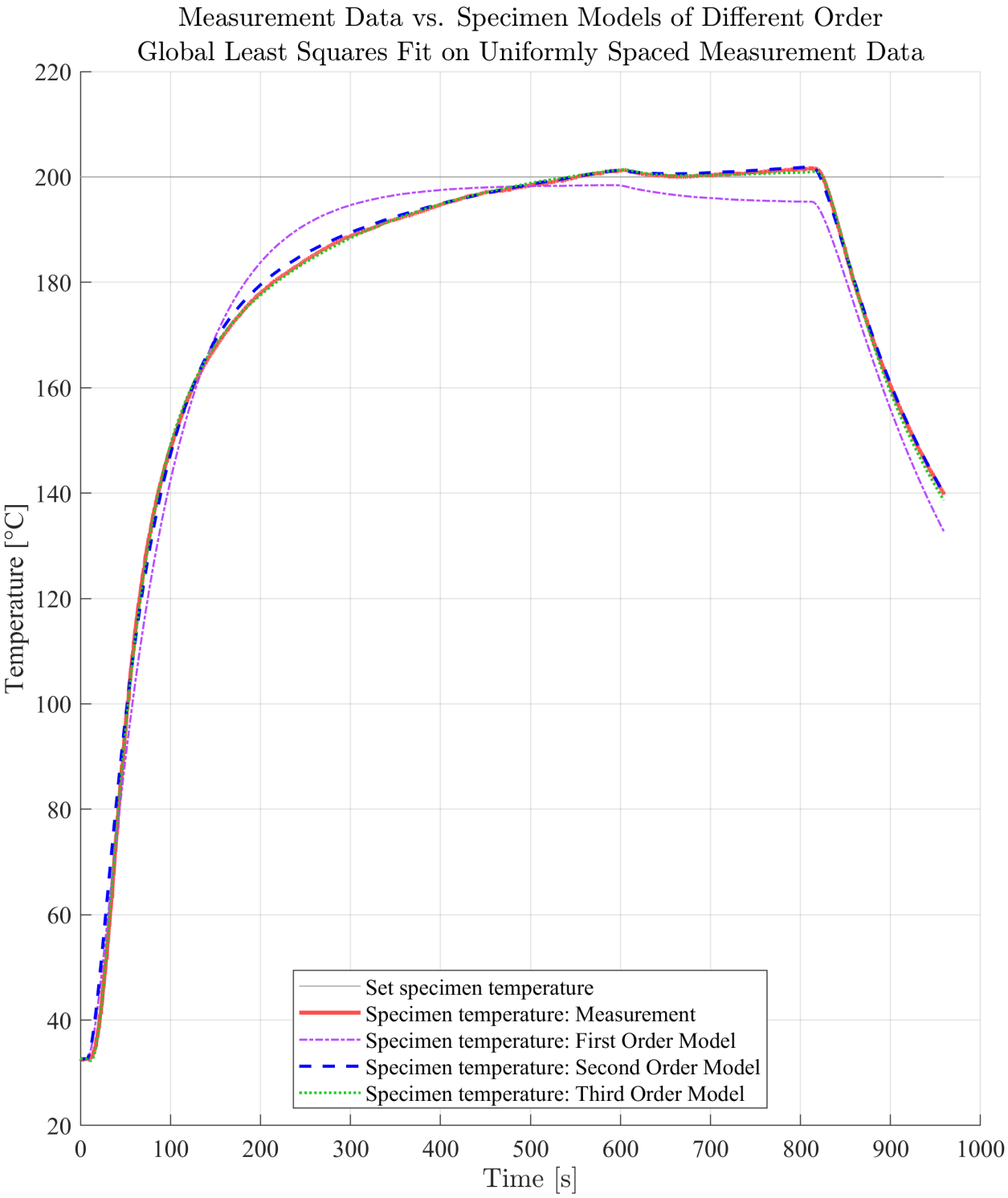


Fig. 6.13: Specimen models of different order fitted to uniformly sampled measurement data at 200 °C experiment

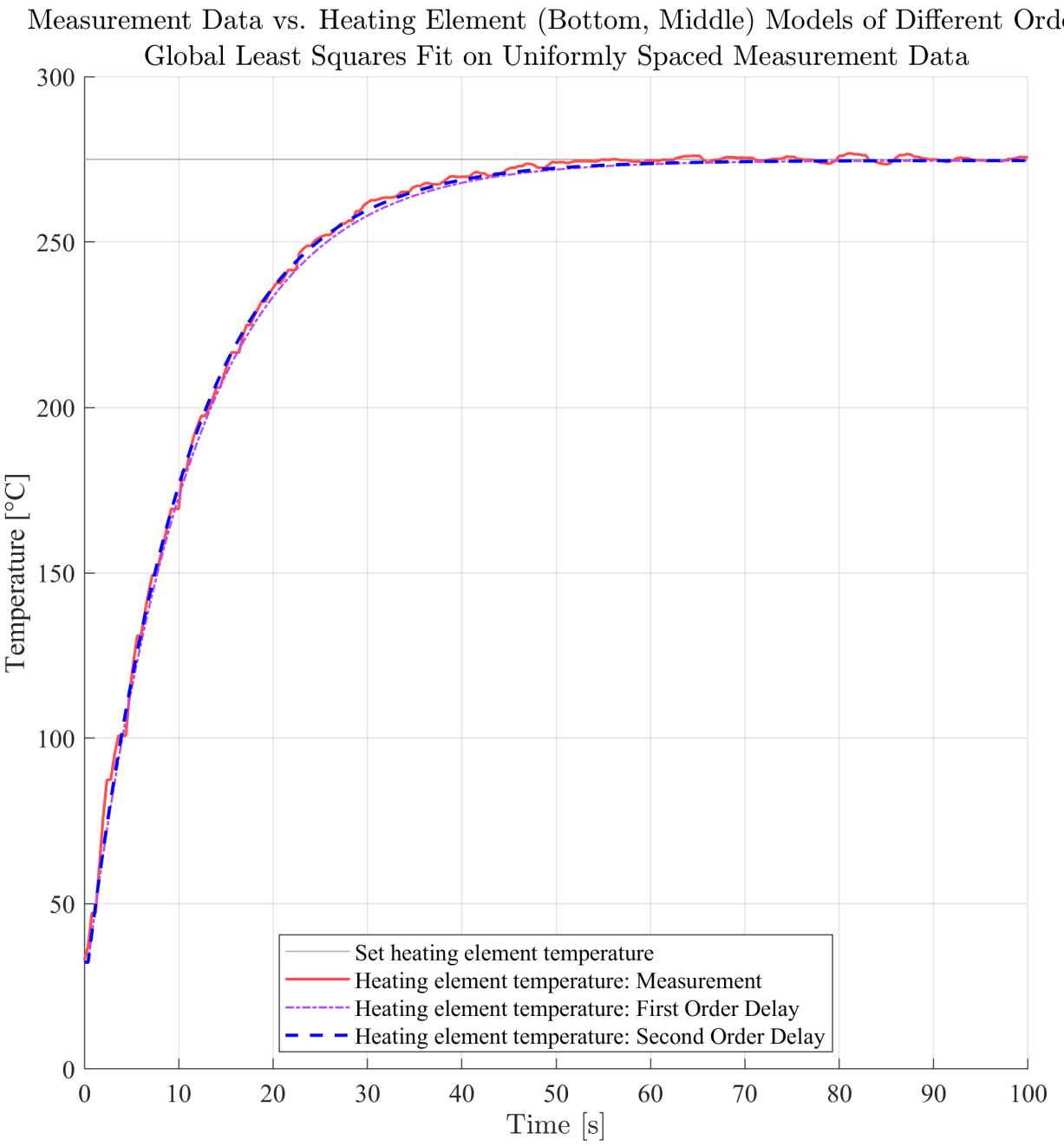


Fig. 6.14: Heating element models of different order fitted to uniformly sampled measurement data at 200 °C experiment

Results of the Parameter Estimation and Structure Verification

Summing up, the model order and therefore structure for the specimen model was identified as $n = 2$ and for the heating element as $n = 1$. Both are primarily the result of the computed summed squared error and the principle of choosing the lowest sensible model order. In addition, the graphical comparison between the model outputs and the real process was also considered. The priority for the specimen model is a good fit in steady-state, while its dynamic response is secondary. However, a squared error criterion is not able to distinguish in which part of the response the largest error occurs or if there are differing error magnitude trends along the observed trajectory. For example, in Figure 6.13 it can be seen that the second order model provides at least the same or an even better fit to the steady-state temperature zone than the third order model, although the sum of squared error suggests otherwise. Therefore a visual assessment is absolutely sensible to assess whether the model fits well in the zones that matter, as was also done in the FEA simulation of the specimen temperature shown in Figure 6.3. When assessing the heating element model, the relative magnitudes of τ_1 and τ_2 also played a role in the model order decision, due to the polynomial nature of the denominator of the second order delay.

6.1.7 Model Accuracy Evaluation and Comparison

The accuracy of the identified models consisting of their order and parameters for the experiment at 200 °C was computed by comparison of the model output with the corresponding measurement or FEA simulation data. The results are summed up in Table 6.7. The steady-state error criterion

Table 6.7: Model accuracy for the experiment at 200 °C

Model	*Sampling type* Used data points	Maximum error $S_1 = \max\|\varepsilon(k)\|$	Average error $S_2 = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k)}$
$H(s) = \dfrac{7.0056 \cdot 10^{-6} + 0.0124s}{7.0375 \cdot 10^{-6} + 0.0196s + s^2}$	*Non-uniform* 122	4.676 K	1.754 K
$H(s) = \dfrac{4.318 \cdot 10^{-5} + 0.0112s}{5.931 \cdot 10^{-5} + 0.02265s + s^2}$	*Uniform* 2399	3.078 K	1.617 K
$G(s) = \dfrac{1}{1 + 12.56s}$	*Non-uniform* 74	4.940 K	3.779 K
$G(s) = \dfrac{1}{1 + 11.06s}$	*Uniform* 1439	4.823 K	1.256 K

of ±2.5 % for the set specimen temperature equals ±5 K for the models at 200 °C. This is satisfied by both $H(s)$ models for the maximum and average error not only in a steady state scenario, but throughout the whole experiment length. Therefore the model order assumption holds true, as the model accuracy is sufficient. However, it can also be seen that the models which were created by identification from the uniformly sampled measurement data possess a superior accuracy. The results of the identification procedure based on non-uniformly sampled data still prove very interesting. Although only 5 % of the amount of data points were used compared to the uniform sampling based identification, the resulting errors appear in very similar value range.

The heating element models are planned to be used only to enable convenient controller design in the *Simulink*™ simulation, as they ensure that the possible controller output change rate is never exceeded. Therefore, they are not subject to the error criteria. However, the error criteria of ±5 K as well as the assumptions of Section 6.1.5 can be assessed in a simulation of the consolidated model

$$S(s) = H(s)G(s) = \frac{4.318 \cdot 10^{-5} + 0.0112s}{5.931 \cdot 10^{-5} + 0.02265s + s^2} \, \frac{1}{1 + 11.06s} \tag{6.16}$$

to get an idea of the heating element accuracy influence for the later control design. The result of this simulation is depicted in Figure 6.15. The error values are listed in Table 6.8, where it can be seen that maximum error criterion does not hold true if the whole time frame is observed. This is

due to a relatively large deviation in the beginning of the heating up process. After 40 s however, the consolidated model satisfies both error criteria. Considering that the criterion of ± 5 K is only required for a steady-state scenario, the consolidated model and thus, the heating element model and its identified order also proves sufficient for its purpose.

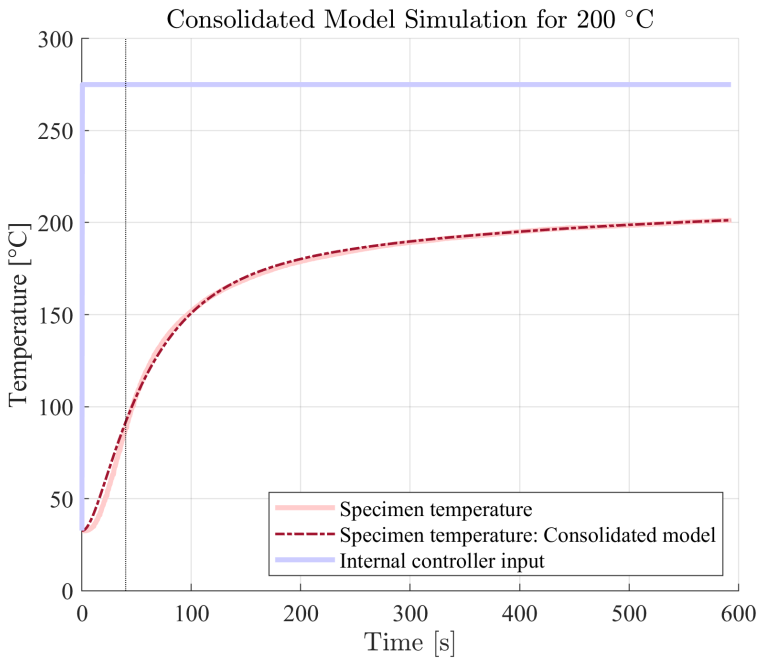


Fig. 6.15: Consolidated model simulation

Table 6.8: Consolidated model accuracy

Consolidated model $S(s)$	Maximum error $S_1 = \max\lvert\varepsilon(k)\rvert$	Average error $S_2 = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k)}$
Total time range	10.959 K	2.1051 K
$t > 40$ s	3.3252 K	0.963 K

Result of the Model Accuracy Evaluation and Comparison

Both identified specimen as well as heating element models were found to satisfy the accuracy criteria for their purpose. Therefore, no order or structure adjustment is required. However, the models created from measurement data will be utilised throughout all further steps and the later validation experiments on the test bed due to two reasons. First, they exhibit a superior accuracy, which is a major factor due to the goal of achieving the best possible accuracy in the sensorless control approach. Second, the FEA simulation data only describes the system adequately for a predefined input and the resulting steady-state temperature. However, this might not hold true for arbitrary input signals, which favours the measurement data for the model construction.

6.1.8 Model Validation by Simulation

The model validation was conducted by utilising the identified models to compute the output of the 100 °C data sets not used in modelling. The resulting accuracy for the validation data is shown in Table 6.9. The results for the specimen model identified from FEA simulation data are depicted in Figure 6.16, those for the model derived from measurement data in Figure 6.17. It can be seen that the model output is slightly higher then the measurements in both cases. This is probably due to process nonlinearities or thermal effects not considered in the modelling. The results for the heating element model validation are shown in Figure 6.16 and Figure 6.17 for the identification based on FEA simulation and measurement data respectively. Both provide an exceptional steady-state fit to the validation data. The output of the model created from measurement data also follows the trajectory excellently. In contrast, the model identified from the FEA simulation data deviates from the trajectory by overshooting it. However, this behaviour can already be observed in the model construction shown on the left side of Figure 6.16.

Table 6.9: Model accuracy for the validation data at 100 °C

Model	Validation data sampling type	Maximum error $S_1 = \max\lvert\varepsilon(k)\rvert$	Average error $S_2 = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\varepsilon^2(k)}$
$H(s) = \dfrac{7.0056 \cdot 10^{-6} + 0.0124s}{7.0375 \cdot 10^{-6} + 0.0196s + s^2}$	Non-uniform	4.193 K	2.954 K
$H(s) = \dfrac{4.318 \cdot 10^{-5} + 0.0112s}{5.931 \cdot 10^{-5} + 0.02265s + s^2}$	Uniform	3.108 K	1.644 K
$G(s) = \dfrac{1}{1 + 12.56s}$	Non-uniform	6.249 K	1.267 K
$G(s) = \dfrac{1}{1 + 11.06s}$	Uniform	4.160 K	0.609 K

The recommendation from the learnings of the validation is to create a separate specimen model for each experiment temperature. This enables the best possible result, as the temperature shift found using the specimen model derived for 200 °C on the 100 °C data sets also shifts the computed steady-state temperature. When utilising the criterion of ±2.5 % at 100 °C, the maximum specimen model error exceeds ±2.5 K in both derived models. Due to the shift-like nature of the phenomenon, the specimen model structure and accuracy should not be the underlying issue, but a rather different proportional gain at the lower temperature level. In contrast, the heating element model has shown a perfectly adequate behaviour for the validation data set. It is therefore suggested to utilise the same heating element model for all experiments regardless of their set temperature.

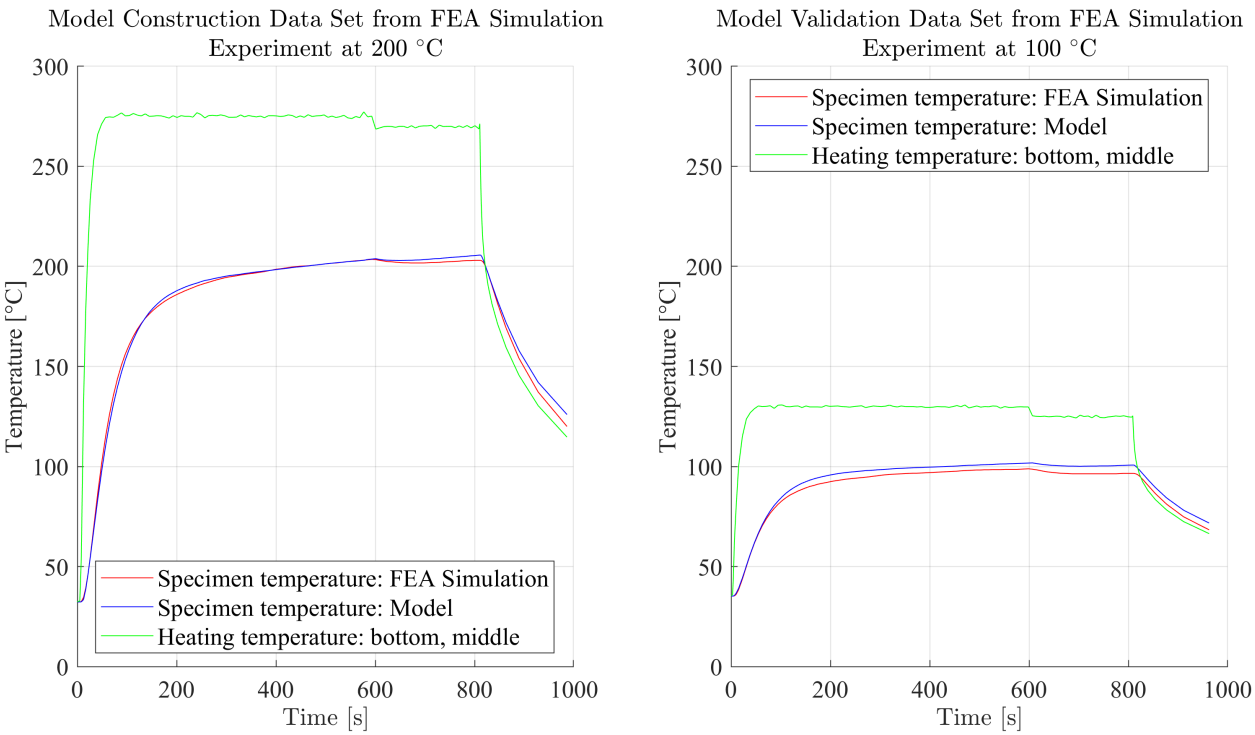


Fig. 6.16: Validation of the FEA simulation data based specimen model

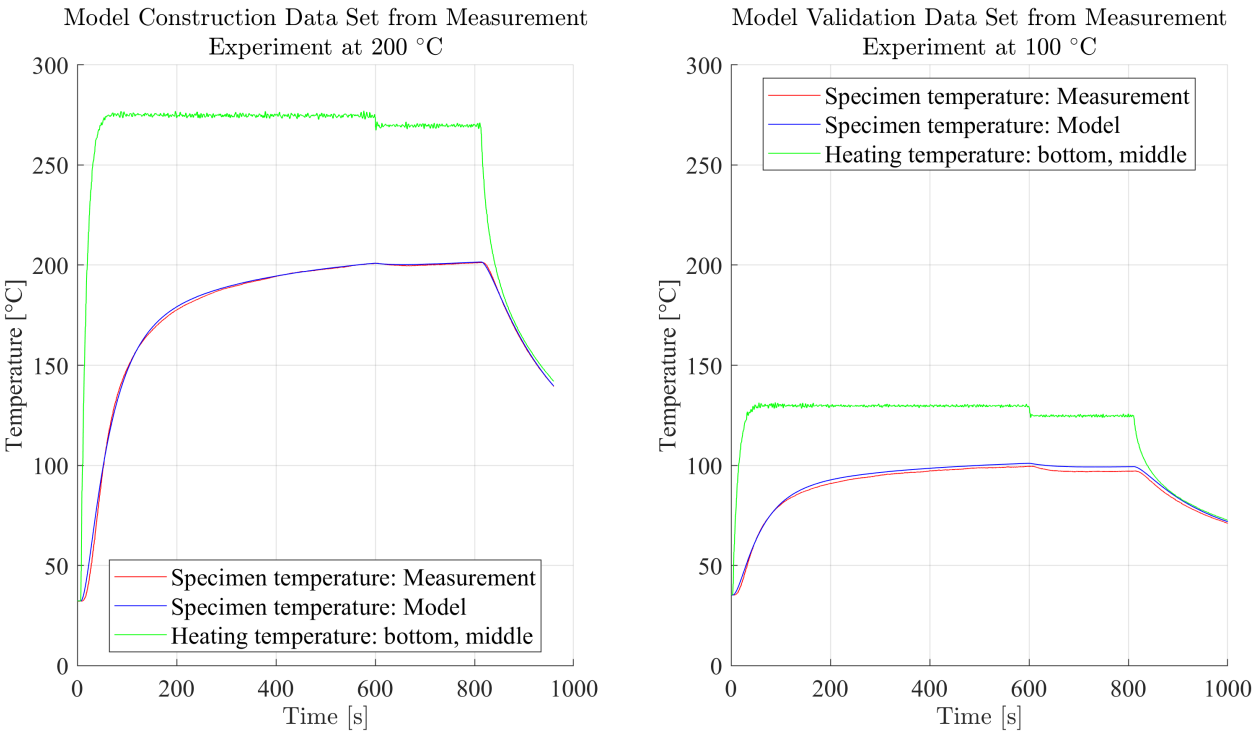


Fig. 6.17: Validation of the measurement data based specimen model

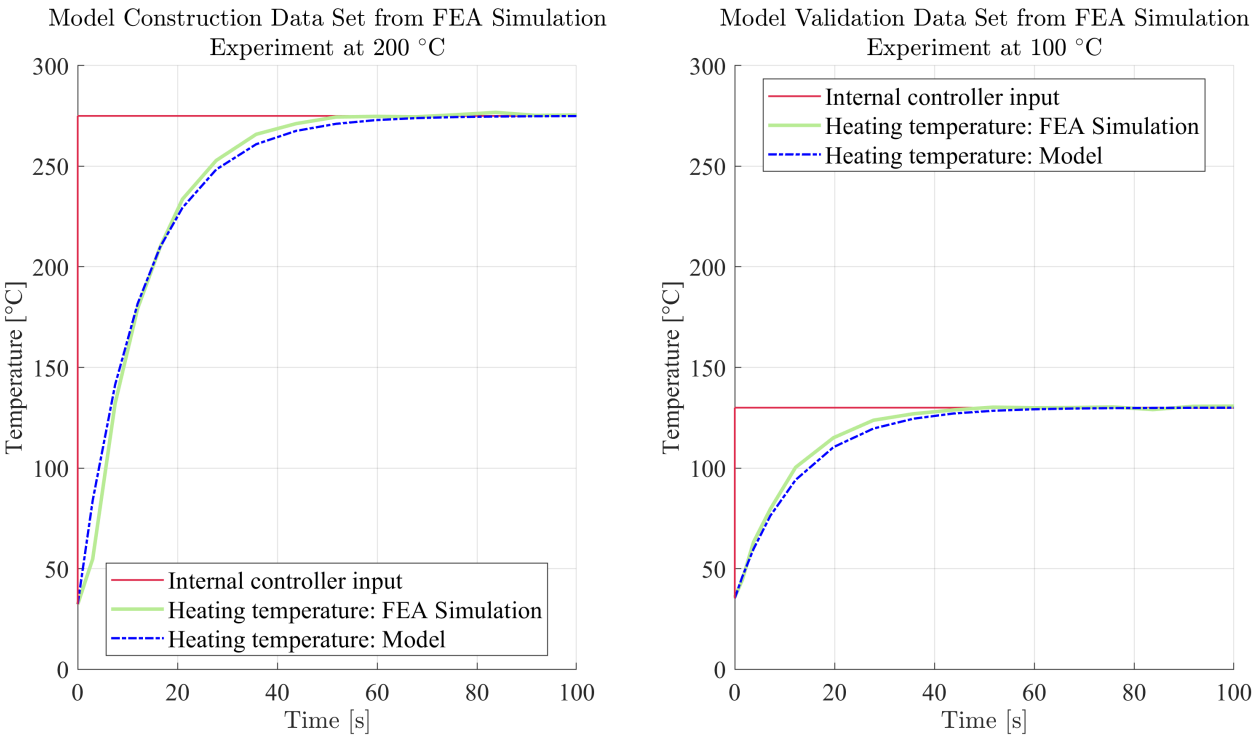


Fig. 6.18: Validation of the FEA simulation data based heating element model

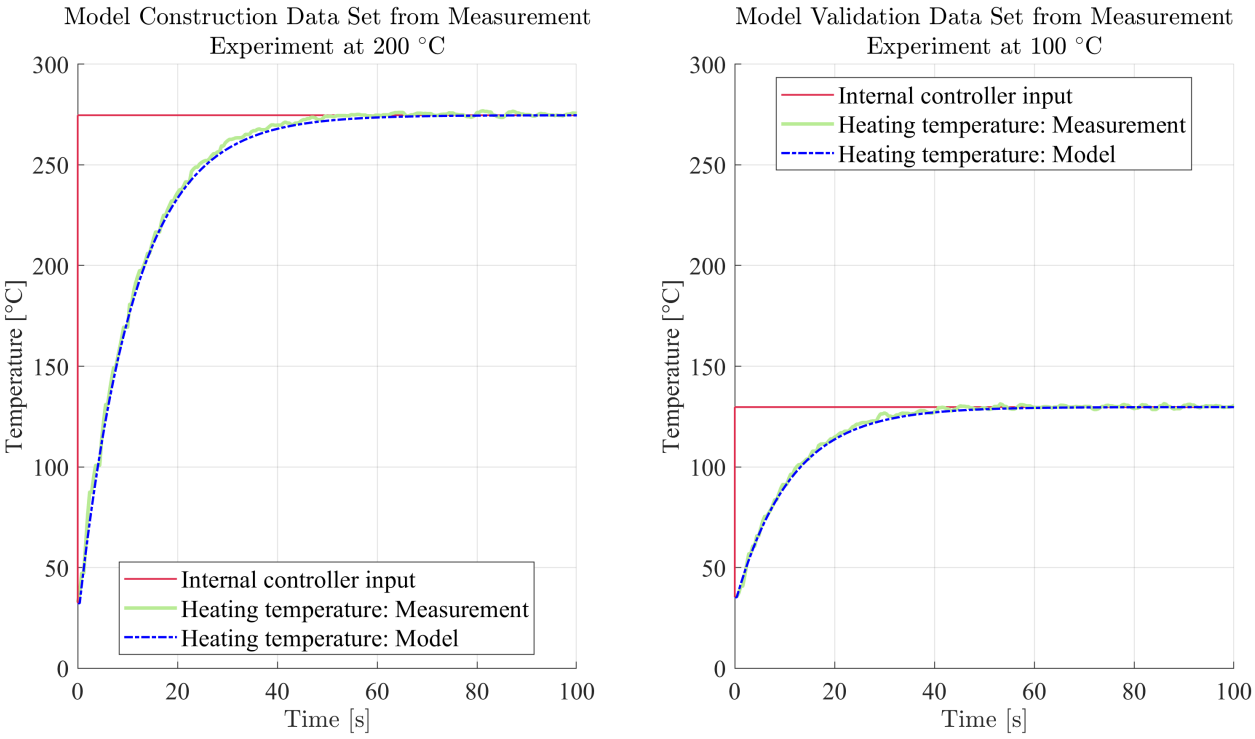


Fig. 6.19: Validation of the measurement data based heating element model

6.2 Open-Loop Temperature Control

6.2.1 Concept

The temperature control at the test bed was realised via open-loop control in which a .csv-file provides the required heating element set temperatures to the internal controller of the test bed. This set up is shown in Figure 6.20. The .csv-file is generated by means of a closed-loop simulation

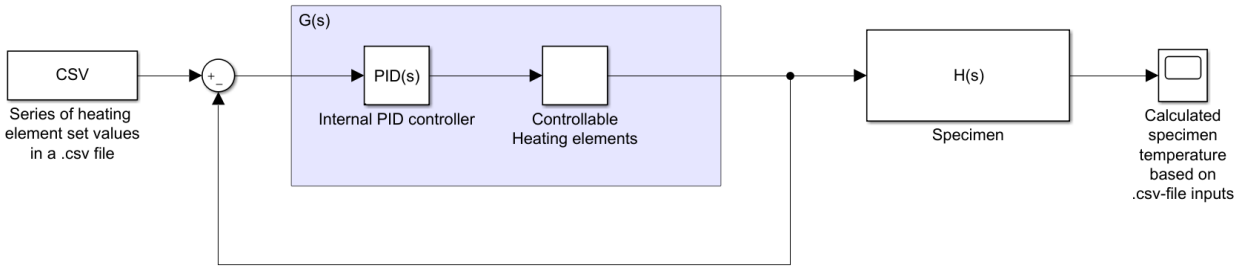


Fig. 6.20: Open-loop control principle for the test bed in *Simulink*™

in *Simulink*™ as depicted in Figure 6.21, in which the desired specimen temperature and heating curve characteristics are designed by controller tuning. The .csv-file then provides the required temperature trajectory for the two controllable heating elements to achieve the designed specimen response. In Figure 6.21, the initial system ambient temperature is subtracted from the specimen set value to shift the control starting point to zero as was done in the model construction. The bottom middle heating element temperature serves as input to the specimen model. The top middle heating element is not part of the simulation, but its set values are simply computed from the bottom middle temperature trajectory by subtracting a set difference specified in Table 6.10 for each data point in the .csv-file, whereas $T_{HE,BM}$ is the bottom middle and $T_{HE,TM}$ top middle heating element temperature. The different temperatures are required for temperature homogeneity in the specimen, the necessary divergence value is provided by the Chair of Metal Forming. The sampling time in *Simulink*™ has to be set to $T_s = 0.5\,\mathrm{s}$, which corresponds to the sampling time of the internal controller. At the end of the simulation, the series of simulated set temperature values for the bottom middle heating element as well as the consequential calculated set values for the top middle heating element are exported as a .csv-file with their corresponding sampling instants. This file can be imported at the test bed, as long as its structure follows the specifications shown in Table 6.11. By utilising the generated files for the temperature control, separate models can be used easily for each experiment. Therefore, the *Simulink*™ simulations and following validation experiments at the test bed are done with the most accurate models available, which are those created by identification from the measurement data. The specimen model used for experiments at 100 °C is

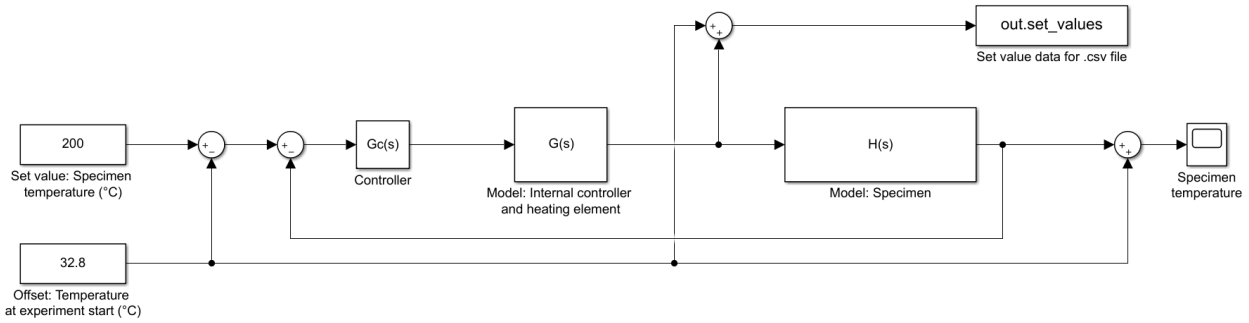


Fig. 6.21: Closed-loop simulation structure in *Simulink*™

Table 6.10: Heating temperature differences for specimen temperature homogeneity

Set specimen temperature	100 °C	200 °C
Required temperature difference $\Delta T = T_{HE,BM} - T_{HE,TM}$	15 K	20 K

Table 6.11: Structure of the .csv-file

Time [s]	Heating temperature: top, middle [°C]	Heating temperature: bottom, middle [°C]
...	...	...
6.5	90.23	105.23
7.0	94.39	109.39
7.5	98.38	113.38
...	...	...

$$H_{100}(s) = \frac{0.01141s + 7.790 \cdot 10^{-5}}{s^2 + 0.02720s + 11.28 \cdot 10^{-5}}, \tag{6.17}$$

which was identified from the measurement data of the experiment at 100 °C by utilising the same model structure as the already defined specimen model

$$H_{200}(s) = \frac{0.01120s + 4.318 \cdot 10^{-5}}{s^2 + 0.02265s + 5.931 \cdot 10^{-5}}, \tag{6.18}$$

for experiments at 200 °C from Section 6.1.7. The heating element model

$$G(s) = \frac{1}{1 + 11.06s} \tag{6.19}$$

is also simply taken from Section 6.1.7 and the same for both experiment temperatures. The proposed open-loop control concept offers the following advantages:

- Adjustment of the specimen temperature control is done easily and handily with the .csv-files.

- The simulation in *Simulink*™ enables changes of controller parameters and models very efficiently, which is a major factor due to the experimental environment.
- The compliance of process constraints can be examined before the real experiment in the *Simulink*™ simulation, minimizing the risk of constraint transgressions at the test bed.
- No on-line calculations are performed, as it would be the case in closed-loop control, which is an advantage if complicated models that require large computational effort are utilised.

6.2.2 Control Requirements

The Chair of Metal Forming has specified a tolerance of $\pm 2.5\,\%$ in the steady-state specimen temperature. The heating up process should take no longer than 800 s. Transient response characteristics take no priority, although the heating process should be as fast as possible. As the four outer heating elements have to be adjusted manually throughout the whole experiment, the time to reach target temperature should be approximately 200 s to give the staff enough time for adjustments during the heating process. In addition, the heating temperature of each element should not exceed 400 °C to prevent any damage. Control overshoot is undesirable, as the process of cooling down takes a long time and the thermal energy spent on the overshoot is wasted.

6.2.3 Controller Choice

As the control design revolves around the controller, the choice of the controller type for the closed-loop simulation should be made before taking further steps. A model predictive control approach as depicted in Figure 6.22 was explored due to easily specifiable value and rate constraints for manipulated variables, the effort minimisation in the algorithm to prevent overshoot, multivariable control and the simple definition of a set value trajectory. However, manipulated variable constraints

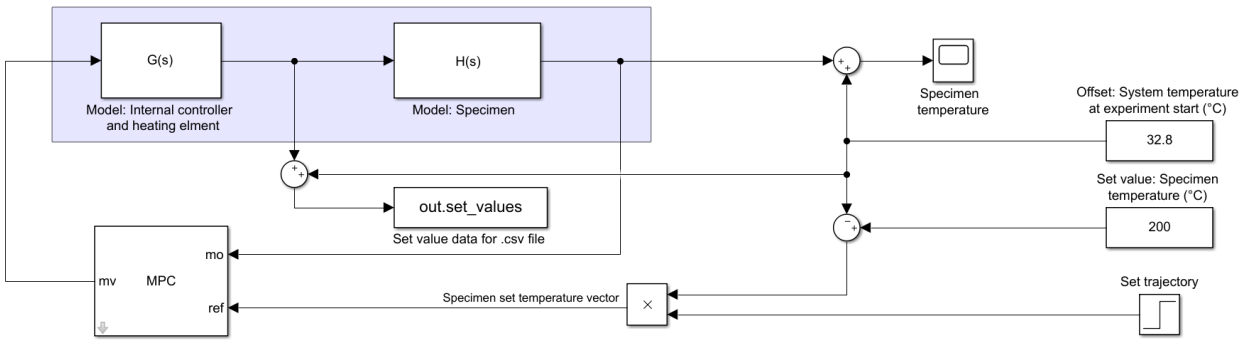


Fig. 6.22: Model predictive control closed-loop simulation in *Simulink*™

are not necessary because all temperatures are simulated and observed in *Simulink*™ before the

program execution on the test bed anyway. Multivariable control and trajectory definition are also not required. Although there are two heating elements to be controlled, the temperature of the top middle heating element has to follow the reference temperature trajectory of the bottom middle heating element with a specified temperature difference. This can be achieved better and much easier by manually tuning a PID control loop as shown in Figure 6.23 for the desired trajectory of the bottom middle heating element instead of utilising multivariable control. The corresponding top middle heating temperature is then simply calculated from this trajectory. In addition, PID control is much more approachable and easier to tune with little background knowledge of control theory in comparison to model predictive control.

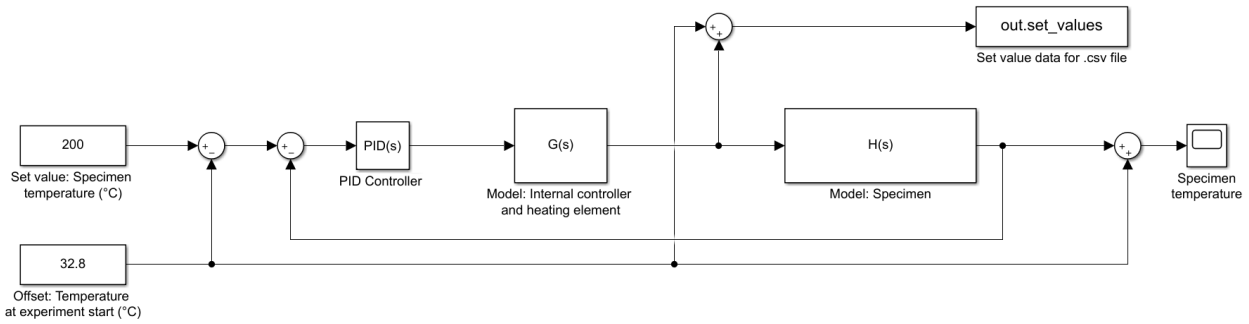


Fig. 6.23: PID control closed-loop simulation in *Simulink*™

The choice of controller therefore resulted in PID control, which is able to satisfy all specified control requirements while being easy and intuitively to implement and tune at the same time. Also, programmable PID control is already available at the internal PLC of the test bed, which possibly enables a future transfer of the open-loop approach to closed-loop temperature control.

6.2.4 Tuning and Simulation

The closed-loop control of the specimen temperature was simulated in *Simulink*™ for the purpose of controller tuning and the generation of the required .csv-file for open-loop control. The simulation with a pure P-controller shows that the resulting model does not exhibit sustained oscillation for any sensible proportional gain K_p, as can be seen in Figure 6.24. This means that the Ziegler-Nichols tuning method from Section 5.2.2 is not applicable. An alternative tuning procedure is the direct synthesis method, which is also described in Section 5.2.2. As there are the transfer functions of the specimen and the heating element available, the consolidated transfer function

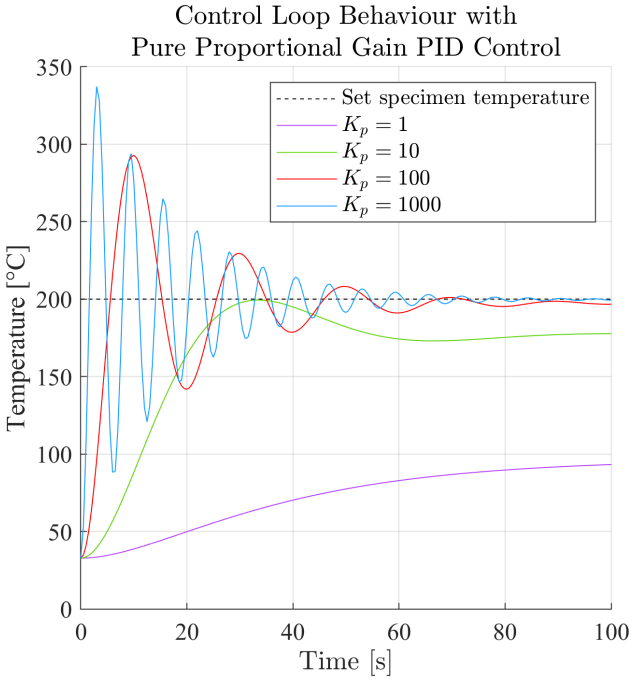


Fig. 6.24: Different magnitudes of proportional gain K_p ($K_i = K_d = 0$)

$$H_{res}(s) = H_{200}(s)G(s) = \frac{1}{1+11.06s}\,\frac{0.0112s+4.318\cdot10^{-5}}{s^2+0.02265s+5.931\cdot10^{-5}} = \frac{0.0112s+4.318\cdot10^{-5}}{11.06s^3+1.251s^2+0.02331s+5.931\cdot10^{-5}} \tag{6.20}$$

for the experiment at 200 °C can be calculated. Equation 6.20 can then expressed with arbitrary parameters by

$$\tilde{G}(s) = H_{res}(s) = \frac{b_0+b_1s}{a_0+a_1s+a_2s^2+a_3s^3} \tag{6.21}$$

for the further mathematical transformations. For easy handling of the numerator dynamics of Equation 6.21, the desired closed-loop transfer function is chosen as

$$\alpha = \left(\frac{Y}{Y_{sp}}\right)_d = \frac{b_0+b_1s}{(\lambda s+1)^2}. \tag{6.22}$$

The second order delay in the denominator of α is utilised to ensure a strictly proper transfer function and its dynamics can be easily adjusted with the constant λ. The required PID controller G_c to achieve the desired closed-loop response is then computed by

$$G_c(s) = \frac{1}{\tilde{G}(s)}\left(\frac{\alpha}{(1-\alpha)}\right), \tag{6.23}$$

in which

$$(1-\alpha) = 1 - \frac{b_0 + b_1 s}{(\lambda s+1)^2} = \frac{(\lambda s+1)^2 - (b_0 + b_1 s)}{(\lambda s+1)^2}. \tag{6.24}$$

A substitution of Equation 6.24 into 6.23 then results in:

$$\begin{aligned}
G_c(s) &= \frac{a_0 + a_1 s + a_2 s^2 + a_3 s^3}{b_0 + b_1 s} \;\; \frac{b_0 + b_1 s}{(\lambda s+1)^2} \;\; \frac{(\lambda s+1)^2}{(\lambda s+1)^2 - (b_0 + b_1 s)} \\
&= \frac{a_0 + a_1 s + a_2 s^2 + a_3 s^3}{(\lambda s+1)^2 - (b_0 + b_1 s)} = \frac{a_0 + a_1 s + a_2 s^2 + a_3 s^3}{\lambda^2 s^2 + (2\lambda - b_1)s + (1 - b_0)} \;\; \frac{\frac{1}{(2\lambda - b_1)}}{\frac{1}{(2\lambda - b_1)}} \\
&= \frac{1}{(2\lambda - b_1)} \frac{a_0 + a_1 s + a_2 s^2 + a_3 s^3}{\frac{\lambda^2 s^2}{(2\lambda - b_1)} + s + (1 - b_0)} \\
&= \frac{1}{(2\lambda - b_1)} \frac{a_0 + a_1 s + a_2 s^2 + a_3 s^3}{s} \;\; \frac{1}{\left(\frac{\lambda^2 s}{(2\lambda - b_1)} + 1 + \frac{1 - b_0}{s}\right)} \\
&= \frac{a_0}{(2\lambda - b_1)} \frac{\left(1 + \frac{a_1}{a_0} s + \frac{a_2}{a_0} s^2 + \frac{a_3}{a_0} s^3\right)}{s} \;\; \frac{1}{\left(\frac{\lambda^2 s}{(2\lambda - b_1)} + 1 + \frac{1 - b_0}{s}\right)} \;\; \frac{\frac{a_1}{a_0}}{\frac{a_1}{a_0}} \\
&= \frac{a_1}{(2\lambda - b_1)} \frac{\left(1 + \frac{a_1}{a_0} s + \frac{a_2}{a_0} s^2 + \frac{a_3}{a_0} s^3\right)}{\frac{a_1}{a_0} s} \;\; \frac{1}{\left(\frac{\lambda^2 s}{(2\lambda - b_1)} + 1 + \frac{1 - b_0}{s}\right)}
\end{aligned} \tag{6.25}$$

The standard transfer function of a PID controller is given by

$$G_{c,PID}(s) = K_p \frac{\tau_i \tau_d s^2 + \tau_i s + 1}{\tau_i s}, \tag{6.26}$$

which, when compared with Equation 6.25, results in

$$K_p = \frac{a_1}{2\lambda - b_1}, \qquad K_i = \frac{K_p}{\tau_i} = K_p \frac{a_0}{a_1}, \qquad K_d = K_p T_d = K_p \frac{a_2}{a_1} \tag{6.27}$$

for the PID controller settings, as derivates higher than two have to be omitted. After variation of λ and simulation of the results, the direct synthesis tuning for the fastest possible achievement of the set specimen temperature of 200 °C is shown in Figure 6.25 and the corresponding parameters in Table 6.12. It can be seen that the identified controller settings lead to an unacceptably long

time to reach the desired steady-state set temperature. In addition, K_d was not utilised, as its value needed to make a noticeable difference in the control loop response would also cause an overshoot exceeding the allowed heating element temperature of 400 °C.

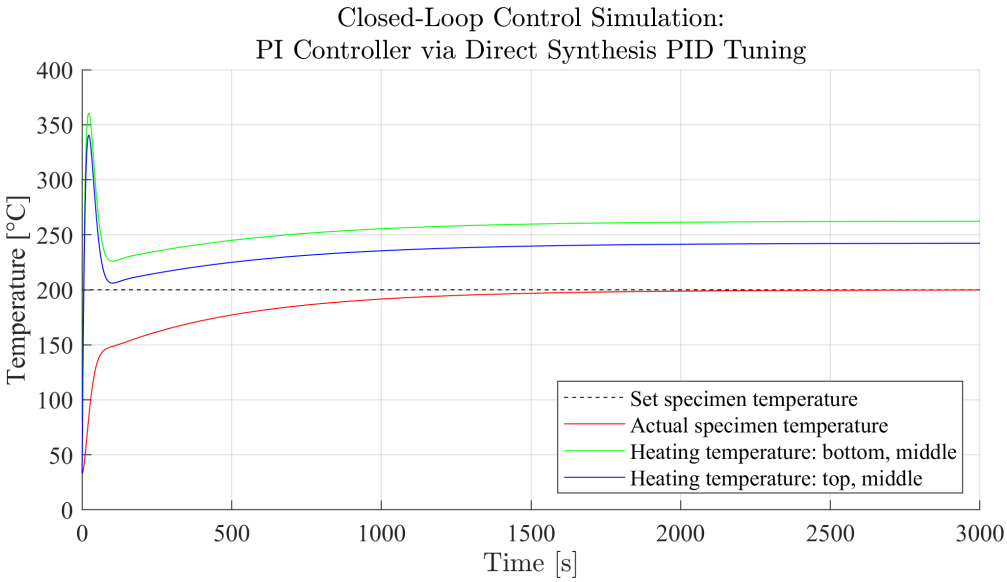


Fig. 6.25: Simulation for $\lambda = 1 \cdot 10^{-4}$ with $K_d = 0$

Table 6.12: Direct synthesis tuning parameters

$\tilde{G}(s) = H_{res}(s) =$	λ	K_p	K_i	K_d
$\frac{0.001013s + 3.904 \cdot 10^{-6}}{s^3 + 0.1131s^2 + 0.002108s + 5.363 \cdot 10^{-6}}$	$1 \cdot 10^{-4}$	2.6350	0.0067	141.38

As the PID settings obtained by the direct synthesis method have proven unsatisfactory, the PID controller is tuned experimentally. The control settings were chosen in a way that the set specimen temperature is reached after 200 s, the specimen temperature stays within a $\pm 2.5\,\%$ of its set value in steady-state and the heating elements do not exceed 400 °C. The resulting settings for the experiments at 100 °C and 200 °C are summed up in Table 6.13. As K_d does not improve the control system response without causing an unacceptable heating temperature overshoot, it was omitted.

Simulation Results utilising PID Control

The closed-loop simulation results for the experiment at 100 °C and 200 °C are shown in Figure 6.26 and 6.27 respectively. It can be seen in Table 6.14 that all specified requirements are fulfilled.

Table 6.13: PID controller settings

Specimen set temperature	Consolidated process model	K_p	K_i	K_d
100 °C	$H_{100}(s)G(s)$	2.33	0.038	0
200 °C	$H_{200}(s)G(s)$	2.00	0.030	0

Although it does not influence the specimen model in the simulation, the temperature curve of the top middle heating element is also shown in Figure 6.26 and 6.27. Its temperature was set to follow that of the bottom middle heating element with a difference of 15 °C and 20 °C in the 100 °C and 200 °C experiment respectively, as that is required for a homogenous specimen temperature and the data is necessary for the generation of the .csv-file.

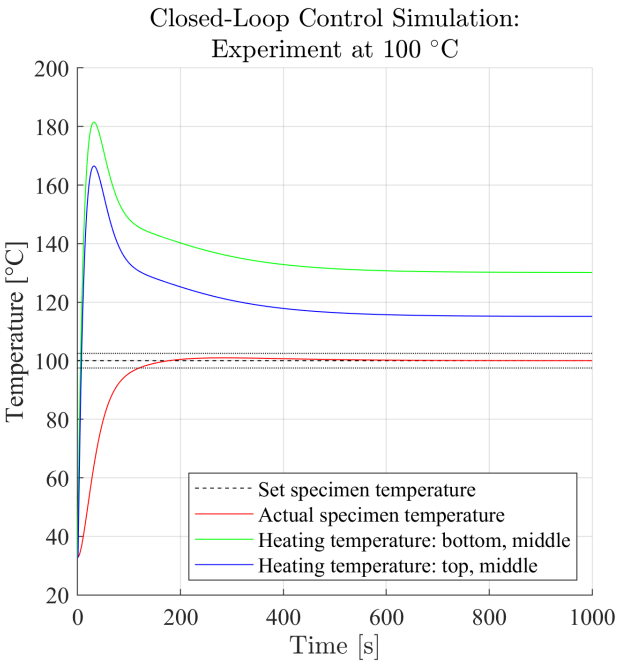


Fig. 6.26: PID control simulation at 100 °C

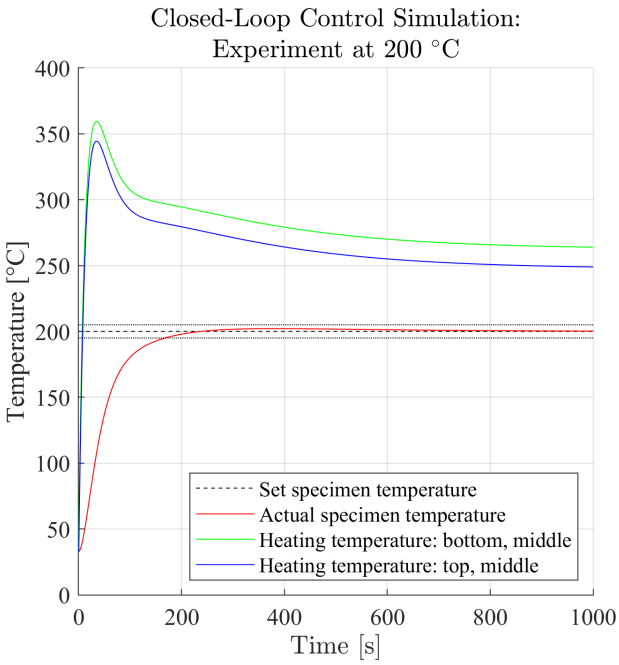


Fig. 6.27: PID control simulation at 200 °C

Table 6.14: Closed-loop simulation characteristics with PID control

Characteristic	Experiment at 100 °C	Experiment at 200 °C
Maximum overshoot e_{max}	0.976 K	2.126 K
Time of max. overshoot T_{max}	284.5 s	382.0 s
Rise time T_r (90%)	72.0 s	99.5 s
Settling time T_ε (97.5%)	120.0 s	168.5 s
Steady-state error $e(t = 1000\,\text{s})$	0.009 K	0.266 K

Simulation Results utilising Model Predictive Control

Although the controller choice resulted in PID control, a closed-loop simulation based on the model predictive control system shown in Figure 6.22 was conducted regardless. The utilised MPC settings are documented in Table 6.15. The bottom middle heating temperature is the manipulated variable, the specimen temperature is the measured output and T_A is the initial system ambient temperature. The specifications for prediction and control horizon differ for the two experiment temperatures because the two different specimen models $H_{100}(s)$ and $H_{200}(s)$ are utilised. The

Table 6.15: MPC specifications

	Set specimen temperature	
	100 °C	200 °C
Prediction horizon P	15	10
Control horizon M	1	2
Manipulated variable weight	0.01	
Manipulated variable upper constraint	$(400\,°\text{C} - T_A)$	
Manipulated variable lower constraint	0	
Measured output weight	1	
Sampling time T_S	0.5 s	

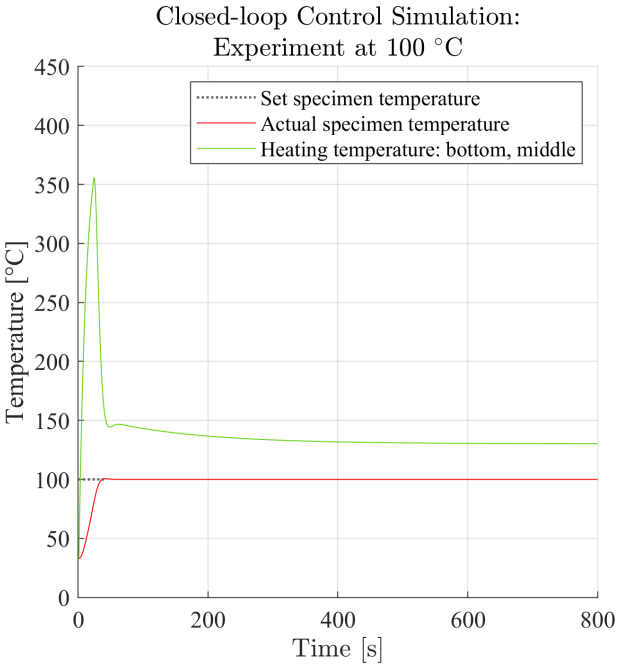


Fig. 6.28: MPC control simulation at 100 °C

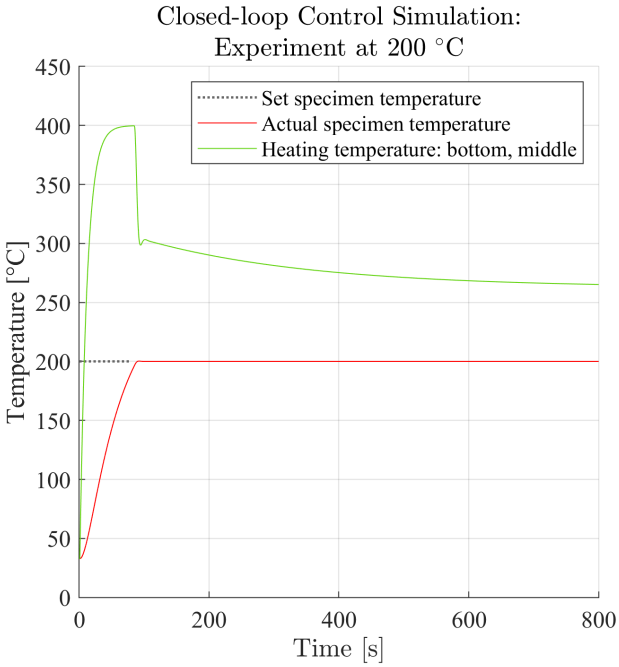


Fig. 6.29: MPC control simulation at 200 °C

Table 6.16: Closed-loop simulation characteristics with model predictive control

Characteristic	Experiment at 100 °C	Experiment at 200 °C
Maximum overshoot e_{max}	0.612 K	0.269 K
Time of max. overshoot T_{max}	42.0 s	92.0 s
Rise time T_r (90%)	28.5 s	72.0 s
Settling time T_ε (97.5%)	33.5 s	84.0 s
Steady-state error $e(t = 1000\,\mathrm{s})$	0 K	0 K

simulation results are depicted in Figure 6.28 and Figure 6.29 for 100 °C and 200 °C respectively and are very promising, as can be deducted from the control characteristics documented in Table 6.16. In comparison to the closed-loop simulation with PID controller, the simulation utilising model predictive control is characterised by a smaller maximum overshoot as well as a steady-state control deviation of 0 K. If the need for extra time to enable manual adjustment of the outer heating elements is ignored, the set specimen temperatures of 100 °C and 200 °C can be reached in 33.5 s and 84.0 s within the temperature tolerance boundaries respectively.

6.2.5 Control System Validation at the Test Bed

The results of the closed-loop simulations for 100 °C and 200 °C were exported as .csv-files and imported to the test bed controller to validate the control system. The resulting measured temperature curves in comparison to the simulation results are shown in Figure 6.30 and 6.32 respectively. Table 6.17 provides all relevant characteristics. It can be seen that the steady-state error criteria is satisfied in both experiments and the measured heating element temperature curves follow the simulated curves almost exactly. However, the transient response of the specimen temperature is

Table 6.17: Open-loop validation experiments characteristics

Characteristic	Experiment at 100 °C	Experiment at 200 °C
Maximum overshoot e_{max}	1.2 K	0.3 K
Time of max. overshoot T_{max}	238.4 s	409.6 s
Rise time T_r (90%)	63.2 s	114.4 s
Settling time T_ε (97.5%)	93.2 s	216.0 s
Steady-state error e	0.5 K / t = 340 s	−0.2 K / t = 560 s

slightly different to the simulation result in both experiments. A possible reason for this deviation aside from model uncertainties and insufficient model accuracy can be found in the manually controlled outer heating elements. In the data used for modelling, the outer heating element temperature is consistently shifted by a fixed value from the controlled heating elements throughout the whole process, as can be seen in Figure 6.31(a) and 6.33(a). This is not the case in the validation experiments, where this constant temperature shift only starts to occur at approximately 70 s for the experiment at 100 °C (Figure 6.31(b)), and at approximately 200 s for 200 °C (Figure 6.31(b)). As the required consistent temperature shift is being achieved in the later stages of the experiment, simulation and measurement results grow more consistent.

The four outer heating elements are functioning as unpredictable disturbances, as their output depends on their simultaneous manual adjustment throughout the whole process and thus, the skill of the operator. It is therefore expected that the outer heating element temperature curves are different each time an experiment is performed. As a result, the transient response of the specimen will likely be different in every experiment. In addition, the four manually controlled outer heating elements also limit the heating up time, as a sensible amount of time is required for the manual adjustments. This is a contradiction to the requirement of heating up as fast as possible.

For a statistical assessment of model parameters, accuracy and the influence of the outer heating element temperature curves, more experiments are required. However, the goal of reaching and holding a set specimen temperature within a defined time range and tolerance boundaries with sensorless open-loop control has been successfully demonstrated by the validation experiments.

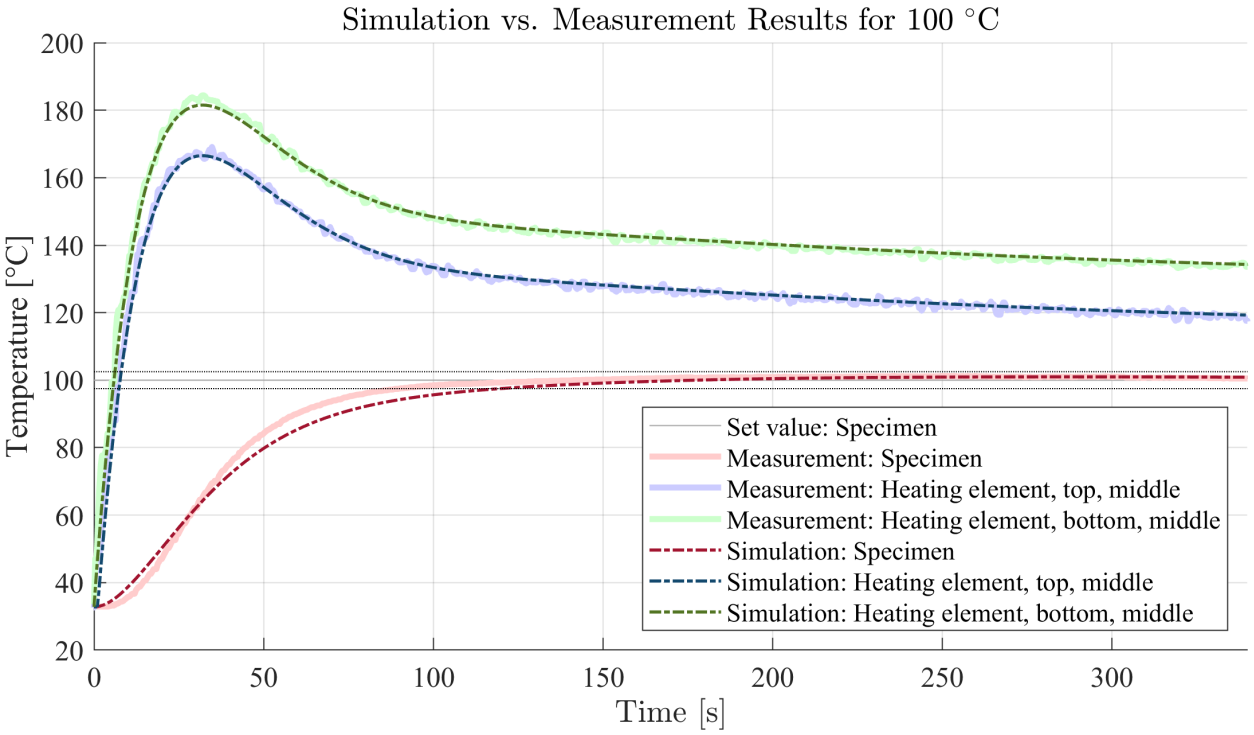
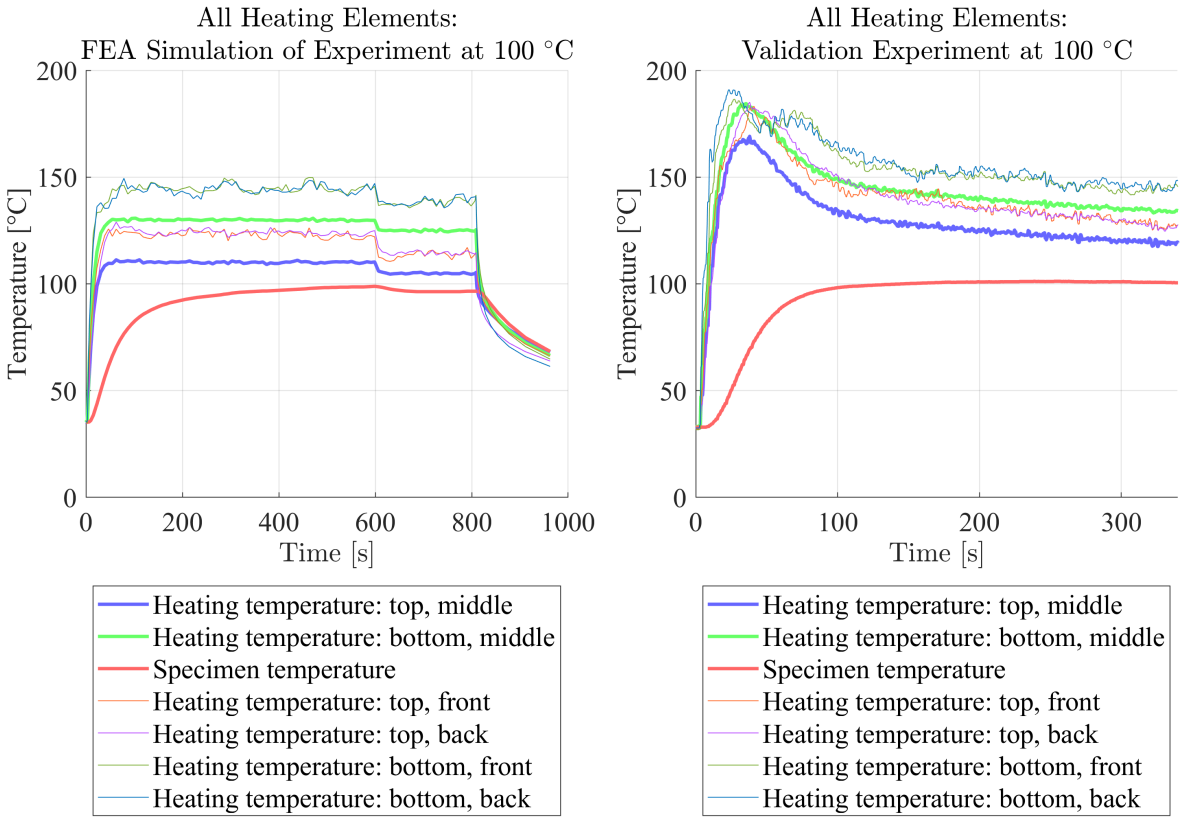


Fig. 6.30: Validation experiment at 100 °C



(a) Data for model identification (b) Validation experiment

Fig. 6.31: Measurement data of all heating elements for experiment at 100 °C

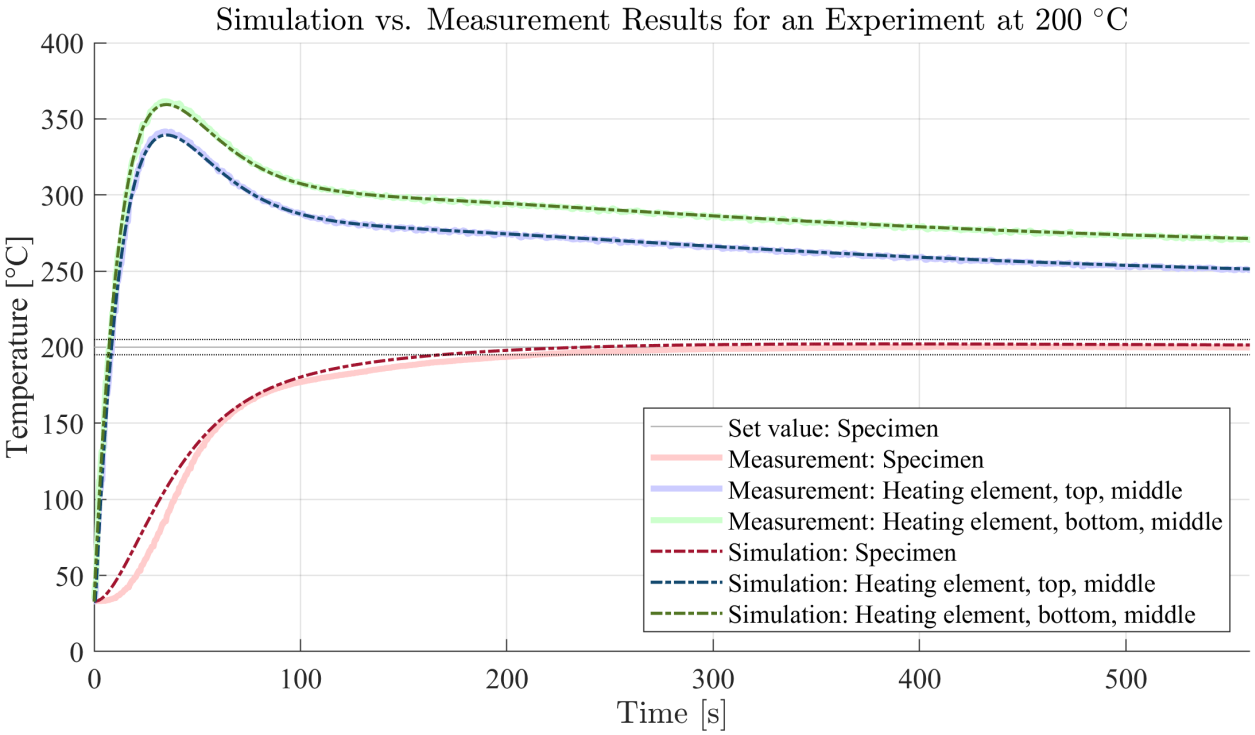
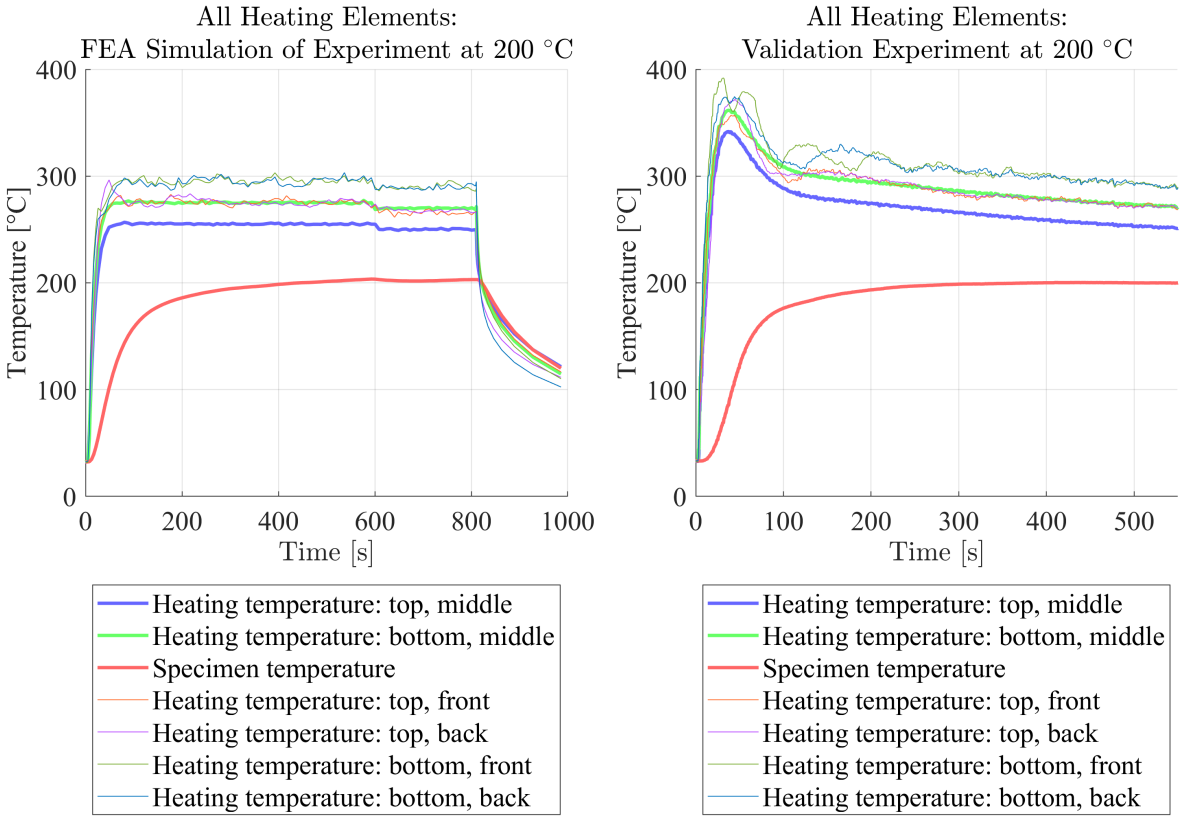


Fig. 6.32: Validation experiment at 200 °C



(a) Data for model identification (b) Validation experiment

Fig. 6.33: Measurement data of all heating elements for experiment at 200 °C

6.3 Closed-Loop Temperature Control Concept

A closed-loop approach for specimen temperature control based on a PID controller cascade system was proposed, but not implemented due to the preference and benefits of the open-loop control system. A cascaded control system utilising PID controllers fits the situation at the test bed well, as the heating elements react extremely fast to changes of the input in comparison to the specimen. Also, since temperature sensors at the heating elements are available and a model is created for the specimen, both feedback variables required for a control cascade are accessible by default.

Using the time-shift properties of the z-transform, a closed-loop temperature control can be realised by transforming the specimen model into discrete time form. The discrete time transfer function can then be executed as code on the *Raspberry Pi*™ installed at the test bed. The control structure proposed for the closed-loop temperature control utilises the internal controller of the test bed and is shown in Figure 6.34.

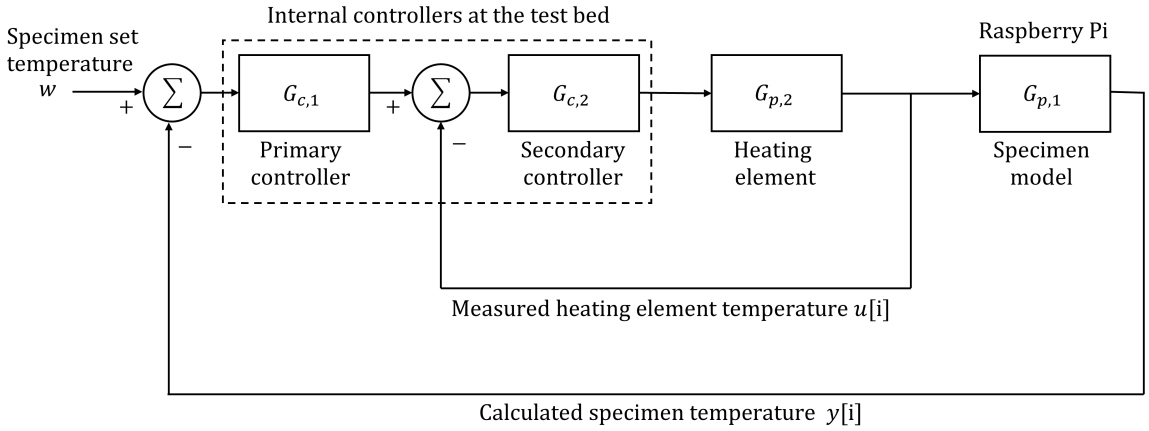


Fig. 6.34: Closed-loop control concept

Converting a transfer function to code is done exemplary for the model used for experiments at 200 °C by subjecting it to the Tustin approximation of Equation 3.14 with a sampling time $T_s = 0.5\,\mathrm{s}$. Its corresponding approximate discrete time form

$$H(z) = \frac{0.002787z^2 + 5.367 \cdot 10^{-6}z - 0.00782}{z^2 - 1.989z + 0.9887} \tag{6.28}$$

is the result. Dividing this discrete time transfer function by the highest power of z yields

$$H(z) = \frac{0.002787 + 5.367 \cdot 10^{-6}z^{-1} - 0.00782z^{-2}}{1 - 1.989z^{-1} + 0.9887z^{-2}} = \frac{\mathrm{Out}}{\mathrm{In}}, \tag{6.29}$$

in which all terms are available in the form of the backwards time shift z^{-j} for $j = 0,1,2,\ldots,n$. This result can then be implemented in *Python*™ code by computing the output `y[i]` for any input `u[i]`, whereas the time shift z^{-j} refers to the value of the input `u[i-j]` or output `u[i-j]` at a previous iteration according to the respective backward shift. The code is shown in Listing 6.3. This was also validated using a Message Queue Telemetry Transport (MQTT) connection between a PC acting as the controller and a *Raspberry Pi*™ functioning as the specimen model.

Listing 6.3: Discrete time model in form of *Python*™ code

```
1     ...
2     if i == 0:
3         ut_1 = 0;
4         yt_1 = 0;
5         ut_2 = 0;
6         yt_2 = 0;
7     elif i == 1:
8         ut_2 = 0;
9         yt_2 = 0;
10        ut_1 = u[i - 1]
11        yt_1 = y[i - 1]
12    else:
13        ut_1 = u[i - 1]
14        yt_1 = y[i - 1]
15        ut_2 = u[i - 2]
16        yt_2 = y[i - 2]
17
18    y[i] = 1.989*yt_1 - 0.9887*yt_2 + 0.002787*u[i] + ...
          5.367*(10^-6)*ut_1 - 0.00782*ut_2
19    ...
```

Chapter 7

Conclusions and Future Work

A sensorless temperature open-loop control system was successfully developed and implemented at the Equal Channel Angular Pressing (ECAP) test bed at the Chair of Metal Forming at the Montanuniversität Leoben. Models for the specimen and heating elements were identified successfully using the global least squares system identification method and it was shown that low-order linear models are sufficient for a reasonably accurate specimen temperature prediction. The use of the global least squares system identification method was proven suitable to estimate parameters for models in state-space representation for both input-output data with uniform as well as non-uniform sampling intervals. The developed open-loop control system utilising a PID controller is designed to satisfy accurate steady-set specimen temperature characteristics as well as suitable transient response required for the test bed operator. This was confirmed by simulation in *Simulink*™ and experimentally at the test bed. As a result, the heating-up time on the test bed was reduced to 30 % of the time required in operation without the control system. A final, constant set point deviation of less than 1 % was also made possible by the control, whereas in operation without the control system the specimen temperature could only be kept constant for and much larger set point deviations were to be expected.

For the Equal Channel Angular Pressing test bed, two considerations for future modifications are given. First, the utilisation of model predictive control should be further evaluated. A MPC system was designed in the process of choosing a suitable regulator for the closed-loop control simulation in Section 6.2.3, due to its benefits of handling process and actuator constraints. The simulation results look very promising due to their extremely fast settling time, minimal overshoot and steady-state error of 0 K, but have not been validated at the test bed. Second, it was found that the manual adjustment of the outer heating elements increases the time required for the heating up process. While it is possible to reach a specimen temperature of 200 °C within 100 s with the right controller settings, this was limited to 200 s in course of this thesis to provide the test bed operator with enough time to follow the given temperature trajectory manually with the outer heating elements. Retrofitting a connection to the controller should therefore be considered, to automate and speed

up the process further, augment specimen temperature homogeneity and eliminate the disturbances resulting from the manual control of the outer heating elements.

In conclusion, the scientific contributions and findings of this thesis are:

- The global least squares system identification method was proven viable for the direct estimation of continuous time model parameters based on available non-uniformly sampled input-output data. It achieved sufficient results in comparison to uniformly sampled data even with only 5 % of the amount of data points of the latter available.
- Sensorless temperature control in metal forming was proven feasible in the case of the Equal Channel Angular Pressing test bed by utilising a linear, low order dynamic model acting as a virtual sensor for the specimen temperature in combination with an open-loop control system.
- For specific model quality requirements as in the case of the specimen model, which should provide high steady-state accuracy while the dynamic response plays a secondary role, it was found that a visual assessment provides information that some error criteria cannot. While squared, average and maximum error give a good idea of the general model fit, they do not provide any locational information of the error along the desired trajectory. A visual comparison should therefore be used to complement those criteria, if the occurring error is subject to different prioritisation depending on the section of the trajectory.

Finally, two major research questions for future work built on the findings in this thesis are raised:

- The issue of how model uncertainties translate between the measurement data, FEA simulation and mathematical model and therefore, how well a dynamic model identified from FEA simulation data represents the real process should be addressed in the future, to further explore the viability of FEA simulation data for virtual sensor creation.
- An investigation into metrics or characteristics of non-uniformly and uniformly sampled input-output data should be conducted with the goal of assessing and comparing their suitability for the purpose of modelling and the resulting model quality.

List of Figures

List of Tables

List of Abbreviations

A/D	Analogue-to-Digital
D/A	Digital-to-Analogue
CPS	Cyberphysical System
DTR	Determinant Ratio Test
DV	Disturbance Variable
ECAP	Equal Channel Angular Pressing
FEA	Finite Element Analysis
IoT	Internet of Things
LTI	Linear Time-Invariant
LS	Least Squares
MIMO	Multiple-Input Multiple-Output
MISO	Multiple-Input Single-Output
MO	Measured Output
MPC	Model Predictive Control
MQTT	Message Queue Telemetry Transport
MV	Manipulated Variable
MQTT	Message Queue Telemetry Transport
PID	Proportional Integral Derivative
PLC	Programmable Logic Controller
SISO	Single-Input Single-Output
UFG	Ultra Fine Grained
ZOH	Zero-Order Hold

List of Code Listings

References

[1] D. Martin, N. Kühl, and G. Satzger. “Virtual Sensors”. In: *Business & Information Systems Engineering* 63.3 (2021), pp. 315–323. ISSN: 2363-7005.

[2] M. Kano and K. Fujiwara. “Virtual Sensing Technology in Process Industries: Trends and Challenges Revealed by Recent Industrial Applications”. In: *Journal of Chemical Engineering of Japan* 46.1 (2013), pp. 1–17.

[3] S. Junseo and K. Hyogon. “Sensorless air flow control in an HVAC system through deep learning”. In: *Applied Sciences (Switzerland)* 9.16 (2019). ISSN: 2076-3417.

[4] X. Wang, R. Luo, and H. Shao. “Designing a soft sensor for a distillation column with the fuzzy distributed radial basis function neural network”. In: *Proceedings of 35th IEEE Conference on Decision and Control*. IEEE, 11-13 Dec. 1996, pp. 1714–1719. ISBN: 0-7803-3590-2.

[5] C. Abeykoon. “Design and Applications of Soft Sensors in Polymer Processing: A Review”. In: *IEEE Sensors Journal* 19.8 (2019), pp. 2801–2813. ISSN: 1530-437X.

[6] A. A. Hussein. “A Sensorless Surface Temperature Measurement Method for Batteries Using Artificial Neural Networks”. In: *2018 IEEE Energy Conversion Congress and Exposition (ECCE)*. IEEE, 23.09.2018 - 27.09.2018, pp. 3560–3564. ISBN: 978-1-4799-7312-5.

[7] C. H. Guzmán. “Implementation of Virtual Sensors for Monitoring Temperature in Greenhouses Using CFD and Control”. In: *Sensors (Basel, Switzerland)* 19.1 (2018).

[8] C. Bohn and H. Unbehauen. *Identifikation dynamischer Systeme: Methoden zur experimentellen Modellbildung aus Messdaten*. Wiesbaden: Springer Vieweg, 2016. ISBN: 978-3-8348-1755-6.

[9] X. Wang, S. Li, and Y. Xi. “Soft sensor modeling for slab temperature estimation”. In: *The 12th IEEE International Conference on Fuzzy Systems, 2003. FUZZ ’03*. IEEE, 25-28 May 2003, pp. 1452–1454. ISBN: 0-7803-7810-5.

[10] R. Schwarz et al. “Sensorless battery cell temperature estimation circuit for enhanced safety in battery systems”. In: *IECON 2015 - 41st Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 9.11.2015 - 12.11.2015, pp. 1536–1541. ISBN: 978-1-4799-1762-4.

[11] H. Morimitsu and S. Katsura. “Sensorless Control of Heat Inflow to a Thermal Display Using a Heat Inflow Observer”. In: *IEEE Transactions on Industrial Electronics* 62.7 (2015), pp. 4288–4297.

[12] D. Xu et al. “A review of sensorless control methods for AC motor drives”. In: *CES Transactions on Electrical Machines and Systems* 2.1 (2018), pp. 104–115.

[13] M. Benadja and A. Chandra. “Sensorless control for wind energy conversion system (WECS) with power quality improvement”. In: *2014 IEEE PES General Meeting — Conference & Exposition*. IEEE, 27.07.2014 - 31.07.2014, pp. 1–5. ISBN: 978-1-4799-6415-4.

[14] R. Rocha. “A sensorless control for a variable speed wind turbine operating at partial load”. In: *Renewable Energy* 36.1 (2011), pp. 132–141. ISSN: 09601481.

[15] M. J. Grimble et al. *Identification of Continuous-time Models from Sampled Data*. London: Springer London, 2008.

[16] T. Beyrouthy, L. Fesquet, and R. Rolland. “Data sampling and processing: Uniform vs. non-uniform schemes”. In: *2015 International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*. IEEE, 17.06.2015 - 19.06.2015, pp. 1–6. ISBN: 978-1-4673-7888-8.

[17] H. Raghavn. “Identification of chemical processes with irregular output sampling”. In: *Control Engineering Practice* 14.5 (2006), pp. 467–480. ISSN: 09670661.

[18] E. K. Larsson and T. Söderström. “Identification of continuous-time AR processes from unevenly sampled data”. In: *Automatica* 38.4 (2002), pp. 709–718. ISSN: 0005-1098.

[19] R Johansson. “Continuous-Time Model Identification and State Estimation Using Non-Uniformly Sampled Data”. In: *IFAC Proceedings Volumes* 42.10 (2009), pp. 1163–1168. ISSN: 14746670.

[20] J. I. Yuz et al. “Identification of continuous-time state-space models from non-uniform fast-sampled data”. In: *IET Control Theory & Applications* 5.7 (2011), pp. 842–855. ISSN: 1751-8644.

[21] F. Ding, L. Qiu, and T. Chen. “Reconstruction of continuous-time systems from their non-uniformly sampled discrete-time systems”. In: *Automatica* 45.2 (2009), pp. 324–332. ISSN: 0005-1098.

[22] K. Bekiroglu et al. “System Identification Algorithm for Non-Uniformly Sampled Data”. In: *Proceedings of the IFAC World Congress. International Federation of Automatic Control. World Congress* 50.1 (2017), pp. 7296–7301.

[23] Y. Liu, L. Xie, and F. Ding. “An auxiliary model based on a recursive least-squares parameter estimation algorithm for non-uniformly sampled multirate systems”. In: *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering* 223.4 (2009), pp. 445–454. ISSN: 0959-6518.

[24] E. Huselstein and H. Garnier. “An approach to continuous-time model identification from non-uniformly sampled data”. In: *Proceedings of the 41st IEEE Conference on Decision and Control, 2002*. IEEE, 10-13 Dec. 2002, pp. 622–623. ISBN: 0-7803-7516-5.

[25] C. M. Close, D. K. Frederick, and J. C. Newell. *Modeling and analysis of dynamic systems*. 3rd ed. Hoboken, NJ: Wiley, 2002. ISBN: 0471394424.

[26] C. A. Kluever. *Dynamic systems: modeling, simulation, and control*. Hoboken, NJ: John Wiley and Sons Inc, 2015.

[27] R. Isermann and M. Münchhof. *Identification of Dynamic Systems: An Introduction with Applications*. Advanced Textbooks in Control and Signal Processing. Berlin, Heidelberg: Springer-Verlag Berlin Heidelberg, 2011.

[28] T. H. Schmitt. *Methoden zur Modellbildung und Simulation mechatronischer Systeme*. Kaufmann, 2019.

[29] J. Adamy. *Nichtlineare Systeme und Regelungen*. 3., aktualisierte Auflage. Berlin: Springer Vieweg, 2018. ISBN: 978-3-662-55684-9.

[30] H. Unbehauen. *Regelungstechnik I: Klassische Verfahren zur Analyse und Synthese linearer kontinuierlicher Regelsysteme, Fuzzy-Regelsysteme*. 15., überarbeitete und erweiterte Auflage. Wiesbaden: Vieweg+Teubner Verlag / GWV Fachverlage GmbH Wiesbaden, 2008.

[31] L. Ljung. *System identification: Theory for the user*. 2. ed., 14. printing. Prentice Hall information and system sciences series. Upper Saddle River, NJ: Prentice Hall PTR, 2012.

[32] L. Ljung and T. Glad. *Modeling of dynamic systems*. Englewood Cliffs, N.J: PTR Prentice Hall, 1994.

[33] H. Unbehauen. *Regelungstechnik II: Zustandsregelungen, digitale und nichtlineare Regelsysteme*. 9., durchges. und korrigierte Aufl., korrig. Nachdr. Vol. / Heinz Unbehauen ; 2. Automatisierungstechnik. Wiesbaden: Vieweg+Teubner, 2009.

[34] D. E. Seborg, T. F. Edgar, and D. A. Mellichamp. *Process dynamics and control*. 3rd ed. Hoboken, N.J.: Wiley, 2011.

[35] J. R. Ragazzini and G. F. Franklin. *Sampled-data Control Systems*. McGraw-Hill series in control systems engineering. McGraw-Hill, 1958.

[36] R. Isermann. *Digital Control Systems: Volume 1: Fundamentals, Deterministic Control*. Second, Revised Edition. Berlin, Heidelberg and s.l.: Springer Berlin Heidelberg, 1989.

[37] B. P. Lathi. *Linear systems and signals*. 2. ed. The Oxford series in electrical and computer engineering. New York: Oxford Univ. Press, 2005.

[38] J. Corriou. *Process control: Theory and applications*. Second edition. Cham: Springer, 2018.

[39] J. Lunze. *Regelungstechnik 2: Mehrgrößensysteme, Digitale Regelung*. 9., überarb. Aufl. 2016. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016.

[40] F. L. Lewis. *Control Systems, Lecture 3*. Ed. by The University of Texas Arlington. 2017. URL: https://lewisgroup.uta.edu/.

[41] G. F. Franklin, J. D. Powell, and M. L. Workman. *Digital control of dynamic systems*. 3rd ed. ; reprinted in 2006 with corrections. Half Moon Bay, CA: Ellis-Kagle Press, 2006.

[42] O. Nelles. *Zustandsraum und digitale Regelung*. Ed. by Universität Siegen. 2018. URL: https://www.mb.uni-siegen.de/mrt/lehre/dr/skript_dr.pdf.

[43] W. Liu. *Introduction to Hybrid Vehicle System Modeling and Control*. 1. Aufl. Wiley, 2013.

[44] H. Thal-Larsen. "Frequency response from experimental nonoscillatory transient-response data". In: *Transactions of the American Institute of Electrical Engineers, Part II: Applications and Industry* 74.2 (1955), pp. 109–114. ISSN: 0097-2185.

[45] L. Ljung. "Black-box models from input-output measurements". In: *IMTC 2001. Proceedings of the 18th IEEE Instrumentation and Measurement Technology Conference. Redis-*

covering Measurement in the Age of Informatics (Cat. No.01CH 37188). IEEE, 21-23 May 2001, pp. 138–146. ISBN: 0-7803-6646-8.

[46] C. M. Woodside. "Estimation of the order of linear systems". In: *Automatica* 7.6 (1971), pp. 727–733. ISSN: 0005-1098.

[47] H. Unbehauen. *Regelungstechnik III: Identifikation, Adaption, Optimierung*. 6., verbesserte Auflage. Studium Technik. Wiesbaden: Vieweg+Teubner Verlag, 2000.

[48] J. Lunze. *Regelungstechnik 1: Systemtheoretische Grundlagen, Analyse und Entwurf einschleifiger Regelungen : mit 425 Abbildungen, 76 Beispielen, 179 Übungsaufgaben sowie einer Einführung in das Programmsystem MATLAB*. 11., überarbeitete und ergänzte Auflage. Lehrbuch. Berlin: Springer, 2016.

[49] M. Harker and G. Rath. "Global least squares for time-domain system identification of state-space models". In: *2018 7th Mediterranean Conference on Embedded Computing (MECO)*. IEEE, 10.06.2018 - 14.06.2018, pp. 1–6. ISBN: 978-1-5386-5683-9.

[50] P. Mathieu and M. Mohammed. "An interpretation of subsace identification methods". In: *IFAC Proceedings Volumes* 38.1 (2005), pp. 904–909. ISSN: 14746670.

[51] J. V. Candy. *Model-based signal processing*. Adaptive and learning systems for signal processing, communications, and control. Hoboken, N.J: IEEE Press, 2010.

[52] K. J. Åström. *Lectures on the Identification Problem: The Least Squares Method*. Research Reports TFRT-3004. Department of Automatic Control, Lund Institute of Technology (LTH), 1968.

[53] P. O'Leary and M. Harker. "Discrete-Time Variable-Projection Method for Fractional Order System Identification". In: *International Conference on Fractional Differentiation and its Applications,* (2016).

[54] N. S. Nise. *Control systems engineering*. Seventh Edition. Hoboken: Wiley, 2015.

[55] W. Plaßmann and D. Schulz, eds. *Handbuch Elektrotechnik: Grundlagen und Anwendungen für Elektrotechniker*. 7., neu bearbeitete Auflage. Wiesbaden: Springer Vieweg, 2016.

[56] W. Bolton. *Instrumentation and control systems*. 2. ed. Amsterdam: Newnes/Elsevier, 2015.

[57] A. Böge and W. Böge, eds. *Handbuch Maschinenbau: Grundlagen und Anwendungen der Maschinenbau-Technik*. 23., überarbeitete Auflage. Wiesbaden: Springer Vieweg, 2017.

[58] R. Kumar, S. K. Singla, and V. Chopra. "Comparison among some well known control schemes with different tuning methods". In: *Journal of Applied Research and Technology* 13.3 (2015), pp. 409–415. ISSN: 16656423.

[59] J. G. Ziegler and N. B. Nichols. "Optimum Settings for Automatic Controllers". In: *Journal of Dynamic Systems, Measurement, and Control* 115.2B (1942), pp. 220–222. ISSN: 0022-0434.

[60] K. Ogata. *Modern Control Engineering*. 5th ed. 2010.

[61] B. W. Bequette. *Process control: Modeling, design, and simulation*. 9th print. Prentice-Hall international series in the physical and chemical engineering sciences. Upper Saddle River, NJ: Prentice Hall PTR, 2010. ISBN: 0-13-353640-8.

[62] A. C. Dimian, C. S. Bildea, and A. A. Kiss. "Plantwide Control". In: *Integrated Design and Simulation of Chemical Processes*. Vol. 35. Computer Aided Chemical Engineering. Elsevier, 2014, pp. 599–647. ISBN: 9780444627001.

[63] R. Isermann. *Digital Control Systems: Volume 2: Stochastic Control, Multivariable Control, Adaptive Control, Applications*. Second, revised edition. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991.

[64] J. Qin and T. Badgwell. "An Overview Of Industrial Model Predictive Control Technology". In: *AIChE Symposium Series* 93 (1997).

[65] J. Richalet. "Industrial applications of model based predictive control". In: *Automatica* 29.5 (1993), pp. 1251–1274. ISSN: 0005-1098.

[66] E. F. Camacho and C. Bordons. *Model Predictive control*. Second Edition. Advanced Textbooks in Control and Signal Processing. London: Springer, 2007.

[67] P. O'Leary and M. Harker. *ODE Solvers: The Initial-, Boundary-, and Interior-Value Problem Solvers, odeSolve and odeSolveResBound*. 2014.

[68] J. E. Gentle. *Matrix Algebra: Theory, Computations and Applications in Statistics*. Second edition. Springer Texts in Statistics. Cham: Springer, 2017.

[69] G. H. Golub and C. F. van Loan. *Matrix computations*. 4. ed. Johns Hopkins studies in mathematical sciences. Baltimore, Md.: Johns Hopkins Univ. Press, 2013. ISBN: 978-1-4214-0794-4.

[70] T. Arens et al. *Mathematik*. 4. Aufl. 2018. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018. ISBN: 978-3-662-56740-1.

[71] M. Harker. *Numerical Analysis in Mechatronics: Lecture Notes*. 2019.

[72] P. O'Leary and M. Harker. "Regularized Reconstruction of a Surface from its Measured Gradient Field". In: *Journal of Mathematical Imaging and Vision* 51.1 (2015), pp. 46–70.

[73] J. Pillow. *Lecture Notes to Singular Value Decomposition: Statistical Modeling and Analysis of Neural Data*. Ed. by Princeton University. 2018. URL: `http://pillowlab.princeton.edu/`.

Appendix A

Mathematical Preliminaries

This appendix describes some fundamental mathematical concepts applied in this thesis as well as any needed uncommon operations.

A.1 Kronecker Product

Kronecker multiplication is denoted by the symbol $\otimes$ and defined by

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1m}B \\ \vdots & \ddots & \vdots \\ a_{n1}B & \dots & a_{nm}B \end{bmatrix} \tag{A.1}$$

for any two matrices $A_{n\times m}$ and $B_{p\times q}$ [68, p. 95]. If $A \in \mathbb{R}^{mn}$ and $B \in \mathbb{R}^{pq}$, then their respective Kronecker product $A \otimes B$ is a $m \times n$ block matrix, whose (i, j) block is a $p \times q$ matrix $a_{ij}B$. [69, p. 27] An example is provided by [69, p. 27]:

$$C = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \otimes \begin{bmatrix} b_{11} & b_{12} & b_{13} \\ b_{21} & b_{22} & b_{23} \\ b_{31} & b_{32} & b_{33} \end{bmatrix} \tag{A.2}$$

$$C = \left[\begin{array}{ccc|ccc} a_{11}b_{11} & a_{11}b_{12} & a_{11}b_{13} & a_{12}b_{11} & a_{12}b_{12} & a_{12}b_{13} \\ a_{11}b_{21} & a_{11}b_{22} & a_{11}b_{23} & a_{12}b_{21} & a_{12}b_{22} & a_{12}b_{23} \\ a_{11}b_{31} & a_{11}b_{32} & a_{11}b_{33} & a_{12}b_{31} & a_{12}b_{32} & a_{12}b_{33} \\ \hline a_{21}b_{11} & a_{21}b_{12} & a_{21}b_{13} & a_{22}b_{11} & a_{22}b_{12} & a_{22}b_{13} \\ a_{21}b_{21} & a_{21}b_{22} & a_{21}b_{23} & a_{22}b_{21} & a_{22}b_{22} & a_{22}b_{23} \\ a_{21}b_{31} & a_{21}b_{32} & a_{21}b_{33} & a_{22}b_{31} & a_{22}b_{32} & a_{22}b_{33} \\ \hline a_{31}b_{11} & a_{31}b_{12} & a_{31}b_{13} & a_{32}b_{11} & a_{32}b_{12} & a_{32}b_{13} \\ a_{31}b_{21} & a_{31}b_{22} & a_{31}b_{23} & a_{32}b_{21} & a_{32}b_{22} & a_{32}b_{23} \\ a_{31}b_{31} & a_{31}b_{32} & a_{31}b_{33} & a_{32}b_{31} & a_{32}b_{32} & a_{32}b_{33} \end{array}\right] \tag{A.3}$$

A.2 Numerical Differentiation

With a sequence of n points of function values, the computation of a Lagrange interpolating polynomial is possible as

$$g(x) = L_1(x)f(x_1) + L_2(x)f(x_2) + \ldots + L_n(x)f(x_n) \tag{A.4}$$

in which the Lagrange polynomials L_j are defined by [70, p. 116, 71, p. 17]

$$L_j(x) = \frac{(x-x_1)\ldots(x-x_{j-1})(x-x_{j+1})\ldots(x-x_n)}{(x_j-x_1)\ldots(x_j-x_{j-1})(x_j-x_{j+1})\ldots(x_j-x_n)}. \tag{A.5}$$

For the goal of achieving a practical method for numerical differentiation, approximations to the derivatives at given nodes, that means approximations for $f'(x_k)$ for $k = 1,\ldots,n$ can be made by evaluating the derivate of the Lagrange polynomials at given points x_k [71, p. 17]:

$$g'(x_k) = L_1'(x_k)f(x_1) + L_2'(x_k)f(x_2) + \ldots + L_n'(x_k)f(x_n) \tag{A.6}$$

Evaluating at each point yields the matrix form

$$\begin{bmatrix} g'(x_1) \\ g'(x_2) \\ \vdots \\ g'(x_n) \end{bmatrix} = \begin{bmatrix} L_1'(x_1) & L_2'(x_1) & \ldots & L_n'(x_1) \\ L_1'(x_2) & L_2'(x_2) & \ldots & L_n'(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ L_1'(x_n) & L_2'(x_n) & \ldots & L_n'(x_n) \end{bmatrix} \begin{bmatrix} f(x_1) \\ f(x_2) \\ \vdots \\ f(x_n) \end{bmatrix} \tag{A.7}$$

and consequently, the numerical differentiation matrix

$$\boldsymbol{D} = \begin{bmatrix} L_1'(x_1) & L_2'(x_1) & \ldots & L_n'(x_1) \\ L_1'(x_2) & L_2'(x_2) & \ldots & L_n'(x_2) \\ \vdots & \vdots & \ddots & \vdots \\ L_1'(x_n) & L_2'(x_n) & \ldots & L_n'(x_n) \end{bmatrix}. \tag{A.8}$$

as a result [71, p. 17]. The concept of this matrix based numerical differentiation introduced in [72] makes it possible to represent and compute the numerical differentiation of a discrete function $y = f(x)$ by [72, p. 3]

$$\boldsymbol{g'} = \boldsymbol{D}\,\boldsymbol{y}. \tag{A.9}$$

A.3 Singular Value Decomposition

The Singular Value Decomposition is a procedure for numerically assessing matrix properties. Three particularly important are the range, null space and rank, especially in the case of discretising linear operators [71, p. 31]. An $n \times m$ matrix $\boldsymbol{A}$ can be factored as [68, p. 161]

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{S}\boldsymbol{V}^T, \tag{A.10}$$

in which the variables are: [68, pp. 161–162]

- $\boldsymbol{U}$ is an $n \times n$ orthogonal matrix,
- $\boldsymbol{V}$ is an $m \times m$ orthogonal matrix,
- and $\boldsymbol{S}$ is an $n \times m$ matrix with non-negative entries.

The elements on the diagonal of $\boldsymbol{S}$ are the singular values of A [68, p. 162, 73, p. 2]:

$$\boldsymbol{S} = \begin{bmatrix} S_1 & & & \\ & S_2 & & \\ & & \ddots & \\ & & & S_n \end{bmatrix} \tag{A.11}$$

Inverses

The Singular Value Decomposition provides an easy method the compute the inverse of a matrix. As $\boldsymbol{U}$ and $\boldsymbol{V}$ are orthogonal, their transposes are their inverses,

$$\boldsymbol{U}^T\boldsymbol{U} = \boldsymbol{U}\boldsymbol{U}^T = \boldsymbol{I} \tag{A.12}$$

and

$$\boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{V}\boldsymbol{V}^T = \boldsymbol{I}. \tag{A.13}$$

If the inverse of a matrix $\boldsymbol{A}$ exists, it can then be computed easily by Singular Value Decomposition as [73, p. 2]

$$\boldsymbol{A}^{-1} = \boldsymbol{V}\boldsymbol{S}^{-1}\boldsymbol{U}^T. \tag{A.14}$$