INAUGURAL-DISSERTATION

zur

Erlangung der Doktorwürde

der

Naturwissenschaftlich-Mathematischen Gesamtfakultät

der

RUPRECHT-KARLS-UNIVERSITÄT
HEIDELBERG

vorgelegt von

Philipp Glaser, M.Sc.

aus Ulm

Tag der mündlichen Prüfung:

Uncertainty Quantification for Complex Engineering Systems

Betreuer: Professor Dr. Vincent Heuveline

Zweitbetreuer: Professor Dr. Carsten Proppe

Acknowledgements

During the development of this work, I received countless support from many people. The actions, conversations, ideas, and advice have influenced this document and my life. I am very grateful for that. First of all, I wish to express my sincere gratitude to my supervisor Prof. Dr. Vincent Heuveline. Through his open and friendly nature, he has created an extremely fruitful environment at the Faculty of Mathematics and Computer Science at the Heidelberg University. The many impulses, discussions, and good advice have significantly enhanced my work and skills. I would also like to thank my second supervisor Prof. Dr.-Ing. Carsten Proppe for his interest in my work and the valuable comments that have helped to improve this dissertation. My thanks are also extended to Dr.-Ing. Kosmas Petridis from the Robert Bosch GmbH for his trust and motivation. I acknowledge his time in our frequent exchanges as well as the support beyond that, which could not have been more valuable. I want to thank Robert Bosch GmbH for supporting this research and providing a great working environment. Both at Bosch and the Institute Engineering Mathematics and Computing Lab in Heidelberg, I had the pleasure of working with many wonderful colleagues who have supported me in many different ways. A special thank goes to Joachim Güttinger for providing the lead application model and the always good cooperation, and to Andreas Zehender, Manfred Kaul and Dieter Hagelocher for their support in setting up the test bench. My sincerest thanks go to my friends who accompanied me on my way. Their continued support and their encouragement made this work possible. Even if some individual persons are not named here, I would like to thank you very much for your kindness towards me. Above all, I would like to thank my family for their unconditional love.

Abstract

Many industrial applications include model parameters for which precise values are hardly available. To better characterize these parameters, deterministic values are replaced by stochastic variables. These can be regarded as parameter uncertainties and potentially have a significant influence on the simulation results. The quantification of such uncertainties plays a crucial role, e.g., for unknown component tolerances or measurement errors. One of the challenges is to gain knowledge about the parameter distribution from experimental data. In this context, Bayesian inference offers an approach to combine numerical simulations with experimental data to obtain a better knowledge of the uncertainties. Many standard methods require a large amount of evaluations to achieve high numerical accuracy. This is a significant drawback, especially when the cost of a single forward simulation is very high. Meta models, such as Polynomial Chaos (PC) extensions, can significantly reduce the number of required evaluations. To validate the described methods and algorithms, in reality, a test bench was developed in the present work, with which a motor characteristic of an electric machine with uncertain physical parameters can be measured. With this test bench, it is possible to define physical reference parameters and to record a corresponding set of measurements. The focus is on the validation of the methods based on real measurements from an industrial application. The numerical results show that the PC approach can significantly reduce the required computing time compared to the original simulation model and thus make the method applicable in practice.

Zusammenfassung

Viele industrielle Anwendungen beinhalten Modellparameter, für die selten präzise Werte vorliegen. Um eine bessere Charakterisierung dieser Parameter zu erhalten, werden die deterministischen Werte durch stochastische Variablen ersetzt. Diese können als Parameterunsicherheiten betrachtet werden und haben potenziell einen signifikanten Einfluss auf die Simulationsergebnisse. Die Quantifizierung solcher Unsicherheiten spielt eine entscheidende Rolle, z.B. bei unbekannten Bauteiltoleranzen oder Messfehlern. Eine Herausforderung besteht unter anderem darin, aus experimentellen Daten Erkenntnisse über die Parameterverteilung zu gewinnen. In diesem Zusammenhang bietet die Bayes'sche Inferenz einen Ansatz die numerische Simulationen mit experimentellen Daten zu kombinieren, um eine bessere Kenntnis der Unsicherheiten zu erhalten. Die Standard-Methoden benötigen häufig viele Auswertungen, um eine hohe numerische Genauigkeit zu erreichen. Dies ist ein großer Nachteil, insbesondere wenn die Kosten für eine einzelne Simulation sehr hoch sind. Metamodelle, wie z.B. Polynomial Chaos (PC) Erweiterungen, können die Anzahl der benötigten Auswertungen dabei deutlich reduzieren. Um die beschriebenrn Methoden und Algorithmen in der Realität zu validieren, wurde in der vorliegenden Arbeit eine Prüfstand entwickelt, mit dem eine Motorcharakteristik einer elektrischen Maschine mit unsicheren physikalischen Parametern gemessen werden kann. Mit diesem Prüfstand ist es möglich, physikalische Referenzparameter zu definieren und einen entsprechenden Satz von Messungen dazu aufzunehmen. Der Fokus liegt auf der Validierung der Methode basierend auf realen Messungen aus einer industriellen Anwendung. Die numerischen Ergebnisse zeigen, dass der PC-Ansatz die erforderliche Rechenzeit im Vergleich zu dem ursprünglichen Simulationsmodell deutlich reduzieren kann und das Verfahren dadurch in der Praxis anwendbar macht.

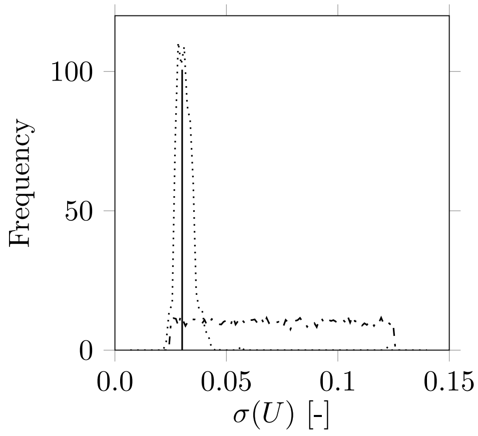
Erratum

Erratum to the thesis ”Uncertainty Quantification for Complex Engineering Systems” by Philipp Glaser.

- page 12: ”several temperatures $[T_i]_{1,\ldots,4}(t)$” should read ”the four temperatures $T_{brush}(t)$, $T_{magnet}(t)$, $T_{winding}(t)$, and $T_{worm}(t)$” and Equation **2.1** should read:

$$\begin{aligned}\boldsymbol{y}(t) &= [\omega(t), I(t), T_{brush}(t), T_{magnet}(t), T_{winding}(t), T_{worm}(t)]^{\mathrm{T}} = \mathcal{M}(t, \boldsymbol{p}, \boldsymbol{u}(t)),\\ \boldsymbol{u}(t) &= [U(t), \tau_{\text{load}}(t)]^{\mathrm{T}}.\end{aligned}$$

- page 13: ”T_{coil}” should read ”T_{winding}”.
- page 16: ”T_{coil}” should read ”T_{winding}” and the following paragraph is added after ”Finally, circles represent the calculated temperatures.”: ”The thermal simulation model calculates twelve temperatures in total, with Figure 2.3 showing the seven selected temperatures relevant to the study. These temperatures are $T_{bearingA}$, $T_{bearingB}$, T_{brush}, $T_{commutator}$, T_{magnet}, $T_{winding}$, and T_{worm}. One considers the ambient temperature T_{amb} as a parameter. In the further course, one will focus on the temperatures T_{brush}, T_{magnet}, $T_{winding}$, and T_{worm}, because one can also measure them with the test bench. As the calculation of the entire thermal part falls under intellectual property, one will only address one exemplary temperature in the following.”.
- page 18: the temperatures ”T_{brush}”, ”T_{magnet}”, ”T_{winding}”, and ”T_{worm}” should read ”$T_{\text{brush,meas}}$”, ”$T_{\text{magnet,meas}}$”, ”$T_{\text{winding,meas}}$”, and ”$T_{\text{worm,meas}}$”.
- page 22: ”Figure 2.4” should read ”Figure 2.7a”.
- page 29: the temperatures ”T_{brush}”, ”T_{magnet}”, ”T_{winding}”, and ”screw T_{screw}” should read ”$T_{\text{brush,meas}}$”, ”$T_{\text{magnet,meas}}$”, ”$T_{\text{winding,meas}}$”, and ”worm $T_{\text{worm,meas}}$”.
- page 30: ”$T_{\circ,\text{meas}}$” should read ”$T_{\text{brush,meas}}$”, ”$T_{\text{magnet,meas}}$”, ”$T_{\text{winding,meas}}$”, and ”$T_{\text{worm,meas}}$”.
- page 31: ”$T_{\text{worm,valid}}$” should read ”$T_{\text{winding,valid}}$”.

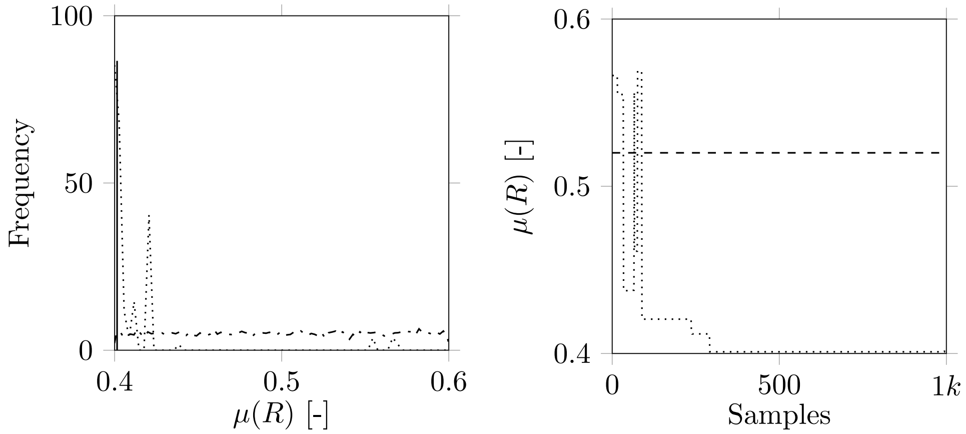


(a) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

Figure 1.1: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the surrogate model with the HMC algorithm.

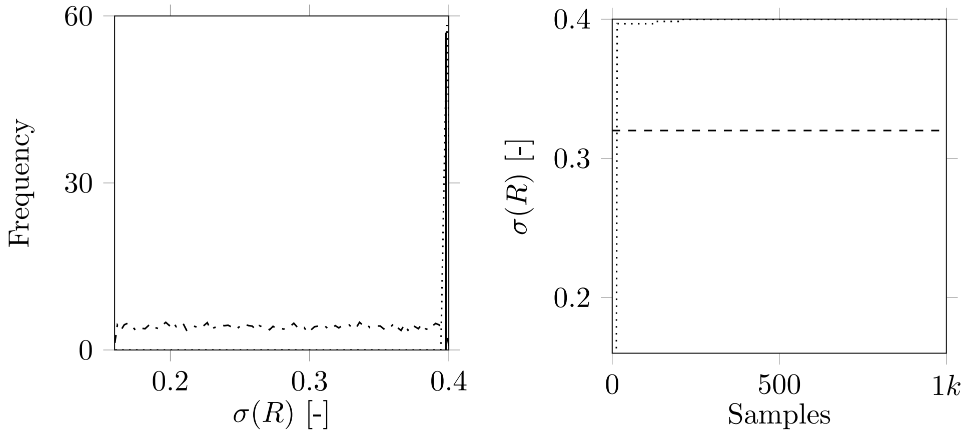
- page 114: Four new figures (see Figure 1.2) are added and show the described results of the resistance.
- page 118: The scale of the Figure 4.20c was updated (see Figure 1.1).
- page 123: The scale of the Figure 4.23 was updated (see Figure 1.3).

Heidelberg, den 18.01.2021



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the resistance R. MLE is displayed with solid line.

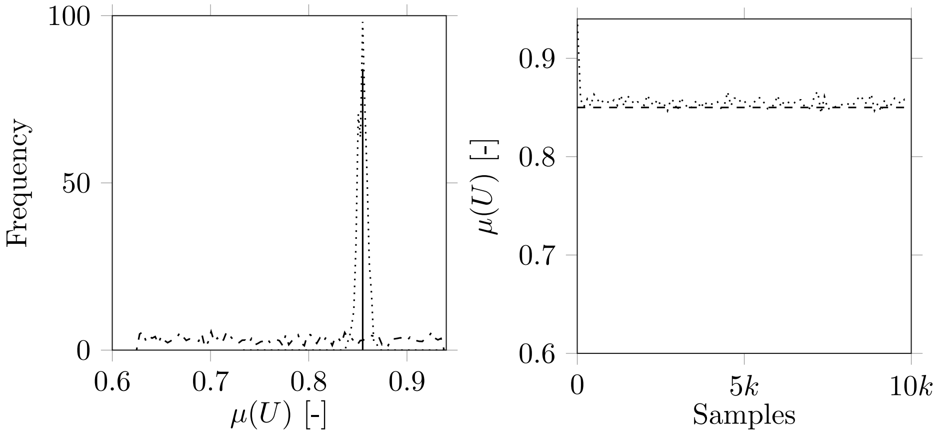
(b) Trace (dotted line) of the Markov chain for $\mu(R)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the resistance R. MLE is displayed with solid line.

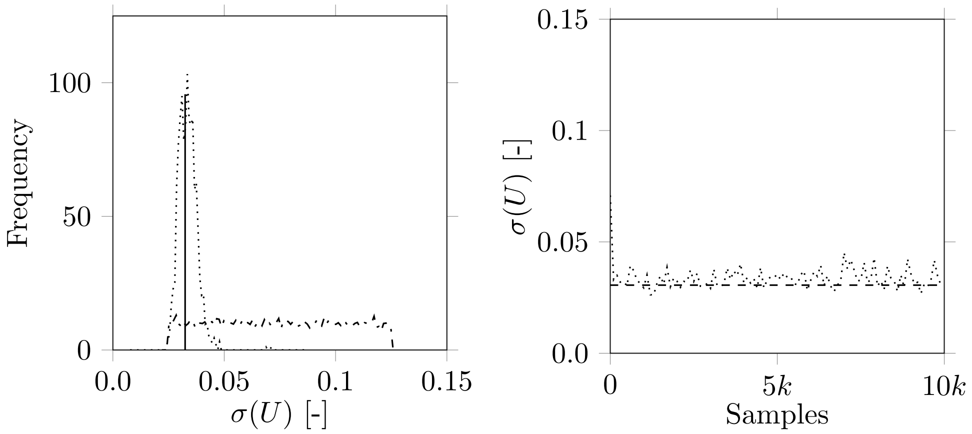
(d) Trace (dotted line) of the Markov chain for $\sigma(R)$. The dashed line shows theoretical value.

Figure 1.2: Numerical result of the MCMC estimation for parameter R with one time point. The computation was performed on the original model.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows the theoretical value.

Figure 1.3: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the surrogate model with No-U-Turn Sampler (NUTS) and enlarged prior assumptions.

Contents

1

Introduction

This work deals with the quantification of uncertainties, and it focuses on enhancing the quality of simulation models. More precisely, one is interested in the propagation of input uncertainties and their impact on the overall system prediction. The windscreen wiper system is the test application in which the used methods show their applicability. Real product measurements support the investigation. The central challenges to be addressed based on the application case are the quantification of input distributions, the efficient propagation of uncertainties through a simulation model, and the handling of measurement series.

Within the development process of a new windshield wiper system, simulation models play an increasing role in all stages. From the design process, layout decisions can be made very efficiently using models. Optimizations of the product functionalities within the application domain can be performed fast without real prototypes. Release decisions based on simulation models with less real testing can significantly reduce costs. In this context, the predictability of the used simulations concerning reliability is crucial, and therefore, the consideration of realistic distributions is an essential element.

Simulation models are currently in use to support the overall product development. In natural science and engineering, detailed models already omit real prototypes. From an economic point of view, the trend of increasing the use of simulations is necessary to master the growing system complexity and to stay competitive. The consideration of uncertainties for the prediction of system behavior is a crucial component to deliver a reliability statement. Uncertainty Quantification (UQ) supports the efficient propagation of input uncertainties, and it provides stochastic information for the computed solutions. As a result, engineers obtain a better insight into their products, and they can make statements on the robustness of their systems. Besides, UQ, in combination with a measurement series, improves the quality of input distributions. In the process of gathering information about inputs, engineers have to rely on domain expert judgments. In some cases, the estimates are not satisfying, and there these methods are highly required by them.

The current chapter provides a better insight into the challenges of engineering systems based on multi-domain components. The main emphasis is on the investigation of UQ methods, and the section introduces its concept in combination with a unique test bench hardware, which ensures the applicability to practical problem classes. Finally, the outline provides details on the contribution and structure of this work.

1.1 Challenges in complex engineering systems

Today the demand for a profound and precise understanding of physical, economic, and other systems is growing. Increasing computing and storage capacities of new hardware foster this trend and enable the development of sophisticated and realistic simulation models. These approximations are often composed of domain-specific parts, and those elements combine themselves to the overall model by interactions, such as dependencies or relationships. If the specific parts expand the system properties by nonlinearities, feedback loops, or memories, one can consider the system as complex. In other words, the property 'complex' cannot be reduced to the presence of particular system characteristics but indicates that the model inputs map to the outputs elaborately. Nowadays, there is a broad field of applications, and categories according to different system characteristics help to classify them. These classifications lead to individual properties that can favor the use of appropriate methods. For example, engineering systems considered later have a strong physical background, and if one examines the windscreen wiper system, then this system has several modeled components that are electrically, mechanically, and thermally motivated.

As a manufacturer of windscreen wiper systems, one remains competitive by optimizing all areas of the value chain, and concerning simulations, the development area is addressed here above all. More precisely, simulation models offer the possibility to provide a statement about a system without, for example, having a real prototype. From an economic point fo view, omitting a physical prototype in the development stage can, in many cases, reduce either the time, the costs, or usually even both. If simulations are not only used to accompany the development, but decisions are also made based on it instead of a test bench, the requirements on the model increase at the same time. The main challenge is to achieve reliable simulation results so that one can make competent decisions. The model has to be able to estimate the real behavior sufficiently well so that the numerical result obtains the same statement. It is necessary to model selective areas in more detail to achieve the stated quality, and this poses a significant challenge. The major issue is not so much to detail the models but to integrate variations as they occur in reality. Specifically, the distributions of relevant sources, like environmental influences, component variations, or production tolerances, are part of the investigation. Thus the transformation from a purely deterministic to a stochastic consideration has to be done. One research task is to increase the efficiency of this computation in such a way that evaluations are feasible from an economic point of view.

The effectively handling of uncertainties in simulation is only one challenge. Achieving realistic distributions for the inputs and parameters of the model is just as demanding. At this point, one can distinguish between two different cases. In the first case, the distributions can be measured directly and with modest effort. During the production of the windscreen wiper drive, geometric variations occur during manufacturing, and these tolerances can be measured straight for each wiper drive. An adequately high number of data points yield proper distributions for the inputs or parameters. In the second case, those necessary measurements are not available for physical or economic reasons. At the same time, it is of high interest to know what the distributions have to look like. This stochastic information is necessary to have a good representation of the reality at the output. This situation is one of the significant challenges for complex engineering systems, and the present work addresses this issue.

The quality of the model approximation additionally depends on model parameters, which are not varying, and the optimization of those values is similarly challenging. The determination of the optimal model parameters is summarized in the literature by the term parameter identification. In other words, the task implies the identification of a parameter set that matches the simulation results with observations based on a predefined metric as closely as possible. So far, the procedure is well known, and there exists a lot of information in the literature. However, the application areas of the models increase due to the stochastic consideration mentioned before, and this influences the parameter identification results. Another challenge associated with this procedure is the selection of the measurement data for the metric. If there is no dedicated measurement series for the identification, an optimal set of signals is needed, and an extension of the known procedure is required to select the relevant data within the entire data set. In summary, the stochastic extension addresses new requirements on parameter identification in this setting.

As a final consideration, the measurement data poses a separate challenge concerning the determination of parameter distributions. In most of the studies in the literature, the validation of the methods under investigation uses only synthetic measurement data. To show the general applicability and function, this is entirely sufficient. When applying to a physical product, it is necessary to access the existing measurement series. The windscreen wiper system provides a large amount of data from the development and manufacturing stages, and these are capable of approximating the parameter distributions. The difficulty resides in the fact that no references are available for the validation of the parameter distributions themselves. Besides, the available measurement setup is incapable of obtaining these values. Therefore, the validation needs appropriate measurement data in this case.

1.2 Uncertainty Quantification

Uncertainty Quantification (UQ) is a multidisciplinary topic and lots of disciplines, such as mathematics, statistics, computer science, and engineering, contribute to the solution. Besides, UQ has a wide range of application cases. Considering the methods, one can say in a strongly simplified way, that UQ offers an end-to-end investigation of inputs and parameters with the outputs concerning their distributions. Additionally, those methods can also be used to identify and study dependencies and correlations within the system.

The literature distinguishes between two principal sources of uncertainty: Aleatoric and epistemic uncertainties. The first category includes all irreducible uncertainties, and they are responsible for the fact that the same experiment carries out slightly different results. In the case of the windshield wiper motor, the load torque is a aleatoric uncertainty because the wiper windshield contact varies due to unevenness and other external influences such as airflows. Epistemic uncertainty is also known as systematic uncertainty, and it describes variations that one can reduce by additional knowledge, but this usually does not happen in practice. A typical case is that the simulation model does not contain specific physical effects in detail for reasons of efficiency, and therefore

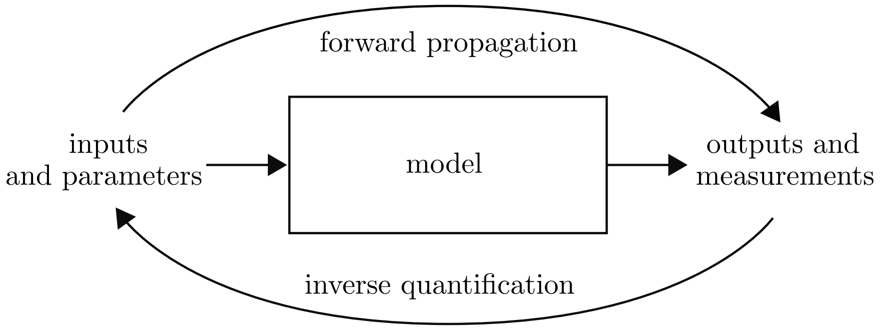


Figure 1.1: Overview scheme of UQ.

variations occur. The measurement system causes another discrepancy. Regarding the windshield wiper drive, a simple measuring device records the current and voltage, and scattering occurs due to the low sampling rate.

There are two major classes of problems in uncertainty quantification. One of them is the forward uncertainty propagation, where the distributions on the inputs and parameters propagate through the simulation model and provide a statistical statement about the outputs. The other class is the inverse uncertainty quantification, where the related methods use measurement data to calibrate the input and parameter distributions. Figure 1.1 shows a visualization of the interaction.

A common approach for the forward uncertainty propagation is the Monte Carlo method. It is based on random realizations of the input or parameter data and requires no modifications to the simulation model and to the solver. The board acceptance of this methodology owes its simplicity in the application. For complex models with a long simulation time, this approach is usually not useful and needs replacement with more efficient methods. Part of this work is investigating new approaches to increase the efficiency. A constraint in this context results from the requirement that no changes to the model equations and the solution method are possible. This assumption considerably restricts the choice of available methods, and at the same time, it increases the applicability to complex models from an industrial point of view. In so-called black-box models, only the inputs, parameters, and outputs are accessible. The solution methods are already integrated and can just be manipulated externally to a limited extent.

Given some experimental measurement data of system outputs and results from a corresponding simulation model, inverse uncertainty quantification assesses the variance between the experiment and the numerical results and gives an estimate on the input and parameter distributions of the model. This approximation of the stochastic moments of the inputs and parameters from the measurements offers entirely new possibilities for the industry. The focus of this work lies in the application and validation of the method based on a real product with an independent test bench.

1.3 Objective and outline

The present work is interdisciplinary and deals with the fields of engineering science and mathematics. The main focus is on the implementation of an inverse uncertainty quantification method for a real application example from the industry. More precisely, the estimation of the stochastic moments on the input and parameter side is the main task and crucial for product development. In the literature, there are already many theoretical investigations and possible extensions, which are practice-oriented. From a scientific point of view, the challenge is the applicability of the mentioned studies for a real application and thus building a bridge between science and industry. The construction of a windshield wiper test bench enables the investigation and validation of the inverse uncertainty quantification. This research includes studies on obtaining probability function, efficiency considerations, and requirements for models and measurements. One major obstacle is the creation of realistic measurement data and the validation of the inverse uncertainty quantification results. In the field of engineering science, this challenge is addressed in this thesis and shown using the test bench. The primary emphasis in mathematics is on solving the inverse problem in an efficient way. Based on current methods from the literature, proposed improvements enable the fulfillment of prerequisites in reality. Besides, by coupling methods from different areas, a more efficient calculation, and generic applicability are achieved.

This work offers three thematic areas, which easily link to the schematic overview of UQ. In addition to the simulation model, the test bench is an additional central element.

Chapter 2 introduces the underlying application that partially serves as a benchmark in this study. The main element is an electric drive of a windshield wiper system, and Section 2.1 motivates the challenges arising from this application scenario. Subsequently, one examines the individual components of the simulation model in more detail and discusses their modeling (see Section 2.2). In addition to the simulation, a real test bench helps to record measurement series. Besides a general overview, Section 2.3 contains additional information on the generation and post-processing of measurements. In the further sections, one describes the parameter identification and validation of the previously presented simulation model.

Chapter 3 focuses on forward propagation. First, one gives an overview of the available methods and explains an accepted approach in detail. The focus is on a method from the literature that promises an increase in efficiency compared to classical methods for many problem classes (see Section 3.1). One introduces the so-called Polynomial Chaos Expansion (PCE) procedure first through a simple application example in Section 3.2 and basic efficiency studies show the performance. One can already derive the requirements for the complex model and limitations of the methodology from the simplified model. Afterward, one applies the algorithms on the test bench model, and Section 3.3 explains the numerical results. Finally, Section 3.4 validates the numerical results of the simulations with the measurement data from the test bench.

Based on the measurement series and the simulation model, Chapter 4 investigates the backward propagation in more detail. The approximation of the stochastic moments of the inputs and parameters from the measurements offers completely new possibilities for the industry. The current state of the art helps to get an overview, and Section 4.1 gives an introduction to Bayesian inference. More precisely, one outlines the underlying idea and discusses the Likelihood function in more detail. It describes the relationship of the input and parameter distributions to the measurements, and its modeling is explained further in Section 4.2. In the following, the modeling of the uncertain inputs and parameters is considered (see Section 4.3), and one proposes method enhancements to increase the efficiency of the investigations without the need to change the complexity of the method applicability (see Section 4.4). Concluding, the numerical results are presented in Section 4.5. These summaries also include a comparison of efficiency in relation to the different methods.

Conclusions and perspectives are covered in Chapter 5.

2

Model and test bench hardware

This chapter introduces the windscreen wiper system mentioned earlier, and the included electric drive is the benchmark problem supporting this study. The overview contains an initial description of how the electric drive works and it involves a detailed consideration of the particular components and their functional relationships. In this context, one considers the application cases and the associated requirements for the product.

A simulation model exists for the system under consideration, and this chapter describes its structure in more detail. The subsequent application of the simulation model requires data, and the associated measurements arisse in the test bench hardware. This chapter describes the design of the test bench, its functionality concerning measurement data, and the connection to the model.

The electric drive is the central part of the test bench, and the design of the entire setup allows reproducing realistic load scenarios. Furthermore, additional measuring instruments ensure detailed observation quality of the physical system and the environment parameters, such as temperatures. Based on the observations, one tests the simulation model with reality and ensures model confidence.

2.1 Windshield wiper drive

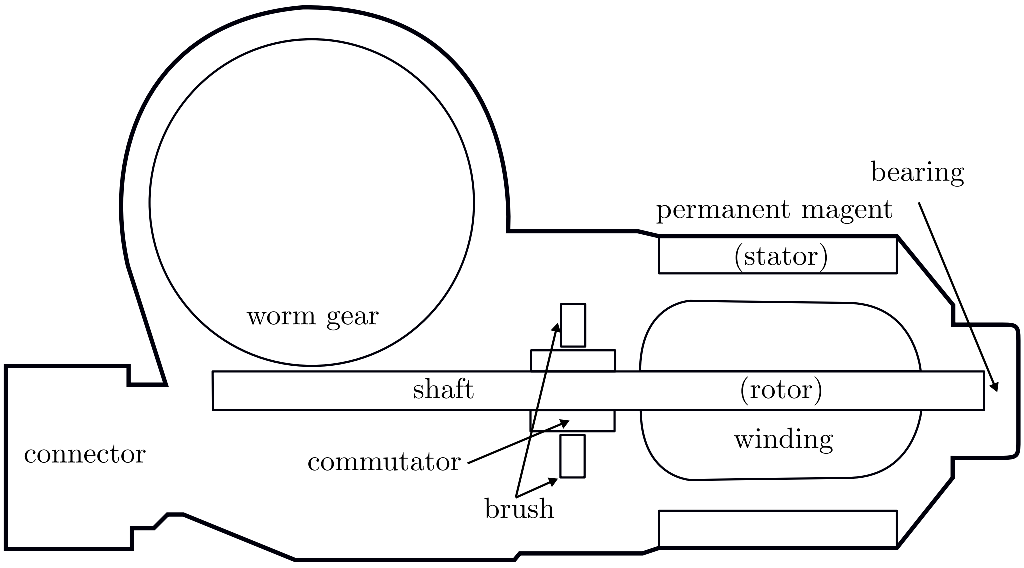


Figure 2.1: Layout of a windshield wiper drive with its main parts.

The windscreen wiper system resides in the engine compartment and consists of a drive unit, rods, and windscreen wipers. Instead of the entire windscreen wiper system, one considers only the drive unit in this thesis. The Figure 2.1 provides a systematic view, and it shows the central parts such as rotor, stator, brushes, commutator, and windings. The drive is a direct-current (DC) motor with brushes which commutates internally. In this type of construction investigated, the rotor is the moving part. It rotates the shaft and thus delivers the mechanical power. The rotor carries conductors through which the applied current flows and which interact with the magnetic field of the fixed stator. In this case, permanent magnets are the cause of the magnetic field. The interaction then generates the forces causing the shaft to rotate. To achieve a constant motor torque, one uses a split ring device called a commutator, and it causes the current to be reversed or zero at any point where the coil passes the level vertical to the magnetic field. The brushes ensure the electrical contacts to the rotating ring, and the mechanical construction consists of spring-loaded carbon contacts. A worm gear transmits the motor torque to a gear wheel, which connects to the wiper linkage.

When it rains, the windscreen wiper system helps to wipe the water from the windscreen and thus actively contributes to driving safety. Reliability is therefore essential, and high demands for tolerance limits and life cycle statements are needed. From a large number of real application scenarios, one derives requirements and appropriate tests. The considered scenarios represent a particular load case of the motor, and one examines them individually. Looking at the input space of the system, each of those load cases can be assigned an area, and these areas have little or no overlap with each other. This classification initially has a weak connection with uncertain variables and is instead an analysis by a test manager. He is responsible for checking the requirements, defining the scenarios, and the necessary tests. Depending on the initial situation, the effect of the corresponding uncertainties changes. The following scenarios present a significant part of the challenges in combination with the involved and varying parameters.

A typical operating range of the windscreen wiper is wiping on a wet surface in combination with a low driving speed of the car. The torque at the drive is modest and quite constant over time. If one changes the setting insignificantly and increases the driving speed, the torque at the drive grows considerably. The air streaming past exerts extraordinary forces on the wiper system, and due to various influences, the airflow can vary exceedingly in this scenario, which also impacts the applied motor torque. If the water does not entirely cover the windscreen, one can observe a similar behavior even at low speeds. In this case, the wiper blades do not glide steadily, and friction forces occur between them and the windscreen. Again, one can expect that the torque varies considerably depending on the moistening. Regardless of the load scenarios, it is advisable to consider the load torque as an uncertain input.

In addition to the applied load torque, the operating voltage is also a significant factor influencing the desired operation of the wiper drive, and the presented scenarios assume that the operating voltage is as constant as possible. The present vehicle models contain a broad spectrum of energy consumers that lead to small variations in the supply voltage of the electric drive. Furthermore, some car manufacturers increase the on-board voltage if the outside temperature sinks below a specific value so that all devices continue to function faultlessly even at quite cold temperatures. Finally, the supply voltage is another uncertain input, but for simplicity, one excludes the last-mentioned scenario in the consideration.

The two uncertain parameters presented here are strongly linked to the scenarios and, detached from them, the investigation also includes uncertainties from production. An example is a winding resistance, which depends on the length of the wire used in the motor. The deviation occurs in the manufacturing process, and one assumes that the winding resistance varies in the mentioned scenarios.

In summary, the uncertain parameters to be considered are the load torque, the voltage, and the winding resistance. For further considerations, one picks a scenario that allows the product to work in a critical range. A confident statement is, of course, very significant in that case. The situation in which the water does not cover the windshield completely offers precisely this critical consideration, and an essential requirement is that the wiping performance remains constant despite all the different effects. From a test manager's perspective, the condition arises that the drive must deliver a constant speed regardless of any uncertainties. The corresponding test case represents that one passes it as long as the motor speed does not drop below a predefined limit value. This statement must be valid for every combination of the uncertain parameters mentioned above.

2.2 Simulation model

The simulation model of the electric drive divides into three schematic parts, which Figure 2.2 illustrates. In the following work, one treats the model as a black-box, and therefore one only uses the relationships of the inputs $\boldsymbol{u}(t)$, the parameters $\boldsymbol{p}$ and the outputs $\boldsymbol{y}(t)$. Internal states $\boldsymbol{x}(t)$ that are not accessible outside the system, represent the dependencies of the electrical, mechanical, and thermal parts. The model presented is provided by the Electric Drives division of Robert Bosch GmbH.

The inputs of the simulation model are the voltage $U(t)$ and the load torque $\tau_{\mathrm{load}}(t)$. The outputs are the motor speed $\omega(t)$, the current $I(t)$, and several temperatures $[T_i]_{i=1,\ldots,4}(t)$ which are explained in detail in Subsection 2.2.3. The global model $\mathcal{M}$ can be defined as:

$$\boldsymbol{y}(t) = [\omega(t), I(t), [T_i]_{i=1,\ldots,4}(t)]^{\mathrm{T}} = \mathcal{M}(t, \boldsymbol{p}, \boldsymbol{u}(t)), \quad \boldsymbol{u}(t) = [U(t), \tau_{\mathrm{load}}(t)]^{\mathrm{T}}. \tag{2.1}$$

The model parameters $\boldsymbol{p}$ which have to be obtained by identification, are highlighted separately in the following by use of the subindex $\circ_{\mathrm{p}}$. The following three subsections use the nomenclature introduced here and provide detailed information on the modeling.

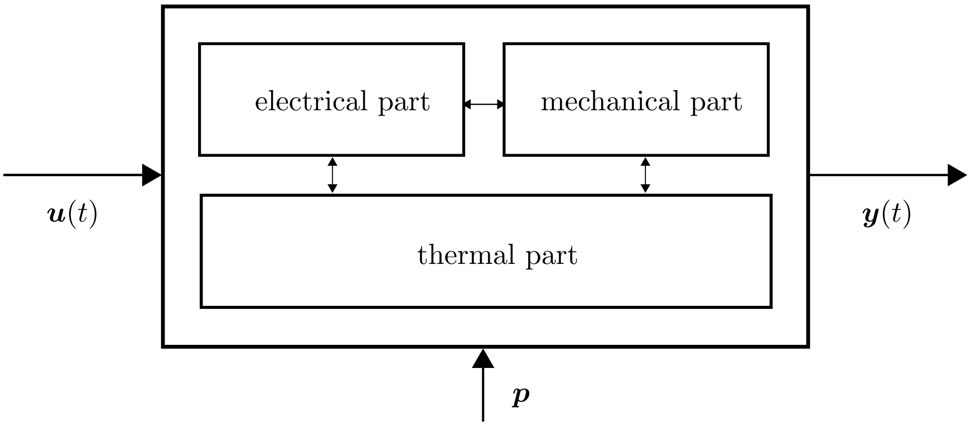


Figure 2.2: Schematic view of the electric drive model with its parts and interconnections.

2.2.1 Electrical model equations

The electrical part of the simulation model is based on an ordinary differential equation [68] and depends on the input voltage $U(t)$. The time derivative of the current $I(t)$ expresses as follows:

$$\frac{d}{dt}I(t) = \frac{1}{L_\mathrm{p}}\Big[- RI(t) - c_\mathrm{e}(t)\omega(t) + \big(U(t) - U_\mathrm{drop}\big)\Big], \tag{2.2}$$

where the inductance L_p is a model parameter. The length of the wire and the temperature of the coil T_coil influence the winding resistance R. Since this depends on the temperature-dependent material constant for a conductor, it grows with increasing temperature. Therefore, one assumes that the resistance adjusts linearly during the temperature change:

$$R = R_\mathrm{p}\big[1 + \alpha(T_\mathrm{coil} - T_\mathrm{amb})\big].$$

The linear temperature coefficient of the resistor is represented by α, one uses the ambient temperature T_amb as a reference, and R_p represents the model parameter of the winding resistance at the reference temperature T_amb.

The product of the so-called back electromotive force constant c_e and the motor speed $\omega(t)$ results in a voltage that opposes the change in current which induced it.

The temperature of the magnet T_{magnet} influences this. Instead of a linear approach, one chooses a polynomial approximation:

$$c_{\text{e}}(t) = \sum_{i=0}^{6} \left(\sum_{j=0}^{6} b_{(i,j)} T_{\text{magnet}}^{j} \right) I^{i}(t),$$

where the coefficients $b_{i,j}$ are specified by an internal layout tool and the model parameter c_{e_p} replaces the first element $b_{(0,0)}$. In this approximation, one assigns the sum of the index i to the current I and the sum of the index j to the temperature T_{magnet}.

The voltage drop U_{drop} depends on the wire used and in the given simulation model, one assumes a polynomial approximation with corresponding coefficients a_i:

$$U_{\text{drop}} = \operatorname{sgn}\big(I(t)\big) \sum_{i=0}^{6} a_i |I(t)^i|.$$

The coefficients are calculated from an internal database, physical values such as wire length, measurements, and expert knowledge. The model parameters for this subsection can be summarized as:

$$\boldsymbol{p}^{\text{elec}} = [R_{\text{p}}, L_{\text{p}}, c_{\text{e}_\text{p}}]^{\text{T}}. \tag{2.3}$$

2.2.2 Mechanical model equations

The mechanical part of the electric drive consists of the motor part and a worm gear. The ordinary differential equation depends on the input load torque $\tau_{\text{load}}(t)$ and one can write the time derivative of the motor speed $\omega(t)$ as follows [68]:

$$\frac{d}{dt}\omega(t) = \frac{1}{J_{\text{p}}} \Big[\big(c_{\text{m}} I(t) - \tau_{\text{loss}} - \tau_{\text{fric}} \big) \eta - \tau_{\text{load}}(t) i_{\text{g}} \Big]. \tag{2.4}$$

The model parameter J_{p} expresses the total inertia of the motor armature, and the gear ratio i_{g} is a constant value proportional to the worm gear ratio. In this case, the torque constant c_{m} is equal to the back electromotive force constant c_{e}, so there is a functional expression that describes how the motor converts current to torque and angular velocity. In the following, one uses the same model parameter c_{m_p}. The armature losses based on the hysteresis loss $\tau_{\text{hysteresis}}$ and the eddy current loss τ_{ecurrent} sum up to:

$$\tau_{\text{loss}} = \Big(\tau_{\text{hysteresis}}\big(I(t), T_{\text{magnet}}\big) + \tau_{\text{ecurrent}}\big(I(t), T_{\text{magnet}}\big) \Big) \epsilon_{\text{iron,p}},$$

which can be adjusted by the model parameter $\epsilon_{\mathrm{iron,p}}$. Similar to the torque constant, this function also depends on the temperature of the magnet T_{magnet}.

The mechanical friction losses can be explained below:

$$\tau_{\mathrm{fric}} = \tau_{\mathrm{fric,air}}\big(\omega(t)\big) + \tau_{\mathrm{fric,motor}},$$

where the air friction loss $\tau_{\mathrm{fric,air}}$ depends on the motor speed and the losses of the motor $\tau_{\mathrm{fric,motor}}$ can be further detailed:

$$\begin{aligned}\tau_{\mathrm{fric,motor}} = \ & \tau_{\mathrm{fric,bearingA}}(T_{\mathrm{bearingA}}, \mu_{\mathrm{A,p}}, \eta) + \\ & \tau_{\mathrm{fric,bearingB}}(T_{\mathrm{bearingB}}, \mu_{\mathrm{B,p}}, \eta) + \\ & \tau_{\mathrm{fric,commutator}}(T_{\mathrm{commutator}}).\end{aligned} \tag{2.5}$$

In the following formulas, the letter "A" relates to the bearing between the worm and the commutator, and the letter "B" relates to the bearing on the pole housing. Both friction losses for the bearings depend on their temperature T_{bearingA}, T_{bearingB}, and the gear meshing efficiency η. The model parameters $\mu_{\mathrm{A,p}}$ and $\mu_{\mathrm{A,p}}$ influence each loss. The motor brushes cause the loss of the commutator, and it depends on the temperature of the commutator $T_{\mathrm{commutator}}$. The relation is modeled using a lookup table based on measurement data.

Looking at the Equation (2.4), the gear meshing efficiency η is a central value for the worm gear and is calculated as [61]:

$$\eta = \frac{\tan(\gamma)}{\tan\left(\gamma + \arctan\left(\frac{\mu_{\mathrm{worm}}(T_{\mathrm{worm}})\epsilon_{\mathrm{worm,p}}}{\cos(\alpha_0)}\right)\right)},$$

where γ is the lead angle of the worm and α_0 expresses the pressure angle. The friction coefficient μ_{worm} depends on the worm temperature T_{worm}, and one also models it using a lookup table based on measured data. The parameter $\epsilon_{\mathrm{worm,p}}$ adjusts the correlation.

The model parameters for this subsection can be combined to:

$$\boldsymbol{p}^{\mathrm{mech}} = [J_{\mathrm{p}}, c_{\mathrm{m_p}}, \epsilon_{\mathrm{iron,p}}, \epsilon_{\mathrm{worm,p}}, \mu_{\mathrm{A,p}}, \mu_{\mathrm{B,p}}]^{\mathrm{T}}, \tag{2.6}$$

where the torque constant $c_{\mathrm{m_p}}$ is equal to the introduced parameter $c_{\mathrm{e_p}}$ in Equation (2.3) and both are merged with the parameter $c_{\mathrm{m,p}}$.

2.2.3 Thermal model equations

Several differential-algebraic equations represent the thermal model of the electric drive. In this subsection, one focuses on a general overview of the subparts, which show the calculation of the different temperatures. One exemplary shows the interconnection

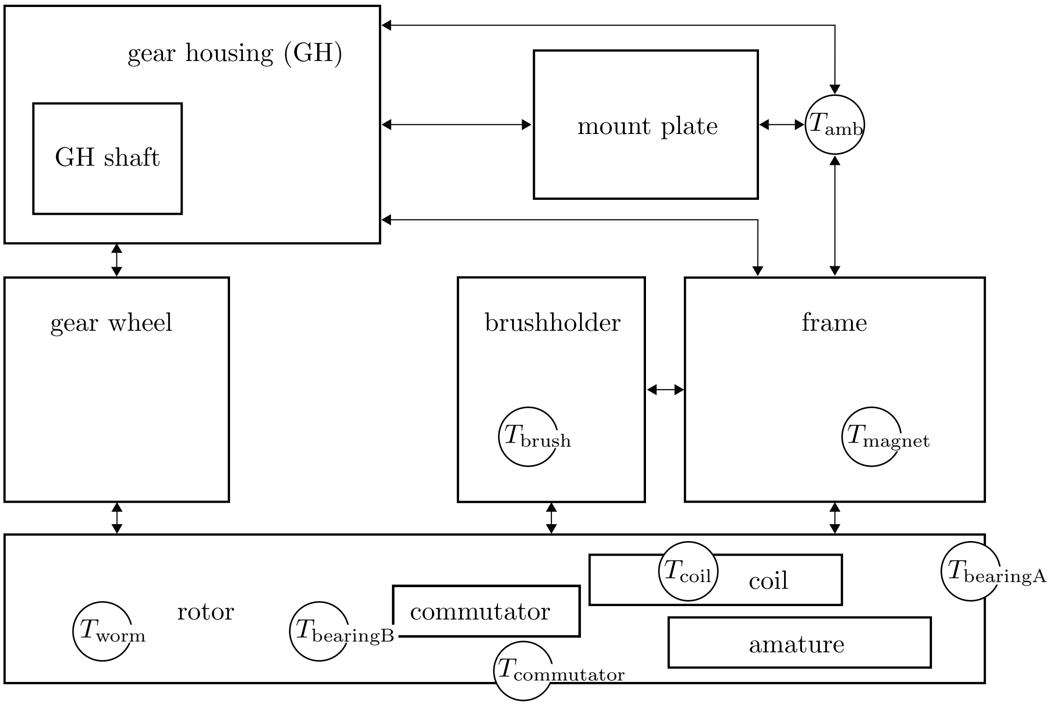


Figure 2.3: Systematic overview of the heat model with its subparts and the resulting temperatures.

with the subsections 2.2.1 and 2.2.2 based on the commutator. Figure 2.3 shows a schematic representation of the entire thermal system. Each square represents a subpart of the model, and the arrows show the interaction between them. If one assumes that the corresponding submodels are physically connected, then the conductivity corresponds to such a connection. More precisely, the gear housing and the gear wheel have an interconnection by the gear housing shaft. Convection is another possibility of energy exchange, and the connection between the brush holder and frame is one example. Finally, circles represent the calculated temperatures.

The calculation of the commutator temperature $T_{\text{commutator}}$ requires the evolved power $P_{\text{commutator}}$ related to the friction caused by the brushes. With $\tau_{\text{commutator}}$ from Equation (2.5) the power can be calculated as follows:

$$\begin{aligned} P_{\text{commutator}} = \quad & \omega(t)\tau_{\text{fric,commutator}} + \\ & I^2(t)R_{\text{Co}}\big(1 + \rho_{\text{Co}}(T_{\text{commutator}} - T_{\text{amb}})\big), \end{aligned}$$

where the resistance of the commutator is defined as R_{Co} and the specific temperature coefficient ρ_{Co} is used.

The model parameters from this subsection are not listed in detail as a vector because they are only briefly needed for parameter identification. If required, the thermal parameters are given below as $\boldsymbol{p}^{\text{ther}}$, and these can be, for example, temperature coefficients for the materials used or heat transfer coefficients.

2.3 Test bench hardware

To obtain measurement data quickly, one often applies noise to existing simulation results and then uses them. To get more realistic data, one builds up a test bench hardware. The main advantage of this configuration is the circumstance that one can predefine input and parameter uncertainties. Based on these distributions and the given test bench hardware, one can automatically perform measurement runs. These data enable the validation of the used methods.

The resulting measurement data and the test bench, in general, are very similar to the data and structures of the so-called endurance testers in the business unit. On these test benches, one checks electric drives under laboratory conditions for wear and other requirements. Moreover, there are test benches that one can use to carry out end-of-line measurements. To do so, one instantly clamps the drives from series production to the test hardware and examines their correct operation. For both types of tests, one has to place the drive manually. The test bench presented in this section is capable of varying various physical parameters without having to replace the electrical drive. Also, more sensors were used to accurately measure the system properties.

The following subsection introduces the test bench hardware and explains the individual requirements. Furthermore, one discusses the method for generating measurement data and its challenges. Based on the previous Section 2.2, one proposes a parameter identification and a validation for a measurement set.

2.3.1 General overview

As already mentioned, the unique feature of the hardware is the ability to define uncertain sources that affect the characteristics of the electric drive. One can perform this procedure automatically without replacing the drive with another, and obtain many measurements based on predefined parameter distributions. It is not necessary to recalibrate the measurement system for each run. Figure 2.4 shows a simplified illustration of the test bench hardware.

The core element is the electric drive marked with the Number ①. The frame mounts the motor with the worm gear, and the power source supplies the drive with electricity.

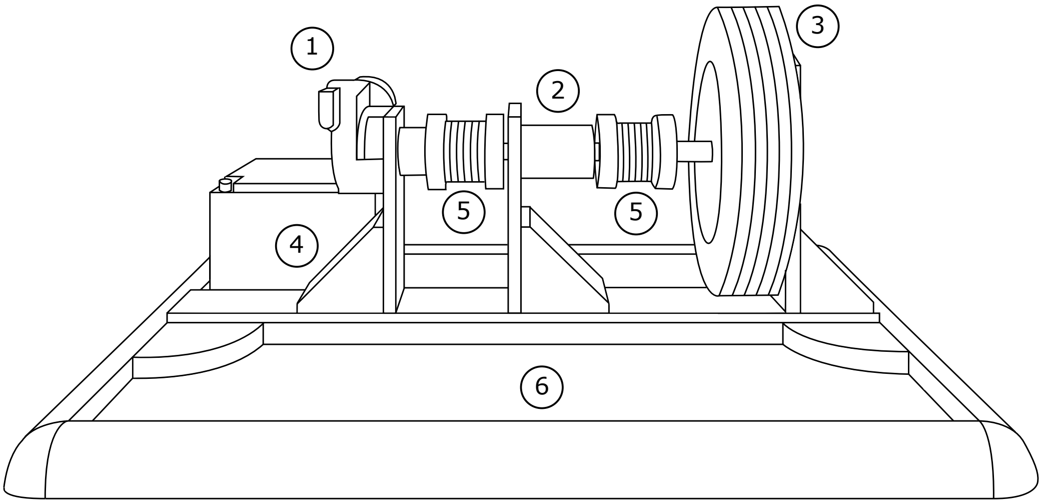


Figure 2.4: Illustration of the test bench hardware [32].

It can handle high current peaks and has a programmable interface. More precisely, a control signal triggers the voltage U settings. Besides, one can adjust the internal resistance R of the power supply, and this emulates a varying winding resistance. In this case, one assumes that a series connection between the internal resistance and the winding exists, so there is not much difference in the applied variance.

Temperature sensors are attached to the drive to check the thermal part of the simulation model. The temperatures of the carbon brush at the negative pole T_{brush}, the magnet T_{magnet}, and the worm T_{worm} are measured using a type K thermocouple installed with an insulating adhesive. An additional infrared sensor is attached to the pole pot so that one can measure the winding temperature T_{winding} contactless.

One uses a metal bellows coupling (see Number ⑤) to connect the electric drive shaft with the other components, and this construction prevents additional transmission of vertical and tangential forces. The motor speed ω and the torque τ are measured with the sensor illustrated with Number ②. This device can process the torque signal without contact. An integrated speed encoder provides two signals with a resolution of 360 pulses per revolution.

Another metal bellow coupling connects the shaft of the sensor with the magnetic powder brake (see Number ③). With this layout, different load scenarios can be displayed without changing the test bench. The load torque τ_{load} can be adjusted via a control signal, and the specification of the magnetic powder brake specifies the minimum torque.

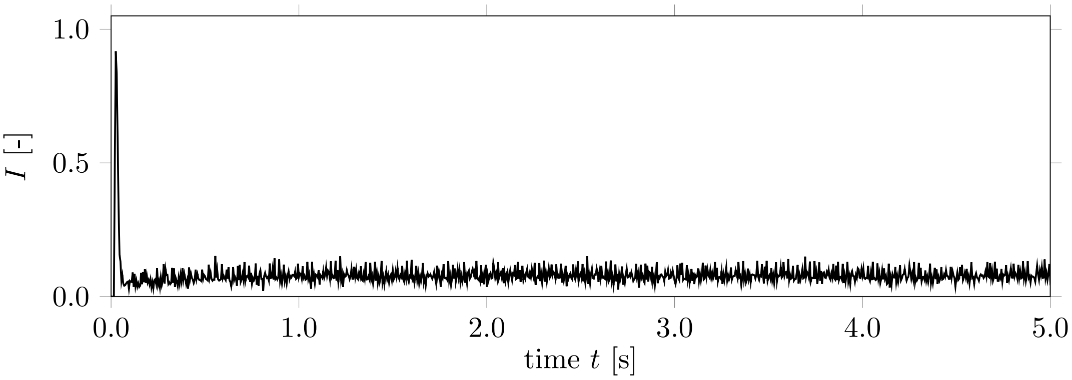
An interface box connects all control and measurement signals, and it pre-processes the data. A dSpace box [19] establishes the connection to the computer and enables the interaction with the software. The battery (see Number ④) in the figure is a symbol for the used power supply.

2.3.2 Obtaining measurement data

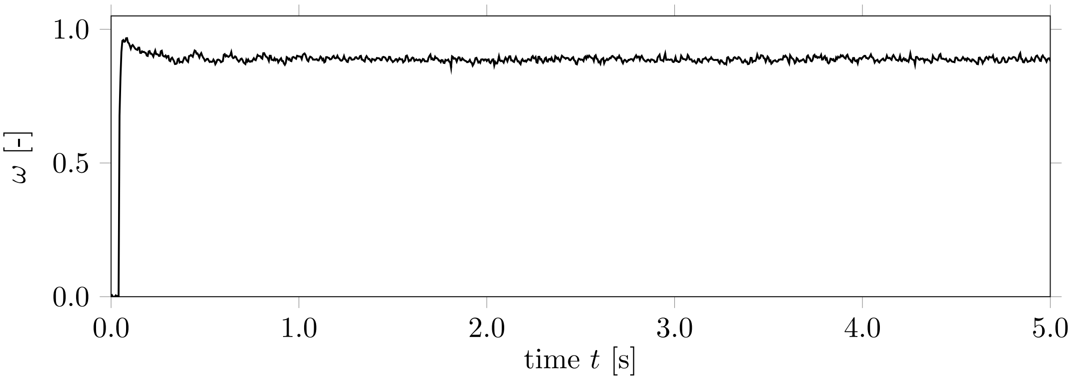
In the further course of the work, all displayed measurement signals and used values are normalized. One divides all affected data and variables by a predefined maximum value. The involved results can be identified by the unit-less representation.

In the following, a measurement run is recorded to get an impression of the data. One assumes that the electric drive is switched on at time $t_{\text{start}} = 0\text{s}$. The signals of the current $I(t)$ and the motor speed $\omega(t)$ are recorded until the time frame $\Delta t = 5\text{s}$ is reached. After the time $t_{\text{end}} = t_{\text{start}} + \Delta t$ the measurement is stopped, and the electric drive is switched off, which means that the voltage is set to zero. The input parameters are assumed to be constant, and the following values are used for the voltage $U = 0.9$, the load torque $\tau_{\text{load}} = 0.055$, and the resistance $R = 0.0$. Figure 2.5 shows the obtained data. The signal values are normalized as described above.

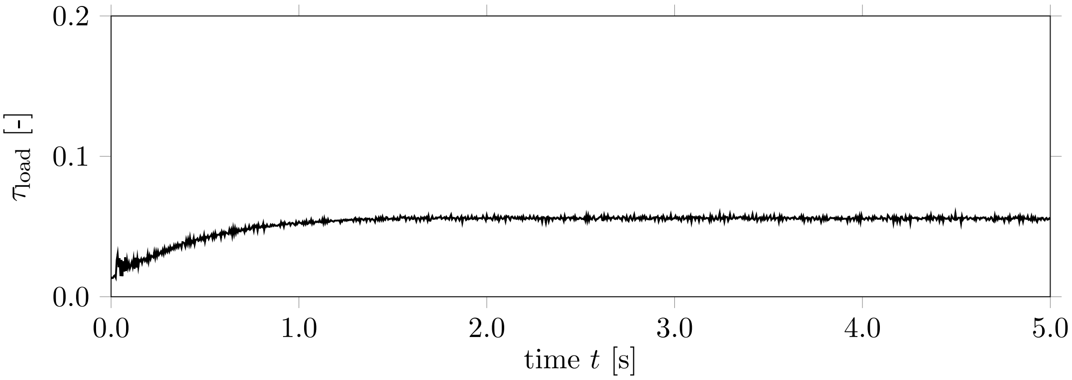
Looking at the current signal over time $I(t)$ (see Figure 2.5a), one can see the startup behavior, which is represented by a large peak at the beginning. Then the current keeps a constant level, and there one can see that the signal has a huge influence of measurement noise. The motor speed over time $\omega(t)$ (see Figure 2.5b) shows a short deceleration before the motor starts rotating with increasing speed. The occurrence of this phenomenon is caused by the fact that the measurement unit of the motor speed has a short time delay since the speed is calculated with a value of the last time step. The signal approaches the so-called stationary state in a short time. This means that the motor speed remains at a certain constant level ω_{steady}. Between $t > 0.1\text{s}$ and $t < 1.5\text{s}$ there are small oscillations around this stationary state to observe. An obvious explanation can be given by tracking the real load torque τ_{load} signal (see Figure 2.5c). It can be seen that the magnetic powder brake used is not capable of delivering the given constant torque trajectory. In practice, the adjusted value is not reached during $t < 1.0\text{s}$. When the electric drive starts moving, the discs inside the brake rotate evenly. The powder between the discs and the coils of the brake is oriented to the applied magnetic force. This starting process of the powder distribution leads to a delay, especially at low speed.



(a) Normalized current signal over time.



(b) Normalized motor speed signal over time.



(c) Normalized load torque signal over time.

Figure 2.5: Test bench hardware measurements of an electric drive during the switch-on process.

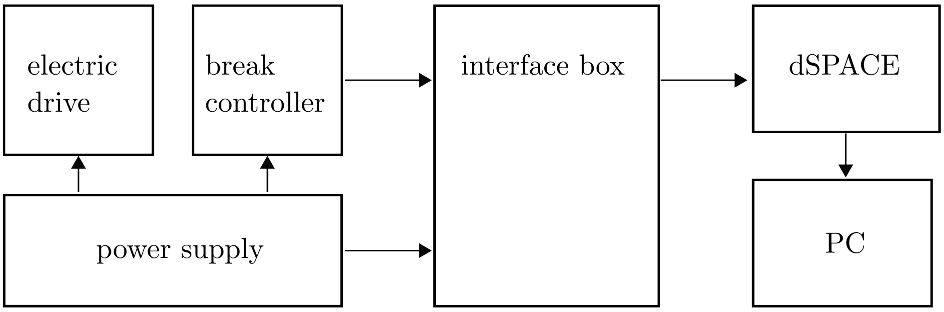


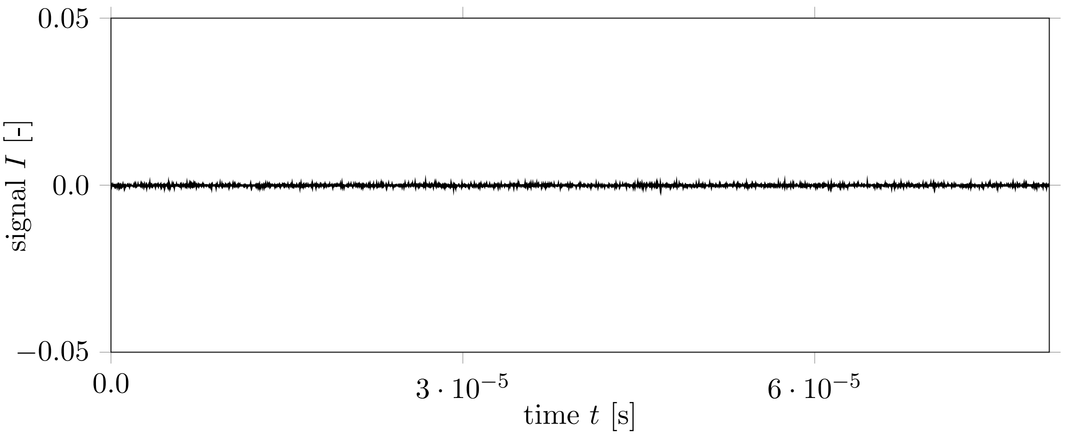
Figure 2.6: Schematic view on the measurement chain.

In the following, one evaluates the current signal in detail, since the measurement noise is far greater than one might initially assume. Therefore, one analyses the effect, and suggests methods for reduction. Based on this, an overall investigation of the test bench hardware is carried out, and a motor characteristic of the electric drive is recorded. With the help of this characteristic, a large part of the dynamic range of the motor can be investigated, and this is used in practice to evaluate the quality of electric motors.

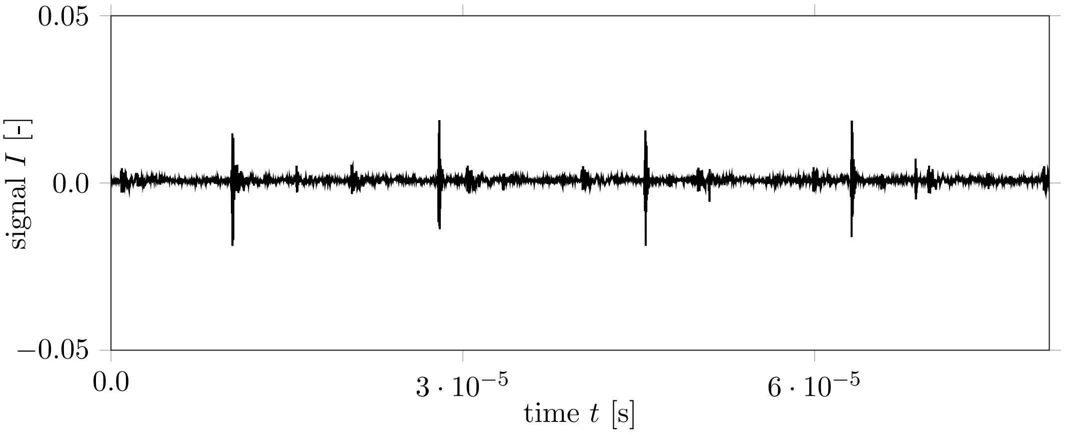
Measurement noise of current signal

Figure 2.6 shows the schematic dependencies of the individual components in such a way that a better overview of the measurement chain for the current signal is achieved. The electric drive is directly connected to the power supply, which has a built-in measuring unit for the current. For a detailed analysis, an additional measuring device is attached to the connection between the power source and the electric drive. For this task, a current clamp is used, which makes it possible to measure the current in a wire without interrupting the connection. In this configuration, the measurement of the signal is recorded by an oscilloscope. The sampling rate of the digital instrument used is 1GS/s, and therefore a time step of 1ns can be resolved. In addition, the internal measuring device of the power supply is connected to the interface box and transmitted to the PC via the dSpace box [19].

The background noise of the measuring instrument is obtained by using the current clamp. In this study, the power supply is turned on, and the electric drive is connected to it. The voltage U is set to zero, and one records a measurement for the background



(a) Signal with all measuring instruments switched off with the exception of power supply.



(b) Signal with all measurement devices switched on.

Figure 2.7: Measurement signal of the current $I(t)$ recorded with a current clamp. In this test scenario, the supply voltage of the electric drive is set to zero.

noise, which one accepts as a reference. Figure 2.4 shows a snapshot of the total signal, and one can observe that the effect is quite small. For comparison purposes, a noise measure $\bar{I}_{\text{noise}}$ is defined as the distance between the minimum and maximum value of the signal $I(t)$. If this measure is applied to the basis value, it is given:

$$\bar{I}_{\text{noise,basis}} = 0.002.$$

After the reference data for the measurement instruments have been obtained, a second measurement is recorded. The difference is that all devices of the test bench hardware are switched on, and they are in the operating mode. The recorded signal is shown in Figure 2.7b, and periodic peaks of different amplitude can be monitored. The predefined noise measure can be quantified as:

$$\bar{I}_{\text{noise}} = 0.056,$$

and compared to the reference $\bar{I}_{\text{noise,basis}}$, it is more than 25 times bigger. The four large peaks cause the largest contribution to the recorded noise. A detailed analysis shows that the measurement chain is not technically decoupled, and the dSpace device is therefore responsible for the large noise contribution. This can easily be seen in Figure 2.8a, where only the dSpace device is turned on. Thus, this equipment is essential for the transmission of the signals to the computer, and the hardware decoupling is not appropriate for cost reasons. A software-based filter is suggested to eliminate the given peaks. For deep insight, a fast Fourier transformation [10, 52] is used to obtain a frequency diagram (see Figure 2.8b).

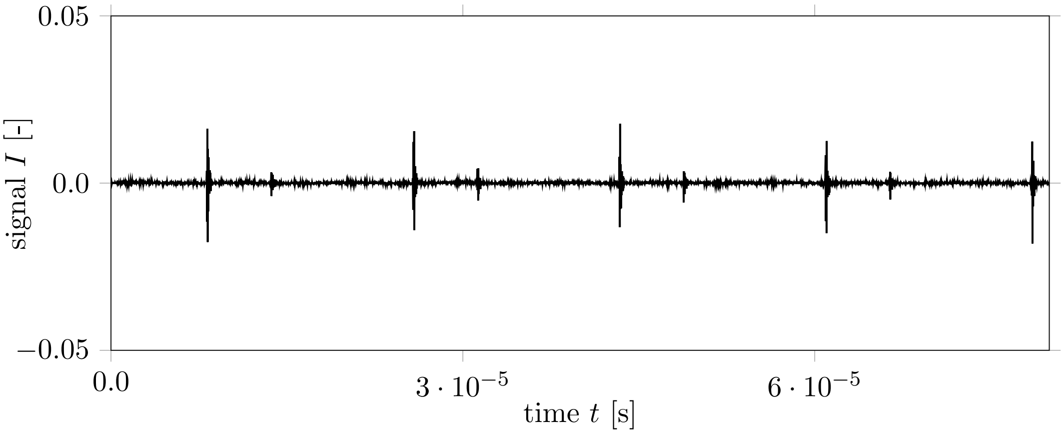
Based on this visualization, two large peaks around $3 \cdot 10^7$Hz can be observed and mapped to the disturbance of the dSpace hardware. In this work, a Butterworth low pass filter [12, 70, 69] is proposed for noise reduction. The infinite impulse response (IIR) filter used can be described as follows:

$$y[n] = \frac{1}{a_0} \left(\sum_{\mathrm{i}=0}^{\mathrm{P_f}} b_{\mathrm{i}} x[n-i] - \sum_{\mathrm{j}=1}^{\mathrm{Q_f}} a_{\mathrm{j}} y[n-j] \right),$$

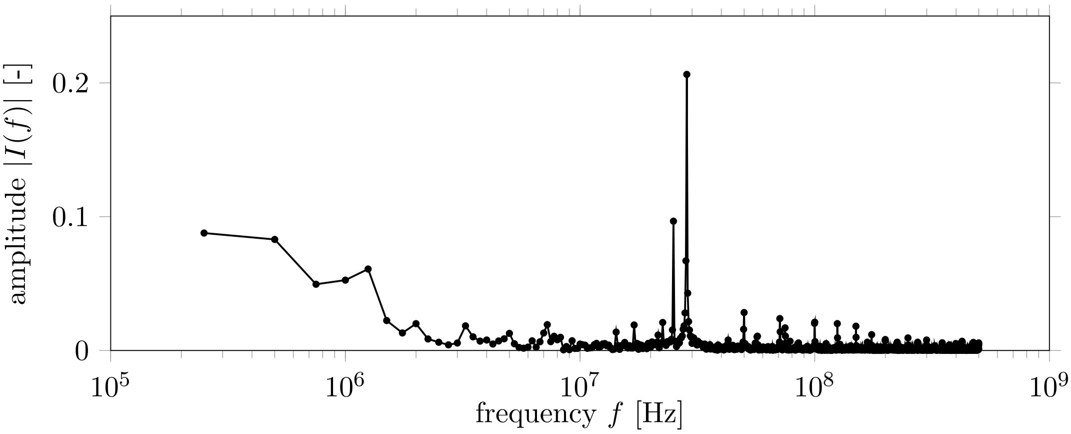
where $x[n]$ is the input and $y[n]$ is the output. The feedback filter coefficients are described as a_{j} with order Q_{f} and the feedforward filter coefficients as b_{i} with order P_{f}. For the design of the coefficients a_{j}, b_{i} a Butterworth filter of second order with the sampling rate $f_{\mathrm{s}} = 1 \cdot 10^9$Hz and the cut-off frequency $f_{\mathrm{cut}} = 2 \cdot 10^7$Hz is used and the coefficients result:

$$\begin{aligned} a = & \; [1.0000, -1.8227, 0.8372], \\ b = & \; [0.0036, 0.0072, 0.0036]. \end{aligned}$$

The filter realization can be applied to the measurement signal of the current. As mentioned at the beginning of this subsection, an oscilloscope was used to record the detailed measurement data. In general, the signals are acquired by the power supply's internal current measuring device. Data acquisition is performed with a sampling time of $t_s = 0.0001$s, which corresponds to a sampling rate of $f_{\mathrm{s}} = 1 \cdot 10^3$Hz. Using the Nyquist-Shannon sampling theorem [38] shows that perfect reconstruction of a band-limited signal is guaranteed when the frequency is less than or equal to $f_{\mathrm{s}}/2$. This means that the resolution of the measurement unit is not detailed enough to capture



(a) Measurement signal of the current $I(t)$ over time.



(b) Fast Fourier transformation based on the given measurement signal.

Figure 2.8: This test scenario covers the case where only the dSpace device is powered and all others are turned off.

the noise phenomenon described. Otherwise, the influential frequency band can be determined. An exemplary calculation could be to assume a measured motor speed ω_{max} at the gearing, and thus the motor speed ω_{motor} equals $i_{\mathrm{g}}\omega_{\mathrm{max}}$. The armature of the motor has L coils, and there is the same number of commutations per revolution. The commutation frequency f_{comm} can be calculated by:

$$f_{\mathrm{comm}} = \frac{\omega_{\mathrm{motor}}}{60} L,$$

that can be used to find an upper limit for the influential frequency band. In this setting, a cutoff frequency of $f_{\mathrm{cut}} = 6 \cdot 10^2\mathrm{Hz}$ is suggested and Figure 2.9 shows the result. Compared to the previous signal path, the noise is significantly reduced, and in the following, the presented filter is applied to the current measurements as a post-processing step.

Identification of the motor characteristics

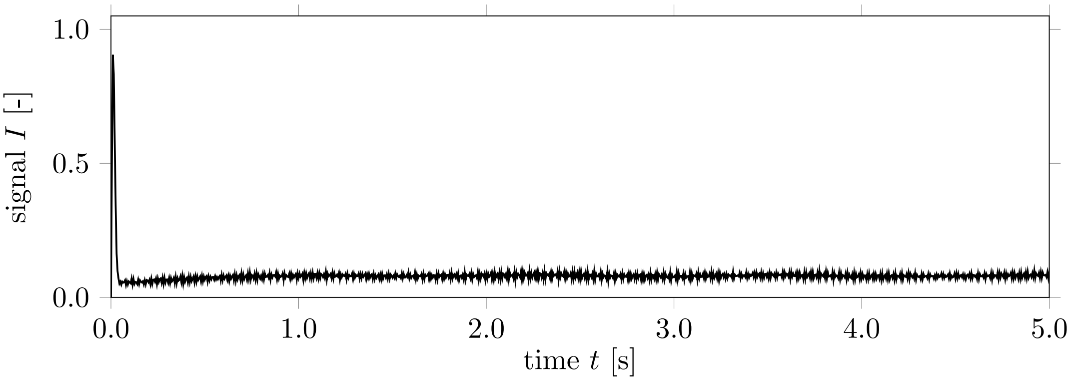
The motor characteristics contain the essential information of a DC permanent magnet motor. This part focuses on the torque-speed and the torque-current curve recorded under a constant operating voltage. The electric drive, mounted on the test bench hardware, was previously tested on another test bench, which one uses for end-of-line measurements. Figure 2.10 shows the results, and one achieves the properties by incrementally increasing the load torque τ_{load}, and acquiring the quasi-stationary values of armature current I, and motor speed ω.

The so-called mechanical characteristic (see Figure 2.10a) shows the dependence between the load torque τ_{load} and the motor speed ω. In short, there is a compromise between how fast the electric drive can accelerate and how much torque it can deliver. The extreme points are, therefore, of particular interest: The motor speed ω_{nl} without a load on the shaft, and the stall torque τ_{stall} where the motor reaches the maximum load while the shaft is no longer rotating.

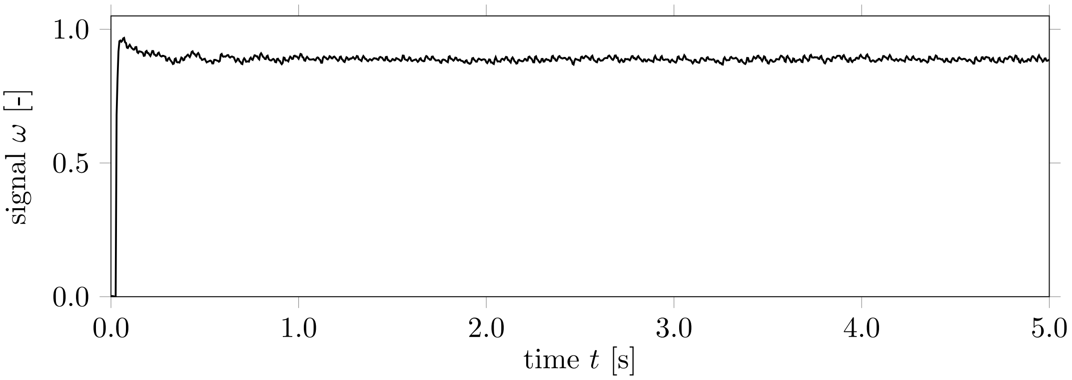
Figure 2.10b represents the electrical characteristic and the maximum value of the motor current I_{max}. If the magnetic saturation of the field poles is reached, the dependence of the current on the torque is linear.

Based on the characteristics shown, the same test is recorded with the test bench hardware and the results are shown in Figure 2.11.

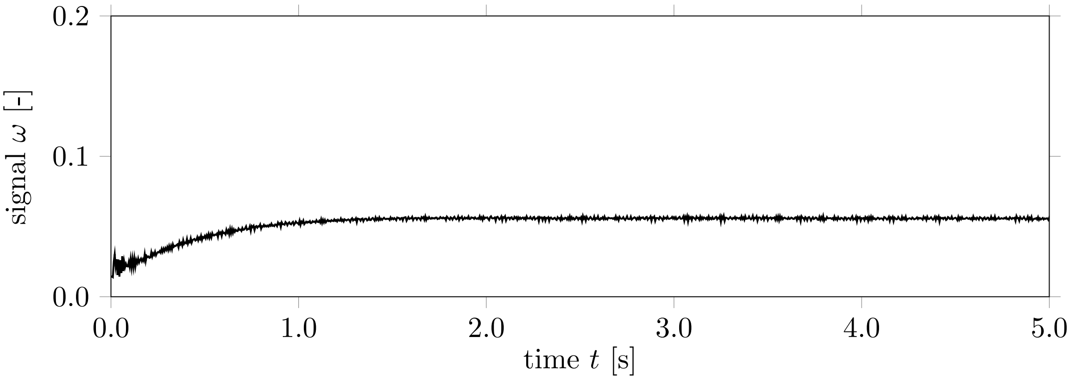
Both characteristics are comparable to the measurement with the standard test bench hardware from the production line. In this measurement, the motor was not operated up to the stall torque. This is partly because the installed brake has a lower nominal load than the motor under investigation. Besides, the magnetic powder brake



(a) Normalized current signal over time.

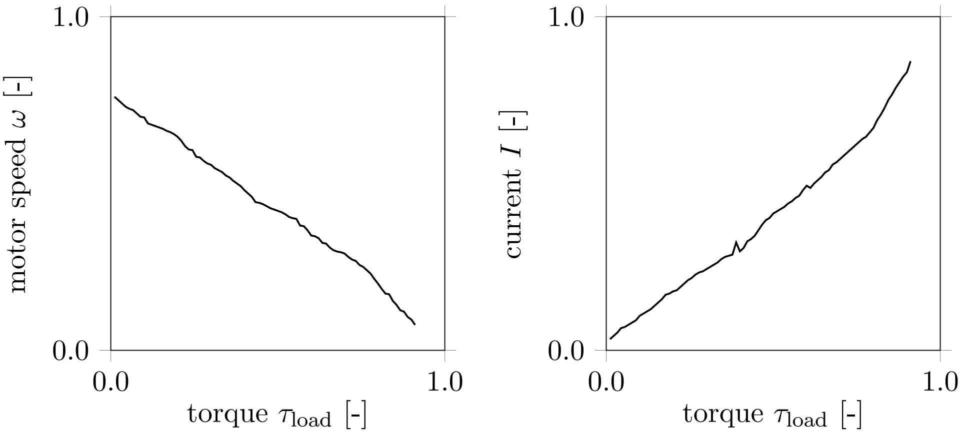


(b) Normalized motor speed signal over time.



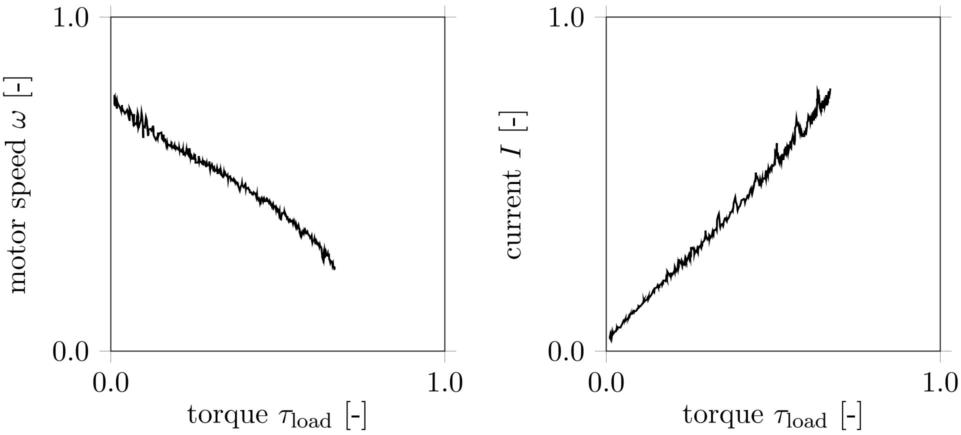
(c) Normalized load torque signal over time.

Figure 2.9: Test bench hardware measurements of a electric drive during the switch-on process with applied filter approach.



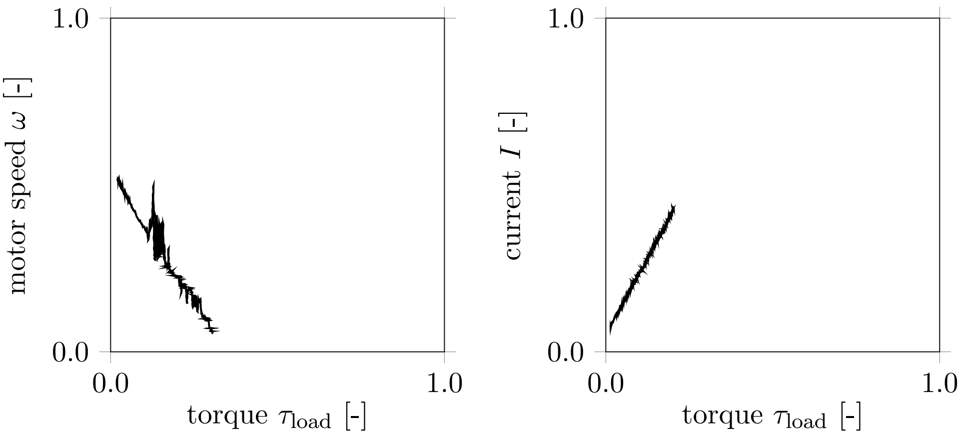
(a) Mechanical motor characteristic. (b) Electrical motor characteristic.

Figure 2.10: Obtained motor characteristics based on test bench hardware from production line.



(a) Mechanical motor characteristic. (b) Electrical motor characteristic.

Figure 2.11: Obtained motor characteristic with $U = 0.84$ by test bench hardware.



(a) Mechanical motor characteristic. (b) Electrical motor characteristic.

Figure 2.12: Obtained motor characteristic with $U = 0.375$ by test bench hardware.

does not block reliably in small speed ranges. This can be indicated if the motor is operated with reduced power. Looking at Figure 2.12, one can observe that the characteristic shows oscillations in the middle range of the applied load torque.

With several load torque values, the electric drive also stimulates the natural frequency of the test bench hardware, as shown in the torque-speed curve. Fortunately, the natural frequency is above the typical load scenarios for the electric drive used. Nevertheless, one has to check the recorded measurements for resonance. Sometimes a resonance may occur, and this happens when the brake powder is compacted, and a corrupt torque load is applied for a short time.

2.3.3 Recording a measurement series

Collecting more than one measurement run requires the comparability of different experiments. Based on the motivation, the test bench hardware is built to reproduce data series from end-of-line measurements. Therefore, the recording of the data must meet certain conditions, and the measurements should be stored correspondingly. In the following, the previous application case of acceleration behavior is used to obtain the measurement series. One sets the time frame to $\Delta t = 5\text{s}$, such that the data covers the start process and a large part of the stationary behavior.

However, it is relatively easy to see and argue that sequential operations of the motor generates thermal power and heats the electric drive. Concerning reproducibility, one should perform each measurement run under consistent boundary conditions. The

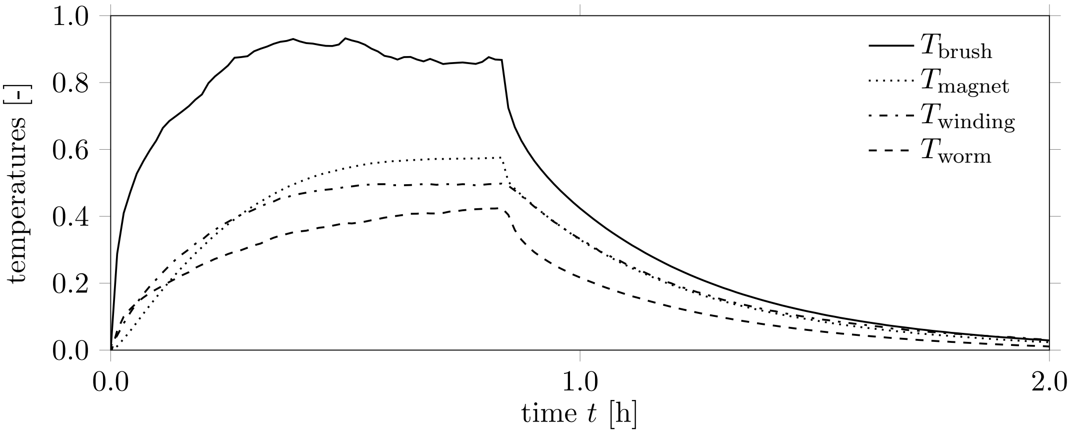


Figure 2.13: Temperature profile for the heating and cooling phase of the electric drive. The signals are recorded with the test bench hardware.

heating of the motor affects the quality of the data set, and one addresses this issue by analyzing the thermal development in detail.

An initial overview is obtained by operating the drive for one hour with the parameter setting, which is given in the following. This means that the voltage $U = 0.9$, the load torque $\tau_{\mathrm{load}} = 0.055$, and the resistance $R = 0.0$ are constant and the temperature profiles of the brush T_{brush}, the magnet T_{magnet}, the winding T_{winding}, and the screw T_{screw} are measured (see Figure 2.13). At this point, it should be mentioned that the resistance parameter of the test bench is additive to the winding resistance of the motor. Looking at the figure, one can see at the beginning of the heating phase and additionally the cooling phase over one hour. It can be assumed that the maximum temperature values have not yet been reached, but that a convergence-like behavior can be observed. From this measurement series, a limited range based on each stationary level is given for each temperature:

$$\begin{aligned}
T_{\mathrm{brush,valid}} &\in [T_{\mathrm{brush,stat}} \pm 0.009], \\
T_{\mathrm{magnet,valid}} &\in [T_{\mathrm{magnet,stat}} \pm 0.005], \\
T_{\mathrm{winding,valid}} &\in [T_{\mathrm{winding,stat}} \pm 0.005], \\
T_{\mathrm{worm,valid}} &\in [T_{\mathrm{worm,stat}} \pm 0.014].
\end{aligned} \tag{2.7}$$

The measurements in the specified range can be recorded by continuously monitoring the temperature levels. One has to perform a warm-up sequence beforehand. The given dynamics of the cooling phase should be used to determine a suitable waiting time between the recording of the runs.

The following introduces an algorithm that allows a series of measurements with parameter variations of the supply voltage U, the load moment τ_{load}, and the resistance R to be automatically recorded. One assumes that one or more of these parameters are uncertain and vary based on a continuous probability distribution. In this setting, Gaussian distributions are selected to describe the predefined variations:

$$\begin{aligned} U_{\text{real}} &= \mathcal{N}(\mu_{\text{U}}, \sigma_{\text{U}}), \\ \tau_{\text{load,real}} &= \mathcal{N}(\mu_{\tau}, \sigma_{\tau}), \\ R_{\text{real}} &= \mathcal{N}(\mu_{\text{R}}, \sigma_{\text{R}}). \end{aligned}$$

Sample pairs for the test bench hardware can then be drawn from the assumed real distributions as follows:

$$(U_{\text{tb}}^{s}, \tau_{\text{load,tb}}^{s}, R_{\text{tb}}^{s}) \in (U_{\text{real}}, \tau_{\text{load,real}}, R_{\text{real}}), \quad s = 1, \ldots, S,$$

These points can be used to start the measurement algorithm. Before triggering a recording, all four temperatures $T_{\circ,\text{meas}}(t)$ must be within the valid set given by Equation (2.7). If one or more temperature values are below the valid range, each uncertain parameter is set to its mean value $(U_{\text{tb}}, \tau_{\text{load,tb}}, R_{\text{tb}}) \in (\mu_{\text{U}}, \mu_{\text{tau}}, \mu_{\text{R}})$ and the electric drive is turned on. During this warm-up period, the temperatures are continuously monitored. When all conditions are met, the motor is stopped for a short time, and a recording of the signals is initiated.

A measurement s starts at the time t_0^s and records the time response of the test bench signals. After Δt, the recording is finished. The electric drive and the magnetic powder brake are then switched off. The resulting data is stored as measurement run s. It contains signal data of the voltage $U_{\text{meas}}^{s}(t)$, the current $I_{\text{meas}}^{s}(t)$, the motor speed $\omega_{\text{meas}}^{s}(t)$, the torque load $\tau_{\text{load,meas}}^{s}(t)$, and the four temperatures $T_{\circ,\text{meas}}^{s}(t)$. All sensor values use the same sampling rate, and therefore a global time vector t^s is appended.

This procedure is repeated S times, and between these measurements, the test bench rests for a few seconds. Depending on the temperature level, another warm-up phase, or, more likely, a cool-down phase may be required before starting the next $s+1$ measurement run to meet the thermal requirements.

Based on this approach, one generates a measurement series with $S = 100$ runs, and one obtains a good overview of the resulting data by considering the mean and standard deviation of the output signals. This statistical information about an exemplary measurement $y_{\text{meas}}(t)$ can be obtained for each time step t as follows:

$$\begin{aligned} \mu(y_{\text{meas}})(t) &= \tfrac{1}{S} \textstyle\sum_{s=1}^{S} y_{\text{meas}}^{s}(t), \\ \sigma(y_{\text{meas}})(t) &= \sqrt{\tfrac{1}{S} \textstyle\sum_{s=1}^{S} \left(y_{\text{meas}}^{s}(t) - \mu\left(y_{\text{meas}}\right)(t)\right)^2}, \quad t \in [t_0^s, t_0^s + \Delta t]. \end{aligned}$$

This calculation is illustrated in Figure 2.14 and the statistical values of the winding temperature $T_{\text{winding,meas}}$, the current I_{meas} and the motor speed ω_{meas} are given. By looking at Figure 2.14a, the stationary level can be observed to be constant over time, and the standard deviation of the signal is within the valid temperature range $T_{\text{worm,valid}}$. More precisely, the extreme values over time must be within the specified range. This requirement is fulfilled, but not shown in the figure. For the given parameter uncertainties, one can expect that the output signals also show deviations in direct comparison to each other which can be observed in the figures 2.14b and 2.14c. It should also be noted that the signals are filtered with the Butterworth filter described in the last subsection. The signal variations are primarily caused by the parameter variations, and the measurement noise has only a small contribution.

2.3.4 Parameter identification

The simulation model described in Section 2.2 is the mathematical description of several variants of the electric drive. One has to adjust the model to the test bench motor, and this is called parameter identification. Much detailed information on this topic is available, e.g., [38, 45]. The following subsection is not intended to give a comprehensive survey of this topic, but rather an overview of the model parameters and the data used for identification and validation. Also, the test bench hardware slightly influences the characteristics of the electric drive, so one has to make small adjustments to the simulation model.

To obtain the optimal model parameters, one assumes that a discrete process $\boldsymbol{f}$ exists that can perfectly reproduce the behavior of the electric motor. The most naive mathematical representation for one output r of this relationship is:

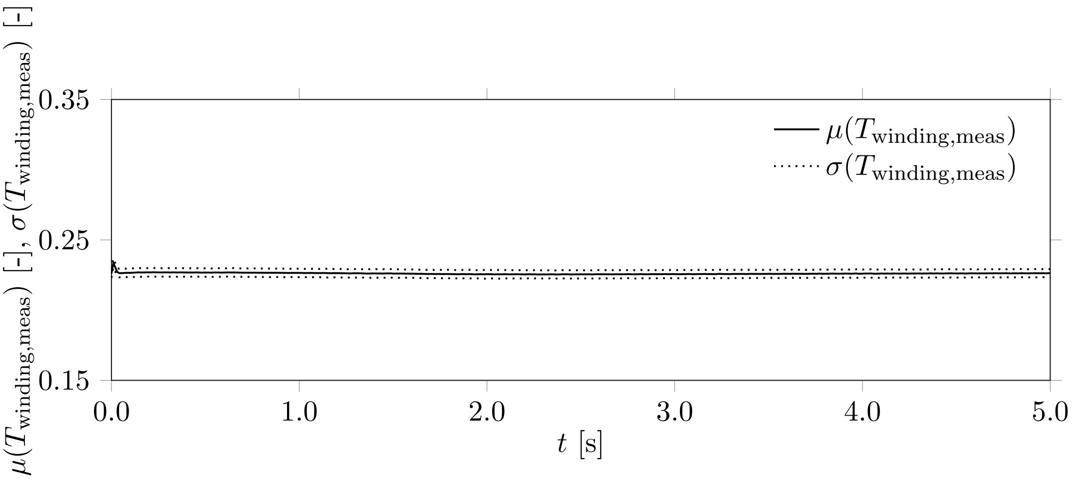
$$\boldsymbol{y}_{\mathrm{r}}(t_k) = \boldsymbol{f}(k, \boldsymbol{\theta}, \boldsymbol{u}(t_k)),$$

where the optimal parameters of the model read as:

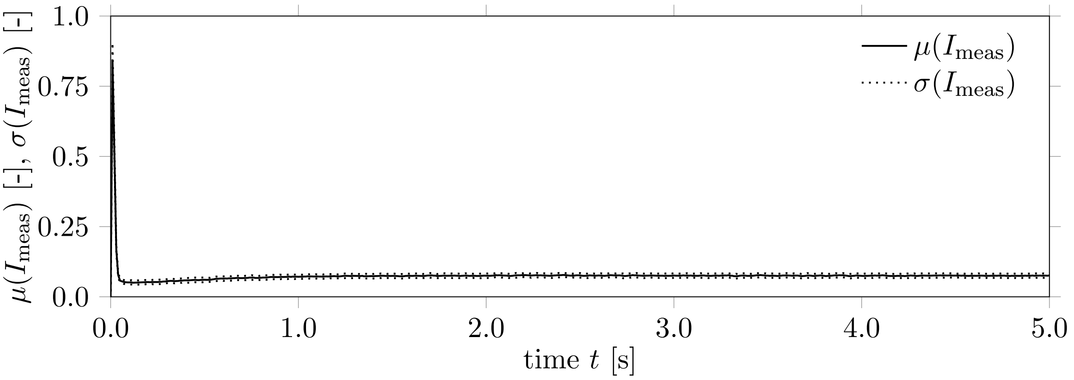
$$\boldsymbol{\theta}^{\mathrm{T}} = (\theta_1, \theta_2, \dots, \theta_N) = \left(\boldsymbol{p}^{\mathrm{elec}^{\mathrm{T}}}, \boldsymbol{p}^{\mathrm{mech}^{\mathrm{T}}}, \boldsymbol{p}^{\mathrm{ther}^{\mathrm{T}}}\right).$$

Measurements enable a comparison with reality, and this can be expressed as $\boldsymbol{y}_{\mathrm{meas}}(t_k)$. The measurement of the optimal process leads to error influences and this corresponds to the optimal assumed process $\boldsymbol{y}_{\mathrm{r}}(t_k)$ with additive errors $\boldsymbol{\eta}(t_k)$ at the outputs. One can easily adapt the model of the motor derived in the Section 2.2 to the discrete process representation as follows:

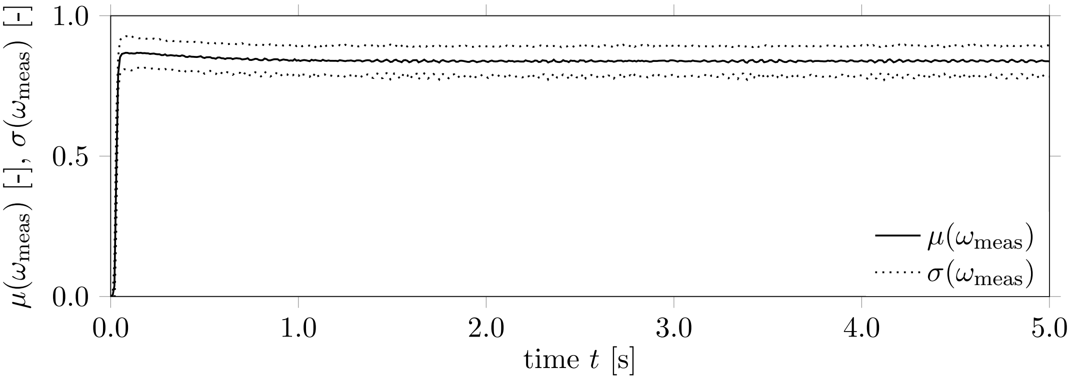
$$\boldsymbol{y}(t_k) = \boldsymbol{\mathcal{M}}(t_k, \bar{\boldsymbol{\theta}}, \boldsymbol{u}(t_k)), \tag{2.8}$$



(a) First and second moment of the winding temperature $T_{\mathrm{winding,meas}}(t)$.



(b) First and second moment of the current $I_{\mathrm{meas}}(t)$.



(c) First and second moment of the motor speed $\omega_{\mathrm{meas}}(t)$.

Figure 2.14: Statistical information on the measurement series for selected signals over time.

where the unknown parameters are denoted as $\bar{\boldsymbol{\theta}} = (\bar{\theta_1}, \bar{\theta_2}, \dots, \bar{\theta_N})^{\mathrm{T}}$.

The basic task of parameter estimation is to find the parameter set $\bar{\boldsymbol{\theta}}$ that comes close to the optimal parameter set $\boldsymbol{\theta}$, with K observations based on the measurements $\boldsymbol{y}_{\mathrm{meas}}(t_k)$. A common approach is to introduce the observation errors between the model and the simulation:

$$\begin{aligned} e_{\mathrm{I}}(t_k) &= y_I(t_k) - y_{\mathrm{I,meas}}(t_k), \\ e_{\omega}(t_k) &= y_{\omega}(t_k) - y_{\omega,\mathrm{meas}}(t_k), \end{aligned} \tag{2.9}$$

and define a cost function based on the sum of the square errors:

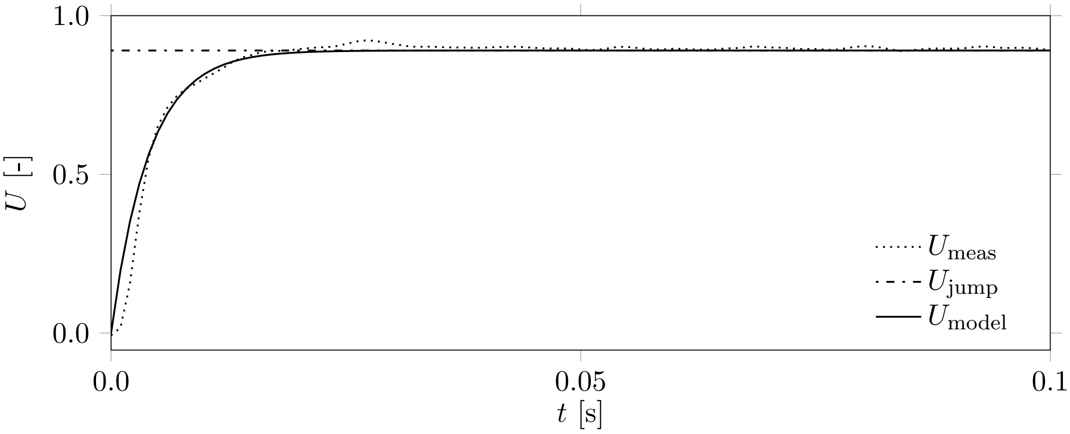
$$E = \sum_{k=0}^{K-1} (e_{\mathrm{I}}(t_k))^2 + \sum_{k=0}^{K-1} (e_{\omega}(t_k))^2 . \tag{2.10}$$

In general, it is necessary to weight the observation errors due to different scales, but in this case, one already receives normalized output signals. The advantage of using such a quadratic cost function is the simplicity associated with the computational effort. Assuming that the additional disturbances are Gaussian noise, the cost function E gives the best estimates of the error variances [38]. A disadvantage could be that large outliers have a strong influence on the total function value, and it is recommended to visualize the error behavior over time.

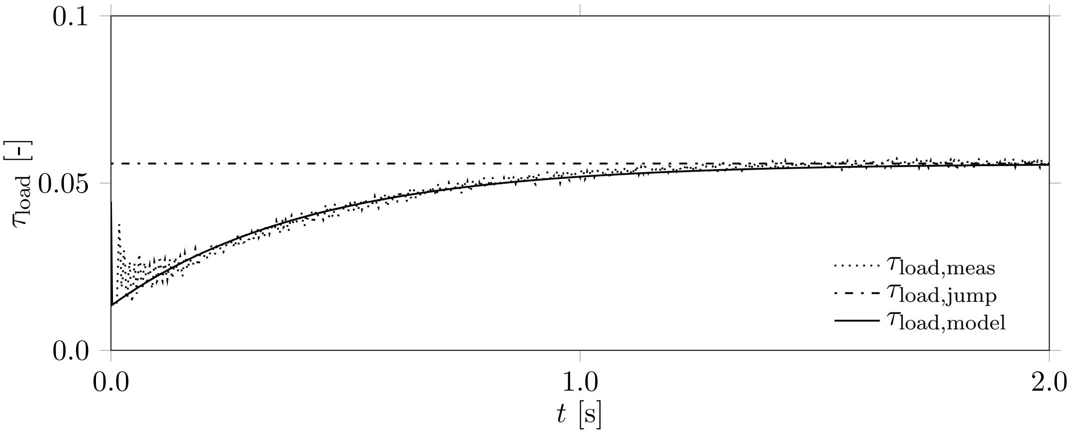
In Subsection 2.3.3 the scenario of switching on the electric drive is measured, and one can theoretically model this behavior by a naive jumping of the input signals from the idle to the desired state. This is shown in Figure 2.15 for the voltage $U(t)$ and the load torque $\tau_{\mathrm{load}}(t)$. The dashed line represents the jumping behavior for both signals, and the solid line shows the measured values. However, this simple approach does not sufficiently model the starting behavior.

Considering Figure 2.15a, one sees that the power supply cannot provide the required constant voltage immediately, and it has inertia at the beginning. This mismatch leads to a considerable error between the measurement and the simulation output, and therefore one proposes an appropriate model. In this case, the timing behavior of the voltage corresponds to a PT1 response of a first-order delay element. If the input signal of this element is a step response, the output behavior has a specific initial slope and approaches asymptotically the final value [46]. One can model the voltage $U_{\mathrm{model}}(t)$ with the following ordinary differential equation:

$$\begin{aligned} T_{\mathrm{U}}\dot{U}_{\mathrm{model}}(t) + U_{\mathrm{model}}(t) &= K_{\mathrm{U}} U_{\mathrm{tb}} H(t), \\ U_{\mathrm{model}}(t_0) &= 0, \qquad\qquad t \in [t_0, t_0 + \Delta t], \end{aligned} \tag{2.11}$$



(a) Signal of the input voltage $U(t)$ over time with focus on the switch on process of the electrical part.



(b) Signal of the input load torque $\tau_{\mathrm{load}}(t)$ over time with focus on the switch on process of the mechanical part.

Figure 2.15: Suitable models for the input signals must be selected based on the test bench measurement data. The solid line shows the actual behavior of the inputs and the dashed line shows the output of the model-based approximation. The dashed line indicates the unsatisfactory behavior of the jump approach previously used.

where the time constant is defined as T_U, the gain factor K_U is set to 1, and $H(t)$ is the Heaviside Step function:

$$H(t) = \begin{cases} 0 & t \leq t_0, \\ 1 & t > t_0. \end{cases}$$

The constant voltage is denoted as U_tb, and the time response is shown in Figure 2.15a.

One observes the similar performance of the startup behavior for the load torque in Figure 2.15b. An unusual characteristic is that the minimum load torque $\tau_\mathrm{load,min}$ of the magnetic powder brake is higher than zero due to constructive reasons. The ordinary differential equation can be considered as:

$$\begin{aligned} T_\tau \dot{\tau}_\mathrm{load,model}(t) + \tau_\mathrm{load,model}(t) &= K_\tau \tau_\mathrm{load,tb} H(t), \\ \tau_\mathrm{load,model}(t_0) &= \tau_\mathrm{load,min}, \qquad\qquad t \in [t_0, t_0 + \Delta t], \end{aligned} \tag{2.12}$$

where the time constant T_τ and the gain factor K_τ are as defined above. The difference is the initial state set to the minimum load moment constant. The model $\tau_\mathrm{load,model}$ can approximate the starting behavior well, as shown in Figure 2.15b.

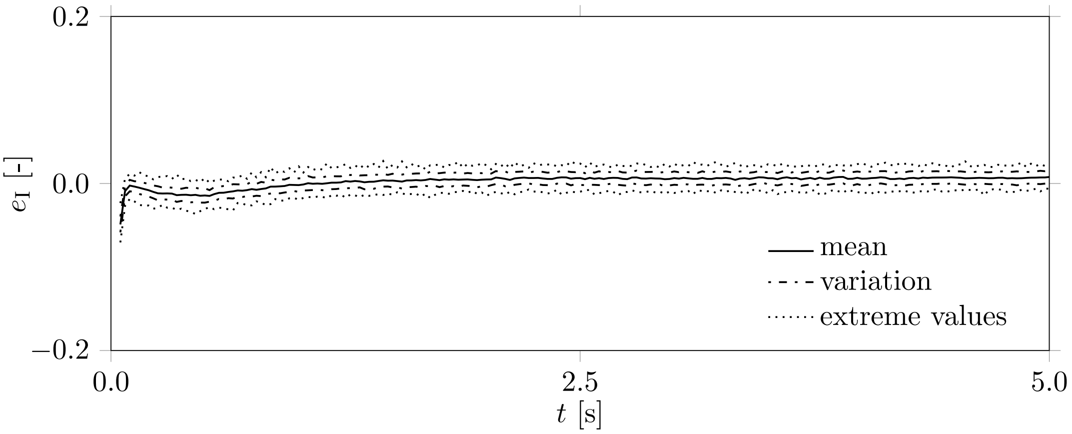
The input models in the equations (2.11) and (2.12) can then be used to obtain a realistic and common signal characteristic that is matched to the test bench hardware. The model of the motor in Equation (2.8) can, therefore, be extended as follows:

$$[y_\mathrm{I}(t_k), y_\omega(t_k)]^\mathrm{T} = \boldsymbol{\mathcal{M}}(t_k, \bar{\boldsymbol{\theta}}, [U_\mathrm{model}(t_k), \tau_\mathrm{load,model}(t_k)]^\mathrm{T}).$$

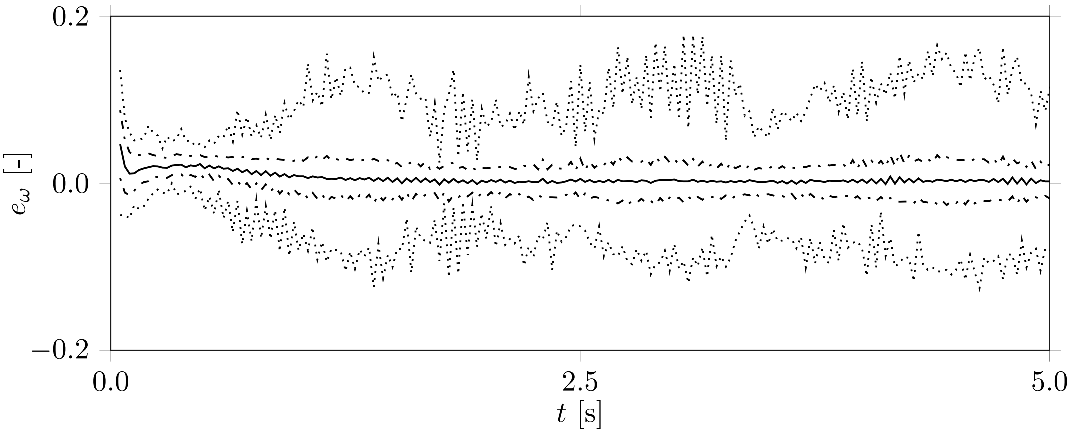
The model parameters $\bar{\boldsymbol{\theta}}$ are identified with a MATLAB [65] package called "lsqnonlin", which can be categorized as a subspace trust region method. It is based on an internally-reflecting Newton method, and details can be found in [15] and [16]. The cost function is defined as described in Equation (2.10) and a measurement run is used for the observed data $\boldsymbol{y}_\mathrm{meas}$.

2.3.5 Model confidence

The procedure described above provides a parameterized simulation model that can predict the quantities of interest realistically for the input space of the used measurement. The assumed uncertainty on the input space leads to a broad operating range, and thus, a functional test is beneficial for the entire measurement series. In this case, the model is treated as a black-box, and no functional relationships within the model are investigated. With this method, it is not possible to consider all conceivable input-output connections. Nevertheless, the model confidence can be increased by using all



(a) Mean and standard deviation of the error e_I with extreme value bounds.



(b) Mean and standard deviation of the error e_ω with extreme value bounds.

Figure 2.16: Statistical overview of the discrepancy of the simulation model for the outputs current I and motor speed ω.

available measurement runs. It should be noted that the number of correct tests makes no statement about the validity of the model.

The function test is performed using the complete measurement series from the last Subsection 2.3.3, and the input-output connection is evaluated with a stochastic measure. A good overview can be achieved with the error measures in Equation (2.9). For each measurement s a corresponding parameter and input set $(U^s_{\mathrm{tb}}, \tau^s_{\mathrm{load,tb}}, R^s_{\mathrm{tb}})$ is known to perform a simulation. The mean and standard deviation of the e_{I} error can be calculated as follows:

$$\begin{aligned}\mu(e_{\mathrm{I}})(t) &= \tfrac{1}{S}\textstyle\sum_{s=1}^{S} y_{\mathrm{I}}(t, U^s_{\mathrm{tb}}, \tau^s_{\mathrm{load,tb}}, R^s_{\mathrm{tb}}) - y_{\mathrm{I,meas}}(t),\\ \sigma(e_{\mathrm{I}})(t) &= \sqrt{\tfrac{1}{S}\textstyle\sum_{s=1}^{S}\Big(y_{\mathrm{I}}(t, U^s_{\mathrm{tb}}, \tau^s_{\mathrm{load,tb}}, R^s_{\mathrm{tb}}) - y_{\mathrm{I,meas}}(t) - \mu\left(e_{\mathrm{I}}\right)(t)\Big)^2},\end{aligned}$$

with $t \in [t^s_0, t^s_0 + \Delta t]$. This can also be applied in a similar way to the e_ω error. The results are visualized in Figure 2.16.

Looking at the mean value of the error for the current e_{I}, one sees that the model provides an excellent approximation of the electric motor over the entire simulation time. Only at the beginning, one can observe a slightly bigger error. This mismatch is at least partly due to a reduced sampling of the measurement data. The error band is very close to the mean value, and one can, therefore, assume that the model can also provide a perfect approximation for the extended test space. This hypothesis is supported by the extreme values shown. There are no significant outliers that could indicate implausible modeling of the quantity of interest I.

In contrast to the current, the motor speed error curve $e_\omega(t)$ initially shows a bias that drops to zero after about two seconds, and this is because the load moment builds up slowly over time. Figure 2.15b shows that it remains stationary after two seconds, and this means that one can only represent the variance of the load moments in the measurement series to a limited extent by the model. Concerning the current, a small standard deviation of the error can also be detected, which leads to a good approximation with the simulation model, especially in the steady-state range. The strong deviations of the extreme values are conspicuous, and this can be explained by looking at the individual measurements. Here, the motor speed may oscillate harmoniously due to certain parameter combinations. As already described, this phenomenon is caused by the magnetic powder brake. In the further course of this work, one excludes measurements with oscillating behavior from the overall series.

3

Forward propagation of uncertainties

The previous Chapter 2 describes the test bench and the corresponding simulation model in detail. This chapter builds on the presented model and focuses on propagating the uncertain parameters through the simulation. Analogous to test bench hardware, one assumes distributions for those parameters. In practice, one uses methods that encapsulate the actual simulation environment and call it several times in a loop. This thesis compares those methods with new approaches that primarily contribute to efficiency improvements. A fast and accurate calculation of the uncertainties at the model output is later the basis for the backward propagation. Beforehand, one uses the test bench hardware to validate the obtained simulation result.

It is common to use Monte Carlo methods [13, 23, 49] to generate samples from a given probability distribution, and therefore, one can evaluate the model without modifying the solver or the problem description. The huge advantage is to use the deterministic simulation environment to get a statement about the stochastic output behavior of the model based on the parameter distribution. The simulation is thus carried out several times at different parameter points, and the stochastic evaluation of the individual simulation results is related to the law of large numbers: The simulation results from a high number of random parameter samples should be close to the arithmetic average [4, 39]. Adding additional evaluations then increases the quality of the stochastic moments, but this also increases the total execution time. Especially when using computing-intensive models, the calculation times limit the maximum number of evaluations.

The simulation tasks can be executed in parallel to solve this problem since the individual model evaluations in the Monte Carlo approach are independent. Extensions like quasi-Monte Carlo [13] or Latin Hypercube sampling [36, 34] achieve further improvements. Those methods perform better or worse, depending on the problem [53, 57]. Nevertheless, dealing with simulation models that have a high computation time is challenging, and one possible option is to simplify the original models. As mentioned earlier, one requirement from an industry perspective is to allow the use of black-box models, which makes standard techniques for model order reduction challenging to apply. Instead, the focus is on so-called surrogate models, which approximate the behavior of the simulation model as closely as possible, and at the same time, their evaluation is, in general, numerically faster. In this context, one assumes uncertain parameters and computes the surrogate model for those conditions. An approach is to use the response surface methodology [9], and the basic idea is to start with a series of design experiments to achieve an optimal result. Another familiar concept is the use of Gaussian processes [1, 54] to fit a surrogate model.

If one considers a model with the general connection between input $\boldsymbol{u}(t)$ and output $\boldsymbol{y}(t)$, one can represent it by the following deterministic assignment:

$$\boldsymbol{y}(t) = \mathcal{M}(t, \boldsymbol{p}, \boldsymbol{u}(t)), \quad t \in [0, T], \tag{3.1}$$

where $\boldsymbol{p} = [p_1, \ldots, p_N] \in \mathbb{R}^N$ defines the vector of the N model parameters and the upper time limit is T. Instead of the two model outputs in the benchmark application, one assumes a vector $\boldsymbol{y} \in \mathbb{R}^Q$ with the output size Q. In the following, one supposes that the model parameters are uncertain, and symbolizes it by the capital letter $\boldsymbol{X}$. In this context, X_i should be random variables with a probability of:

$$Pr(X_i \leq x_i).$$

It is presumed that the cumulative distribution function $F_{X_i}(x_i) = Pr(X_i \leq x_i)$ is a non-negative Lebesgue integrable function of x_i and is defined on a real interval of (a, b) where the following conditions are met: $-\infty \leq a < b \leq \infty$, $F_{X_i}(a) = 0$ and $F_{X_i}(b) = 1$. The probability density function (PDF) $f_{X_i}(x_i)$ on (a, b) can then be specified by:

$$f_{X_i}(x_i) = \begin{cases} \frac{dF_{X_i}(x_i)}{dx_i} & \forall x_i \in (a, b), \\ 0 & \forall x_i \notin (a, b). \end{cases} \tag{3.2}$$

This work uses a Polynomial Chaos Expansion (PCE) [74] to approximate the original simulation model. This method increases the calculation efficiency of the stochastic moments regarding the relevant output values. The approach works well when it comes to probabilistic uncertainties in the model parameters, and the method then represents the evaluation of the uncertainty in the dynamic system. One applies different techniques for the adjustment of the surrogate model coefficients, and one closes with a brief comparison concerning the benchmark problem.

In the following Section 3.1 the basic idea of PCE is introduced, and methods for the calculation of the corresponding coefficients are presented. A sparse grid approach leads to a further improvement in the computational costs. The addressed procedure is demonstrated with a simple application example, and a suitable measure for the comparison of the obtained results with a reference is presented in Section 3.2. Based on the knowledge gained, the methods are applied to the benchmark problem (see Section 3.3), and suitable output quantities, which are referred to in the remainder as the quantity of interests (QoIs), are presented. Finally, Section 3.4 compares the simulation-based results with the measurements performed by the test bench hardware of the electric drive.

3.1 Polynomial Chaos expansion

Wiener [72] introduced the first idea of PCE decomposition in the 1930s, and he used Hermite polynomials for random variables to represent Gaussian processes. Based on his definition, one can consider the PC as a functional approximation of a given random variable by the basic functions of a finite number of other random variables. More precisely, a general second-order random process $X(\theta)$ can be represented in the form of orthogonal polynomials with a discrete description:

$$\begin{aligned} X(\theta) &= A_0 H_0 \\ &+ \sum_{i_1}^{\infty} A_{i_1} H_1(\xi_{i_1}(\theta)) \\ &+ \sum_{i_1}^{\infty} \sum_{i_2}^{i_1} A_{i_1 i_2} H_2(\xi_{i_1}(\theta), \xi_{i_2}(\theta)) \\ &+ \sum_{i_1}^{\infty} \sum_{i_2}^{i_1} \sum_{i_3}^{i_2} A_{i_1 i_2 i_3} H_2(\xi_{i_1}(\theta), \xi_{i_2}(\theta), \xi_{i_3}(\theta)) \\ &+ \dots, \end{aligned} \tag{3.3}$$

where H_n are Hermite polynomials based on Gaussian variables $\boldsymbol{\xi} = (\xi_{i_1}(\theta), \xi_{i_2}(\theta), \dots)$ with zero mean and unit variance. This representation has a spectral convergence property if the given random variable $X(\theta)$ is smooth and is finite-dimensional. The number of terms is determined by the size of X and the truncated polynomial degree. It is associated with the requirements of the random variable, such as the property that the square integrability is guaranteed for the associated probability space. An extension for non-Gaussian measurements was published by Cameron et al. [14] and the development in the engineering field was promoted by Ghanem et al. [30]. Xiu et al. [74] adapted the method to make it applicable to non-Gaussian random processes and called it generalized Polynomial Chaos (gPC).

In this section, the focus of the discussion is on gPC, and the mathematical framework of the described methods is mainly quoted from [74, 43, 62]. Starting from the definition in the one-dimensional case, the extension to the multidimensional case is presented, and suitable procedures for the determination of the gPC coefficients are shown.

3.1.1 Definition in the one-dimensional case

One assume a real-valued random variable $X(\theta)$ with a probability space $(\Theta, 2^{\Theta}, \mathcal{P})$, where Θ is the sample space, 2^{Θ} is the σ-algebra and $\mathcal{P}$ is some probability measure. More detailed information on the assumptions of the probability space can be found in [59]. The random variable can be treated as a function of θ, which can be considered a sample of Θ, and based on Equation (3.3) the one-dimensional case with the more general Wiener-Askey PCE [74] can be written as follows:

$$X(\theta) = \sum_{i}^{\infty} c_i \phi_i(\xi(\theta)), \tag{3.4}$$

where c_i are the spectral coefficients, and $\{\phi_i\}_{i=0}^{\infty}$ denote the Wiener-Askey polynomials in terms of the so-called hyperparameter $\xi(\theta)$ defined on the support Θ. The dependence of the hyperparameter ξ on θ should be dropped to simplify the notation.

The spectral stochastic representation can be seen as a Fourier-like decomposition [43], and the orthogonality relation of the basis functions can be expressed as:

$$\langle \phi_j(\xi), \phi_k(\xi) \rangle = \mathbb{E}\left[\phi_j(\xi)\phi_k(\xi)\right] = \mathbb{E}\left[\phi_k^2(\xi)\right] \delta_{jk}, \quad j, k \in \mathbb{N}_0, \tag{3.5}$$

with the inner product in the Hilbert space of the variables $\langle \cdot, \cdot \rangle$, and the Kronecker delta δ_{jk} defined as:

$$\delta_{jk} = \begin{cases} 1 & j = k, \\ 0 & j \neq k. \end{cases}$$

Table 3.1: Correspondence between the type of gPC basis and probability distribution for the continuous and discrete case [74].

PDF of random variable	gPC basis polynomials	Support
Beta	Jacobi	$[a, b]$
Gamma	Laguerre	$[0, \infty]$
Gaussian	Hermite	$(-\infty, \infty)$
Uniform	Legendre	$[a, b]$
Poisson	Charlier	$[0, 1, 2, \dots]$
Binomial	Krawtchouk	$[0, 1, 2, \dots, n]$
Negative Binomial	Meixner	$[0, 1, 2, \dots]$
Hypergeometric	Hahn	$[0, 1, 2, \dots, n]$

One assumes that the cumulative distribution function $F_{\mathrm{X}}(\xi)$ of the random variable X is absolutely continuous with respect to the Lebesque measure. By use of the definition in Equation (3.2), the corresponding PDF exists, such that the relation $dF_{\mathrm{X}}(\xi) = f_{\mathrm{X}}(\xi)d\xi$ holds on the support Θ. The inner product of Equation (3.5) can be modified in the following way:

$$\langle \phi_j(\xi), \phi_k(\xi) \rangle = \int_{\Theta} \phi_j(\xi)\phi_k(\xi) f_{\mathrm{X}}(\xi) d\xi, \quad j, k \in \mathbb{N}_0. \tag{3.6}$$

Depending on the standardization of the polynomials $\phi_j(\xi)$, a different weighting function than the PDF can be used, and therefore this part will be referred to as $w(\xi)$ in the following. One can also use this procedure in the discrete case analogously. Table 3.1 contains the correspondence between the random variable distribution, and the type of generalized PC basis polynomials, including the support Θ for the hyperparameter ξ.

3.1.2 Extension to the multi-dimensional case

Regarding the model in Equation (3.1), one has to consider multiple random parameters, and therefore, it is assumed that the hyperparameter ξ can be extended to a random vector $\boldsymbol{\xi} = \{\xi_1, \dots, \xi_N\}$ with independent components. According to the model, one names the considered stochastic process Y, like the output vector. In this case, a scalar is used without loss of generality. The vector case can be obtained by applying the approach component-wise using the same polynomial base for each component.

Let $\boldsymbol{i} = \{i_1, \ldots, i_N\} \in \mathbb{N}_0^N$ denotes a multi-index with $|\boldsymbol{i}| = i_1 + \cdots + i_N$. Based on Equation (3.4), and the assumption that the parameters were supposed to be independent, the stochastic process $Y(\boldsymbol{\xi})$ can be defined as follows:

$$Y(\boldsymbol{\xi}) = \sum_{|\boldsymbol{i}|=0}^{\infty} y_{\boldsymbol{i}} \psi_{\boldsymbol{i}}(\boldsymbol{\xi}), \tag{3.7}$$

where $y_{\boldsymbol{i}}$ are the unknown deterministic PC coefficients and $\psi_{\boldsymbol{i}}$ are the multivariate polynomials. The series in Equation (3.7) is often associated with the term PCE in literature [74, 8].

The use of a random vector for the hyperparameters $\boldsymbol{\xi}$ results in a joint PDF, due to the independence of the random variables, which can be written in the following way:

$$f_{\boldsymbol{X}}(\boldsymbol{\xi}) = \prod_{j=1}^{N} f_{\mathrm{X_j}}(\xi_j), \tag{3.8}$$

where $f_{\mathrm{X_j}}(\xi_j)$ is the marginal PDF of the random variable X_j. One can assume a set of orthogonal polynomials $\Phi_k^{(j)}$ for $k \in \mathbb{N}$ with respect to $f_{\mathrm{X_j}}$ in terms of Equation (3.5):

$$\langle \Phi_k^{(j)}(\xi_j), \Phi_l^{(j)}(\xi_j) \rangle = \mathbb{E}[\Phi_k^{(j)}(\xi_j)\Phi_l^{(j)}(\xi_j)] = \delta_{kl},$$

where δ_{kl} again is the Kronecker delta. The polynomials are subsequently itemized such that N sets of univariate polynomials with various degrees are obtained, and the N-variate gPC basis functions can be developed as:

$$\psi_{\boldsymbol{i}}(\boldsymbol{\xi}) = \prod_{j=1}^{N} \Phi_{i_j}^{(j)}(\xi_j), \quad \boldsymbol{i} = \{i_1, \ldots, i_N\}.$$

The orthogonality relation from Equation (3.5) then takes the form:

$$\begin{aligned} \langle \psi_{\boldsymbol{j}}(\boldsymbol{\xi}), \psi_{\boldsymbol{k}}(\boldsymbol{\xi}) \rangle &= \mathbb{E}[\psi_{\boldsymbol{j}}(\boldsymbol{\xi})\psi_{\boldsymbol{k}}(\boldsymbol{\xi})] = \mathbb{E}[\psi_{\boldsymbol{j}}^2(\boldsymbol{\xi})]\delta_{\boldsymbol{jk}} \\ &= \textstyle\int_{\Theta^N} \psi_{\boldsymbol{j}}(\boldsymbol{\xi})\psi_{\boldsymbol{k}}(\boldsymbol{\xi}) w(\boldsymbol{\xi}) d\boldsymbol{\xi}, \end{aligned} \tag{3.9}$$

for $j \neq k$, and the discrete case can be written as:

$$\langle \psi_{\boldsymbol{j}}(\boldsymbol{\xi}), \psi_{\boldsymbol{k}}(\boldsymbol{\xi}) \rangle = \sum_{\boldsymbol{\xi}} \psi_{\boldsymbol{j}}(\boldsymbol{\xi})\psi_{\boldsymbol{k}}(\boldsymbol{\xi}) w(\boldsymbol{\xi}).$$

As mentioned in the previous one-dimensional part, the weighting function $w(\boldsymbol{\xi})$ corresponds to the selected PC base. See [74] for more information. In addition, there is a one-to-one correspondence between the multi-index i and a one-dimensional index j. Based on Equation (3.7), the mapping results:

$$Y(\boldsymbol{\xi}) = \sum_{j=0}^{\infty} y_j \psi_j(\boldsymbol{\xi}), \quad \text{for } j \neq k. \tag{3.10}$$

Given the infinite summation in Equation (3.10), an approximation $Y^P(\boldsymbol{\xi})$ must be proposed to address the problem concerning numerical computation. One approach leads to the truncation of the sum at the finite term P:

$$Y(\boldsymbol{\xi}) \approx Y^P(\boldsymbol{\xi}) = \sum_{j=0}^{P} y_j \psi_j(\boldsymbol{\xi}). \tag{3.11}$$

The total number of expansion terms $(P+1)$ is influenced by the dimension N of the random vector $\boldsymbol{\xi}$, and the total order p of the multivariate polynomials $\psi_j(\boldsymbol{\xi})$. This can be defined by the following relation:

$$(P+1) = \frac{(N+p)!}{N!p!}. \tag{3.12}$$

Due to the orthogonality of the basis polynomials, the mean value $\mu_{\mathrm{Y^P}}$ and the variance $\sigma^2_{\mathrm{Y^P}}$ can be derived from the given representation:

$$\begin{aligned} \mu_{\mathrm{Y^P}} &= \mathbb{E}\left[Y^P(\boldsymbol{\xi})\right] = \mathbb{E}\left[\textstyle\sum_{j=0}^{P} y_j \psi_j(\boldsymbol{\xi})\right] = \textstyle\sum_{j=0}^{P} y_j \mathbb{E}\left[\psi_j(\boldsymbol{\xi})\right] \\ &= \textstyle\sum_{j=0}^{P} y_j \langle \psi_j(\boldsymbol{\xi}), \psi_0(\boldsymbol{\xi}) \rangle = y_0, \end{aligned}$$

and

$$\begin{aligned} \sigma^2_{\mathrm{Y^P}} &= \mathrm{Var}\left[Y^P(\boldsymbol{\xi})\right] = \mathbb{E}\left[\left(Y^P(\boldsymbol{\xi}) - \mu_{\mathrm{Y^P}}\right)^2\right] = \mathbb{E}\left[\left(\textstyle\sum_{j=0}^{P} y_j \psi_j(\boldsymbol{\xi}) - y_0\right)^2\right] \\ &= \mathbb{E}\left[\left(\textstyle\sum_{j=1}^{P} y_j \psi_(\boldsymbol{\xi})\right)^2\right] = \textstyle\sum_{j=1}^{P} \sum_{k=1}^{P} y_j y_k \mathbb{E}\left[\psi_j(\boldsymbol{\xi}) \psi_k(\boldsymbol{\xi})\right] \\ &= \textstyle\sum_{j=1}^{P} \sum_{k=1}^{P} y_j y_k \langle \psi_j(\boldsymbol{\xi}) \psi_k(\boldsymbol{\xi}) \rangle = \textstyle\sum_{j=1}^{P} y_j^2 \mathbb{E}\left[\psi_j^2(\boldsymbol{\xi})\right], \end{aligned}$$

To fit the model structure in Equation (2.1), the random variable approach must be extended. As already mentioned, this can be done by component-wise applying the approach to the truncated random vector $\boldsymbol{Y}^P$. This results in the following notation:

$$\boldsymbol{Y}^P(\boldsymbol{\xi}) = \sum_{j=0}^{P} \boldsymbol{y}_j \psi_j(\boldsymbol{\xi}), \tag{3.13}$$

where $\boldsymbol{y}_j = [y_{1j}, \ldots, y_{Nj}]^{\mathrm{T}} \in \mathbb{R}^N$ are the coefficients of the j-th PC basis.

A further adaptation of the PCE is the extension to a truncated random process over time, $\boldsymbol{Y}^P(t, \boldsymbol{\xi})$, and this can be achieved by splitting the random process into a deterministic and a stochastic part. The PCE of the random process can be considered to be a generalization as applied before, and therefore, Equation (3.13) can be modified:

$$\boldsymbol{Y}^P(t, \boldsymbol{\xi}) = \sum_{j=0}^{P} \boldsymbol{y}_j(t) \psi_j(\boldsymbol{\xi}), \tag{3.14}$$

where the deterministic part $\boldsymbol{y}_j(t)$ is the coefficients of the j-th PC base over time, and can be referred to as stochastic modes of the model.

3.1.3 Determination of the PC coefficients

This subsection introduces approaches for calculating the PC coefficients to be determined so that the PCE can approximate the results of the model. This work focuses on methods that one can use for black-box models. One often classifies them as non-intrusive, but the distinction between intrusive and non-intrusive is not clearly defined [31].

The methods presented are only an extract from the available approaches in the literature. Based on the above requirements from the model, the focus is on non-intrusive spectral projection, which is partly extracted from Eldred et al. [21].

Non-intrusive spectral projection

The Galerkin projection [17, 43] introduces the truncated spectral expansions into the model equations, and the PC coefficients are determined so that the residual is orthogonal to the polynomial basis. The non-intrusive spectral projection (NISP) projects the orthogonal basis function directly against the PCE of the model outputs.

Based on PCE in Equation (3.14) the NISP method projects the model output $\boldsymbol{Y}^P(t,\boldsymbol{\xi})$ against the set of basis functions $\psi_k(\boldsymbol{\xi})$ using the inner product:

$$\langle \boldsymbol{Y}^P(t,\boldsymbol{\xi}), \psi_k(\boldsymbol{\xi})\rangle = \left\langle \sum_{j=0}^{P} \boldsymbol{y}_j(t)\psi_j(\boldsymbol{\xi}), \psi_k(\boldsymbol{\xi}) \right\rangle .$$

The orthogonal property in Equation (3.9) provides the following relation:

$$\begin{aligned} \mathbb{E}\left[\boldsymbol{Y}^P(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right] &= \textstyle\sum_{j=0}^{P} \mathbb{E}\left[\boldsymbol{y}_j(t)\psi_j(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right] \\ &= \textstyle\sum_{j=0}^{P} \boldsymbol{y}_j(t)\mathbb{E}\left[\psi_j(\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right] \\ &= \textstyle\sum_{j=0}^{P} \boldsymbol{y}_j(t)\mathbb{E}\left[\psi_j^2(\boldsymbol{\xi})\right]\delta_{jk}. \end{aligned}$$

The coefficients of the PCE can then be obtained due to orthogonality as:

$$\boldsymbol{y}_j(t) = \frac{\mathbb{E}\left[\boldsymbol{Y}^P(t,\boldsymbol{\xi})\psi_j(\boldsymbol{\xi})\right]}{\mathbb{E}\left[\psi_j^2(\boldsymbol{\xi})\right]}, \quad \text{for } j = 0, \dots, P,$$

where $\mathbb{E}\left[\psi_j^2(\boldsymbol{\xi})\right]$ can be considered as a normalizing factor which can be analytically determined with respect to the selected base polynomials. Getting a good approximation for the numerator is the most challenging part, and is associated with the solution of the following integral over the parameter space Θ^N:

$$\mathbb{E}\left[\boldsymbol{Y}^P(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right] = \int_{\Theta^N} \boldsymbol{Y}^P(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})w(\boldsymbol{\xi})d\boldsymbol{\xi}, \tag{3.15}$$

where in this case $\boldsymbol{Y}^P(t,\boldsymbol{\xi})$ can be considered as the evaluation of the stochastic process $\boldsymbol{Y}(t,\boldsymbol{\xi})$ at a certain point in the parameter space $\boldsymbol{\xi}$.

Given the complex function $\boldsymbol{Y}(t,\boldsymbol{\xi})$, it is difficult to analytically evaluate the integral in Equation (3.15) especially when the parameter space Θ^N is large. For this reason, one usually approximates the multi-dimensional integral by use of numerical methods that one can find in many disciplines, and one can reference several approaches in the literature. When focusing on a large parameter space, the number of function evaluations within a discrete method depends strongly on the dimension N of the parameter space. Using a one-dimensional integral, the number of required evaluations increases by the N-th power. In this subsection, one introduces common methods related to the UQ framework. This investigation includes the standard approaches: Monte Carlo integration, deterministic quadrature, and the sparse grid cubatures for NISP to handle larger parameter dimensions N.

Monte Carlo integration

Monte Carlo is a technique for numerical integration with random numbers. In particular, the method is based on randomly selected points of the parameters at which the integration is evaluated [55]. Let $\boldsymbol{\xi}^s = [\xi_1^s, \ldots, \xi_N^s] \in \Theta^N$ be a sample for s of the parameter space generated from the joint PDF $f_{\boldsymbol{X}}(\boldsymbol{\xi})$ based on the Equation (3.8). Then the integral in Equation (3.15) with the given complex stochastic process can be approximated as follows:

$$\begin{aligned} \mathbb{E}\left[\boldsymbol{Y}(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right] &= \textstyle\int_{\Theta^N} \boldsymbol{Y}(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})w(\boldsymbol{\xi})d\boldsymbol{\xi} \\ &\approx \tfrac{1}{S_{\mathrm{MC}}} \textstyle\sum_{s=1}^{S_{\mathrm{MC}}} \boldsymbol{Y}(t,\boldsymbol{\xi}^s)\psi_k(\boldsymbol{\xi}^s)w(\boldsymbol{\xi}^s), \end{aligned} \tag{3.16}$$

where S_{MC} is the number of Monte Carlo samples. If $\mathbb{E}\left[\boldsymbol{Y}(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right]$ exists, then the weak law of large numbers implies that for any arbitrarily small ϵ:

$$\lim_{S_{\mathrm{MC}}\to\infty} Pr\left(\left|\frac{1}{S_{\mathrm{MC}}}\sum_{s=1}^{S_{\mathrm{MC}}} \boldsymbol{Y}(t,\boldsymbol{\xi}^s)\psi_k(\boldsymbol{\xi}^s)w(\boldsymbol{\xi}^s) - \mathbb{E}\left[\boldsymbol{Y}(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right]\right| \geq \epsilon\right) = 0.$$

In other words, if the number of sample points S_{MC} used is large enough, then there is a small probability that the approximation deviates much from the investigated integral. The strong law of large numbers is similar in this case. As long as S_{MC} is large enough, the approximation resulting from a Monte Carlo integration, is close to $\mathbb{E}\left[\boldsymbol{Y}(t,\boldsymbol{\xi})\psi_k(\boldsymbol{\xi})\right]$ as desired. One can denote the approximation error as $(S_{\mathrm{MC}})^{-\frac{1}{2}}$ [23], and it is independent of the N dimension of the parameter space. This characteristic of Monte Carlo integration is a significant advantage over the methods below in terms of large parameter space. However, the convergence rate $\mathcal{O}\left((S_{\mathrm{MC}})^{-\frac{1}{2}}\right)$ is comparatively moderate. The application of this method can be found in studies of Debusschere [18].

Deterministic quadrature

This method uses numerical quadrature [21, 18] to evaluate the multi-dimensional integral in Equation (3.16). The simplest way to approximate these multi-dimensional integrals is to use a tensor product of one-dimensional quadrature rules, e.g., Gauss-Hermit, Gauss-Legendre, Gauss-Laguerre, and Gauss-Jacobi [22].

One can now assume a sequence of one-dimensional quadrature operators:

$$\mathcal{U}_i(f)(\boldsymbol{\xi}) = \sum_{j=1}^{S_Q^i} f(\xi_i^j) w_i^j, \tag{3.17}$$

where for each $i \in \{1, \dots, N\}$ a sequence $\{\xi_i^1, \dots, \xi_{\mathrm{i}}^{S_Q^i}\}$ for the quadrature on Θ_i exists. The number of quadrature points S_{Q}^i is fixed and one assume that f is continuous on Θ_i. One of the most significant rules is the Gaussian quadrature. Applied to Equation (3.17) all polynomials with degree less or equal to $2S_Q^i - 1$ are integrated exactly for each $i = 1, \dots, N$. Concentrating on the integral in Equation (3.15) the highest order of $\boldsymbol{Y}^P(t, \boldsymbol{\xi})$ and $\psi_k(\boldsymbol{\xi})$ are P each, and therefore the integral involves polynomials with at least order $2P$ in each dimension. To achieve good accuracy with Gaussian quadrature, a minimum order of $P + 1$ is recommended.

For the multivariate case, the full tensor product quadrature formula can be written using the multi-index $\boldsymbol{i}$ as:

$$\begin{aligned} \mathcal{Q}_{\boldsymbol{i}}^N(f)(\boldsymbol{\xi}) &= (\mathcal{U}_{i_1} \otimes \cdots \otimes \mathcal{U}_{i_N})\,(f)(\boldsymbol{\xi}) \\ &= \sum_{j_1=1}^{S_Q^{i_1}} \cdots \sum_{j_N=1}^{S_Q^{i_N}} f(\xi_{i_1}^{j_1}, \dots, \xi_{i_N}^{j_N}) \left(w_{i_1}^{j_1} \otimes \cdots \otimes w_{i_N}^{j_N} \right), \end{aligned} \tag{3.18}$$

where, based on this general description, one can assume that the order of all random parameters is S_Q for the sake of simplicity. Table 3.2 shows the correspondence between probability distributions and quadrature formulas, including weights. Then, it is easy to see that the evaluation of the above equation requires $(S_Q)^N$ evaluations. This representation means that the number of grid points for models with a small parameter space is very low, and the procedure is very efficient. For higher dimensions of N, the grid points grow exponentially fast, and other approaches such as sparse grids are better suited.

Sparse grid cubatures

In order to treat high-dimensional parameter spaces with the NISP method, sparse

Table 3.2: Correspondence between the probability distribution and the quadrature rule.

PDF of random variable	Quadrature	Weight	Support
Beta	Gauss-Jacobi	$(1-\xi)^{\alpha}(1+\xi)^{\beta}$	$[-1, 1]$
Gamma	Gauss-Laguerre	$e^{-\xi}$	$[0, \infty]$
Gaussian	Gauss-Hermite	$e^{-\xi^2}$	$(-\infty, \infty)$
Uniform	Gauss-Lengendre	1	$[-1, 1]$

grids are proposed in this context [73]. These were introduced by Smolyak [60] and are used for the first time in the context of multivariate integration [28, 51]. The following summary is mainly based on [50] and should give a short overview. The Smolyak algorithm is a linear combination of product formulas and is based on a one-dimensional interpolation formula as given in Equation (3.17), and then the difference quadrature formula can be defined by as:

$$\Delta_k(f)(\boldsymbol{\xi}) = (\mathcal{U}_k - \mathcal{U}_{k-1})\,(f)(\boldsymbol{\xi}),$$

with $\mathcal{U}_0(f)(\boldsymbol{\xi}) = 0$, and the multi-dimensional case can be obtained by Smolyak's construction:

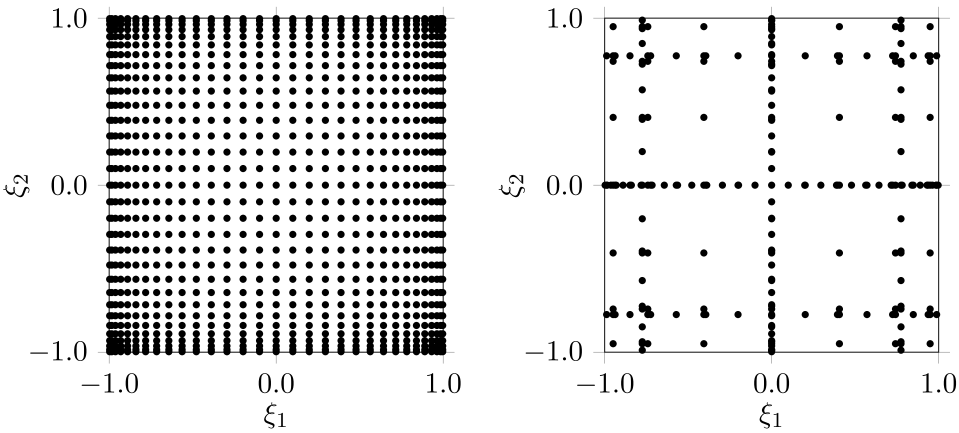
$$\mathcal{Q}_{\boldsymbol{i}}^{N}(f)(\boldsymbol{\xi}) = \sum_{j=1}^{N} \sum_{|\boldsymbol{k}| \le i_j + N - 1} (\Delta_{k_1} \otimes \cdots \otimes \Delta_{k_N})\, f(\boldsymbol{\xi}),$$

where $\boldsymbol{k} \in \mathbb{N}^N$, and the tensor product of multiple quadrature formulas are defined as the sum of all possible combinations, as demonstrated in Equation (3.18). Smolyak's formula then can be written as:

$$\begin{aligned} \mathcal{Q}_{\boldsymbol{i}}^{N}(f)(\boldsymbol{\xi}) = \; & \textstyle\sum_{j=1}^{N} \sum_{i_j \le |\boldsymbol{k}| \le i_j + N - 1} h_{ijk}\, (\mathcal{U}_{k_1} \otimes \cdots \otimes \mathcal{U}_{k_N})\,(f)(\boldsymbol{\xi}), \\ & \text{with } h_{ijk} = (-1)^{i_j + N - |\boldsymbol{k}| - 1} \begin{pmatrix} N-1 \\ |\boldsymbol{k}| - i_j \end{pmatrix}. \end{aligned}$$

Besides the algorithm with Smolyak's construction and the combination technique, one has to reduce the function evaluations and to improve the given grids of the one-dimensional quadrature formulas $\mathcal{U}_i(f)(\boldsymbol{\xi})$. One assumes the one-dimensional difference grid:

$$\Omega_i = \Gamma_i^1 \backslash \Gamma_{i-1}^1$$



(a) Full grid with 961 nodes.

(b) Sparse grid with 221 nodes.

Figure 3.1: Comparison of two-dimensional grids based on the same one-dimensional grid points.

with a underlying grid $\Gamma_i^N = \{\boldsymbol{\xi}_i^j : 1 \leq j \leq S_{\mathrm{G}}^i\} \in \Theta^N$ and $\Gamma_0^N = \emptyset$. For the non-nested case one can set $\Omega_i = \Gamma_i^1$. The multivariate formula of Smolyak then forms a so-called sparse grid:

$$\Gamma_i^N = \bigcup_{|\boldsymbol{k}| \leq i+N-1} (\Omega_{k_1} \otimes \cdots \otimes \Omega_{k_N}),$$

where the grid is given by the union over the pairwise disjoint nodal sets Ω_i.

Figure 3.1 shows a comparison between the full grid and the presented sparse grid approach. In this case, a two-dimensional parameter space is assumed, and the number of grid points for the full grid is almost five times greater than for the sparse version. It can also be observed that the grid based on the Smolyak algorithm is a subset of the full grid. This provides an opportunity for higher dimensional random parameter spaces in combination with the PC expansion, and the application of sparse grids can be found in [3, 73, 21, 42, 59].

3.2 Investigation with a simplified simulation model

This section explains how to use the introduced gPC method using a simplified simulation model. In addition to a basic understanding of the model equation dependencies, the simulation time is also considerably short, and therefore, many evaluations can be

Table 3.3: Parameter distributions for the simple simulation model.

Parameter	Tolerance range r_t	Mean value μ	Standard deviation σ
L_p	±10%	$1.1e^{-2}$	$3.7e^{-4}$
R	±5%	$4.0e^{-1}$	$6.7e^{-3}$
c_e	±5%	$2.0e^{-2}$	$3.3e^{-4}$
J_p	±10%	$1.0e^{-6}$	$3.3e^{-8}$
c_m	±5%	$1.0e^{-2}$	$1.7e^{-4}$
d	±20%	$2.0e^{-2}$	$1.3e^{-3}$

performed for reference purposes without high computing costs. Based on the model of the electric drive the equations (2.2) and (2.4) can be simplified as follows:

$$\begin{aligned} \tfrac{d}{dt}I(t) &= \tfrac{1}{L_\mathrm{p}}\big[-RI(t) - c_\mathrm{e}\omega(t) + U(t)\big], \\ \tfrac{d}{dt}\omega(t) &= \tfrac{1}{J_\mathrm{p}}\big[c_\mathrm{m}I(t) - d\omega(t) - \tau_\mathrm{load}(t)\big]. \end{aligned}$$

The outputs of the simulation model are the two states, more precisely, the current $I(t)$ and the motor speed $\omega(t)$ over time.

Before applying the UQ method to the simulation model, it is necessary to clarify which output variables are significant, and one should make a reasonable statement about the influence of uncertain parameters. A popular scenario is the operation of the electric motor with constant voltage and load torque. In this context, the stationary values of the two output variables I_stat and ω_stat, the so-called QoIs, are of great importance for further investigations. Another scenario refers to the mechanical motor characteristic in Figure 2.11a. There the QoI is called stall torque τ_stall. If the motor speed is zero, the applied load torque value is taken and defined as stall torque. This value is a commonly used indicator of wiper drive performance.

For the simplified simulation model, it must be taken into account that the parameters are different due to manufacturing tolerances. All introduced parameters are modeled with normal distributions, and it is assumed that they are independent of each other. The standard deviation can be approximated over the tolerance range in Table 3.3 and the sigma level $\sigma_\mathrm{l} = 3$:

$$\sigma = \frac{r_\mathrm{t}\mu}{\sigma_\mathrm{l}}. \tag{3.19}$$

Based on the six uncertain model parameters, the limits of the PCE with the full-grid approach can be easily recognized by the required simulations in connection with the grid level (see Table 3.4). For example, the definition of the polynomial order $P = 3$

Table 3.4: Number of required simulations to fit the coefficients of the PCE.

Polynomial order P	1	2	3	4	5	6
Full grid	64	729	4,096	15,625	46,656	117,649
Sparse grid	13	97	533	2,381	9,113	30,869

leads to a reduction of the required simulations by a factor of eight related to the sparse grid approach.

To obtain a first reference, one performs a Monte Carlo-based sample with 10,000 points and visualizes the mean values and standard derivatives of the QoIs over time (see figures 3.2a and 3.3a). As already mentioned, one uses stationary values for further investigations. The figures show that the selected stationary time $t_{\mathrm{sta}} = 1.5$ is in a suitable range. One assumes this value because the mean value and the standard deviation converge after 1.0 second and remain constant over time.

In the following, the distribution of QoIs based on parameter uncertainties in the stationary domain is of major interest. One evaluates the available simulation results at a certain point in time t_{sta} with the help of a histogram and generate PDFs from them, which represent the figures 3.2b and 3.3b. Those results are the basis for the analysis of the PCE approaches, and besides the visual comparison, the introduction of a probability measure is necessary.

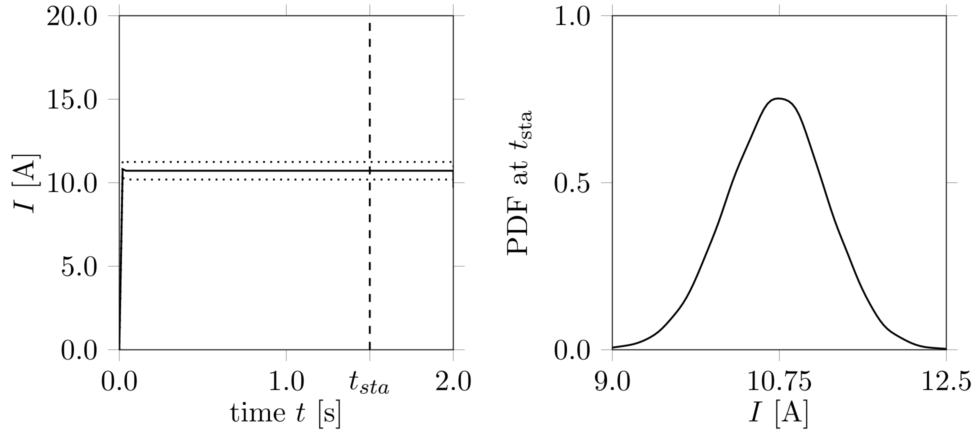
A common method for comparing two samples of one-dimensional probability distributions is the Kolmogorov-Smirnov test [41]. The construction for this approach are two empirical distribution functions $F_{\mathrm{MC,n}}(X_i)$ and $F_{\mathrm{PCE,m}}(X_i)$ for n or m independent and identically distributed observations X_i. The cumulative distribution function $F_{\mathrm{MC,n}}(X_i)$ is a step function that increases by $1/n$ at each of the n data points, and it0 is defined as:

$$F_{\mathrm{MC,n}}(x) = \frac{1}{n}\sum_{i=1}^{n} I_{[-\infty,x]}(X_i),$$

where $I_{[-\infty,x]}(X_i)$ is the indicator function. The Kolmogorov-Smirnov statistic for the two empirical distribution functions is given as:

$$D_{\mathrm{n,m}} = \sup_x \left| F_{\mathrm{MC,n}}(x) - F_{\mathrm{PCE,n}}(x) \right|, \tag{3.20}$$

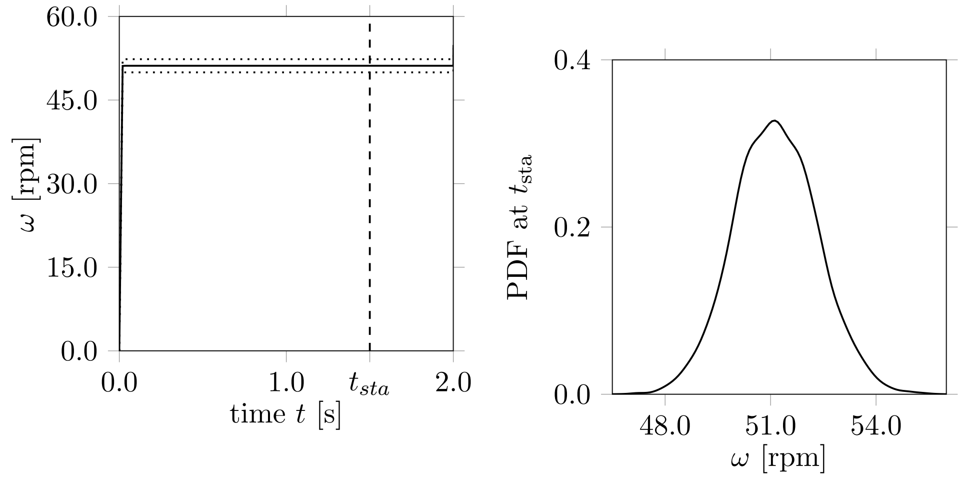
where sup is the supremum of the distances in the set. If $F_{\mathrm{MC,n}}(X_i)$ and $F_{\mathrm{PCE,m}}(X_i)$ have the same underlying distribution function, then $D_{n,m}$ almost surely converges to zero with the assumption that n, m becomes infinite. Also, one can calculate the



(a) Mean value (solid line) and standard derivation (dotted line) for QoI $I(t)$.

(b) PDF obtained from histogram for QoI I at static time point t_{sta}.

Figure 3.2: Statistical overview for output I with assumed parameter uncertainties and Monte Carlo sampling with 10,000 sample points.



(a) Mean value (solid line) and standard derivation (dotted line) for QoI $\omega(t)$.

(b) PDF obtained from histogram for QoI ω at static time point t_{sta}.

Figure 3.3: Statistical overview for output ω with assumed parameter uncertainties and Monte Carlo sampling with 10,000 sample points.

Table 3.5: Kolmogorov-Smirnov test for QoI $I(t_{\mathrm{sta}})$.

	$F_{\mathrm{MC,n}}(I)$, $F_{\mathrm{PCE\|fg,n}}(I)$	$F_{\mathrm{MC,n}}(I)$, $F_{\mathrm{PCE\|sg,n}}(I)$	$F_{\mathrm{PCE\|fg,n}}(I)$, $F_{\mathrm{PCE\|sg,n}}(I)$
$D_{n,n}$	0.0074	0.0074	0.0044
p-value	0.7067	0.6988	0.2778

Table 3.6: Kolmogorov-Smirnov test for QoI $\omega(t_{\mathrm{sta}})$.

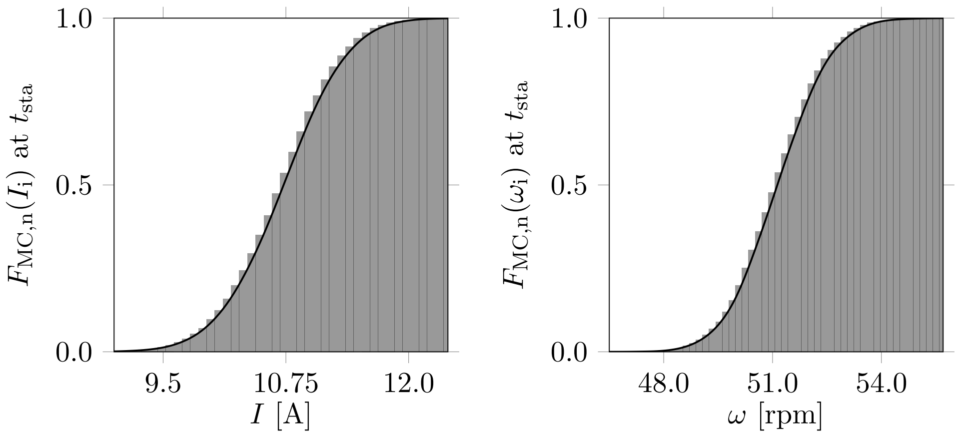
	$F_{\mathrm{MC,n}}(\omega)$, $F_{\mathrm{PCE\|fg,n}}(\omega)$	$F_{\mathrm{MC,n}}(\omega)$, $F_{\mathrm{PCE\|sg,n}}(\omega)$	$F_{\mathrm{PCE\|fg,n}}(\omega)$, $F_{\mathrm{PCE\|sg,n}}(\omega)$
$D_{n,n}$	0.0080	0.0078	0.0048
p-value	0.5977	0.6394	0.1920

p-value using the test statistics $D_{n,m}$, and the corresponding sample sizes [47]. The lower the p-value, the greater the statistical proof that the null hypothesis must be rejected, and one can assume that the distributions are not close to each other.

The cumulative density functions of the two QoIs $I(t_{sta})$ and $\omega(t_{sta})$ are derived from the PDFs of the Monte Carlo sample (see figures 3.4a and 3.4b). In addition, a PCE with a Gauss-Hermite quadrature rule (abbreviated with fg) and a Smolyak Sparse Grid (sg) approach is used to approximate the simulation model. The corresponding CDFs are obtained by evaluating the surrogate PCE models with 10,000 samples and the results are presented in the figures 3.4c, 3.4d, 3.4e and 3.4f. As a first impression, one can assume that the PCE surrogate model is able to approximate the original model by considering only the CDFs in the figures. For reliable statements, the results are compared with the application of the Kolmogorov-Smirnov test.

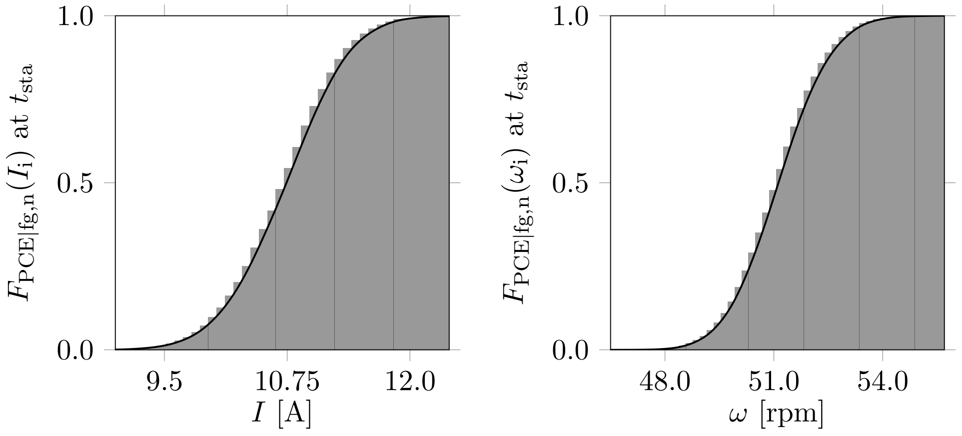
In this case, one assumes that the results of the Monte Carlo sample are the reference, and one wants to compare them with the PCE using different sampling approaches. To achieve better comparability, one resamples the surrogate with a full grid and a sparse grid based on the identical parameter values from the Monte Carlo study. The results are presented in the tables 3.5 and 3.6, where both approaches are compared with the Monte Carlo sample and with each other.

For the simplified simulation model, the difference between the PCE with Gauss-Hermite square rule and the PCE with Smolyak sparse grid is very little for both QoIs, and one can immediately determine this result by the p-value. If the p-value is close to



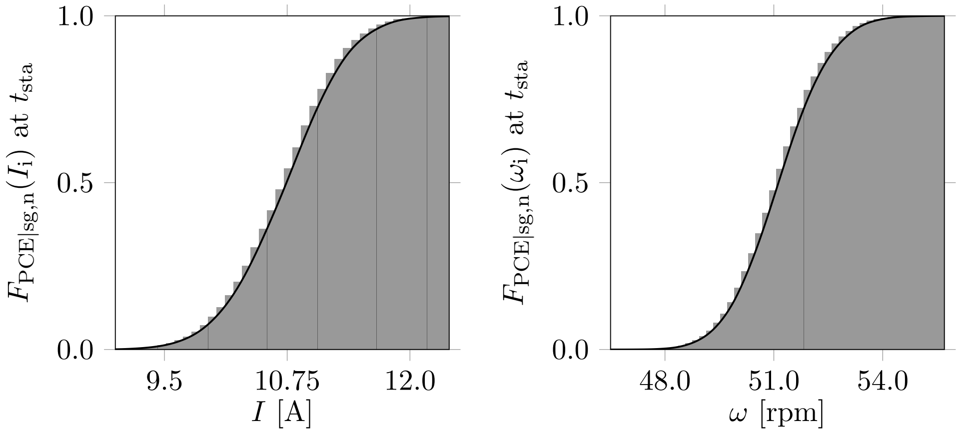
(a) CDF $F_{\mathrm{MC,n}}(I)$ for QoI $I(t_{\mathrm{sta}})$.

(b) CDF $F_{\mathrm{MC,n}}(\omega)$ for QoI $\omega(t_{\mathrm{sta}})$.



(c) CDF $F_{\mathrm{PCE|fg,n}}(I)$ for QoI $I(t_{\mathrm{sta}})$.

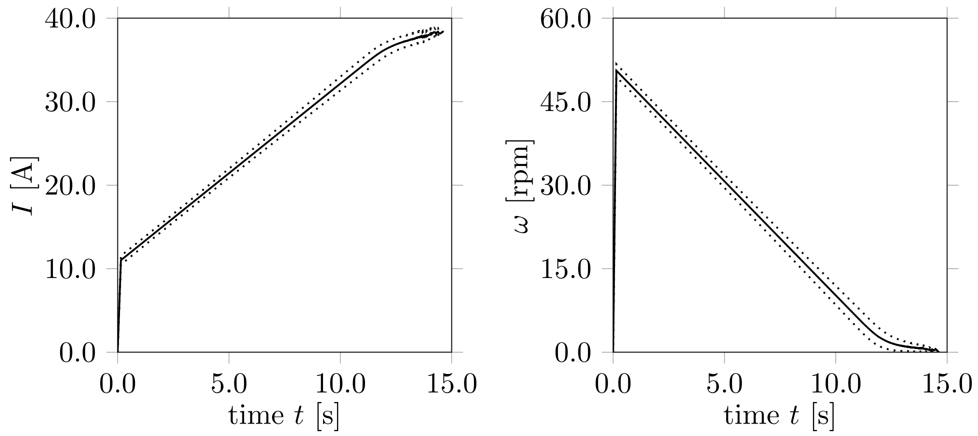
(d) CDF $F_{\mathrm{PCE|fg,n}}(\omega)$ for QoI $\omega(t_{\mathrm{sta}})$.



(e) CDF $F_{\mathrm{PCE|sg,n}}(I)$ for QoI $I(t_{\mathrm{sta}})$.

(f) CDF $F_{\mathrm{PCE|sg,n}}(\omega)$ for QoI $\omega(t_{\mathrm{sta}})$.

Figure 3.4: Cumulative distribution functions (CDF) for both QoIs with kernel density estimation (solid line) and empirical distribution functions (grey bars).



(a) Mean value (solid line) and standard derivation (dotted line) for $I(t)$.

(b) Mean value (solid line) and standard derivation (dotted line) for $\omega(t)$.

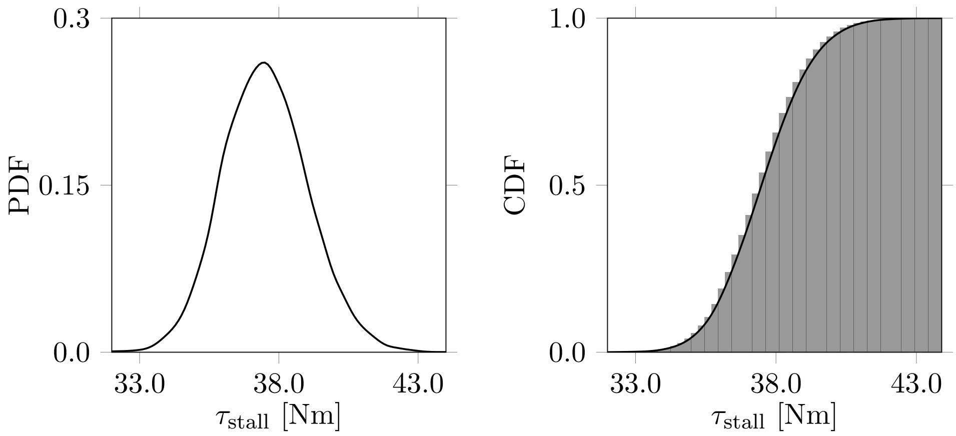
Figure 3.5: Statistical overview for outputs I and ω with assumed parameter uncertainties based on increasing load torque and Monte Carlo sampling with 10,000 sample points.

zero, one assumes that the two groups were drawn from separate distributions, and those may distinguish in the median, the variability, or the shape of the distribution. A reduction of the evaluations to the original model concerning a sparse grid is therefore reasonable. In comparison to the Monte Carlo reference sample, both PCE surrogate models perform well in a given acceptance range, which means that the supremum of the distance set is quite small.

As already mentioned, the stall torque τ_{stall} is another important QoI and will be used in the later investigation to examine further the quality of a PCE surrogate model based on the selected sample and polynomial order P.

Based on the previous investigation, one performs a Monte Carlo sample of 10,000 runs, and Figure 3.5 shows the mean and standard deviation values of the model outputs. The results are obtained by continuously increasing the load torque over time to zero speed. If the motor blocks, the simulation model stops, and the required QoI of the stall torque τ_{load} corresponds to the applied load torque at the moment the motor is blocked. The PDF and the CDF are shown at Figure 3.6, and they serve as a reference for the investigations on the PCE surrogate models.

For the construction of the PCE coefficients, one uses the same samples for the full and sparse grid approach as in the first investigation. In contrast to the stationary



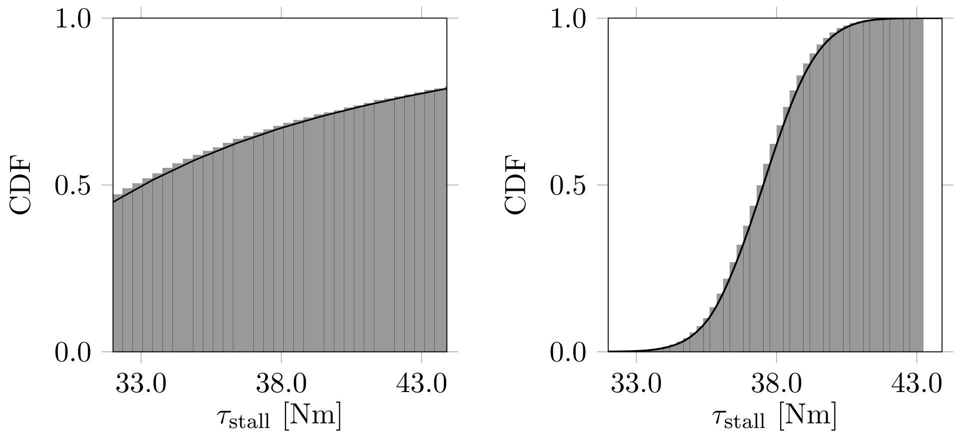
(a) PDF obtained from histogram for QoI τ_{stall}.

(b) CDF obtained from histogram for QoI τ_{stall}.

Figure 3.6: Statistical overview for QoI stall torque τ_{stall} based on Monte Carlo sampling with 10,000 sample points.

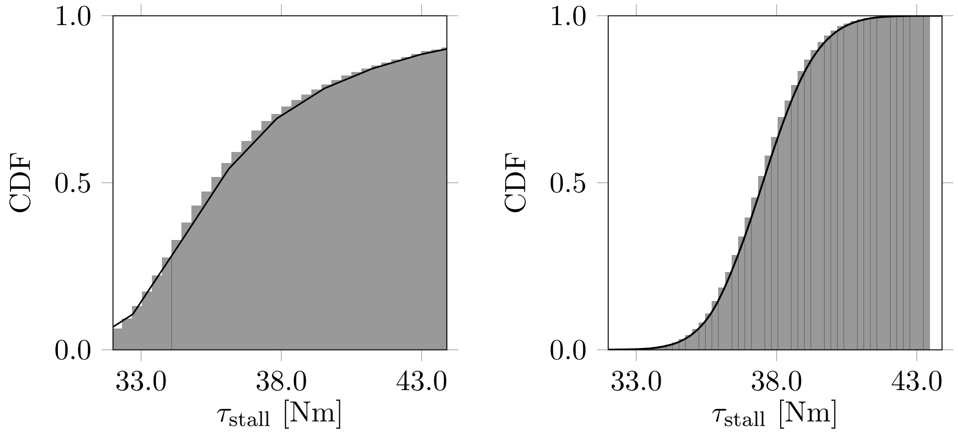
setting, the dynamic QoI increases the requirement on the surrogate models, which can be seen quantitatively in Figure 3.7. The full grid approach can approximate the distribution of the stall torque for the Monte Carlo sampling reference. However, the sparse grid approach does not converge as fast as expected concerning the number of grid points required. The proposed distribution deviation is, especially for a small grid level, many times higher than the reference value. It clearly shows that the results achieved must be checked. One can perform this investigation by using different grid levels to get a feeling for the convergence state of the applied method. One can confirm the observations by use of the previous Kolmogorov-Smirnov statistics in Equation (3.20). Based on Table 3.7, one can state that the D-value does not decrease significantly, and the p-value is almost high enough to accept the approximation concerning the Monte Carlo reference. If one examines the results in Table 3.8, one can see that the D-value decreases continuously over the mesh plane used. Furthermore, the p-value is close to zero, so that the surrogate model is not able to approximate the distribution of the QoI sufficiently.

Instead of comparing the surrogate model with the reference solution, one can look at the PCE coefficients to obtain an indication of the convergence of the solution. If the coefficients for the high-order PCE are relatively large, this means that the grid level used is not sufficient for the approximation, and it should be increased by at least one. In this case, the sparse grid approach is not able to ensure a suitable fit for the



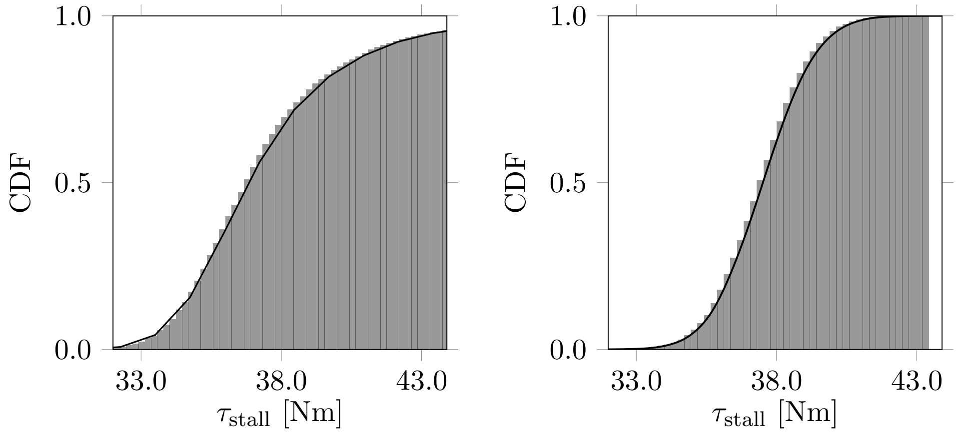
(a) CDF $F_{\text{PCE}|\text{sg},1}(\tau_{\text{stall}})$.

(b) CDF $F_{\text{PCE}|\text{fg},1}(\tau_{\text{stall}})$.



(c) CDF $F_{\text{PCE}|\text{sg},2}(\tau_{\text{stall}})$.

(d) CDF $F_{\text{PCE}|\text{fg},2}(\tau_{\text{stall}})$.



(e) CDF $F_{\text{PCE}|\text{sg},3}(\tau_{\text{stall}})$.

(f) CDF $F_{\text{PCE}|\text{fg},3}(\tau_{\text{stall}})$.

Figure 3.7: Cumulative distribution functions (CDF) for fg and sg approaches with increasing order and the estimation (solid line) and empirical distribution functions (grey bars) are shown.

Table 3.7: Kolmogorov-Smirnov test for QoI τ_{stall} with full grid PCE.

	$F_{\text{MC}}(\tau_{\text{stall}})$, $F_{\text{PCE}\vert\text{fg},1}(\tau_{\text{stall}})$	$F_{\text{MC}}(\tau_{\text{stall}})$, $F_{\text{PCE}\vert\text{fg},2}(\tau_{\text{stall}})$	$F_{\text{MC}}(\tau_{\text{stall}})$, $F_{\text{PCE}\vert\text{fg},3}(\tau_{\text{stall}})$
D	0.0146	0.0099	0.0108
p-value	0.2369	0.7112	0.6042

Table 3.8: Kolmogorov-Smirnov test for QoI τ_{stall} sparse grid PCE.

	$F_{\text{MC}}(\tau_{\text{stall}})$, $F_{\text{PCE}\vert\text{sg},1}(\tau_{\text{stall}})$	$F_{\text{MC}}(\tau_{\text{stall}})$, $F_{\text{PCE}\vert\text{sg},2}(\tau_{\text{stall}})$	$F_{MC}(\tau_{stall})$, $F_{\text{PCE}\vert\text{sg},3}(\tau_{\text{stall}})$	$F_{MC}(\tau_{stall})$, $F_{PCE\vert sg,4}(\tau_{stall})$
D	0.5473	0.3915	0.2039	0.0925
p-value	< 0.0001	< 0.0001	< 0.0001	< 0.0001

PCE model. In this case, the sparse grid approach is not able to ensure an appropriate approximation to the PCE model. In the further course of this work, it is mandatory to review the surrogate model, and a helpful approach is to consider the magnitudes of the coefficients, especially, in high orders.

3.3 Numerical results for the benchmark problem

Based on the previous section, one can now apply the obtained knowledge to the benchmark problem concerning eight uncertain parameters. One uses the complex simulation model in Section 2.2, and assume that the thermal parameters $\boldsymbol{p}^{\text{ther}}$ are constant. Table 3.9 lists the uncertainties associated with the simulation model, and one defines the tolerance ranges beforehand. The examined values are chosen as Gaussian distribution, and each mean value and tolerance range is estimated with the help of expert knowledge. The standard deviation for all parameter distribution is calculated using Equation (3.19). All values are normalized, and in this case, each parameter distribution is divided by its mean value.

Both QoIs of the simple simulation model are also highly relevant for the model under investigation. As a reference, a Monte Carlo sampling with $50,000$ simulation runs is carried out for the distribution of the motor speed in the stationary case, and for the distribution of the stall torque. In the industrial context, it is common to use Latin Hypercube Sampling (LHS) [53, 34] instead of a standard Monte Carlo sampling.

Table 3.9: Uncertain parameters for the benchmark simulation model.

Parameter	Description	Tolerance range r_t
R_p	Resistance	±10%
$c_{\mathrm{m_p}}$	Constant	±5%
L_p	Inductance	±5%
J_p	Total inertia of motor armature	±10%
$\epsilon_{\mathrm{iron,p}}$	Adaption parameter for armature losses	±5%
$\epsilon_{\mathrm{worm,p}}$	Adaption parameter for worm friction coefficient	±20%
$\mu_{\mathrm{A,p}}$	Friction parameter for bearing A	±20%
$\mu_{\mathrm{B,p}}$	Friction parameter for bearing B	±20%

The primary goal is to reduce the number of simulation runs required for a comparative QoI quality.

The number of simulation runs used is as low as possible concerning the computing costs of the evaluation. In general, it is difficult to determine the best amount of runs a priori, and it is even difficult to quantify the quality without a reference solution. Therefore, one defines two experimental designs as 50 and 500, and uses them as the standard evaluation of the QoIs. The numbers are obtained based on the simulation time for a run and the assumption that the simulation tasks are executed sequentially.

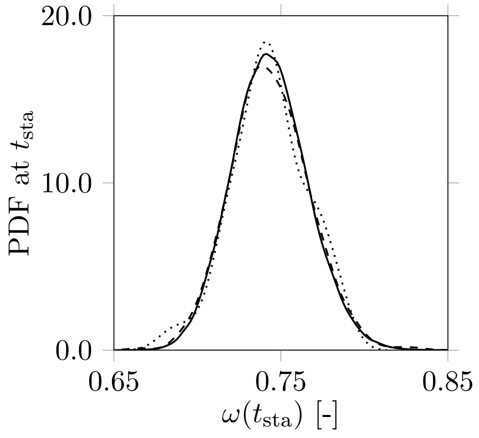
The LHS results with the different design of the experiments are compared to the MC reference in Figure 3.8. Looking at the stationary motor speed, one can see that the LHS approximation with 50 samples can also approximate the reference well. Only in the tail area, the approximation is not as good as desired. Primarily, if one uses the results for evaluations in boundary regions, the accuracy is not sufficient. The results with 500 simulation runs show a more exact estimation of these ranges.

The discrepancies between the reference and the LHS results are used to quantify the quality of the stochastic approximation. An implementation is based on the histograms of the PDFs, and the square error is calculated for each bar with the given equation:

$$\epsilon^{i}_{\mathrm{F_{MC},F_{LHS}}} = (F^{i}_{\mathrm{MC}} - F^{i}_{\mathrm{LHS}})^2 \tag{3.21}$$

where $i = 1, \dots, B$ is the index of the bins of the histogram. The error measure is calculated for both QoIs and visualized in Figure 3.9.

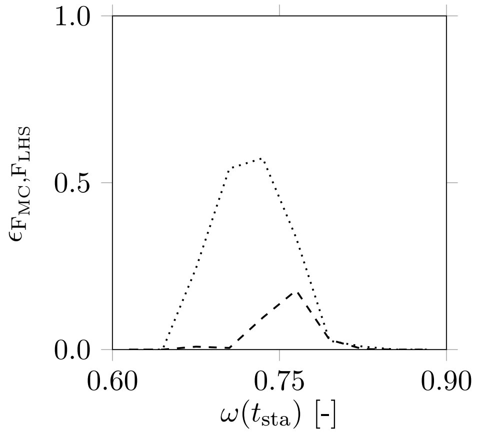
In addition to considering histograms, the Kolmogorov-Smirnov test is used, as mentioned in the previous section. One assumes that the results of the Monte Carlo sample are also the reference. The results are presented in Table 3.10. Based on the



(a) PDF for QoI $\omega(t_{\mathrm{sta}})$.

(b) PDF for QoI τ_{stall}.

Figure 3.8: Comparison of LHS (dotted line: 50 samples, dashed line: 500 samples) with MC reference (solid line) for both QoIs.



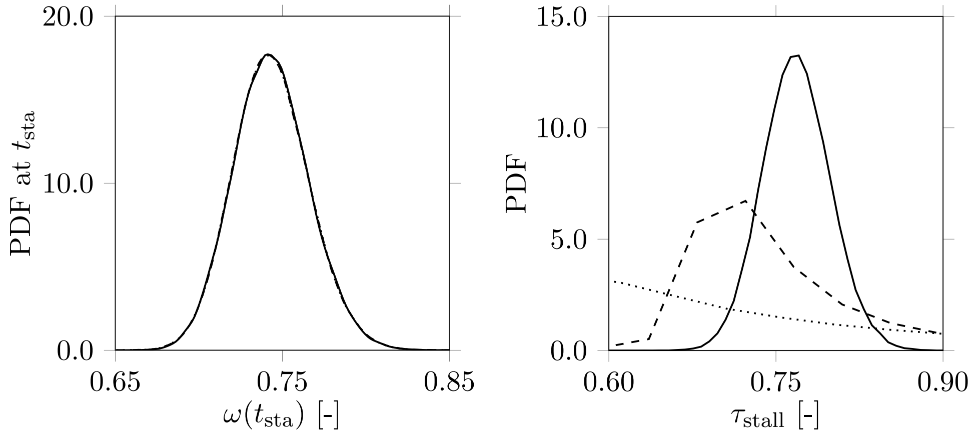
(a) Error measure ϵ for QoI $\omega(t_{\mathrm{sta}})$.

(b) Error measure ϵ for QoI τ_{stall}.

Figure 3.9: Error measure ϵ of LHS (dotted line: 50 samples, dashed line: 500 samples) with respect to MC reference for both QoIs.

Table 3.10: Kolmogorov-Smirnov test for both QoIs with LHS.

	$F_{\mathrm{MC}}\big(\omega(t_{\mathrm{sta}})\big)$, $F_{\mathrm{LHS}\vert 50}\big(\omega(t_{\mathrm{sta}})\big)$	$F_{\mathrm{MC}}\big(\omega(t_{\mathrm{sta}})\big)$, $F_{\mathrm{LHS}\vert 500}\big(\omega(t_{\mathrm{sta}})\big)$	$F_{\mathrm{MC}}(\tau_{\mathrm{stall}})$, $F_{\mathrm{LHS}\vert 50}(\tau_{\mathrm{stall}})$	$F_{\mathrm{MC}}(\tau_{\mathrm{stall}})$, $F_{\mathrm{LHS}\vert 500}(\tau_{\mathrm{stall}})$
D	0.0648	0.0177	0.0610	0.0293
p-value	0.9847	0.9978	0.9924	0.7890



(a) PDF for QoI $\omega(t_{\mathrm{sta}})$.

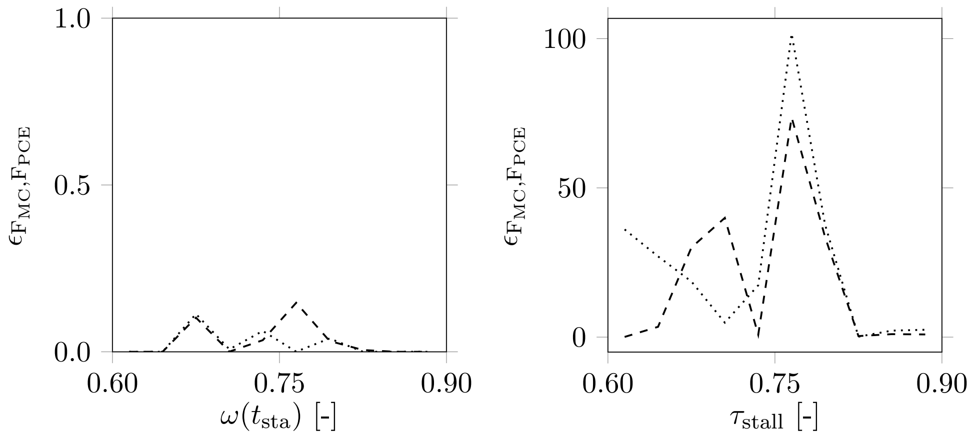
(b) PDF for QoI τ_{stall}.

Figure 3.10: Comparison of PCE surrogate (dotted line: 1st order surrogate model, dashed line: 2nd order surrogate model) with MC reference (solid line) for both QoIs.

test statistics D and p, both QoIs are well approximated. The difference between the LHS results of 50 and 500 samples is remarkable only for the stall torque.

In the previous section, one approximates the PDFs with a PCE surrogate model using a suitable sparse grid approach. The first exploration is performed with orders one and two, resulting in 17 and 161 simulations on the original model. The PCE coefficients are calculated analogously to the simple model, and then a Monte Carlo sampling is performed on the surrogate model with $100,000$ samples to obtain the PDF (see Figure 3.10) for both QoIs.

The QoI of the stationary motor speed $\omega(t_{\mathrm{sta}})$ is considered quantitatively very close to the histogram of the reference. Using the previous qualitative measure with the quadratic error in Equation (3.21), one can not find a significant difference between the applied sparse grids. Compared to the result obtained with LHS and 500 simulation runs,



(a) Error measure ϵ for QoI $\omega(t_{\mathrm{sta}})$. (b) Error measure ϵ for QoI τ_{stall}.

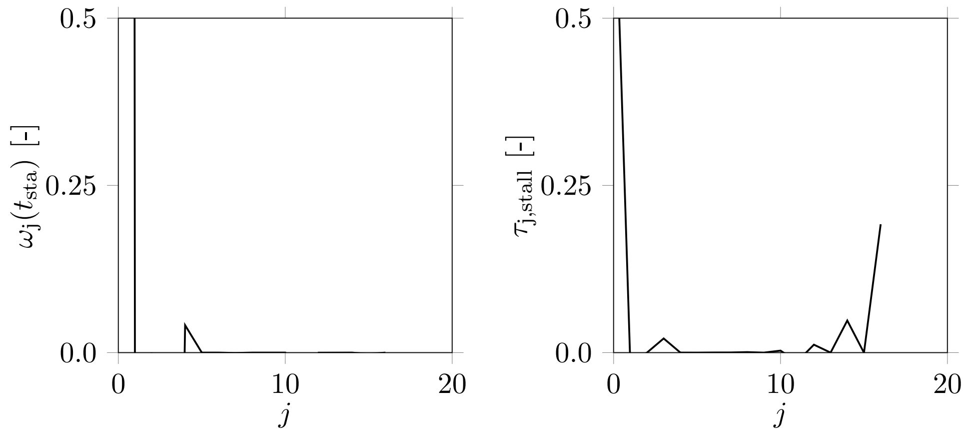
Figure 3.11: Error measure ϵ of PCE surrogate (dotted line: 1st order surrogate model, dashed line: 2nd order surrogate model) with respect to MC reference (solid line) for both QoIs.

Table 3.11: Kolmogorov-Smirnov test for both QoIs with PCE surrogate.

	$F_{\mathrm{MC}}\big(\omega(t_{\mathrm{sta}})\big)$, $F_{\mathrm{LHS}\vert 50}\big(\omega(t_{\mathrm{sta}})\big)$	$F_{\mathrm{MC}}\big(\omega(t_{\mathrm{sta}})\big)$, $F_{\mathrm{LHS}\vert 500}\big(\omega(t_{\mathrm{sta}})\big)$	$F_{\mathrm{MC}}(\tau_{\mathrm{stall}})$, $F_{\mathrm{LHS}\vert 50}(\tau_{\mathrm{stall}})$	$F_{\mathrm{MC}}(\tau_{\mathrm{stall}})$, $F_{\mathrm{LHS}\vert 500}(\tau_{\mathrm{stall}})$
D	0.0040	0.0043	0.5429	0.3918
p-value	0.6511	0.5747	< 0.0001	< 0.0001

the PCE surrogate provides a better approximation to the reference (see Figure 3.11a). This statement is consistent with the Kolmogorov-Smirnov test in Table 3.11. For this QoI, one can assume that the PCE surrogate model provides a better approximation of the distribution than the LHS and, besides, with fewer evaluations of the original simulation model.

It is obvious that the PCE surrogate model for the QoI of the stall torque τ_{stall} is not converged with order one and two (see Figure 3.10b). One can also support this observation by the error measurement and the Kolmogorov-Smirnov test. In this case, it is easy to judge the quality of the surrogate model by comparing it with the reference. In general, this is not possible, and as already mentioned, the coefficients of the PCE surrogate model can be an indicator of convergence.



(a) PCE coefficients for QoI $\omega(t_{\text{sta}})$. (b) PCE coefficients for QoI τ_{stall}.

Figure 3.12: PCE coefficients of surrogate model for both QoIs.

These coefficients for both QoIs are shown in Figure 3.12, and for the stationary motor speed $\omega(t_{\text{sta}})$, the coefficients decrease rapidly with increasing index j. This behavior might indicate that the surrogate model is converged. If one considers the coefficients of the stall torque, one should note that with a high index j, the coefficients $\tau_{j,\text{stall}}$ grow again. In this case, one should assume that the chosen order is not sufficient to solve the problem. It is strongly recommended to increase the order of the PCE.

In this case, if one changes the order from level two to three, it means an increase of the evaluations on the original model from 161 to $1,105$ simulations. Despite the very high number of simulations on the original model, the surrogate model is still not able to approximate the distribution adequately. A possible extension that will not be further investigated here is the use of anisotropic grids. Thereby, one can only increase the grid levels of the relevant parameters and thus saves further simulations.

3.4 Validation with the test bench hardware

So far, one validates the introduced approaches with a reference generated by Monte Carlo samples with a large number of runs on the simulation model. Based on the Subsection 2.3.3, one uses a measurement series from the test bench instead. In this setting, one assumes that the parameters voltage U_{real}, winding resistance R_{real}, and load torque $\tau_{\text{load,real}}$ are normally distributed with predefined variations. In the following, one limits the number of measurements to $S = 100$, and in addition to the steady-state of the motor speed $\omega(t_{\text{sta}})$, one also considers the dynamic range of the current $I(t_{\text{dyn}})$.

Table 3.12: Uncertain parameters for the test bench hardware.

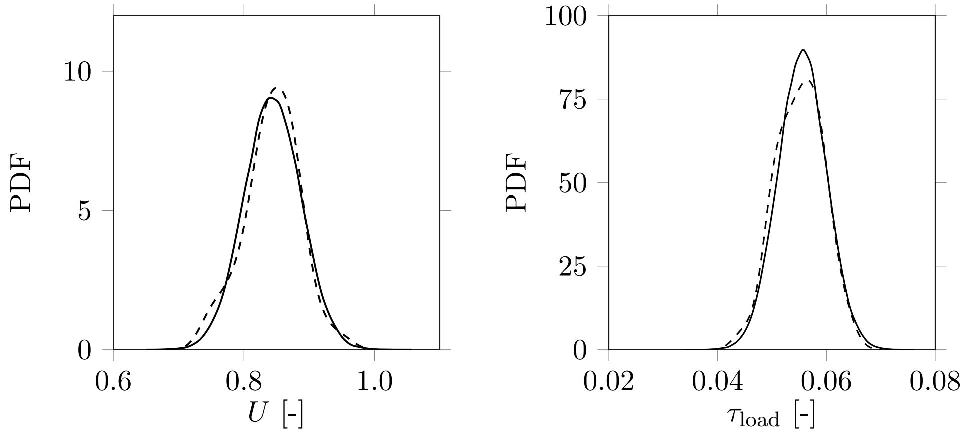
Parameter	Mean value μ	Standard deviation σ
U_{real}	0.84	0.044
R_{real}	0.44	0.08
τ_{real}	0.042	0.0033

The stall torque is not further considered concerning the test bench. This adjustment is mainly because one investigates the long-term behavior of the drive with the test bench.

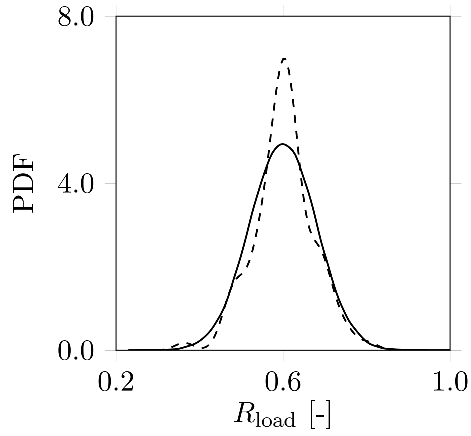
The statistical information of the test bench parameters is given in Table 3.12, and it is used to obtain the test series. Since only 100 measurements are available, it is necessary to check whether the measurement series can approximate the theoretically assumed distributions well.

For the selection of the parameter values, one draws 100 random pairs, and Figure 3.13 shows their distributions. In comparison to the theoretically assumed distribution, one sees that the number of experiments is sufficient to approximate the distribution related to the first and second statistical moments. The error of the mean values $e_{\mu(\text{U})} = 1.0\%$, $e_{\mu(\tau)} = 0.5\%$, and $e_{\mu(\text{R})} = 0.8\%$ are very small. From this point of view, the number of random points is sufficient. If one now considers the error of the standard deviations which are $e_{\sigma(\text{U})} = 9.1\%$, $e_{\sigma(\tau)} = 11.3\%$, and $e_{\sigma(\text{R})} = 14.7\%$, one can observe a larger variation. Therefore, one can pretend that the variation is acceptable in the following use case.

Figure 3.14 shows the LHS and PCE surrogate model results of the motor speed $\omega(t_{sta})$ for the steady-state compared to the test bench measurements and the corresponding error measurements. The PDF visualization shows that both methods can approximate the real variation from the measurement in an acceptable way. If one switches from the first analysis of the PDFs to the error measurements, one quickly realizes that there is a slight discrepancy in the measured distributions. On the one hand, this can be caused by the model error. On the other hand, the distribution information of the test bench is based on only 100 measurements, and one can assume that the statistical moments have not yet completely converged. This observation is also reflected in the Kolmogorov-Smirnov test, and the values in Table 3.13 show only a very small dependency of the distributions on each other. In the comparison of the two methods on the test bench results, there is no significant difference in quality. Regarding the



(a) Input voltage of the motor. (b) Input torque load of the motor.



(c) Input winding resistance of the motor.

Figure 3.13: Distribution of the test bench parameters (dotted line: measurement data, dashed line: theoretical specification).

Table 3.13: Kolmogorov-Smirnov test for the motor speed $\omega(t_{\text{sta}})$ with LHS and PCE surrogate model.

	$F_{\text{meas}}\big(\omega(t_{\text{sta}})\big)$, $F_{\text{LHS}\vert 50}\big(\omega(t_{\text{sta}})\big)$	$F_{\text{meas}}\big(\omega(t_{\text{sta}})\big)$, $F_{\text{LHS}\vert 500}\big(\omega(t_{\text{sta}})\big)$	$F_{\text{meas}}\big(\omega(t_{\text{sta}})\big)$, $F_{\text{PCE}\vert 1}\big(\omega(t_{\text{sta}})\big)$	$F_{\text{meas}}\big(\omega(t_{\text{sta}})\big)$, $F_{\text{PCE}\vert 2}\big(\omega(t_{\text{sta}})\big)$
D	0.1600	0.1420	0.1445	0.1426
p-value	0.3608	0.0694	0.0308	0.0344

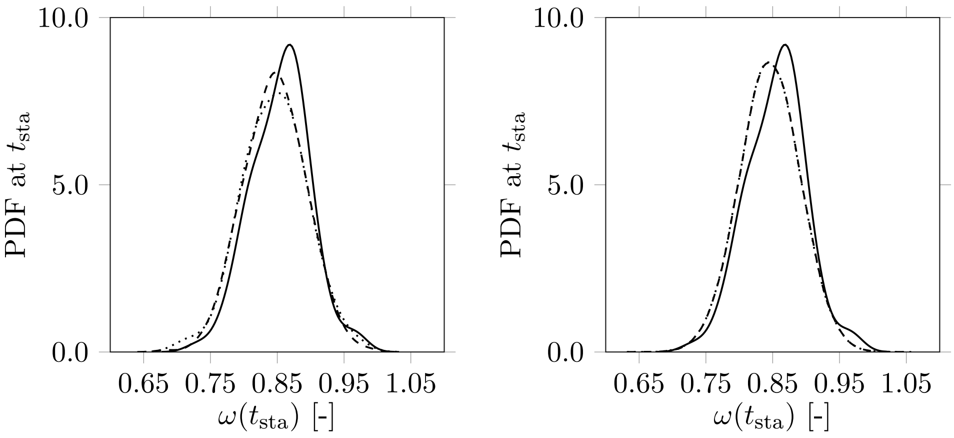
Table 3.14: Kolmogorov-Smirnov test for the current $I(t_{\text{dyn}})$ with LHS and PCE surrogate model.

	$F_{\text{MC}}\big(I(t_{\text{dyn}})\big)$, $F_{\text{LHS}\vert 50}\big(I(t_{\text{dyn}})\big))$	$F_{\text{MC}}\big(I(t_{\text{dyn}})\big)$, $F_{\text{LHS}\vert 500}\big(I(t_{\text{dyn}})\big)$	$F_{\text{MC}}\big(I(t_{\text{dyn}})\big)$, $F_{\text{PCE}\vert 1}\big(I(t_{\text{dyn}})\big)$	$F_{\text{MC}}\big(I(t_{\text{dyn}})\big)$, $F_{\text{PCE}\vert 2}\big(I(t_{\text{dyn}})\big)$
D	0.1100	0.0960	0.0945	0.0957
p-value	0.8147	0.4262	0.3339	0.3201

simulation evaluations used on the original model, the PCE surrogate model achieves this quality with only seven simulations.

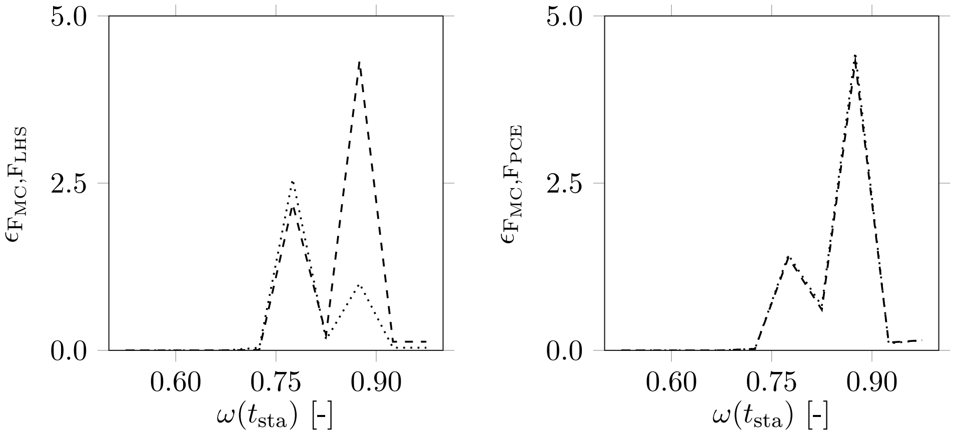
If one considers the current in the dynamic time domain, similar observations can be made (see Figure 3.15), and both methods can approximate the distribution assumption by the test bench. A discrepancy also shows up here, especially in the lower value range. Another hypothesis to explain the difference shown is again the small number of measurements that led to the reference. Also, one expects a more notable impact of noise for the current in contrast to the motor speed. Looking at the Kolmogorov-Smirnov test in Table 3.14, the probability that the compared distributions are identical is in the same order of magnitude.

Finally, one can state that the simulation model can approximate the test bench characteristics and the corresponding windshield wiper drive well. One can already see at this point that the number of measurements can only provide a statistical statement to a limited extent, and therefore deviations between the measured and numerical distributions can be expected. Concerning the input distribution of the winding resistance, there is a notable difference from the theoretical specification compared to the measured data, which could be an indication that this distribution is difficult to classify.



(a) PDF based on LHS (dotted line: 50 samples, dashed line: 500 samples) for QoI $\omega(t_{\text{sta}})$.

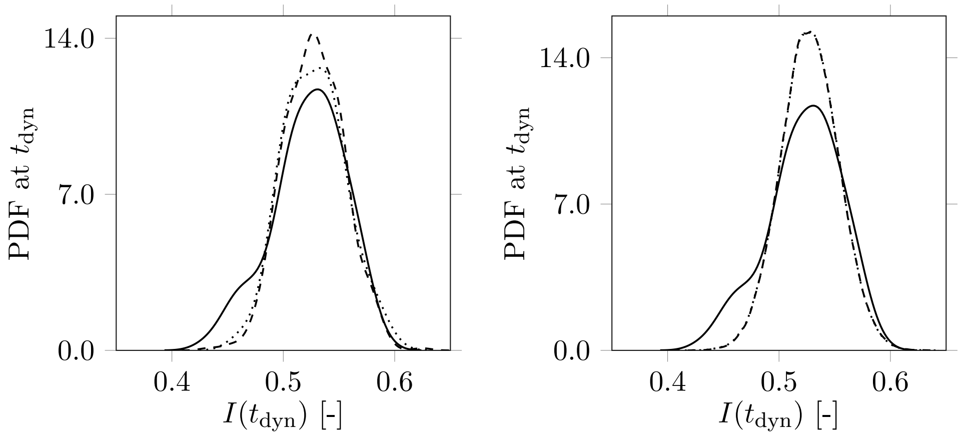
(b) PDF based on PCE surrogate model (dotted line: $P = 1$, dashed line: $P = 2$) for QoI $\omega(t_{\text{sta}})$.



(c) Error measure ϵ based on LHS (dotted line: 50 samples, dashed line: 500 samples) for QoI $\omega(t_{\text{sta}})$.

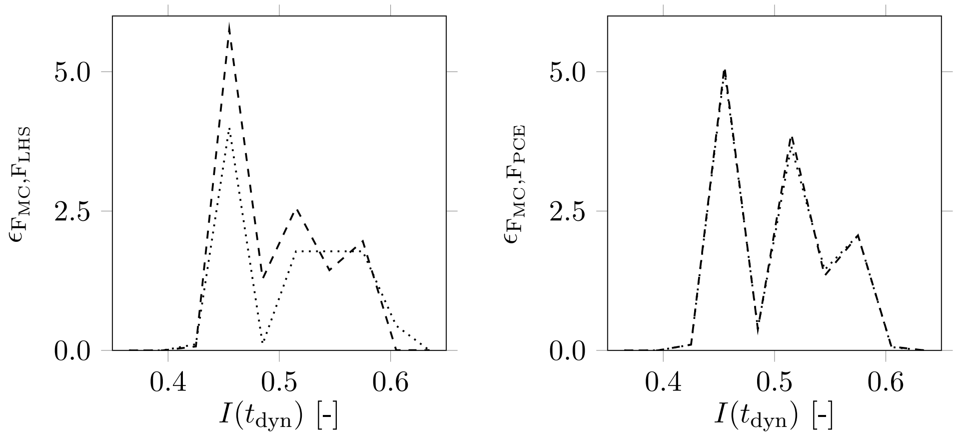
(d) Error measure ϵ based on PCE surrogate model (dotted line: $P = 1$, dashed line: $P = 2$) for QoI $\omega(t_{\text{sta}})$.

Figure 3.14: Comparison of LHS and PCE surrogate model results and test bench hardware measurements (solid line).



(a) PDF based on LHS (dotted line: 50 samples, dashed line: 500 samples) for QoI $I(t_{\mathrm{dyn}})$.

(b) PDF based on PCE surrogate model (dotted line: $P = 1$, dashed line: $P = 2$) for QoI $I(t_{\mathrm{dyn}})$.



(c) Error measure ϵ based on LHS (dotted line: 50 samples, dashed line: 500 samples) for QoI $I(t_{\mathrm{dyn}})$.

(d) Error measure ϵ based on PCE surrogate model (dotted line: $P = 1$, dashed line: $P = 2$) for QoI $I(t_{\mathrm{dyn}})$.

Figure 3.15: Comparison of LHS and PCE surrogate model results and test bench hardware measurements (solid line).

4

Backward propagation of uncertainties

The output distributions in Chapter 3 are calculated based on assumed input and parameter distributions, and the simulation results can provide a suitable approximation to the reference measurements. In practice, however, it is often difficult to estimate the distribution of parameters and inputs due to the lack of data and knowledge. In some cases, one can use quality specifications, but frequently, experts make the estimates. For engineering applications, they usually assume either a uniform or a Gaussian distribution. Moreover, a subset of the distributions is hard to estimate because required measurements are either quite expensive or not possible at all. In this case, backward propagation can help to obtain an approximation of such parameter distributions.

In general, the backward propagation includes a variety of methods, and very roughly, one can make a distinction based on the parameter properties. If the parameters have a physical significance, then one speaks of parameter estimation. On the other hand, one denotes it as parameter calibration when they have no or little physical reference. Regardless of the classification, the overall aim is identical: one wants to find the optimal parameters which bring the results of the simulation model and the available measurement data in line. In a mathematical sense, one can describe this class as inverse problems, and they are often ill-posed [67]. In other words, this means that there may not be a unique solution. One can set up an optimization to find a solution estimate for the inverse problems based on a deterministic or statistical method. One can achieve the deterministic solution by introducing a measure between the simulation results and the measurement data. This procedure is well known, and one focuses mainly on the discrepancy between the model prediction and the observed data. This error then needs to be minimized by the use of an optimization algorithm. One can find an overview of suitable methods in [58, 37, 45, 40].

In contrast, one can achieve a statistical approximation by reformulation to an inference problem. It implies that one tries to regularize the ill-posedness of the inverse problem. One usually expresses this regularization as prior distributions, and the goal is to bring the information from the observation into the estimated parameter distributions. A possible approach is, among many others, the Bayesian inference [27, 63, 2, 25]. It offers the possibility to improve the knowledge of parameter distributions based on a numerical simulation model and observed measurement data. Based on a prior parameter guess, a posterior distribution can be estimated. This method requires, in general, excessive computational effort. One way to overcome this obstacle is to use a surrogate model to reduce the computational cost.

In this context, the recorded measurement data must be pre-processed to highlight the essential parts that one can use them for further investigation. The main advantage of the test bench setup is the ability to pre-define parameter distributions, and this reference data can be used to validate the obtained posterior distributions from the Bayesian inference approach. The focus is on building a suitable Likelihood function based on the knowledge of a global sensitivity analysis. Also, one investigates measurements over time and multiple outputs.

The remainder of this chapter is as follows. Section 4.1 focuses on general methods of Bayesian inference and especially, on possible extensions regarding PCE. In addition to the determination of the posterior distributions of the uncertain parameters, one uses a surrogate PC model to accelerate the evaluation. The resulting coefficients are reused to calculate Sobol's indices, which enable a global sensitivity analysis. Its results are used to model the Likelihood function in Section 4.2. To describe this problem, one still needs to model the physical parameters appropriately, and Section 4.3 outlines this procedure. Section 4.4 discusses the combination of the standard approach for Bayesian interference and the PC surrogate model in more detail. It shows possibilities for increasing efficiency. One can find the numerical results of the benchmark problem in Section 4.5. For this purpose, there are three parts to the investigation. Within the first two investigations, the focus is on a standard approach for Bayesian interference in combination with the PC surrogate model, and one investigates the influence of the parameters and measurement data. In the third part, the focus is on the efficiency-enhancing measures.

4.1 Bayesian inference

In this work, one uses the methods of Bayesian statistics for parameter calibration, which means that one reformulates the given inverse problem in a similar way as in [71]. The basic idea of the Bayesian paradigm is to model uncertainties in a probabilistic way using observational data. Based on the deterministic mapping in Equation (3.1), and the assumption that the model parameters $\boldsymbol{p}$ are uncertain, one changes the notation to $\boldsymbol{\theta}$. As already described, the PDFs $f_{\theta_\mathrm{i}}(x_\mathrm{i})$ are assumed for $i = 1, \ldots, N$ with independent scalar components x_i (finite noise assumption). As a result, one represents the model outputs by a random vector:

$$\boldsymbol{Y}(t) = \mathcal{M}(t, \boldsymbol{\theta}, u(t)), \quad t \in [0, T].$$

Initially known information about the model parameters $\boldsymbol{\theta}$ can be described with a so-called prior distribution density $f_{\boldsymbol{\theta}}(\boldsymbol{x})$.

Another necessary consideration is that one has to take into account further error influences in the modeling. A classical approach is to include additive noise in the model, which is suitable for a wide range of practical applications:

$$\boldsymbol{Y}^{\mathrm{obs}}(t) = \mathcal{M}(t, \boldsymbol{\theta}, u(t)) + \boldsymbol{\epsilon}(t), \tag{4.1}$$

where the components of $\boldsymbol{\epsilon}(t)$ are independently and identically distributed random variables over time with the density p_ϵ. The PDF of $\boldsymbol{\epsilon}$ is assumed to be normally distributed, for example, with mean $\mu = 0$ and variance $\sigma = \boldsymbol{\Sigma}$. In this simple case, $\boldsymbol{\epsilon}$ contains both the model error and the measurement error that can arise from sensor noise or non-modeled physical effects.

One aim of the inverse problem is to find the posterior distribution. In contrast to deterministic parameter identification, the stochastic method provides a statement about the variability of the solution. Of course, one can reduce the distribution statement to a deterministic point that is optimal. This transformation can be achieved by using the expected value and additionally obtaining a quality measure of its standard deviation. Specifically, this means updating $p(\boldsymbol{\theta})$ through obtained observations $\boldsymbol{Y}^{\mathrm{obs}}$ to infer knowledge about the corresponding PDF of $\boldsymbol{\theta}$ with the given joint PDF:

$$p(\boldsymbol{\theta}, \boldsymbol{Y}^{\mathrm{obs}}) = p(\boldsymbol{\theta}|\boldsymbol{Y}^{\mathrm{obs}})p(\boldsymbol{Y}^{\mathrm{obs}}) = p(\boldsymbol{Y}^{\mathrm{obs}}|\boldsymbol{\theta})p(\boldsymbol{\theta}),$$

where the connection between the model parameters $\boldsymbol{\theta}$ and the gathered observations $\boldsymbol{Y}^{\text{obs}}$ is expressed by the Likelihood function $p(\boldsymbol{Y}^{\text{obs}}|\boldsymbol{\theta})$. The conditional probability $p(\boldsymbol{\theta}|\boldsymbol{Y}^{\text{obs}})$ is called the posterior PDF of $\boldsymbol{\theta}$ and displays the information about $\boldsymbol{\theta}$ after obtaining an observation $\boldsymbol{Y}^{\text{obs}}$. Considering the inverse problem, this can be expressed with the Bayesian statistics as follows:

$$p(\boldsymbol{\theta}|\boldsymbol{Y}^{\text{obs}}) = \frac{p(\boldsymbol{Y}^{\text{obs}}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{Y}^{\text{obs}})}. \tag{4.2}$$

One describes the denominator by the marginal probability $p(\boldsymbol{Y}^{\text{obs}})$ of the observations and can express it as:

$$p(\boldsymbol{Y}^{\text{obs}}) = \int p(\boldsymbol{\theta}, \boldsymbol{Y}^{\text{obs}})d\boldsymbol{\theta} = \int p(\boldsymbol{Y}^{\text{obs}}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}. \tag{4.3}$$

Based on equations (4.2) and (4.3) the posterior density for the model parameters can be written as:

$$p(\boldsymbol{\theta}|\boldsymbol{Y}^{\text{obs}}) = \frac{p(\boldsymbol{Y}^{\text{obs}}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{\int p(\boldsymbol{Y}^{\text{obs}}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}}.$$

For convenience purpose only the Likelihood function can be defined as $L(\boldsymbol{\theta}) := p(\boldsymbol{Y}^{\text{obs}}|\boldsymbol{\theta})$, which may be viewed as a function of the model parameters $\boldsymbol{\theta}$.

The calculation of the posterior distribution summarizes in the following steps. Based on the parameters that are part of the calibration problem, a suitable prior distribution is selected. For this purpose, one uses all relevant information such as the type of distribution, or minimum and maximum values. Besides, one has to develop a suitable Likelihood function which represents the dependence between the model and the observed data. The final step is to calculate the posterior distribution. A popular approach is the Markov Chain Monte Carlo (MCMC) method [48, 33, 49] which uses correlated Monte Carlo samples to update the unknown distributions, and it is also the chosen approach for the benchmark problem.

4.1.1 Prior distributions selection

The prior parameter distributions $p(\boldsymbol{\theta})$ are an excellent possibility to include first knowledge in the calculation [26], and there may be a considerable degree of subjectivity involved. The most challenging aspect is to set the range of prior distributions so that the observation area covers the model behavior. There are three major categories of priors in applied research.

The most meaningful results concerning the variations, in reality, can often be obtained when prior knowledge of the available parameter distributions is available.

The first category often referred to as informative priors, and they contain numerical information that is essential for estimating the model. One can derive these priors from expert knowledge, preliminary data analysis, or literature. If the information contents of the a priori knowledge are less accurate, it is called a weak-informative prior and is part of the second category. In other words, these priors are strong enough to pull the data away from insufficient areas. If no preliminary information is available, one tries to design the prior distributions in such a way that they have only a small influence on the result. This category forms the third category, and one can achieve it with uniform parameter distributions known as non-informative priors. Concerning the physical parameters, uniform distributions are popular. Expert knowledge or natural constraints then define their boundaries.

4.1.2 Likelihood function

The Likelihood function is central to the process of estimating the unknown parameter distributions. It describes the probability of a random variable that contains another random variable. In a concrete case, this can be a probability of a parameter value in combination with given measurement data. It is possible, by introducing a probability structure into the parameter space that a parameter value has a high probability value for a given specified observation data and still has a small probability, or inversely.

The Likelihood function $L(\boldsymbol{\theta})$ depends on the assumptions made by the error classes and their distributions. One can achieve the specification by modeling the error influences beforehand. Looking at Equation (4.1), one assume $\boldsymbol{\epsilon}$ is an additive error. The components are independent and identically distributed, and $\epsilon_{\text{i}} \sim \mathcal{N}(0, \sigma^2)$ are expected with a fixed standard deviation. The Likelihood $L(\theta)$ function for an uncertain parameter θ and a fixed time t^* describes itself as follows:

$$L(\theta, t^*) = p(Y^{\text{obs}}|\theta) = \prod_s f_\epsilon \left(\boldsymbol{Y}^{\text{s}}_{\text{meas}}(t^*) - \mathcal{M}\left(t^*, \theta, u(t^*)\right) | \mu_\epsilon, \sigma_\epsilon\right),$$

where $\boldsymbol{Y}^{s}_{\text{meas}}(t^*)$ corresponds to the measurements series with an overall number of S records, $S > 0, S \in \mathbb{N}$, and $\mu_\epsilon, \sigma_\epsilon$ express the assumed error statistics. The additive noise is supposed to be a Gaussian distribution. One can rewrite the Likelihood function in the following way:

$$L(\theta, t^*) = \left(\frac{1}{\sqrt{2\pi\sigma_\epsilon^2}}\right)^{\text{S}/2} \exp\left(-\frac{\sum_{\text{s}=1}^{\text{S}} \left(\boldsymbol{Y}^{\text{s}}_{\text{meas}}(t^*) - \mathcal{M}\left(t^*, \theta, u(t^*)\right)\right)^2}{2\sigma_\epsilon^2}\right). \quad (4.4)$$

In most cases, the uncertain parameter is multi-dimensional. Therefore, in the following several parameters are considered. In addition, the Likelihood function in

Equation (4.4) is to be extended from a single point in time t^* to a time range $t \in [t_0, t_{\mathrm{K}-1}]$:

$$
\begin{aligned}
L(\boldsymbol{\theta}, t) = & \left(\frac{1}{\sqrt{2\pi\sigma_\epsilon^2}}\right)^{(\mathrm{S}+\mathrm{T})/2} \\
& \exp\left(-\frac{\sum_{\mathrm{s}=1}^{\mathrm{S}} \sum_{\mathrm{k}=0}^{\mathrm{K}-1} \left(\boldsymbol{Y}_{\mathrm{meas}}^{\mathrm{s}}(t_{\mathrm{k}}) - \mathcal{M}\left(t_{\mathrm{k}}, \boldsymbol{\theta}, u(t_{\mathrm{k}})\right)\right)^2}{2\sigma_\epsilon^2}\right),
\end{aligned}
$$

or more conveniently:

$$
L(\boldsymbol{\theta}, t) = \prod_{\mathrm{k}} \left(\prod_{\mathrm{s}} f_\epsilon \left(\boldsymbol{Y}_{\mathrm{meas}}^{\mathrm{s}}(t_{\mathrm{k}}) - \mathcal{M}\left(t_{\mathrm{k}}, \boldsymbol{\theta}, u(t_{\mathrm{k}})\right) | \mu_\epsilon, \sigma_\epsilon\right) \right), \tag{4.5}
$$

with $k \in [0, K-1]$ and $s \in [1, S]$. If the measurement data contains a large number of time samples or runs, the complexity of the given Likelihood function increases rapidly. A reduction strategy which one presents in the following section, keeps the calculation costs at a low level.

4.1.3 Integration of the PCE surrogate model

From an industrial point of view, high-level industrial models generally have a long simulation duration due to their complexity. The parameter estimation algorithms that one applies in the further course, usually require a high number of model evaluations. To make them applicable from an economic point of view, this section will discuss the integration of a surrogate model within the overall concept.

To reduce the calculation time, one needs an alternative model that posses the main feature of the original model but with much less computational overhead, for example, a surrogate model. Hence, one can replace the evaluations on the original model by a PCE surrogate model. To fit such a model, or in other words, to generate the PC expansion coefficients, a probability space must be defined for the model parameters. Based on the distribution classes for the prior $p(\boldsymbol{\theta})$, the hyperparameters of the PC surrogate model in Equation (3.14) can be obtained:

$$
\boldsymbol{Y}^{\mathrm{P}}(t, \boldsymbol{\xi}(\boldsymbol{\theta})) = \sum_{\mathrm{j}=0}^{\mathrm{P}} \boldsymbol{y}_{\mathrm{j}}(t) \psi_{\mathrm{j}}(\boldsymbol{\xi}(\boldsymbol{\theta})).
$$

The following section addresses the mapping of the physical parameters $\boldsymbol{\theta}$ to the hyperparameters $\boldsymbol{\xi}$ of the surrogate model. So far, the Likelihood function in Equation (4.5) is only considered for a single model output. The approximations of the outputs of the surrogate model are used component by component with the following notation:

$$Y_{\mathrm{i}}^{\mathrm{P}}(t, \boldsymbol{\xi}(\boldsymbol{\theta})) = \sum_{\mathrm{j}=0}^{\mathrm{P}} y_{\mathrm{ij}}(t)\psi_{\mathrm{j}}(\boldsymbol{\xi}(\boldsymbol{\theta})), \quad i = 1, \dots N.$$

The Likelihood function $L(\boldsymbol{\theta}, t)$ in Equation (4.5) can be postulated for each model output i and the evaluation on the original model can be approximated by the surrogate model above:

$$\tilde{L}_{\mathrm{i}}(\boldsymbol{\theta}, t) = \prod_{\mathrm{k}} \left(\prod_{\mathrm{s}} f_{\epsilon}\Big(\boldsymbol{Y}_{\mathrm{i,meas}}^{\mathrm{s}}(t_{\mathrm{k}}) - \sum_{\mathrm{j}=0}^{\mathrm{P}} y_{\mathrm{ij}}(t_{\mathrm{k}})\psi_{\mathrm{j}}(\boldsymbol{\xi}(\boldsymbol{\theta})) \;\Big|\; \mu_{\epsilon}, \sigma_{\epsilon} \Big) \right), \tag{4.6}$$

for $i = 1, \dots N$. The overall Likelihood function $\tilde{L}(\boldsymbol{\theta}, t)$ can be obtained by multiplying each Likelihood function $\tilde{L}_{\mathrm{i}}(\boldsymbol{\theta}, t)$ in the above Equation (4.6):

$$\tilde{L}(\boldsymbol{\theta}, t) = \prod_{\mathrm{i}} \tilde{L}_{\mathrm{i}}(\boldsymbol{\theta}, t). \tag{4.7}$$

4.1.4 Markov Chain Monte Carlo methods

This subsection is intended to outline the idea of Markov Chain Monte Carlo (MCMC) methods, and one suggests the following references [66, 7, 44] for further reading. MCMC methods include a set of algorithms for sampling from a probability distribution. One constructs a Markov chain with the same stationary distributions as the desired one. After a large number of steps, one uses the state of the chain as a sample for the desired distribution. The quality of the distribution increases with the number of steps.

In general, Markov chains are a fundamental class of stochastic models for sequences of non-independent random variables. One can describe such a model of a time-discrete Markov chain with finite states by its state-space, initial distribution, and transition matrix. The state-space of a Markov chain is a finite set of all possible states of the model and can be written as $E = 1, 2, \dots, l$, where $l \in \mathbb{N}$ is an arbitrary finite number. The initial probability that the system is in the state $i \in E$ can now be expressed as γ_{i}. The following assumption applies:

$$\gamma_{\mathrm{i}} \in [0, 1], \quad \sum_{\mathrm{i}=1}^{\mathrm{l}} \gamma_{\mathrm{i}} = 1, \tag{4.8}$$

and the initial distribution of the Markov chain can be summarized with the vector $\boldsymbol{\gamma} = \{\gamma_1, \dots, \gamma_{\mathrm{l}}\}^{\mathrm{T}}$. Then the transition of the system from a state i to j during one time step can be represented by the probability $p_{\mathrm{ij}} \in [0,1]$ with $i, j \in E$. The one-step transition matrix $\boldsymbol{P}$ is formed by the probabilities p_{ij}:

$$\boldsymbol{P} = p_{\mathrm{ij}}, \quad p_{\mathrm{ij}} > 0, \quad \sum_{\mathrm{i=1}}^{\mathrm{l}} p_{\mathrm{ij}} = 1,$$

for $i, j = [1, \ldots, l]$.

A Markov chain can be now represented as follows [5]: Let $\boldsymbol{X}_0, \boldsymbol{X}_1, \cdots : \omega \to E$ be a sequence of random variables which are defined on $(\omega, \mathcal{F}, \mathbb{P})$ and mapping into the set $E = \{1, \ldots, l\}$. Then $\boldsymbol{X}_0, \boldsymbol{X}_1, \ldots$ is called a Markov chain with initial distribution $\boldsymbol{\alpha}$ and the transition matrix $\boldsymbol{P}$, if:

$$P(\boldsymbol{X}_0 = i_0, \boldsymbol{X}_1 = i_1, \ldots, \boldsymbol{X}_{\mathrm{n}} = i_{\mathrm{n}}) = \gamma_{\mathrm{i}_0} p_{\mathrm{i}_0\mathrm{i}_1} \cdots p_{\mathrm{i}_{\mathrm{n}-1}\mathrm{i}_{\mathrm{n}}},$$

for arbitrary $n = 0, 1, \ldots$ and $i_0, i_1, \ldots, i_n \in E$.

With the support of Markov chains, one can now generate algorithms for the approximation of the desired posterior distribution [6, 24]. This procedure takes advantage of the fact that it is often much more comfortable to simulate Markov chains with stationary distribution than the distribution itself.

It is assumed that a Markov chain $\boldsymbol{X}_0, \boldsymbol{X}_1, \ldots$ can be built with the following properties: It has the state space E and an irreducible and aperiodic transition matrix $\boldsymbol{P}$, so that $p_{\mathrm{ij}} = \frac{1}{|E||}$ if $i, j \in E$ are neighbors and otherwise $p_{\mathrm{ij}} = 0$. Then one can build a path $x_0, x_1, \ldots$ of the Markov chain as follows [49]:

1. Choose an acceptable initial configuration $x_0 \in E$.

2. Generate proposal $x_{\mathrm{n}}\prime$ from underlying distribution.

3. Accept $x_0\prime$ as a sample with the probability $\alpha(x_0\prime | x_{\mathrm{n}})$.

4. If the sample is accepted, set $x_{\mathrm{n+1}} = x_{\mathrm{n}}\prime$, else set $x_{\mathrm{n+1}} = x_{\mathrm{n}}$.

This accept-rejection scheme controls the sequence in a way that one samples the target distribution after a certain point in the chain, and this approach is called Metropolis-Hasting MCMC. Metropolis [48] and Hastings [33] developed the Metropolis-Hastings algorithm, and it is one of the most efficient rules [64]. The reason for employing MCMC in Bayesian applications is the challenge to compute the marginal probability $p(\boldsymbol{Y}^{\mathrm{obs}})$ of the observations, and therefore, an essential feature of the algorithm is the ability to obtain a sample without knowing this normalization factor.

4.2 Modeling of the Likelihood function

As already mentioned, the Likelihood function in Equation (4.7) considers not only several measurements and outputs but also signals over time. Now the question arises whether it is necessary to use all measurement data to build the Likelihood function. If one assumes that not every point in time implies an increase in the information content, only a representative subset of the points in time should be selected and used.

In this case, one can apply a sensitivity analysis (SA) for identifying the time ranges that contain most information for the parameter estimation. One should note that the time ranges for each parameter under consideration can be different depending on the selected output. For example, the winding resistor has a stronger sensitivity during switch-on than during stationary movement. One can further assume that the sensitivity of the winding resistance to the current is higher than to the motor speed. Among the numerous techniques on sensitivity analysis, one can divide those into two classes: local and global. Here, the focus is on the global SA, and in particular, one uses the Sobol' indices to gather information about the given simulation model. As a post-processing step of the PCE coefficient calculation, an analytical calculation of the Sobol' indices is possible with the PCE surrogate model. This characteristic is a great advantage compared to the other approaches, and therefore, one applies this method in the present context.

4.2.1 Sobol' indices

For a global SA, one needs distributions of the parameters. These values are the actual goal of the entire investigation, and therefore, they are not yet available for this first step. On starts from the lower and upper estimates for the parameter limits, and selects uniform distributions to fulfill this requirement. If one assume all parameters $\boldsymbol{\theta}$ in Equation (2.1), those can be represented by hyperparameters $\bar{\xi}$. These are independent random variables defined on the n-dimensional unit cube:

$$K^{\mathrm{n}} = \{\bar{\boldsymbol{\xi}} : 0 \leq \bar{\xi}_{\mathrm{i}} \leq 1, i = 1, \ldots, n\}. \tag{4.9}$$

The model responses are random variables, whose total variances $D^{(\mathrm{i})}$ reads:

$$D^{(\mathrm{i})}(t) = Var[y_{\mathrm{i}}(t, \boldsymbol{\xi})] = \int_{\mathrm{K^n}} y_{\mathrm{i}}^2(t, \boldsymbol{\xi}) d\boldsymbol{\xi} - y_{\mathrm{i},0}(t), \quad i = 1, \ldots, Q, \tag{4.10}$$

where the time-depend constant $y_{\mathrm{i},0}(t)$ is the mean value of the model output i. It is possible to decompose the total variance in the above equation [62], and one can write the partial variances as follows:

$$D^{(\mathrm{i})}_{\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}(t) = \int_{\mathrm{K}^\mathrm{s}} y^2_{\mathrm{i},\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}(t,\xi_{\mathrm{j}_1},\dots,\xi_{\mathrm{j}_\mathrm{s}})d\xi_{\mathrm{j}_1},\dots,\xi_{\mathrm{j}_\mathrm{s}}, \qquad (4.11)$$

$$1 \leq j_1 < \cdots < j_\mathrm{s} \leq n, s = 1,\dots,n.$$

The summation of all partial variances is equal to the value of the total variance for each model output y_i. Based on the total variances $D^{(\mathrm{i})}$ in Equation (4.10) and the partial variances $D^{(\mathrm{i})}_{\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}$ in Equation (4.12) the Sobol' indices can computed as:

$$S^{(\mathrm{i})}_{\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}(t) = \frac{D^{(\mathrm{i})}_{\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}(t)}{D^{(\mathrm{i})}(t)}, \quad i = 1,\dots,Q,$$

and one can consider each index $S^{(i)}_{j_1,\dots,j_s}$ as a sensitivity measure indicating the ratio of the total deviation for uncertainties in the set of parameters $\{j_1,\dots,j_s\}$. The Sobol' indices $S^{(i)}_{j_1,\dots,j_s}$ are a well-known estimate for the sensitivity of the model output to its model parameters. In practice, one usually calculates the first-order and total sensitivity indices. An efficient calculation algorithm is called the FAST method [56], and the sensitivity indices obtained correspond to the first-order indices.

Based on [62], one proposes the PC expansion for the computation of Sobol' indices. The required statistics, according to Equation (3.11), can be stated as:

$$\begin{aligned} \bar{Y}^{(\mathrm{i})}(t) &= E\left[\mathcal{M}(t,\boldsymbol{\theta})\right] = \tilde{y}_{\mathrm{i},0}(t), \\ D^{(\mathrm{i})}_{\mathrm{PC}}(t) &= Var\left[\sum_{\mathrm{j}=0}^{\mathrm{P}} \tilde{y}_{\mathrm{i},\mathrm{j}}(t)\psi^{(\mathrm{i})}_{\mathrm{j}}(\boldsymbol{\xi})\right] = \sum_{\mathrm{j}=1}^{\mathrm{P}} \tilde{y}^2_{\mathrm{i},\mathrm{j}}(t) E\left[\left(\psi^{(\mathrm{i})}_{\mathrm{j}}(\boldsymbol{\xi})\right)^2\right]. \end{aligned}$$

It is possible to calculate the sensitivity indices based on the above statistics by deriving a Sobol' decomposition [62]. One can define the PC-based Sobol' indices as:

$$S^{(\mathrm{i})}_{\mathrm{PC},\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}(t) = \sum_{\boldsymbol{j}\in\mathcal{J}_{\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}} \frac{\tilde{y}^2_{\mathrm{i},\boldsymbol{j}}(t)E\left[\psi^2_{\mathrm{i},\boldsymbol{j}}\right]}{D^{(\mathrm{i})}_{\mathrm{PC}}(t)}, \quad i = 1,\dots,Q, \qquad (4.12)$$

with the following set $\mathcal{J}_{\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}}$ of $\boldsymbol{j}$ tuples such that only the indices $(j_1,\dots,j_\mathrm{s})$ are nonzero:

$$\mathcal{J}_{\mathrm{j}_1,\dots,\mathrm{j}_\mathrm{s}} = \left\{\boldsymbol{j} : \begin{array}{lll} \text{æ}_k > 0 & \exists k = 1,\dots,n, & k \in (j_1,\dots,j_\mathrm{s}) \\ \text{æ}_l = 0 & \exists k = 1,\dots,n, & k \notin (j_1,\dots,j_\mathrm{s}) \end{array}\right\}.$$

Using Equation (4.12), the Sobol' indices can be computed effortless. This can be done by reusing the PCE coefficients $\{\tilde{y}_\mathrm{i}(t)\}^Q_{\mathrm{i}=0}$ from Section 3.1.

4.2.2 Numerical results of the global sensitivity analysis

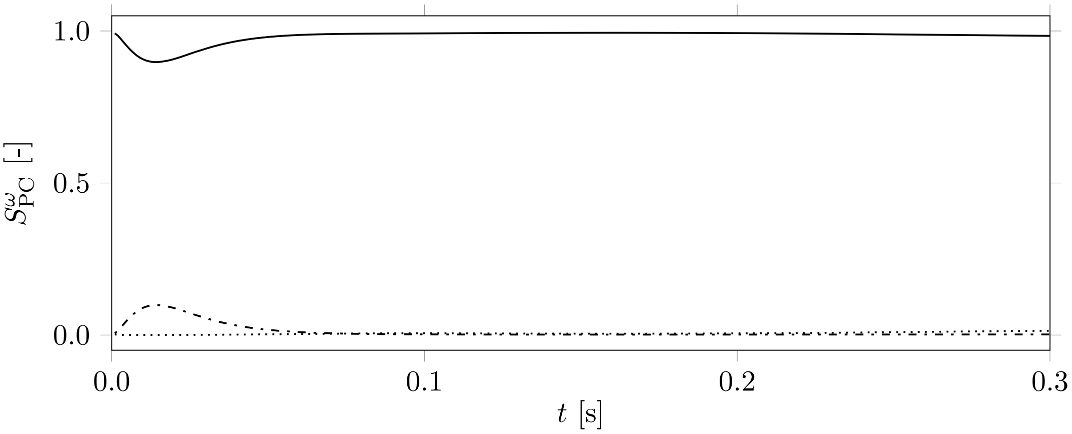
In the global SA presented here, one considers both model outputs (I and ω), and examines the voltage U, the winding resistance R, and the load torque τ_{load} as uncertain parameters. The Sobol' indices are calculated according to the Equation (4.12). Therefore, one has to define lower and upper limits for the three parameters. One approach would be to choose extensive parameter ranges and probably reach the physical limits of the model. This procedure is only useful if one does not know much about the distribution of the parameters. In practice, it is also possible to rely partly on some expert statements to better estimate the parameter limits.

In the following, one examines the two scenarios. Figure 4.1 shows the results for the large parameter space, and Figure 4.2 bases on the narrowed parameter space. If one compares the two results, the Sobol' indices do not differ significantly over time. Despite one can make a reasonable qualitative statement about their time-related influence on the QoI in both cases. One can see that the additional information of the experts shows a stronger influence of the resistance R. The reduced parameter space is examined in more detail below.

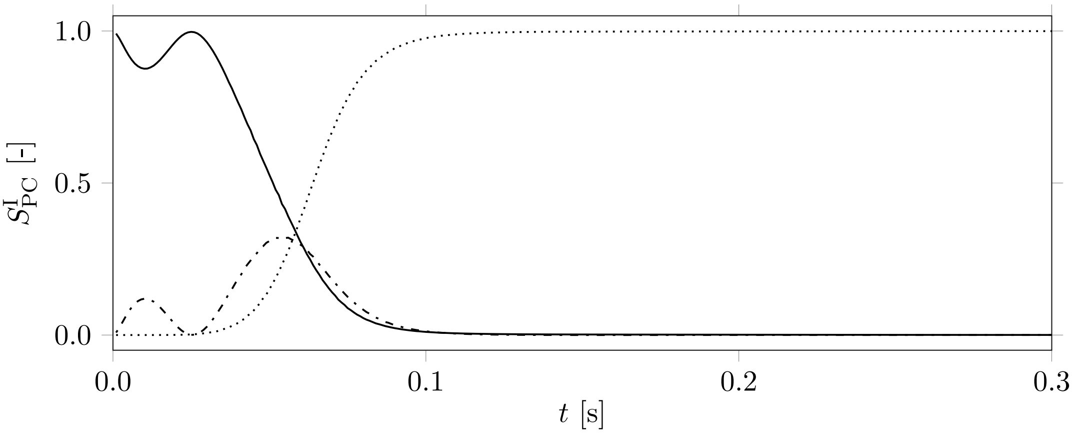
Looking at S^{ω}_{PC}, it is evident that the uncertainty in the voltage U has a considerable influence on the output speed ω. In comparison, the resistance R has less influence at the startup, and τ_{load} does not affect ω at all. The time-domain can be divided based on the signal characteristics of the Sobol' indices, and thus, the observation can be separated into a dynamic $t_{\text{dyn}} = [0, t_{\text{steady}})$ and a static $t_{\text{sta}} = [t_{\text{steady}}, T]$ time-domain. The distinction is chosen at the transition point of the output variables to the steady-state, which one calls t_{steady}. In this case, one achieves this separation at $t_{\text{steady}} = 0.1$ for the sensitivities. In summary, one can state that U is mainly sensitive in the steady-state range, and R in the dynamic range for the motor speed ω. At this point, one should note that the sensitivity of a parameter correlates directly with its Sobol' indices.

Considering the Sobol' indices S^{I}_{PC} of the output I, an analogous behavior with respect to the time ranges is observed in Figure 4.1b and Figure 4.2b. In the dynamic range mainly U and R are very sensitive while τ_{load} is almost exclusively sensitive in the static range.

Based on these results, one can assume that the areas with high sensitivity provide a high level of information for the estimation of parameter distributions. This knowledge is discussed in the following subsection and is also integrated into the Likelihood function.

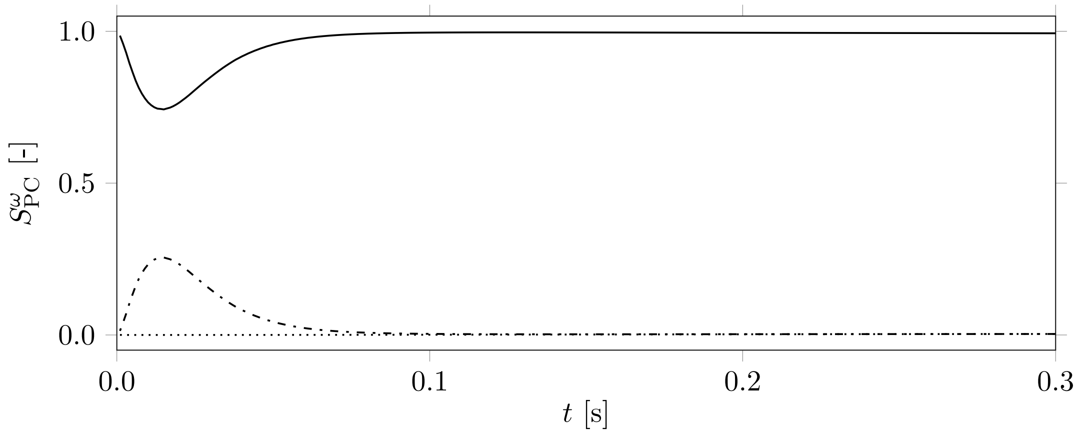


(a) Sobol' indices S^{ω}_{PC} for QoI ω.

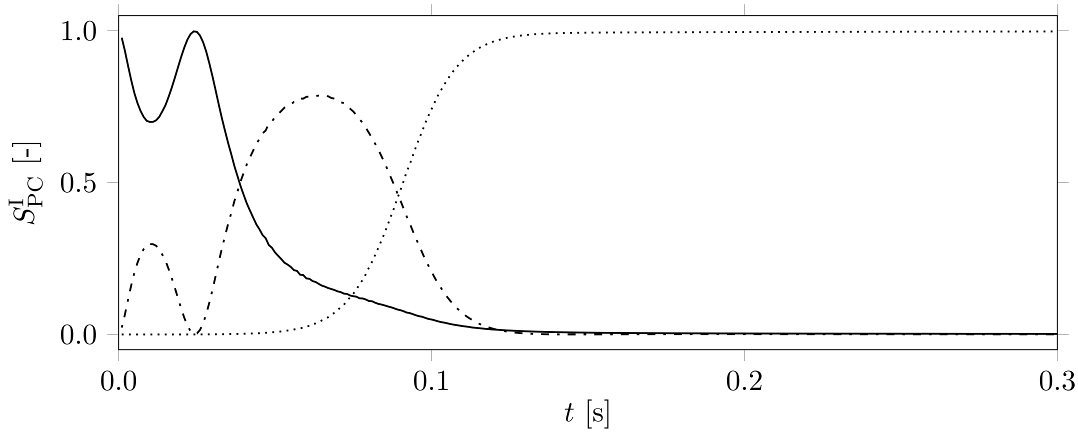


(b) Sobol' indices $S^{\mathrm{I}}_{\mathrm{PC}}$ for QoI I.

Figure 4.1: Sobol' indices of the parameter voltage U (solid line), winding resistance R (dash-dotted line) and load torque τ_{load} (dotted line) for both QoIs. The underlying PCE surrogate model uses a large parameter space.



(a) Sobol' indices S_{PC}^{ω} for QoI ω.



(b) Sobol' indices $S_{\mathrm{PC}}^{\mathrm{I}}$ for QoI I.

Figure 4.2: Sobol' indices of the parameter voltage U (solid line), winding resistance R (dash-dotted line) and load torque τ_{load} (dotted line) for both QoIs. The underlying PCE surrogate model uses a smaller parameter space, which can be obtained by considering expert knowledge.

4.2.3 Likelihood function for a one-dimensional scenario

In the one-dimensional scenario, one only considers the voltage U as uncertain. The other parameters represent a constant value close to the assumed mean value. Starting from the last subsection, one can look at the global sensitivities for both outputs (current I and motor speed ω). As already noticed, the voltage mainly influences the motor speed, and one expects the most informative content of the measurement series in the stationary time domain. As only one output is considered, one can relay on Equation (4.6), and in the first case, only one point in time in the stationary range is selected for the measurement data:

$$\tilde{L}_{\omega}(U, t^{*}_{\mathrm{sta}}) = \prod_{\mathrm{s}} f_{\epsilon}\Big(\boldsymbol{\omega}^{\mathrm{s}}_{\mathrm{meas}}(t^{*}_{\mathrm{sta}}) - \sum_{\mathrm{j}=0}^{\mathrm{P}} \omega_{\mathrm{j}}(t^{*}_{\mathrm{sta}})\psi_{\mathrm{j}}(\boldsymbol{\xi}^{\mathrm{s}}(U)) \;\Big|\; \mu_{\epsilon}, \sigma_{\epsilon}\Big), \tag{4.13}$$

where $t^{*}_{\mathrm{sta}} \in t_{\mathrm{sta}}$ is one time point in the stationary time domain.

Furthermore, the analysis can be extended to several points in time, so that a higher amount of information can be used to optimize the posterior distribution. The central question here is whether additional measurement data in the stationary area offer added value in the Likelihood construction or this information is redundant. On the one hand, one assumes that more information would lead to an increase in the quality of the Likelihood function. On the other hand, redundant information can only slow down the algorithm as a whole without improving quality or even worsen the approximation. The corresponding Likelihood function for the investigation results from this:

$$\tilde{L}_{\omega}(U) = \prod_{\mathrm{k}} \left(\prod_{\mathrm{s}} f_{\epsilon}\Big(\boldsymbol{\omega}^{\mathrm{s}}_{\mathrm{meas}}(t_{\mathrm{k}}) - \sum_{\mathrm{j}=0}^{\mathrm{P}} \omega_{\mathrm{j}}(t_{\mathrm{k}})\psi_{\mathrm{j}}(\boldsymbol{\xi}^{\mathrm{s}}(U)) \;\Big|\; \mu_{\epsilon}, \sigma_{\epsilon}\Big) \right), \tag{4.14}$$

where $k \in [0, 1, \ldots, K-1]$ is the index of the discrete time points, $t_{\mathrm{k}} \in [t_{\mathrm{steady}}, T]$ and $K = 10$.

4.2.4 Likelihood function for a multi-dimensional scenario

In addition to the voltage U, the resistance R and the load torque τ_{load} can also be described as uncertain distributions. As already mentioned in the previous subsection, the parameters R and τ_{load} are mainly sensitive at the QoI I (see Figure 4.2).

The load torque τ_{load} is similar to the voltage U concerning the SA for the QoI I, and based on Equation (4.14) the Likelihood function can be set up in the following two variants:

$$\tilde{L}_{\mathrm{I}}(\tau_{\mathrm{load}}, t^{*}_{\mathrm{sta}}) = \prod_{\mathrm{s}} f_{\epsilon}\Big(\boldsymbol{I}^{\mathrm{s}}_{\mathrm{meas}}(t^{*}_{\mathrm{sta}}) - \sum_{\mathrm{j}=0}^{\mathrm{P}} I_{\mathrm{j}}(t^{*}_{\mathrm{sta}})\psi_{\mathrm{j}}(\boldsymbol{\xi}^{\mathrm{s}}(\tau_{\mathrm{load}})) \;\Big|\; \mu_{\epsilon}, \sigma_{\epsilon}\Big),$$

where $t^*_{\mathrm{sta}} \in t_{\mathrm{sta}}$ is also one single sample in the stationary time domain and:

$$\tilde{L}_{\mathrm{I}}(\tau_{\mathrm{load}}) = \prod_{\mathrm{k}} \left(\prod_{\mathrm{s}} f_\epsilon \Big(\boldsymbol{I}^{\mathrm{s}}_{\mathrm{meas}}(t_{\mathrm{k}}) - \sum_{\mathrm{j}=0}^{\mathrm{P}} \omega_{\mathrm{j}}(t_{\mathrm{k}}) \psi_{\mathrm{j}}(\boldsymbol{\xi}^{\mathrm{s}}(\tau_{\mathrm{load}})) \;\Big|\; \mu_\epsilon, \sigma_\epsilon \Big) \right), \tag{4.15}$$

where $t_{\mathrm{k}} \in [t_{\mathrm{steady}}, T]$ and the number of discrete time points is limited to $K = 10$.

The winding resistance R is particularly sensitive in the dynamic range, and therefore, it differs slightly from the other two variables. One determines the highest information content for the Likelihood function before the steady-state (based on the SA):

$$\tilde{L}_{\mathrm{I}}(R) = \prod_{\mathrm{k}} \left(\prod_{\mathrm{s}} f_\epsilon \Big(\boldsymbol{I}^{\mathrm{s}}_{\mathrm{meas}}(t_{\mathrm{k}}) - \sum_{\mathrm{j}=0}^{\mathrm{P}} \omega_{\mathrm{j}}(t_{\mathrm{k}}) \psi_{\mathrm{j}}(\boldsymbol{\xi}^{\mathrm{s}}(R)) \;\Big|\; \mu_\epsilon, \sigma_\epsilon \Big) \right), \tag{4.16}$$

where $t_{\mathrm{k}} \in [0, t_{\mathrm{steady}}]$.

One can derive several scenarios for a global Likelihood function $\tilde{L}(\boldsymbol{\theta})$ from the Likelihood functions for the individual parameters. In the following, one considers the three most important cases for the investigation.

Firstly, one states the Likelihood function with only two parameters:

$$\tilde{L}(\boldsymbol{\theta}, t^*_{\mathrm{sta}}) = \tilde{L}_\omega(U, t^*_{\mathrm{sta}}) \tilde{L}_{\mathrm{I}}(\tau_{\mathrm{load}}, t^*_{\mathrm{sta}}), \quad \boldsymbol{\theta} = [U, \tau_{\mathrm{load}}]^{\mathrm{T}}. \tag{4.17}$$

Another scenario also consists of both parameters, but with an extended time-frame:

$$\tilde{L}(\boldsymbol{\theta}) = \tilde{L}_\omega(U) \tilde{L}_{\mathrm{I}}(\tau_{\mathrm{load}}), \quad \boldsymbol{\theta} = [U, \tau_{\mathrm{load}}]^{\mathrm{T}},$$

$$\tilde{L}(\boldsymbol{\theta}) = \tilde{L}_\omega(U) \tilde{L}_{\mathrm{I}}(\tau_{\mathrm{load}}) \tilde{L}_{\mathrm{I}}(R), \quad \boldsymbol{\theta} = [U, \tau_{\mathrm{load}}, R]^{\mathrm{T}}. \tag{4.18}$$

4.3 Modeling of physical parameters

The model used is intended to approximate the physical parameter distributions. In most cases, it is worthwhile to include the existing knowledge about the distribution information in the modeling of the parameters. This kind of method is then called hierarchical models [29]. More precisely, one uses a distribution assumption for the uncertain parameter instead of a non-parametric Bayesian model. One assumes a Gaussian distribution for the physical model by using the example of the voltage U, and the parametric model composes as follows:

$$\tilde{U}(\mu_{\mathrm{U}}, \sigma_{\mathrm{U}}, \xi_{\mathrm{U}}) = \mu_{\mathrm{U}} + \xi_{\mathrm{U}} \sigma_{\mathrm{U}}, \tag{4.19}$$

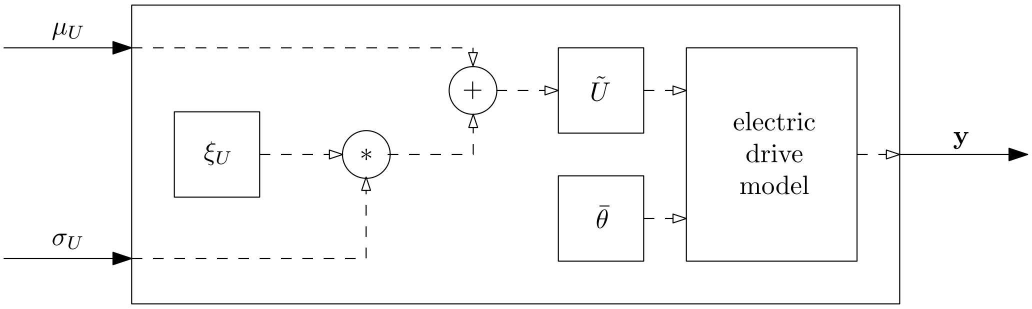


Figure 4.3: Interconnection between the parameters of the hierarchical model [32].

with the two hyperpriors μ_{U} and σ_{U}. The additional parameter ξ_{U} equals a Gaussian distribution with expected value zero and standard deviation one. Figure 4.3 shows the interconnection of all introduced parameters.

From a physical point of view, one can assume all investigated model parameters to be Gaussian distributed, and therefore, one can handle those analogously to the voltage U. One supposes that all distributions of the physical parameters $\boldsymbol{\theta}$ belong to the class of Gaussian distributed values. Besides, one can rewrite the parametric input as follows:

$$\begin{aligned} \boldsymbol{\theta} = [U, \tau_{\mathrm{load}}, R]^{\mathrm{T}}, \quad \textit{with} \qquad & U(\mu_{\mathrm{U}}, \sigma_{\mathrm{U}}, \zeta_{\mathrm{U}}) = \mu_{\mathrm{U}} + \sigma_{\mathrm{U}}\zeta_{\mathrm{U}}, \\ & \tau_{\mathrm{load}}(\mu_{\tau}, \sigma_{\tau}, \zeta_{\tau}) = \mu_{\tau} + \sigma_{\tau}\zeta_{\tau}, \\ & R(\mu_{\mathrm{R}}, \sigma_{\mathrm{R}}, \zeta_{\mathrm{R}}) = \mu_{\mathrm{R}} + \sigma_{\mathrm{R}}\zeta_{\mathrm{R}}, \end{aligned}$$

where the hyperpriors for the mean values μ_{U}, μ_{τ}, μ_{R} and the standard deviations σ_{U}, σ_{τ}, σ_{R} are specified. Also, one introduces new auxiliary variables ζ_{U}, ζ_{τ}, ζ_{R}, and they are all Gaussian distributed with expected value zero and standard deviation one.

The transformation $\boldsymbol{\xi}^{\mathrm{s}}(\cdot)$ within the PCE mentioned in the Likelihood functions in equations (4.14), (4.15) and (4.16) can be mapped as:

$$\begin{aligned} \xi_{\mu,\mathrm{U}}(\mu_{\mathrm{U}}) &= \frac{\mu_{\mathrm{U}} - E[P(\mu_{\mathrm{U}})]}{\sqrt{Var(P(\mu_{\mathrm{U}}))}}, & \xi_{\sigma,\mathrm{U}}(\sigma_{\mathrm{U}}) &= \frac{\sigma_{\mathrm{U}} - E[P(\sigma_{\mathrm{U}})]}{\sqrt{Var(P(\sigma_{\mathrm{U}}))}}, \\ \xi_{\mu,\tau}(\mu_{\tau}) &= \frac{\mu_{\tau} - E[P(\mu_{\tau})]}{\sqrt{Var(P(\mu_{\tau}))}}, & \xi_{\sigma,\tau}(\sigma_{\tau}) &= \frac{\sigma_{\tau} - E[P(\sigma_{\tau})]}{\sqrt{Var(P(\sigma_{\tau}))}}, \\ \xi_{\mu,\mathrm{R}}(\mu_{\mathrm{R}}) &= \frac{\mu_{\mathrm{R}} - E[P(\mu_{\mathrm{R}})]}{\sqrt{Var(P(\mu_{\mathrm{R}}))}}, & \xi_{\sigma,\mathrm{R}}(\sigma_{\mathrm{R}}) &= \frac{\sigma_{\mathrm{R}} - E[P(\sigma_{\mathrm{R}})]}{\sqrt{Var(P(\sigma_{\mathrm{R}}))}}, \end{aligned}$$

$$\xi_{\zeta,\mathrm{U}}(\zeta_{\mathrm{U}}) = \zeta_{\mathrm{U}}, \quad \xi_{\zeta,\tau}(\zeta_{\tau}) = \zeta_{\tau}, \quad \xi_{\zeta,\mathrm{R}}(\zeta_{\mathrm{R}}) = \zeta_{\mathrm{R}}.$$

The PCE basis function can be defined in an abstract way:

$$\psi_\mathrm{j}\left(\boldsymbol{\xi}\left(\boldsymbol{\theta}^{(\mathrm{s})}\right)\right)=\psi_\mathrm{j}\left(\xi_{\mu,\mathrm{U}},\xi_{\sigma,\mathrm{U}},\xi_{\zeta,\mathrm{U}}^{(\mathrm{s})},\xi_{\mu,\tau},\xi_{\sigma,\tau},\xi_{\zeta,\tau}^{(\mathrm{s})},\xi_{\mu,\mathrm{R}},\xi_{\sigma,\mathrm{R}},\xi_{\zeta,\mathrm{R}}^{(\mathrm{s})}\right),$$

where $j=0,\dots,P$ and $s=1,\dots,S$.

4.4 Efficiency improvement strategies

The Metropolis-Hastings algorithm, one can find in Section 4.1, relies on a random-walk and may take a long time to converge, depending on the model. Due to the presented extension by the surrogate model, one can reduce the evaluation within the model considerably. In this section, one focuses on efficiency improvements within the algorithm instead of the model evaluation. The strategies for increasing efficiency will focus on hybrid Monte Carlo methods and enhance the interaction with the surrogate model.

4.4.1 Advanced Markov Chain Monte Carlo methods

The advanced methods of the MCMC expressed in the title refer mainly to the introduction of auxiliary variables so that the next candidates are not selected randomly. In contrast, some other methods completely bypass the Likelihood function and are named as Approximate Bayesian Computation. Though, in the further course of this chapter, the focus is on the first-mentioned extensions.

One adaptation of the MCMC that is still widely used is the so-called Hamiltonian Monte Carlo (HMC) method [20, 11]. There, one uses Hamiltonian dynamics to obtain a sample from the distribution. It is necessary to express the density function by a potential energy function and introduce so-called momentum variables to transform the original variables in the new setting. This adaptation allows the simulation of a Markov chain, which in each iteration relies on the Hamiltonian dynamics and calculates a proposal for the Metropolis update.

One can link the distribution from which one wants to take a sample to a potential energy function. The concept of a canonical distribution from statistical mechanics is the basis of the given relation, and it is mainly adopted from [11]. The canonical ensemble over the states x of a physical system of an energy function $E(x)$ yields the following probability density function:

$$P(x)=\frac{1}{Z}\exp\frac{-E(x)}{T},$$

where T is the temperature of the system and Z is a normalizing constant. If one replaces the energy function with Hamilton's $H(p,q)$, where q is the space coordinate, and p is the momentum, then the equation can be rewritten as:

$$P(q,p) = \frac{1}{Z} \exp \frac{-H(q,p)}{T}. \tag{4.20}$$

The Hamiltonian can represent the total energy of the system, which is the sum of kinetic $K(p)$ and potential energy $U(q)$. The joint density then results:

$$P(q,p) = \frac{1}{Z} \exp \frac{-K(p)}{T} \exp \frac{-U(q)}{T}, \tag{4.21}$$

where K is an energy function of p alone, while U is an energy function of q alone. One will use q to represent the model parameters of interest and utilize p as related to Hamiltonian dynamics. Using the following potential energy function:

$$U(q) = -\log\left(p(q)L(q|\boldsymbol{Y}_{\text{meas}})\right), \tag{4.22}$$

one can express the posterior distribution as a canonical distribution. This statement is given for the case $T = 1$ with the use of the prior information $p(q)$ and the Likelihood function $L(q|\boldsymbol{Y}^{\text{obs}})$. When choosing the kinetic energy function with the related momentum variable p one is more independent and with regard to HMC it is usual to assume a quadratic kinetic energy [11]. This results in a zero-mean multivariate Gaussian distribution for p and in the ordinary case the individual components i of p are assumed to be independent with variance m_i:

$$K(p) = \sum_i \frac{p_i^2}{2m_i}. \tag{4.23}$$

If one considers the HMC method, it is possible to separate each iteration within the algorithm into two phases. In the first phase, one only changes the momentum, and in the second, both position and momentum are adjustable. The canonical distribution $P(q,p)$ remains invariant in the phases [11]. As a first step, one assumes independent momentum variables, and for their distribution, one supposes a mean value of zero and variance of m_i. Afterward, in the second step, a Metropolis update is performed. A suggestion for the new step takes place using the Hamiltonian dynamic. Based on the current state (q,p), one calculates the Hamiltonian dynamics for L steps with a chosen step size ϵ. One common method for obtaining the results is the Leapfrog integration, and both introduced parameters of this integration algorithm are essential for achieving a good performance of the overall computation. Giving the obtained trajectory, one selects the last values of the momentum variable and negates them. The new state

(q^*, p^*) is used as a proposal $x_{\mathrm{n}'}$ within the MCMC algorithm in Subsection 4.1.4 and it will be accepted with the probability:

$$\alpha = \min\left[1, \exp\left(-U(q^*) + U(q) - K(p^*) + K(p)\right)\right]. \tag{4.24}$$

Due to the property of the Leapfrog integrator to preserve the volume and the time reversibility of the Metropolis proposal, (q^*, p^*) is a legitimate proposal. Assuming that one can calculate the Hamiltonian dynamic accurately, the acceptance probability α is one. This characteristic is since the energy within the Hamiltonian system is preserved. The error resulting from the discretization has a significant influence on the acceptance then and depends strongly on the introduced step size parameter ϵ.

If one examines the performance of the HMC method more closely [35], one can see that not only the step size ϵ but also the number of steps L have a significant influence. If one chooses ϵ too large, the high simulation error leads to many rejections of the proposals, whereas a small step size increases the calculation time considerably. One can make similar observations for L. If the step length is too large, HMC generates trajectories that repeat their steps. If the step size is too small, samples are generated, which show a strong correlation and these samples are close to another. This behavior corresponds then strongly to a random walk and does not reflect the desired performance.

The key is to adjust the above two parameters of the algorithm to achieve the best possible performance. The challenge is that the parameters are highly sensitive to the problem to be solved, and there exists no optimal initial setting or initial parameter values. To set the step size ϵ and the number of steps L, one, needs experience and usually additional runs. To enhance the applicability of the methodology in the industrial context, it is necessary to automate the choice of the algorithmic parameters of the HMC. Hoffman and Gelman suggest one possible solution, and they call it the "No-U-Turn Sampler" (NUTS) method [35].

One can find all detailed information in [35], and in the following, one will only briefly outline the basic idea of the NUTS method. In the first step, one attempts to select the number of leapfrog steps L. Using the method, one generates a proposal with the initial position based on the parameters from the previous iteration. Then, the algorithm generates an independent moment vector from a standard normal distribution, and it develops the Hamiltonian simulation forward and backward in time to ensure reversibility. This procedure results in a balanced binary tree whose nodes correspond to the position and momentum states. Each iteration within the algorithm increases this tree depth by one, thus doubling the number of leapfrog steps. The iteration ends when the sub-trajectory starts doubling itself from the left to the right nodes of any balanced subtree within the entire binary tree. In the second step, Hoffman and Gelman [35] propose to use a stochastic optimization with vanishing adaptation to adjust the step size ϵ for the NUTS and HMC methods. With these extensions, one achieves a real good automatic adjustment of the HMC algorithm parameters, and as a result, the applicability enhances considerably. In terms of calculation costs, the central part of the computation is the calculation of the gradients within the leapfrog method. In the following, one proposes a combination with the previously introduced surrogate model to increase efficiency.

4.4.2 Derivations using the surrogate model

For the implementation, one has to approximate the Hamilton equations by discretization of the time using the step size ϵ. As mentioned before, one uses the leapfrog method for this purpose, and the equations for updating position and velocity are [11]:

$$p_i\left(t+\frac{\epsilon}{2}\right)=p_i(t)-\frac{\epsilon}{2}\frac{\partial U\left(q(t)\right)}{\partial q_i}, \tag{4.25}$$

$$q_i(t+\epsilon)=q_i(t)+\epsilon\frac{p_i(t+\frac{\epsilon}{2})}{m_i}, \tag{4.26}$$

$$p_i(t+\epsilon)=p_i(t)-\frac{\epsilon}{2}\frac{\partial U\left(q(t+\epsilon)\right)}{\partial q_i}. \tag{4.27}$$

Looking at the NUTS algorithm as a whole, one can assume that the calculation of gradients, in particular, addresses a significant part of the overall calculation. Starting from the previous equations (4.25), (4.26), and (4.27), one has to calculate the gradient of the potential energy function with respect to the position variables. With Equation (4.22), one can express the gradient as:

$$\frac{\partial U\left(q\right)}{\partial q_i}=\frac{\partial}{\partial q_i}-\log\left(p(q)L(q|\boldsymbol{Y}_{\text{meas}})\right).$$

One can now exemplarily take the Likelihood function for one parameter θ and S measurements from Equation (4.4) and calculate the gradient for the Log-Likelihood as follows:

$$\begin{aligned}-\frac{\partial}{\partial\theta}\log\left(L(\theta|\boldsymbol{Y}_{\text{meas}})\right)=&-\frac{\partial}{\partial\theta}\log\left(\left(\frac{1}{\sqrt{2\pi\sigma_\epsilon^2}}\right)^{\mathrm{S}/2}\right.\\ &\left.\exp\left(-\frac{\sum_{s=1}^{\mathrm{S}}\left(\boldsymbol{Y}_{\text{meas}}^{\mathrm{s}}-\mathcal{M}(\theta)\right)^2}{2\sigma_\epsilon^2}\right)\right)\\ =&\frac{1}{\sigma_\epsilon^2}\sum_{s=1}^{\mathrm{S}}\left(\boldsymbol{Y}_{\text{meas}}^{\mathrm{s}}-\mathcal{M}(\theta)\right)\frac{\partial\mathcal{M}(\theta)}{\partial\theta}.\end{aligned} \tag{4.28}$$

Similarly, the gradient of the Log-Likelihood function for the variance σ_ϵ results as:

$$\begin{aligned}-\frac{\partial}{\partial \sigma_\epsilon}\log\left(L(\theta|\boldsymbol{Y}_{\text{meas}})\right) = &-\frac{\partial}{\partial \sigma_\epsilon}\log\left(\left(\frac{1}{\sqrt{2\pi\sigma_\epsilon^2}}\right)^{\mathrm{S}/2}\right.\\ &\left.\exp\left(-\frac{\sum_{s=1}^{\mathrm{S}}\left(\boldsymbol{Y}_{\text{meas}}^{\mathrm{s}}-\mathcal{M}(\theta)\right)^2}{2\sigma_\epsilon^2}\right)\right) \\ =&\frac{1}{\sigma_\epsilon^3}\sum_{\mathrm{s}=1}^{\mathrm{S}}\left(\boldsymbol{Y}_{\text{meas}}^{\mathrm{s}}-\mathcal{M}(\theta)\right)^2-\frac{S}{2\sigma_\epsilon^2}.\end{aligned} \tag{4.29}$$

Both equations (4.28) and (4.29) involve model evaluations at parameter values, whereas the gradient for the parameter also requires the partial derivatives of the model function. One can easily calculate this gradient using a finite difference method. However, this requires many more evaluations on the model, and this procedure means a loss of efficiency. To make the method operate effectively, one proposes to use the Chaos polynomials from the surrogate model to calculate the derivatives.

Taking Equation (3.13) into account, one can approximate the derivative of the model with the replaced surrogate as follows:

$$\frac{\partial\mathcal{M}(\theta)}{\partial\theta}\approx\frac{\partial\boldsymbol{Y}^P(\xi(\theta))}{\partial\theta}=\frac{\partial}{\partial\theta}\sum_{j=0}^{P}\boldsymbol{y}_j\psi_j(\xi(\theta)) \tag{4.30}$$

where $\boldsymbol{\xi}$ represents the PCE hyperparameters and those are related to the model parameter θ given the transformation in Section 4.3. The derivative of the surrogate model thus results in:

$$\frac{\partial}{\partial\theta}\sum_{j=0}^{P}\boldsymbol{y}_j\psi_j(\xi(\theta))=\sum_{j=0}^{P}\boldsymbol{y}_j\frac{\partial}{\partial\xi}\psi_j(\xi(\theta))\frac{\partial}{\partial\theta}\xi(\theta). \tag{4.31}$$

Depending on the chosen probability distribution for the parameter θ one gets corresponding gPC basis polynomials (see Table 3.1), which then can be differentiated. Furthermore, one has to consider the inner derivative that results from introducing the hyperparameters. If one assumes a Gaussian distribution as an example, one receives the following statements for the derivatives:

$$\begin{aligned}\frac{\partial}{\partial\xi}\psi_j(\xi(\theta)) =&\frac{\partial}{\partial\xi}H_j(\xi(\theta))=jH_{j-1}(\boldsymbol{\xi}(\theta)),\\ \frac{\partial}{\partial\theta}\xi(\theta) =&\frac{1}{\sigma_\theta},\quad \text{with } \xi(\theta)=\frac{\theta+\mu_\theta}{\sigma_\theta},\end{aligned} \tag{4.32}$$

where H_n are Hermite polynomials with order n. One transforms the parameter θ into the hyperparameter ξ and thus one gets the splitting into the mean value μ_θ and the standard deviation σ_θ.

4.5 Numerical results

In this section, one can find the application cases with one uncertain parameter and the realization for several parameter variations. The focus is mainly on the different properties of the Likelihood function, the surrogate model for the approximation of the original model, and the optimal amount of information from the measurements.

The underlying measurement data are determined as described in Chapter 2. Depending on the scenario, one varies the corresponding parameters and records measurement signals of the duration $\delta t = 10$. The number of measurements is $S = 100$, and all of them passed the mentioned temperature test.

One can divide the procedure into five steps. First, one starts with the analysis of the measurements and extracts the statistical data. Also, the theoretical distributions are compared with the assumed ones. In the second step, one defines the distribution assumptions for the parameters and determines them according to the hierarchical approach. A posteriori distribution of the parameters on the original model can then be calculated. These values will later serve as a reference distribution. In the fourth step, the simulation model is exchanged with a suitable surrogate model, and the posterior distributions are approximated on this basis. In the last step, the obtained results are compared and linked to the predefined and measured distributions.

4.5.1 One-dimensional scenario

As already introduced in Subsection 4.2.3, the uncertain voltage U is considered in the one-dimensional scenario. This parameter is approximated based on the measured data of the motor speed ω. One employs Equation (4.13) with a stationary time t^*.

In the first step, one analyzes the available measurement data which can be used to extract the distribution of the parameter U. Those variations do not have to correspond exactly to the specification of the Gaussian distribution since one only considers $S = 100$ random points. In this case, one defines the statistical moments with mean value $\mu(U_{\text{real}}) = 0.84$ and standard deviation $\sigma(U_{\text{real}}) = 0.044$. Figure 4.4 shows the distribution used for the test bench. The mean value is $\mu(U_{\text{tb}}) = 0.85$ and the standard deviation is $\sigma(U_{\text{tb}}) = 0.043$ which results in the following error deviations: $e_{\mu(\text{U})} = 0.58\%$ and $e_{\sigma(\text{U})} = 0.79\%$. These values are very good for the present study and are used below as a reference to evaluate the quality of the numerical approximations.

During the next step, it is necessary to consider the measured data of the motor speed ω. One obtains the stationary data by measuring the motor speed at the time $t^* = 5\text{s}$ for each measurement in the overall series. This procedure results in the

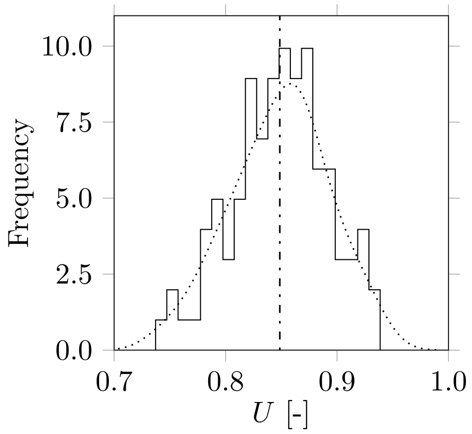


Figure 4.4: Distribution for parameter U based on test bench sampling of a theoretically assumed distribution. Using the histogram data (solid line) a PDF is approximated (dotted line) and the mean value (dash-dot line) is visualized.

histogram which Figure 4.5a shows. The distribution of the output ω results mainly from the variation of the parameter U. One also expects that the measurements are influenced by noise. The influence of the measurement noise ϵ can be approximated by determining the variation of each measurement in the stationary range:

$$\sigma_{\epsilon}^{\mathrm{s}} = \sigma\Big(\omega^{\mathrm{s}}(t_0), \omega^{\mathrm{s}}(t_1), \ldots, \omega^{\mathrm{s}}(t_{\mathrm{T}})\Big), \tag{4.33}$$

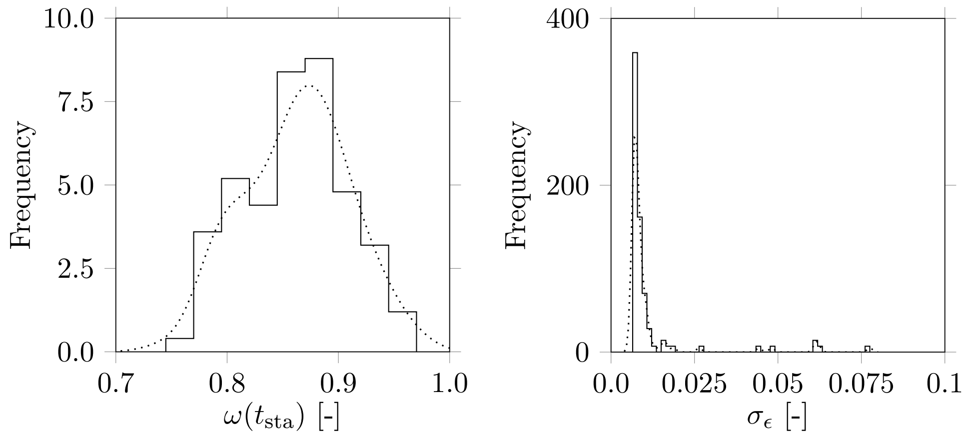
where $s \in [1, \ldots, S]$ are the measurements and $t_0, t_1, \ldots, t_{\mathrm{T}}$ are time points in the stationary range. The histogram of the standard deviation of the measurement is shown in Figure 4.5b. Based on the maximum Likelihood estimate (MLE), the measurement noise is approximated with $\sigma_{\epsilon} = 0.0116$.

There are some additional assumptions needed for the two hyperpriors μ_{U} and σ_{U} of the hierarchical model in Equation (4.19). One assumes that less information about the distribution is available, and experts can give an estimate of the lower and upper limits. Therefore, one selects a non-informative priority in the form of a uniform distribution for both parameters:

$$P(\mu_{\mathrm{U}}) = \mathcal{U}(0.63, 0.94), \tag{4.34}$$

$$P(\sigma_{\mathrm{U}}) = \mathcal{U}(0.025, 0.125). \tag{4.35}$$

The Likelihood function introduced in Equation (4.4) contains the mean μ_{ϵ} and the standard deviation σ_{ϵ} of the additive error ϵ. One assumes that there is no bias, and



(a) Histogram of the measured output ω at a stationary time area (solid line) and the corresponding PDF (dotted line).

(b) Histogram of the output variation ω at a stationary time area (solid line) and the corresponding PDF (dotted line).

Figure 4.5: Analysis of the available measurement data of the test bench for the one-dimensional scenario.

the mean value μ_ϵ can, therefore, be expected to be zero. The prior of the standard deviation σ_ϵ is modeled as a half-normal distribution as follows:

$$P(\sigma_\epsilon) = \frac{\sqrt{2}}{\sigma^* \sqrt{\pi}} \exp\left(-\frac{\sigma_\epsilon^2}{2\sigma^{*2}}\right). \tag{4.36}$$

The recent investigation consists of two calculations. In the first case, one performs an MCMC computation on the original model. In the second case, one replaces the simulation model by a suitable surrogate model. The quality of the posterior distributions is then of interest for the evaluation of the approaches. One can directly compare the obtained results with the assumed distributions of the test bench by using some appropriate quality metrics. It is also interesting to consider the number of calculations on the original model and to examine the average computing time of both approaches.

Figure 4.6 shows the numerical results of the posteriors with an MCMC approach using the original simulation model. In this case, a Markov chain with $1,000$ samples is generated, and the resulting traces are visualized in the figures 4.6b and 4.6d. In this analysis, one omits the so-called burn-in, which means that one considers all samples, even those at the beginning. As a result, the obtained trace allows the observation of a possible convergence of the chain. However, if one extracts the distribution information from the chain, it is recommended to dispose the samples at the beginning. According to figures 4.6b and 4.6d, it can be seen that the chains converge very quickly to a constant value. Compared to the theoretically assumed statistical values used to generate the measurement data, these results provide a good approximation.

The PDFs can be calculated based on the chains and are displayed in the figures 4.6a and 4.6c. One can determine the hyperpriors with the highest probability via the MLE, and then, they serve to reconstruct the underlying hierarchical distribution. The variance of the distribution function, on the other hand, indicates to what extent this estimate is reliable. The smaller the variation, the more likely the Markov chain is converged. In this case, however, the variation is very inadequate, and one can assume that the chain is already converged.

The following approximations can be derived from the numerical results:

$$\begin{aligned} \mu_{\mathrm{MLE}}(U) &= 0.840, \\ \sigma_{\mathrm{MLE}}(U) &= 0.0450. \end{aligned}$$

In a direct comparison with the measured stochastic information of the parameter U, the results obtained are qualitatively very satisfactory.

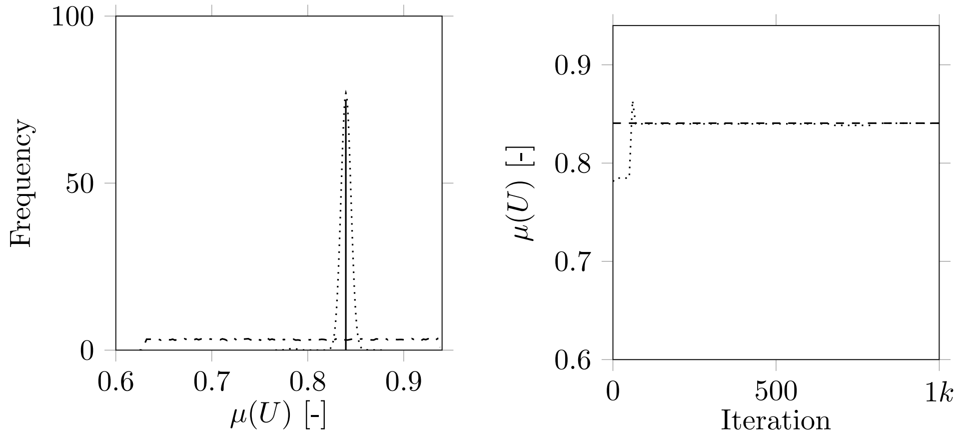
A further measure of the consideration is the efficiency of the method in addition to the quality. To obtain a reliable statement and to minimize possible disturbances, one executes the algorithms several times and calculates the mean value. In the following, the different approaches are presented to determine the efficiency of the methods.

The simulation model has an evaluation time of about two seconds. This value can also vary depending on the selected parameter distributions. In comparison, the computation efforts for the introduced methods, such as the evaluations within the MCMC algorithm, are much smaller than the execution of the simulation model. Besides, the simulation model must be evaluated very frequently in the implemented workflow. Therefore, one assumes that a consideration of the simulation time and the number of required evaluations represent a reasonable estimate for the efficiency. One should also note that all investigations are performed on the same computer to avoid the influence of different hardware. The algorithms are executed several times, and only the average time is used for the presented analysis.

The hierarchical model (see Equation (4.19)) requires the simulation model to be re-evaluated within each proposal of every single measurement. One performs $N = 100$ variations for each proposal. The average simulation time is:

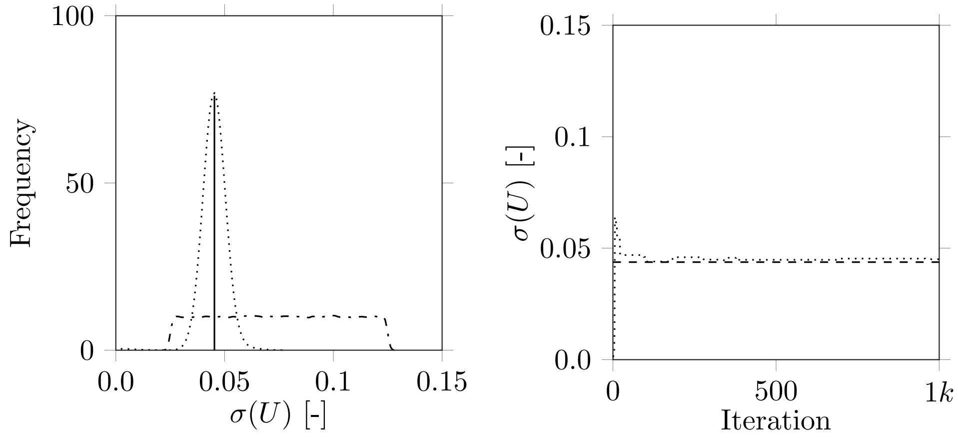
$$\Delta t_{\mathrm{sim}} = 2.11\,[\mathrm{s}],$$

resulting in an evaluation time for one proposal of $\Delta t_{\mathrm{prop}} = 210$ seconds.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows the theoretical value.

Figure 4.6: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the original model.

For performance reasons, it was necessary to parallelize the sequential evaluation at this point. One achieves this requirement by distributing the individual calculations over 25 CPUs. This approach reduced the average calculation time for a proposal to about $\Delta t_{\text{prop}} = 9.08$ seconds. Of interest for the complete evaluation is the time required for the average number of proposals per iteration:

$$\Delta t_{\text{iter}} = 43.52\,[\text{s}].$$

Another way to speed up the calculation is to replace the original model with a PCE surrogate model. One can find the numerical results in Figure 4.7. The traces (see figures 4.7b and 4.7d) behave in the same way as in the previous study. The only difference is the higher number of iterations. This behavior occurs because the approximation of the model most likely causes a significant error that negatively affects the convergence of the chain. One can see that the chains converge very quickly to the predefined value range. Compared to the evaluation of the original model, one can show here that the measurement oscillates much more strongly. This behavior can be well observed in the figures 4.7a and 4.7c. The deviations are reflected in the PDFs by a larger variation of the distribution. This study also shows a good convergence, and one can obtain the following statistical information from the results:

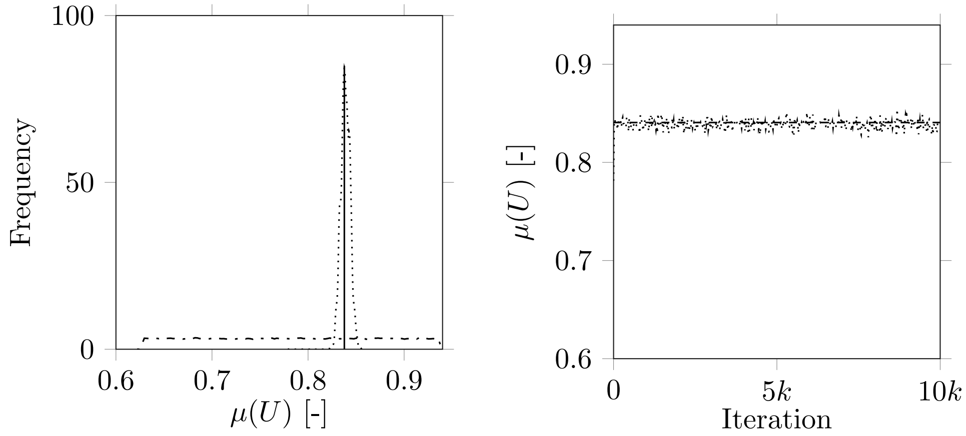
$$\begin{aligned} \mu_{\text{MLE}}(U) &= 0.838, \\ \sigma_{\text{MLE}}(U) &= 0.0452. \end{aligned}$$

Considering the efficiency of the surrogate model approach, an average $t = 0.0375$ seconds is needed for a proposal to be evaluated. At this point, one can considerably simplify the calculation with vectors, and omit a statement for single model evaluation. However, the surrogate model requires more evaluations, and the average time for an iteration adds up:

$$\Delta t_{\text{iter}} = 1.09\,[\text{s}].$$

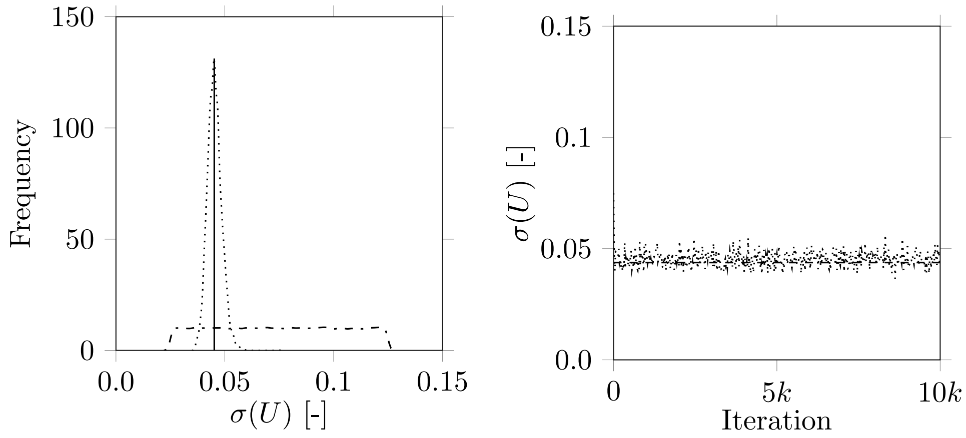
In addition to the average calculation time for an iteration, it is necessary to consider the simulation time on the original model for generating the surrogate model. In this case, three parameters are required for the hierarchical approach ($N = 3$), and one uses a PCE with the order $P = 3$. This selection leads to 105 necessary evaluations on the original model. Since these simulations can also be parallelized, the total time is $t_{\text{surrogate}} = 12.55\,[\text{s}]$ seconds for 25 CPUs.

Finally, one can consider the variance of the Likelihood function of the two investigations. Note that the prior assumptions based on Equation (4.36) are selected as



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows theoretical value.

Figure 4.7: Numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the surrogate model.

Table 4.1: Overview of numerical results for one-dimensional scenario.

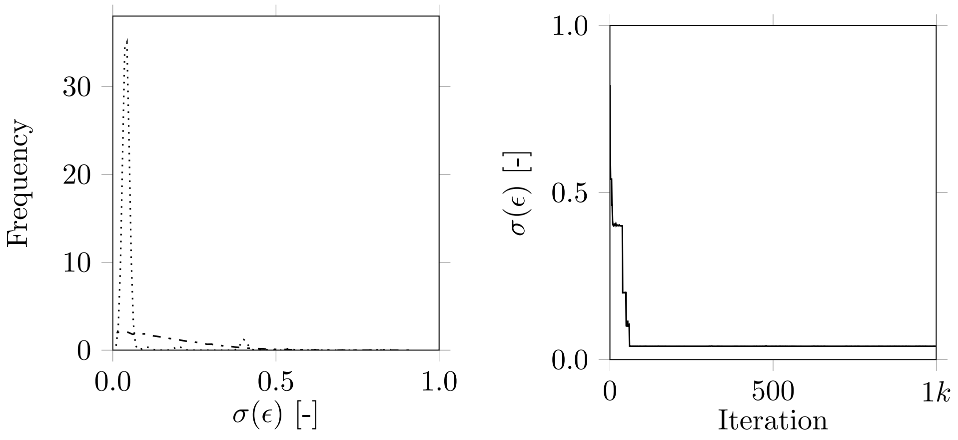
Simulation model	S	K	Δt_{iter}	$\mu(U)$	$\sigma(U)$
Original model	100	1	43.52	0.840	0.0450
Surrogate model	100	1	1.09	0.838	0.0452
Surrogate model	10	10	1.49	0.842	0.0429

follows. For the MCMC algorithm based on the original model, $\sigma^* = 0.2$ is selected. For the approximation with the PCE surrogate model, a larger variance $\sigma^* = 0.5$ is selected for the prior distribution. The numerical results are summarized in Figure 4.8. As with the mean and the standard deviation of the parameter U, one can see the analog behavior of convergence. While convergence with the original model is achieved quickly (see figures 4.8b and 4.8a), with the surrogate model it takes longer for the trace values to decrease (see figures 4.8d and 4.8c).

In some situations, the number of available measurements is smaller than the presented investigations, but there are several time points. In another study, one reduces the number of measurement runs to $S = 10$ and increases the number of discrete time points in the stationary range to $K = 10$. The numerical results were obtained with the Likelihood function in Equation (4.14) and are shown in Figure 4.9.

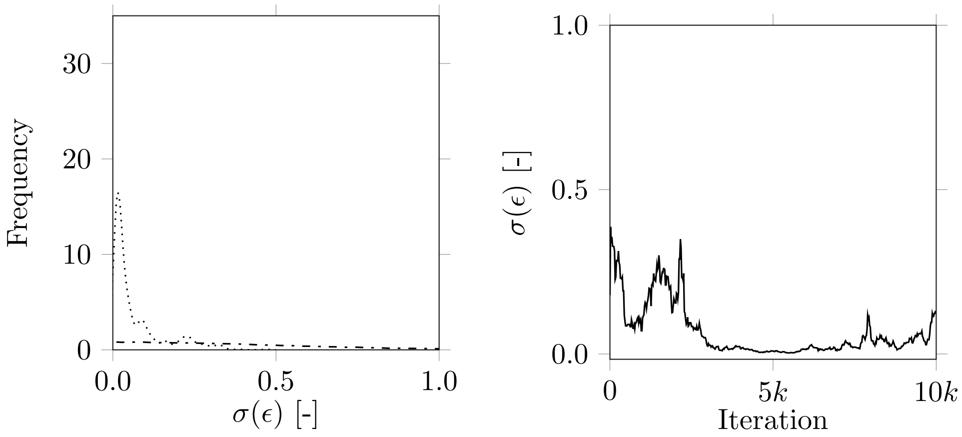
Based on the results, one can observe an improvement of the approximation. It turns out that the hyperparameter $\sigma(U)$ converges only after about $3,000$ iterations. This performance can be plausibly justified by the smaller number of measurements used. At the same time, the estimation of the additive error is considerably improved by the addition of further time points (see Figure 4.10).

Table 4.1 summarizes the one-dimensional scenario. One can show that the MCMC approach is an appropriate candidate for real measurements of the test bench, and in combination with the original model, one can achieve excellent results. It was also shown that the use of a PCE approach to build a surrogate model is beneficial to reduce the required computing time. In this scenario, it was possible to reduce the evaluation by a factor of 40 in contrast to the original simulation model. The number of measurements used also depends on the convergence speed of the standard deviation. If one uses more time points for each measurement, one can observe a faster convergence of the additive error.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the error ϵ based on the original model.

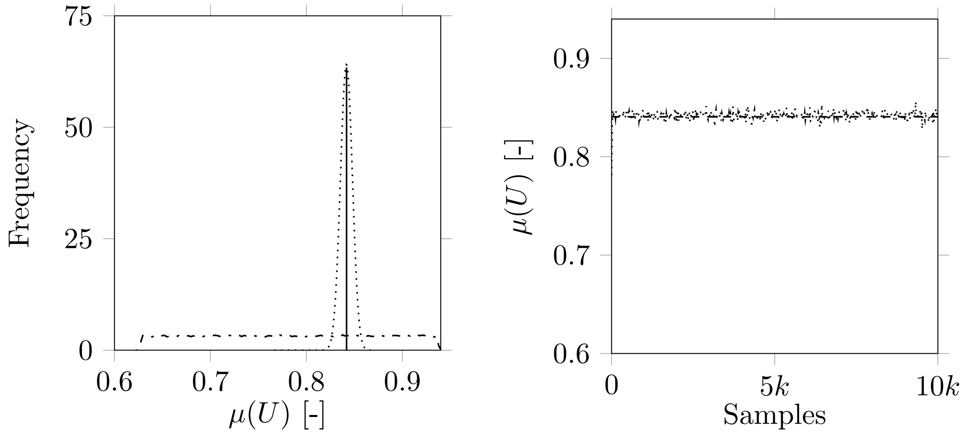
(b) Trace of the Markov chain for $\sigma(\epsilon)$ based on the original model.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the error ϵ based on the surrogate model.

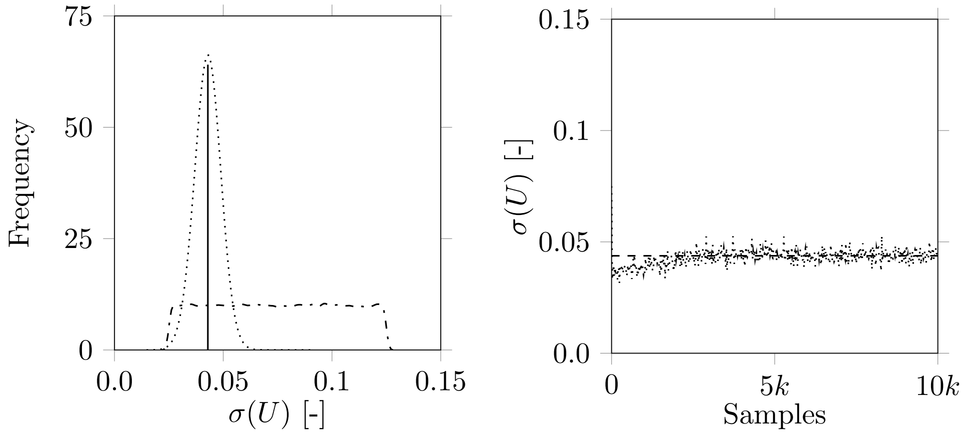
(d) Trace of the Markov chain for $\sigma(\epsilon)$ based on the surrogate model.

Figure 4.8: Standard deviation for the additive error assumption for the chosen Likelihood function for the MCMC evaluation on the original and the surrogate model.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

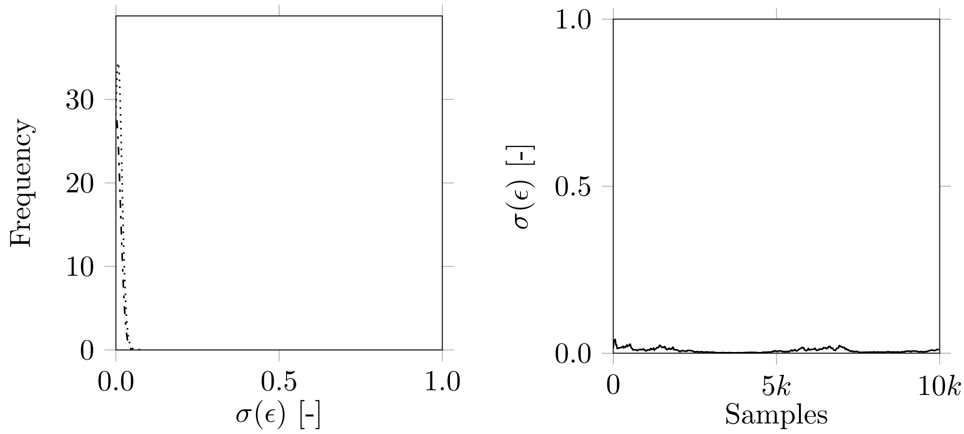
(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows theoretical value.

Figure 4.9: Numerical result of the MCMC estimation for the one-dimensional scenario with time series. The computation was performed on the surrogate model.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the error ϵ based on the original model.

(b) Trace of the Markov chain for $\sigma(\epsilon)$ based on the original model.

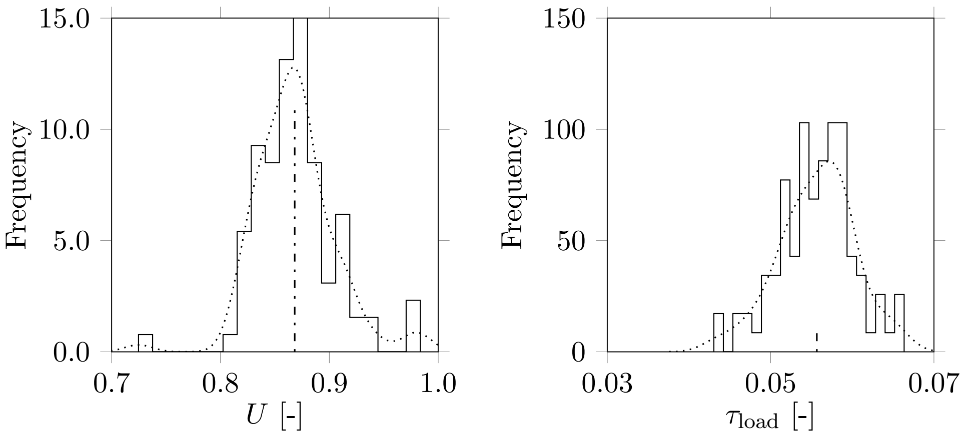
Figure 4.10: Standard deviation for the additive error assumption for the chosen Likelihood function for the MCMC evaluation on the surrogate model with time series.

4.5.2 Multi-dimensional scenario

In the multi-dimensional scenario, one first considers the two parameters U and τ. This sequence is since both uncertain parameters have the highest sensitivity in the stationary range. One examines the resistance R separately.

As in the previous subsection, one examines the available measurements. For the voltage U, the same values from the one-dimensional case were assumed. One selects a normal distribution with the mean value $\mu(\tau_{\text{real}}) = 0.0556$ and the standard deviation $\sigma(\tau_{\text{real}}) = 0.0044$ for the load torque τ. For the design of experiment on the test bench hardware, one selects 100 sample points, and Figure 4.11 visualizes the resulting distributions. The statistical information for the parameter U is $\mu(U_{\text{tb}}) = 0.86, \sigma(U_{\text{tb}}) = 0.036$ and the deviations are $e_{\mu(\text{U})} = 2.90\%$, $e_{\sigma(\text{U})} = 16.06\%$. For the parameter τ the following values are obtained: $\mu(\tau_{\text{tb}}) = 0.0557$, $\sigma(\tau_{\text{tb}}) = 0.0046$, $e_{\mu(\text{U})} = 0.19\%$, and $e_{\sigma(\text{U})} = 4.41\%$. The standard deviation of U shows that the amount of measurements is not sufficient to describe the predefined distributions well. This situation should be taken into account when evaluating the results.

In addition to the output ω, I is also considered in this investigation, and the stationary data at the time $t^* = 6s$ for both outputs are considered. The resulting



(a) Uncertain parameter U.

(b) Uncertain parameter τ.

Figure 4.11: Distribution for uncertain parameters based on test bench samples of theoretically assumed distributions. Using the histogram data (solid line) a PDF is approximated (dotted line) and the mean value (dash-dotted line) is visualized.

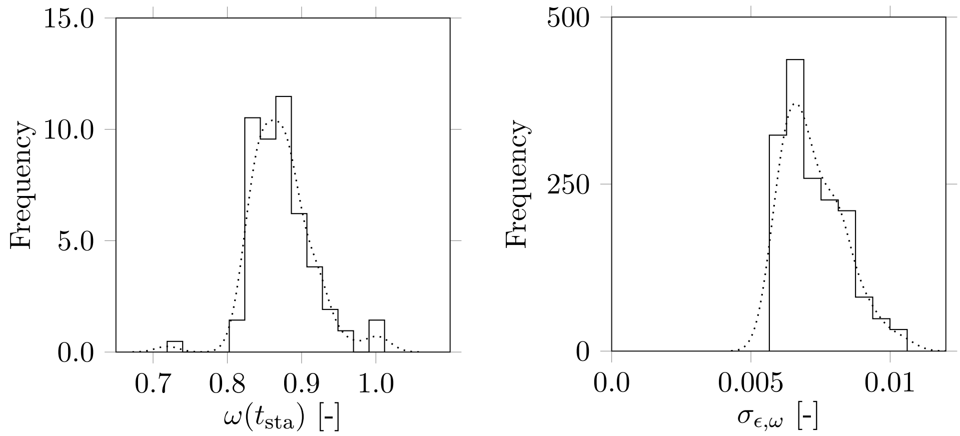
histograms are shown in Figure 4.12. It is assumed that both output distributions (see figures 4.12a and 4.12c) are caused by the uncertain parameters.

Equation (4.33) can be used to calculate the measurement error σ_ϵ for ω, and the resulting variation is shown in Figure 4.12b. This procedure can be applied directly to the output I and Figure 4.12d shows the result. The measurement noise is estimated with $\sigma_{\epsilon,\omega} = 0.0011$ and $\sigma_{\epsilon,\mathrm{I}} = 0.0007$.

Based on the equations (4.34) and (4.35) the assumptions on the distributions for the hyperpriors can be extended. For the parameter U, one uses the same uniform distributions as in the one-dimensional scenario. For the parameter τ, a non-informative prior is also selected based on the hierarchical model. The resulting uniform distributions are:

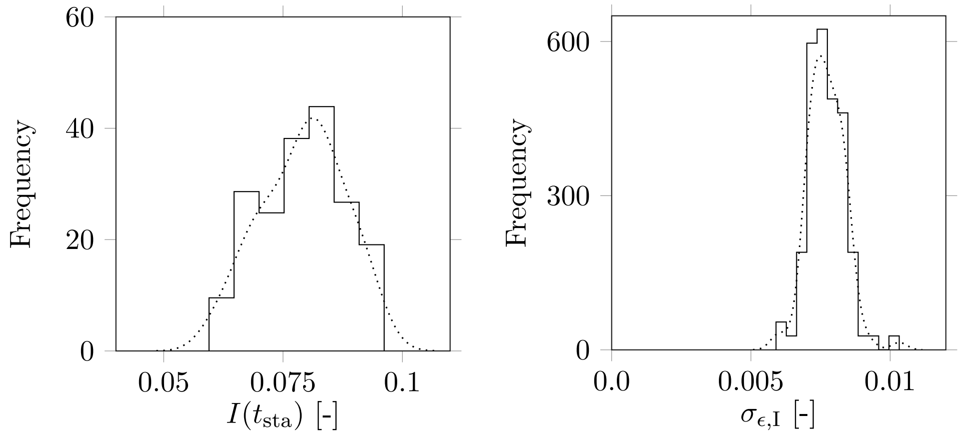
$$\begin{aligned} P(\mu_\tau) &= \mathcal{U}(0.033, 0.078), \\ P(\sigma_\tau) &= \mathcal{U}(0.0, 0.0133). \end{aligned}$$

The two Likelihood functions for the output ω and I are combined. Both have the standard deviation of the additive error ϵ. In the following, these errors based on Equation (4.17) are noted as $\sigma_{\epsilon,\omega}$ and $\sigma_{\epsilon,\mathrm{I}}$. For the prior assumptions, one also use half-normal distributions from the one-dimensional scenario (see Equation (4.36)).



(a) Histogram of the measured output ω at a stationary time area (solid line) and the corresponding PDF (dotted line).

(b) Histogram of the output variation ω at a stationary time area (solid line) and the corresponding PDF (dotted line).



(c) Histogram of the measured output I at a stationary time area (solid line) and the corresponding PDF (dotted line).

(d) Histogram of the output variation I at a stationary time area (solid line) and the corresponding PDF (dotted line).

Figure 4.12: Analysis of the available measurement data of the test bench for the multi-dimensional scenario.

As in the previous subsection, the investigation consists of two parts: First, one performs an MCMC calculation on the original model, and one run a second calculation on the surrogate model. In both cases, one can compare the numerical results with the measured distributions of the test bench. The figures 4.13 and 4.14 show the numerical results for the approximated parameters. These numerical results were calculated on the original model, and the Markov chain is created with $1,000$ samples, but in this particular case, one defines a burn-in of 200 samples. This is also shown by the fact that the respective traces in the figures 4.13b, 4.13d, 4.14b and 4.14d have already reached a stationary value, and the chains are therefore converged. The comparison with the test bench also shows a good approximation of the distributions.

The Figures 4.13a and 4.13c show the prior and posterior distributions of the used hyperpriors of the parameter U. Compared to the one-dimensional scenario, one can observe a slightly greater variance of the two parameters here. Considering the hyperpriors of the parameter τ, one can recognize a similar behavior, and due to the small variation of the posterior distributions, a hypothesis of convergence is possible (see figures 4.14a and 4.14c). The results based on the MLE are given as:

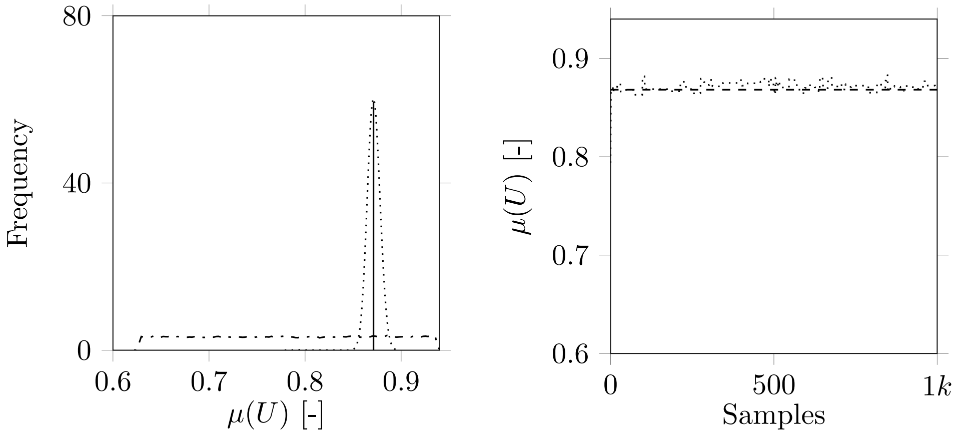
$$\begin{aligned}
\mu_{\text{MLE}}(U) &= 0.844, \\
\sigma_{\text{MLE}}(U) &= 0.0441, \\
\mu_{\text{MLE}}(\tau) &= 0.056, \\
\sigma_{\text{MLE}}(\tau) &= 0.0045.
\end{aligned}$$

As before, the computation of the solution was accelerated by parallelization with 25 CPUs. Then the average calculation time for an iteration on the original model is:

$$\Delta t_{\text{iter}} = 74.23\,[\text{s}].$$

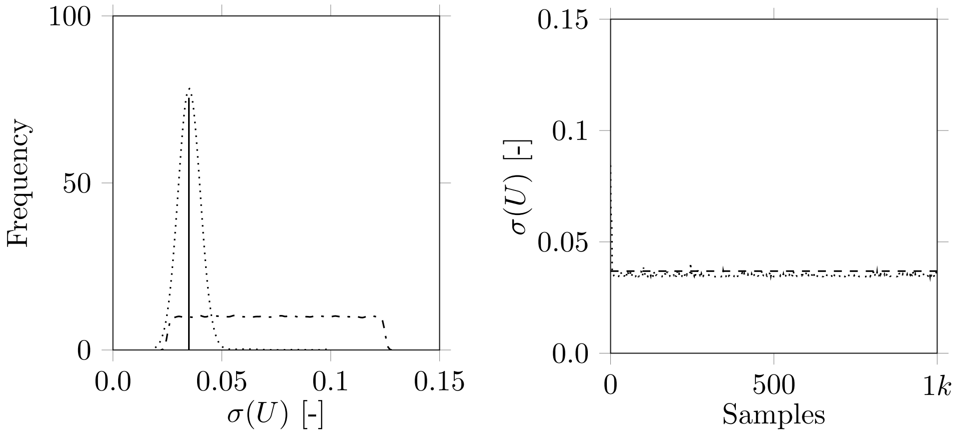
Compared to the one-dimensional scenario, the time required for an iteration has almost doubled, while the average evaluation time for a proposal has remained roughly the same. This circumstance is since more proposals are rejected in an iteration due to the extended parameter space. Besides, one needs more time to evaluate the entire Likelihood function.

In the following, one replaces the original model by the surrogate model and repeats the calculations. The figures 4.15 and 4.16 summarize the numerical results. One observes that the number of iterations has increased again by a factor of 10 compared to the original model. The traces of the individual hyperpriors can be found in the figures 4.15b, 4.15d, 4.16b and 4.16d. One should emphasize that the variation of the parameter $\sigma(U)$ has increased, and this also corresponds to the observation in a



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

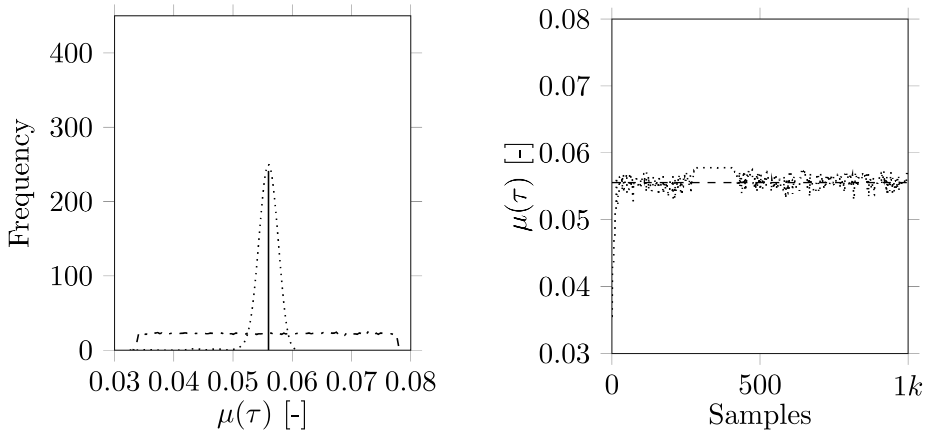
(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

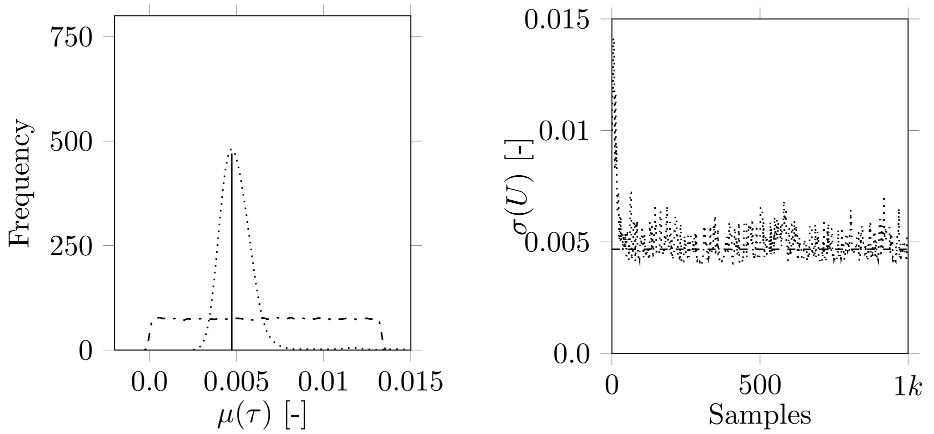
(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows theoretical value.

Figure 4.13: Numerical result of the MCMC estimation for parameter U with one time point. The computation was performed on the original model.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the load torque τ. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(\tau)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the load torque τ. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(\tau)$. The dashed line shows theoretical value.

Figure 4.14: Numerical result of the MCMC estimation for parameter τ with one time point. The computation was performed on the original model.

one-dimensional scenario. The two parameters $\mu(\tau)$ and $\sigma(\tau)$ show a much smaller distribution of the traces. Concerning the assumed parameter values, one can see that the estimates are already very close to the actual solution at the beginning. Thus, one can assume that results that are far away from the solution, are rejected, and the variances are greatly reduced. This is directly visible within the PDFs in the figures 4.15a, 4.15c, 4.16a, and 4.16c. The approximate values are:

$$\begin{aligned}\mu_{\mathrm{MLE}}(U) &= 0.867,\\ \sigma_{\mathrm{MLE}}(U) &= 0.0399,\\ \mu_{\mathrm{MLE}}(\tau) &= 0.056,\\ \sigma_{\mathrm{MLE}}(\tau) &= 0.0046.\end{aligned}$$

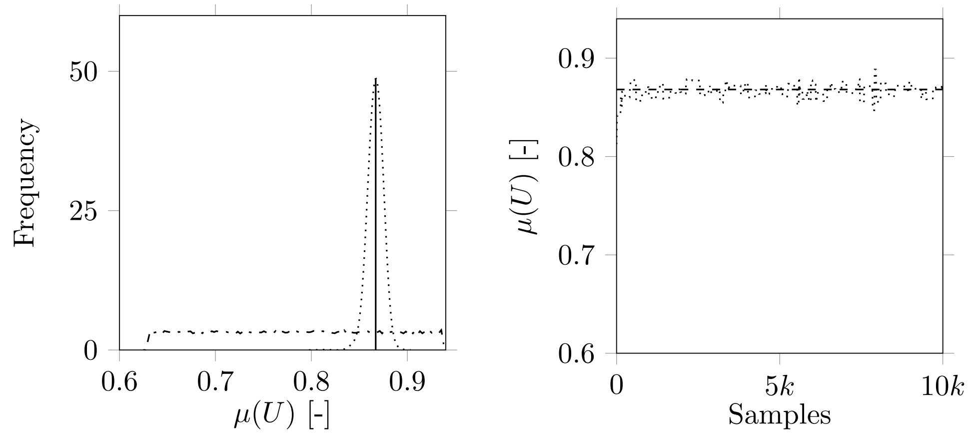
There is a time increase per iteration, just like in the evaluation of the original model. On average the resulting time is:

$$\Delta t_{\mathrm{iter}} = 1.24\,[\mathrm{s}].$$

The time required to calculate the surrogate model $t_{\mathrm{surrogate}}$ does not change and must be added to the total time once. This results in a total simulation time of $t_{\mathrm{all,original}} = 74230\,[s]$ for the original model and $t\mathrm{all,surrogate} = 12413\,[s]$ for the surrogate model. The use of PCE leads to a six times faster evaluation.

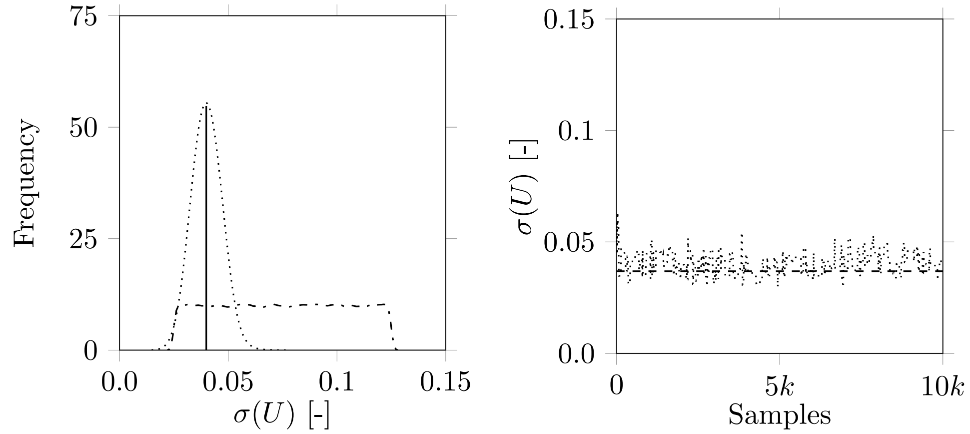
The analysis includes the traces of the individual additive errors at this point. Figure 4.17 shows the standard deviations of the additive errors for the original model. One should note that the distribution $\sigma_{\epsilon,\omega}$ is greater than the value of the one-dimensional scenario σ_ϵ (see Figure 4.17b). The PDF in Figure 4.17a illustrates this behavior. One should note that this study uses a different measurement series, and the increase of the parameter space itself describes a new problem class. The trace of $\sigma_{\epsilon,\mathrm{I}}$ in Figure 4.17d and the corresponding PDF in Figure 4.17c show a realistic trend. The posterior distributions are determined by the previous assumptions of the one-dimensional scenario, and the prior variances are chosen as $\sigma^*_\omega = 0.2$ and $\sigma^*_\mathrm{I} = 0.3$.

For the approximation with the PCE surrogate model, one uses slightly larger variations of $\sigma^*_\omega = 0.4$ and $\sigma^*_\mathrm{I} = 0.5$ as in the prior study. Figure 4.18 shows the numerical results. If one compares the PDF of the surrogate model from Figure 4.18c with the PDF distribution of the original model, similar behavior can be observed. This statement is also supported by the trace in Figure 4.18d. In contrast, the distribution of $\sigma_{\epsilon,\omega}$ shows some differences. Considering Figure 4.18b, the trace decreases over the number of iterations and this leads to a PDF close to zero (see Figure 4.18a). This outcome is consistent with the observation from the one-dimensional scenario but does



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

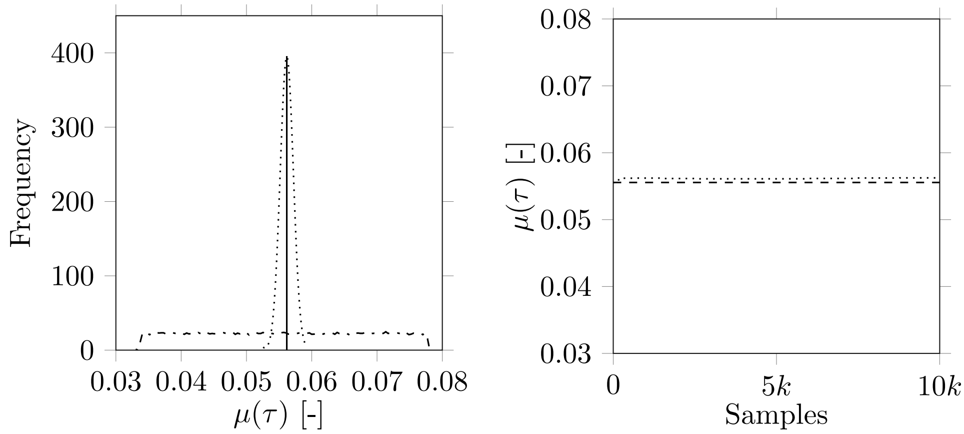
(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U.MLE is displayed with solid line.

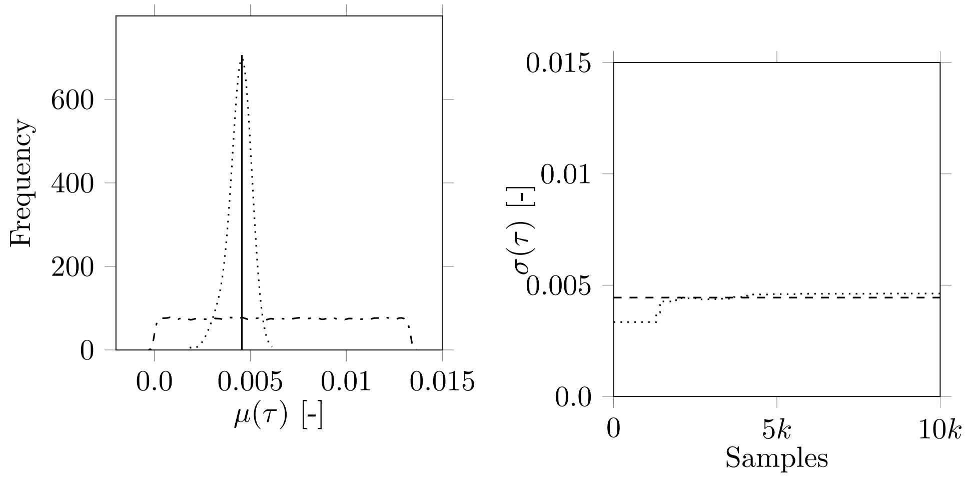
(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows theoretical value.

Figure 4.15: Numerical result of the MCMC estimation for parameter U with one time point. The computation was performed on the surrogate model.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the load torque τ. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(\tau)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the load torque τ. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(\tau)$. The dashed line shows theoretical value.

Figure 4.16: Numerical result of the MCMC estimation for parameter τ with one time point. The computation was performed on the surrogate model.

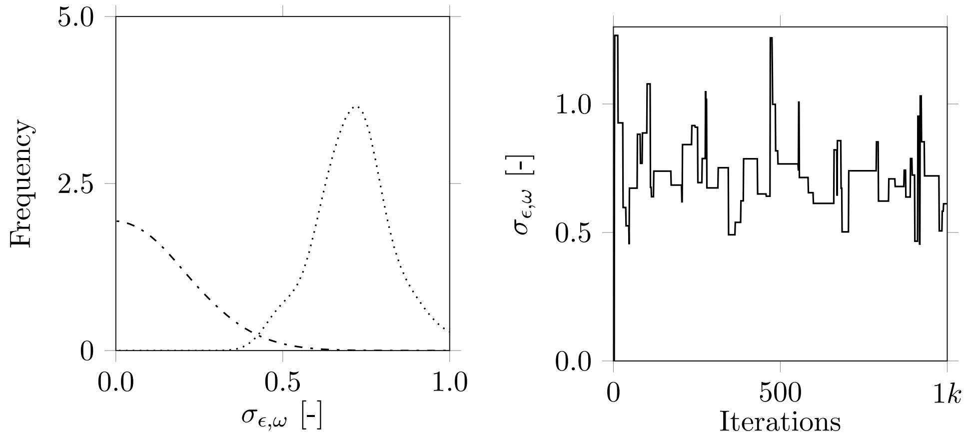
Table 4.2: Overview of numerical results for multi-dimensional scenario.

Simulation model	Δt_{iter}	$\mu(U)$	$\sigma(U)$	$\mu(\tau)$	$\sigma(\tau)$
Original model	74.23	0.871	0.0349	0.056	0.0047
Surrogate model	1.24	0.867	0.0399	0.056	0.0046

not agree with the observations from the multi-dimensional scenario in combination with the original model. One hypothesis could be that the surrogate model does not reflect special properties. Thus a softer response can be achieved.

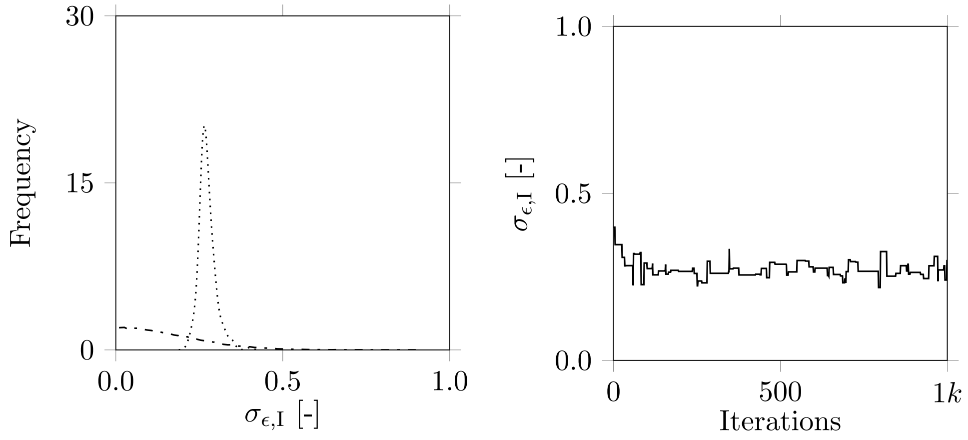
Table 4.2 summarizes the considerations for the multi-dimensional scenario. By the one-dimensional scenario, one shows in this study that it is possible to add further parameters and that the numerical results provide a very reliable approximation of the assumed distributions. In addition to the extension of the parameter space, one can show that one can extend the Likelihood function to several measurement outputs of the system and that estimate an additive error for each assumption. The use of a PCE surrogate model helps to improve the efficiency of the entire calculation. A significant advantage of the approach presented is the application of a sensitivity analysis in combination with the modeling of Likelihood functions. With this step, it is possible to include the relevant measurement information where the most sensitive parameters have the highest contribution in the investigation and thus achieve fast convergence of the procedure.

The resistance R is considered separately, as previously mentioned at the beginning of this subsection. From the sensitivity analysis in Figure 4.1, one can see that the parameter only influences the considered outputs in the dynamic range. This observation is already included in the Likelihood function for the parameter R in Equation (4.16). One can extend the investigation of the multi-dimensional case with the parameters U and τ by Equation (4.18). The calculation of the posterior distributions is similar to the previous analysis.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the error ϵ_ω.

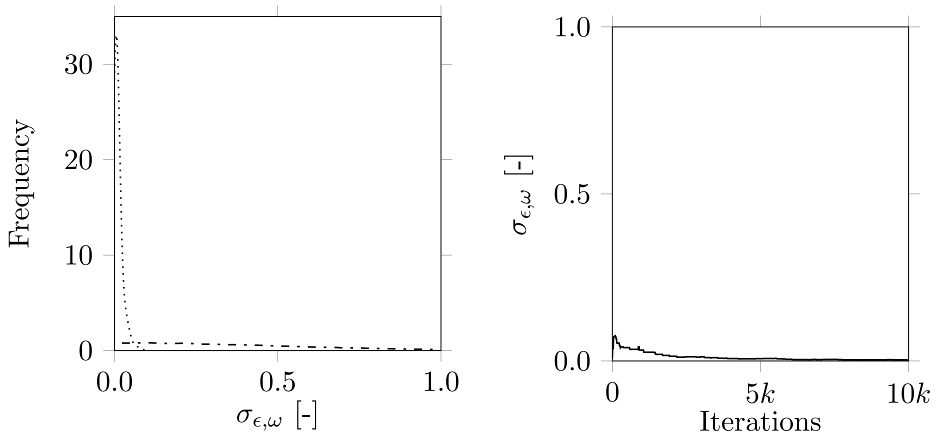
(b) Trace of the Markov chain for $\sigma_{\epsilon,\omega}$.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the error ϵ_I.

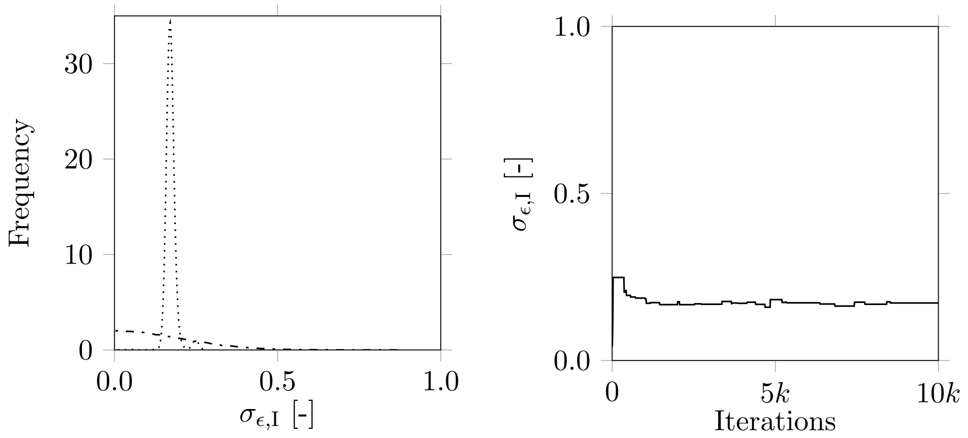
(d) Trace of the Markov chain for $\sigma_{\epsilon,\mathrm{I}}$.

Figure 4.17: Standard deviation for the additive error assumption for the chosen Likelihood function for the MCMC evaluation on the original model.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the error ϵ_ω.

(b) Trace of the Markov chain for $\sigma_{\epsilon,\omega}$.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the error ϵ_I.

(d) Trace of the Markov chain for $\sigma_{\epsilon,\mathrm{I}}$.

Figure 4.18: Standard deviation for the additive error assumption for the chosen Likelihood functions for the MCMC evaluation on the surrogate model.

The numerical results for the parameters U and τ are very close to the results in the figures 4.13 and 4.14 and provide a good approximation. However, the hyperparameters for the resistance R converge to the minimum or maximum value of the prior distributions. For this parameter, no statement about the underlying distribution is possible with this setup. This case is well suited to illustrate the limitations of the method. It may happen that the sensitivity of the parameter is not sufficient, and the impact of the measurement noise is higher than the predicted sensitivity. Due to this general condition, it may not be possible to reconstruct the parameter distribution by selecting other Likelihood functions.

4.5.3 No-U-Turn sampler scenario

This subsection focuses on the efficiency evaluation of the methods used in comparison with the NUTS algorithm and assesses the numerical results. Starting from the previous scenario, one reduces the number of measurement points to $S = 40$, and one further restricts the information of the prior assumptions for a better comparison of the efficiency. In this scenario, one limits oneself to the consideration of the uncertain parameter U. The selection of a stationary time for the evaluation is not affected.

The analysis uses the same measurement data as in the one-dimensional case (see Subsection 4.5.1). The only difference is the reduction of the data points used, which leads to slightly different stochastic moments. The mean value is $\mu(U_{\mathrm{tb}}) = 0.87$ and the standard deviation is $\sigma(U_{\mathrm{tb}}) = 0.033$. Resulting in the following error deviations: $e_{\mu(\mathrm{U})} = 3.6\%$ and $e_{\sigma(\mathrm{U})} = 24.1\%$. One can see that the mean value changes only slightly due to the reduction. Whereas the standard deviation shows a larger change, and one must consider this discrepancy in the final evaluation. As before, one uses the motor speed ω at a stationary time $t^* = 5$ secounds for each measurement in the overall series.

One reduces the information on the hyperpriors in Equation (4.19), and therefore, one increases the interval of the uniform distributions. The two hyperpriors are:

$$\begin{aligned} P(\mu_{\mathrm{U}}) &= \mathcal{U}(0.5, 1.25), \\ P(\sigma_{\mathrm{U}}) &= \mathcal{U}(0.013, 0.25). \end{aligned}$$

The prior of the standard deviation σ_ϵ is modeled as an inverse gamma distribution instead of a half-normal distribution:

$$P(\sigma_\epsilon) = \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{1}{\sigma_\epsilon}\right)^{\alpha+1} \exp\left(-\frac{\beta}{\sigma_\epsilon}\right),$$

where $\alpha = 10$ is the shape parameter, $\beta = 2$ is the scale parameter, and $\Gamma(\cdot)$ denotes the Gamma function.

This investigation consists of four different calculations. The first two cases correspond to the reference in terms of calculation time, and one obtains them by applying the HMC algorithm. The difference between the two cases is the model used. In one instance, one uses the original simulation model, whereas, in the other, one applies the PCE surrogate model. The last two cases refer to the evaluation with the NUTS algorithm. Once more, one distinguishes between the simulation on the original model and the simulation on the surrogate model.

Figure 4.19 shows the numerical results of the posteriors using an HMC approach with the original simulation model. In contrast to the results with the Metropolis-Hastings approach in Figure 4.6, one can see the reduction of the iterations. One can justify this reduction by the fact that the distance between the iteratively generated points is typically greater in the HMC method. In this case, one uses a burn-in period of 25 iterations, which one must add to the total sum. A visual examination shows that the results are very close to the values one wants to identify. More precisely, one can derive the following approximations from the numerical results:

$$\begin{aligned} \mu_{\mathrm{MLE}}(U) &= 0.850, \\ \sigma_{\mathrm{MLE}}(U) &= 0.0308. \end{aligned}$$

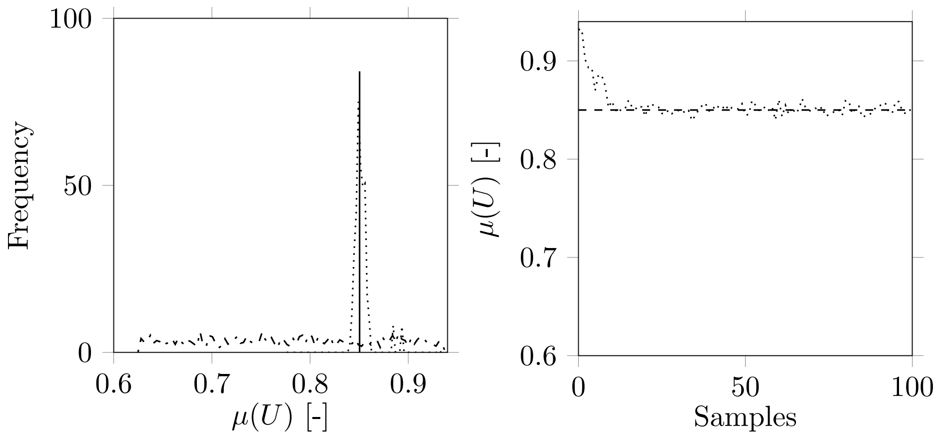
Considering the efficiency, one uses the same hierarchical model as in the one-dimensional case, but one only performs $N = 40$ variations for each proposal. Using the same time for an average simulation, one obtains an evaluation time for one proposal of $\Delta t_{\text{prop}} = 84.4$ seconds. In this calculation, 40 CPU cores parallelize the computation and the average time for one iteration results in:

$$\Delta t_{\text{iter}} = 1449.05\,[\mathrm{s}].$$

Due to the significantly longer calculation time for an iteration, one can already see a big difference to the previously used method: The price for an iteration with the HMC method is much higher, but the algorithm is much more efficient and needs fewer iterations in total. The gradient calculation mainly attributes these additional costs. For the presented calculation, one uses a simple forward finite difference formula of the following form:

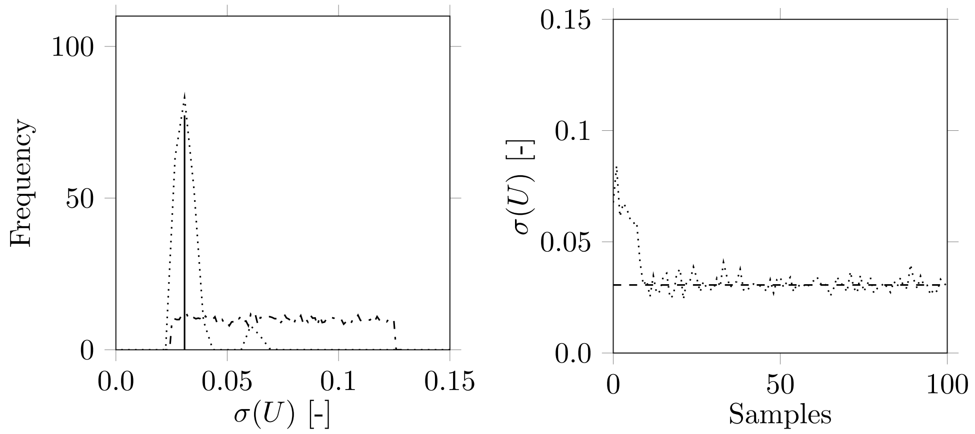
$$\frac{\partial \mathcal{M}(q)}{\partial q} = \frac{\mathcal{M}(q+\epsilon) - \mathcal{M}(q)}{\epsilon},$$

where ϵ is the increment to q for determining the function gradient.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows the theoretical value.

Figure 4.19: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the original model with the HMC algorithm.

The following step is to replace the original simulation model with the PCE surrogate model. Figure 4.20 shows the numerical results, and one can see a very fast convergence. For this calculation, one uses 500 burn-in iterations and another $9,500$ evaluation for the posterior estimation. The numerical results provide very similar results to those from the previous investigation:

$$\begin{aligned}\mu_{\mathrm{MLE}}(U) &= 0.850,\\ \sigma_{\mathrm{MLE}}(U) &= 0.0302.\end{aligned}$$

From an efficiency point of view, the evaluation time of the surrogate model remains the same compared to the one-dimensional scenario. In total, the time required for one iteration changes:

$$\Delta t_{\mathrm{iter}} = 3.04\,[\mathrm{s}].$$

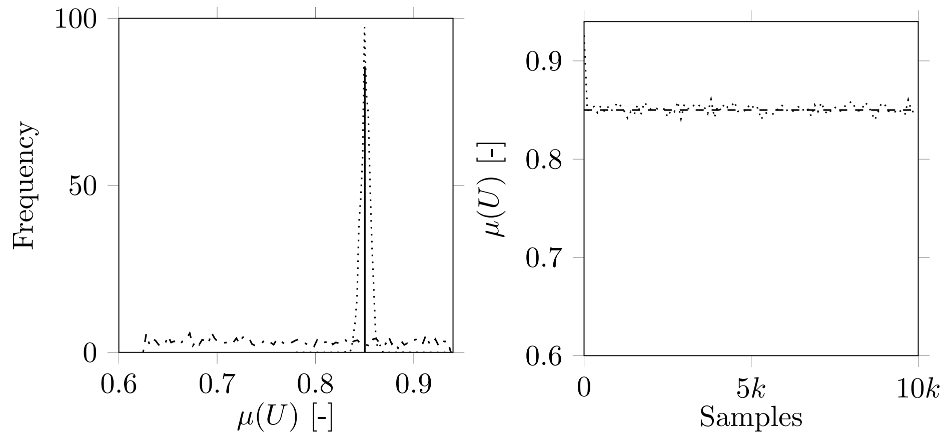
This measure includes the evaluations of the simulation model for the gradient calculation. At this point, of course, one must also take into account the time needed to determine the coefficients of the surrogate model. With a parallelization of 40 cores, one gets an evaluation time of $t_{\mathrm{surrogate}} = 7.84$ seconds.

Finally, one replaces the HMC algorithm with the NUTS method and discusses the results obtained in the following paragraphs. Figure 4.21 shows the numerical results of this method. One performs the calculation on the original model and uses the same procedure for the gradients as in the HMC investigation. The burn-in period is 25 iterations, and for the whole calculation of the posterior distribution, one uses further 100 iterations. A first visual inspection shows a large variation in the mean and standard deviation of the approximation in the beginning, and one achieves a stationary behavior after 25 iterations in addition to the burn-in phase. The numerical results are as follows:

$$\begin{aligned}\mu_{\mathrm{MLE}}(U) &= 0.857,\\ \sigma_{\mathrm{MLE}}(U) &= 0.0318.\end{aligned}$$

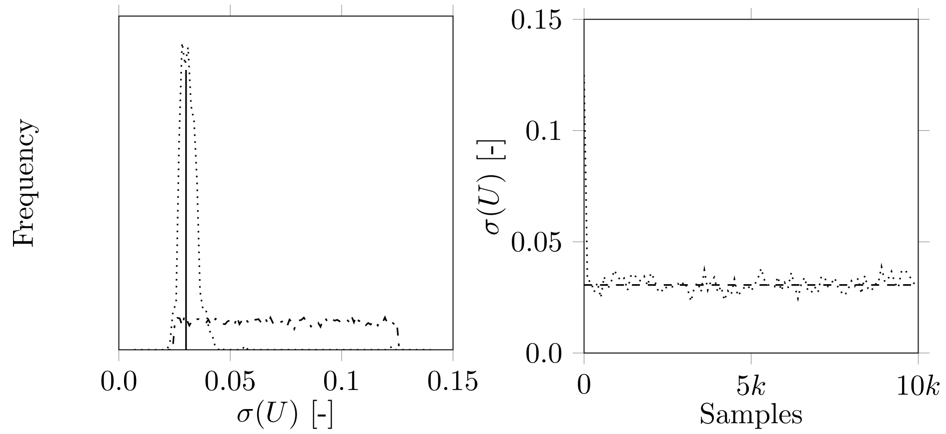
If one compares the values obtained with the previous investigations, one cannot endorse a significant difference in quality. Besides, the algorithm automatically adjusts the step size and the number of leapfrog steps. To be able to judge the adjustment of the two parameters, it is again necessary to consider the average time for one iteration. This evaluation time results in:

$$\Delta t_{\mathrm{iter}} = 2,448.60\,[\mathrm{s}].$$



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows the theoretical value.

Figure 4.20: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the surrogate model with the HMC algorithm.

It becomes apparent that the time for an iteration increased significantly. But the comparison is not entirely equitable at this point. One can establish this statement by the fact that at the beginning, convergence to the actual value cannot be seen, and consequently, much more evaluations on the original model are necessary to determine the initial values and the gradients. If one looks at the times for the iteration individually, one can recognize a decrease in the duration with increasing iterations. Since this method is only an intermediate step, no further investigation is performed.

In the last step of this consideration, one again replaces the original simulation model by the surrogate model. As before, one uses 500 burn-in iterations for the calculation and approximates the posterior distribution with additional $9,500$ iterations. Figure 4.22 visualizes the numerical results, and one can already see a convergence to the real values on a quantitative level. A qualitative view results in the posteriors:

$$\begin{aligned}\mu_{\text{MLE}}(U) &= 0.850,\\ \sigma_{\text{MLE}}(U) &= 0.0304.\end{aligned}$$

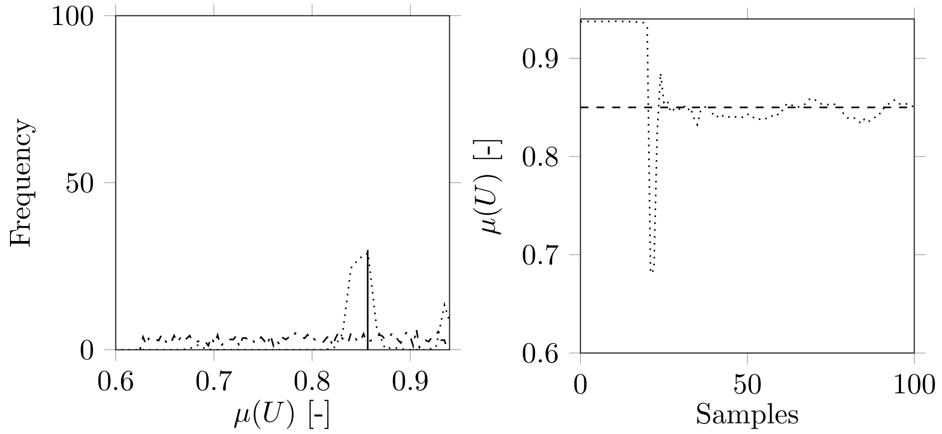
Interesting in the recent evaluation is the use of the surrogate model. Contrary to the previous investigations, the gradients are derived directly from the structure of the surrogate model (see Subsection 4.4.2), and thus, one may expect a further increase in efficiency. If one now considers the time required for an iteration, one obtains:

$$\Delta t_{\text{iter}} = 0.135\,[\text{s}].$$

The time needed for the calculation of the surrogate model parameters remains unchanged from the previous one at $t_{\text{surrogate}} = 7.84$.

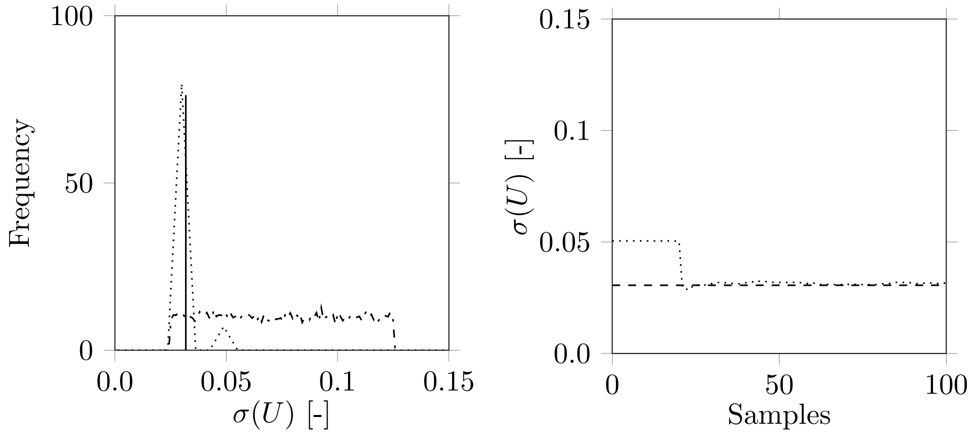
Concluding, the NUTS method allows an excellent approximation of the posterior distributions. With the presented addition of the PCE surrogate model and the integration of the gradients within the NUTS procedure, one achieves a very good performance. Nevertheless, the structure of the methods guarantees high applicability because all introduced parameters in the algorithms used are automatically adapted to the problem class.

If one compares the results of the NUTS approach using the original model (see Figure 4.21) with the use of the surrogate model (see Figure 4.22), one can identify a marginal bias. The discrepancy between the theoretical value and the numerical result does not have a strong correlation to the approximation order P of the surrogate model, as one might first assume. The assumption that P correlates weakly with the bias, gets obvious, when looking at the forward propagation study in Chapter 3. The results there show that the surrogate model has got proper approximation properties



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

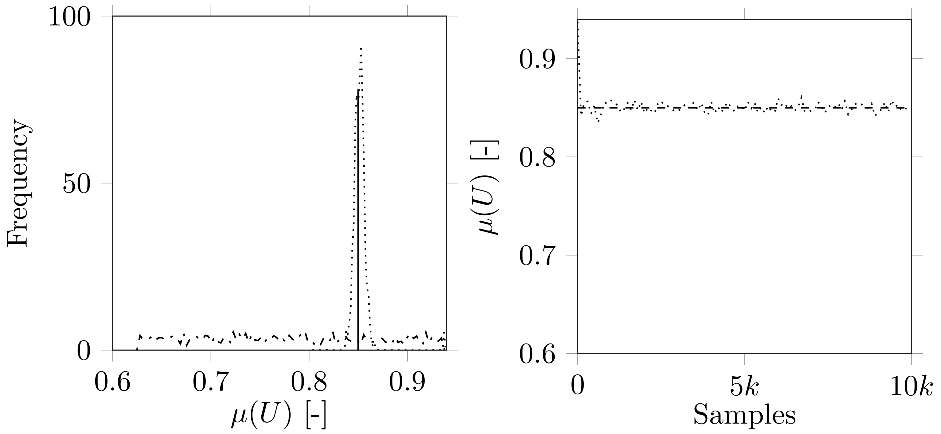
(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

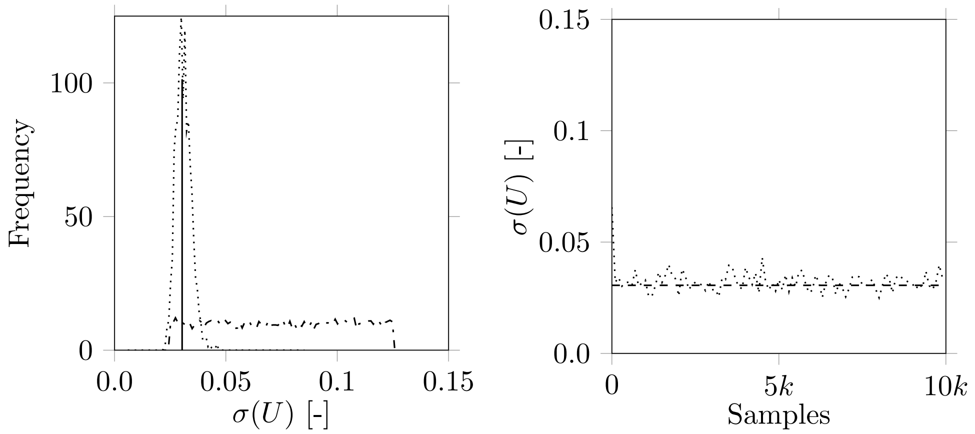
(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows the theoretical value.

Figure 4.21: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the original model with No-U-Turn Sampler (NUTS).



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



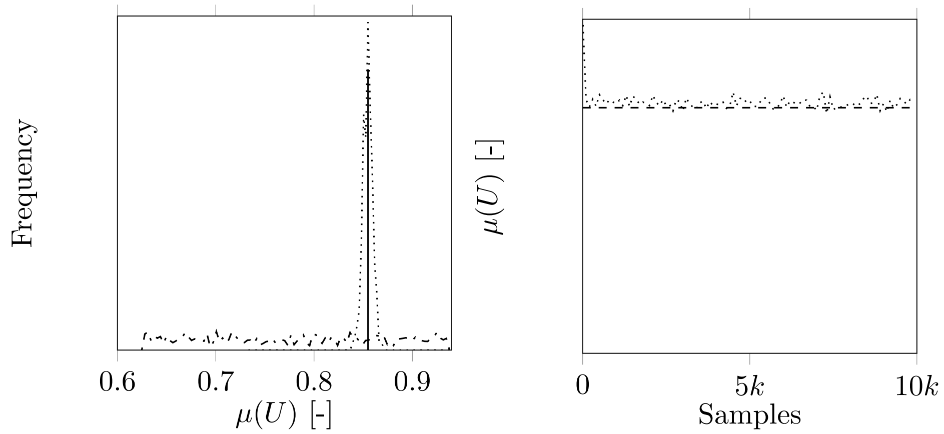
(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows the theoretical value.

Figure 4.22: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the surrogate model with No-U-Turn Sampler (NUTS).

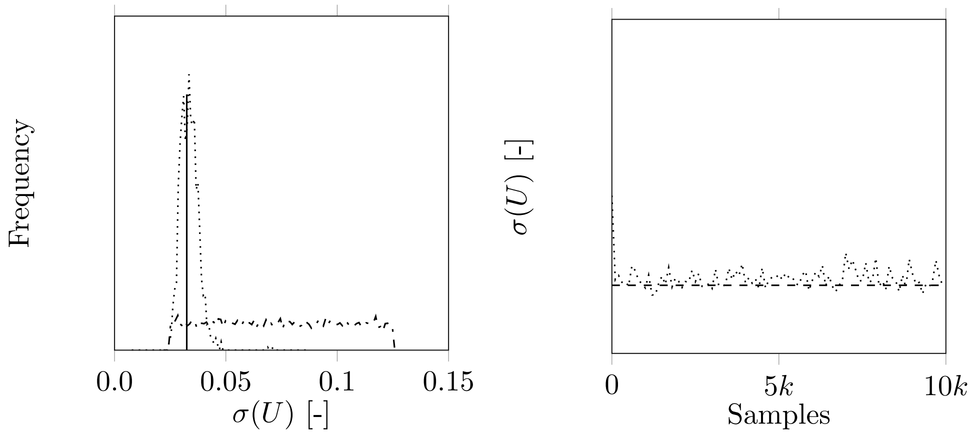
at the output variable ω (see Figure 3.10). The shown posterior distributions in this chapter use the surrogate model with approximation order $P = 2$, and in the context of this investigation, one gets comparable results by reducing P to one. The bias arises mainly from the assumptions for the surrogate model. Since no detailed parameter distributions are available beforehand, one builds the PCE model with the prior assumptions. Figure 4.23 illustrates this hypothesis and shows the numerical results with 25% enlarged priors. The two traces for the mean value and the standard deviation of U indicate a significant bias to the theoretical values compared to the previous results. In this study, one can assess the sensitivity of the prior assumptions to the quality of the posterior distributions. If one wants to investigate the influence of the approximation order, one should consider the dynamic range of the motor speed ω.

To complete the analysis, one considers the estimated errors analogous to the previous investigations. One can say that in all four variants, the error converges to a similar value, and this value is in the range of the one-dimensional scenario. When comparing the original and the surrogate simulation model, one can hardly see any difference in the standard deviation of σ_ϵ. Since one refers to the variants with the surrogate model approach as an approximation of the original model, one can assume a significant error which the posterior results should then reflected. But if one includes the two posterior distributions of the voltage U in this consideration, one can see that the inaccuracies of the surrogate model become apparent in an increased variation of the two posterior results.



(a) PDF (prior with dash-dotted and posterior with dotted line) for the mean value of the voltage U. MLE is displayed with solid line.

(b) Trace (dotted line) of the Markov chain for $\mu(U)$. The dashed line shows theoretical value.



(c) PDF (prior with dash-dotted and posterior with dotted line) for the standard deviation of the voltage U. MLE is displayed with solid line.

(d) Trace (dotted line) of the Markov chain for $\sigma(U)$. The dashed line shows the theoretical value.

Figure 4.23: The numerical result of the MCMC estimation for the one-dimensional scenario with one time point. The computation was performed on the surrogate model with No-U-Turn Sampler (NUTS) and enlarged prior assumptions.

5

Conclusion

This thesis aims to apply the inverse Uncertainty Quantification to a real application example from the industry. The challenges here lie in determining the input and parameter distributions of the simulation model based on recorded observations, the efficient calculation of the methods introduced, the requirements for the model and the measurement series, and the generalization of the process. In concrete terms, this means that the present work combines several approaches and introduces extensions to achieve the inverse Uncertainty Quantification for an application example from the industry. One finds the steps accomplished for reaching this goal in the individual chapters. Chapter 2 focuses on the requirements for the simulation model and the configuration of the test bench for the validation. One can find the extension of the simulation models by stochastic information in Chapter 3. This analysis includes the validation of the forward Uncertainty Quantification. Chapter 4 presents the numerical results for the parameter distributions based on the validated simulation model and measurements from the test bench.

One introduces the windshield wiper drive being the real application example in Chapter 2. The investigation shows the necessary steps to get to the relevant parameters, whose statistical information one determines in the further course. In parallel, one defines the construction of the test bench, and this hardware makes it possible to record measurement data that scatter due to different physical quantities. The unique feature of the device is that one can automatically generate the measurements from parameter distributions. The union of the physical measurements and the information of the assumed distributions for the particular physical quantities makes it possible that one can use the information for the validation of the algorithms in the context of the quantification of uncertainties. To enable this characteristic, it was also necessary to keep the measurement conditions in the same range. Besides, one shows how the measurement data could be prepared in such a way that UQ can be applied at all. Three uncertain physical quantities are taken into account in the present work.

Chapter 3 examines methods for the stochastic extension of the simulation model, and the challenge lies in the fact that one cannot change the solver of the simulation model based on its generalizability. The classical approaches require a high number of evaluations on the model, and from an efficiency point of view, an optimization of the calculation is necessary. Based on the simulation model, one creates a surrogate model with PCE and investigates various approaches for determining the coefficients. One can show that the sparse grid offers an excellent possibility to adapt the surrogate model to the original model with few evaluations. A contribution of this work is the comparison of the statistical information between the simulation results and the measurements of the test bench hardware. In summary, one can show that the proposed surrogate model approach provides a suitable approximation to reality.

One addresses the main challenge of the inverse Uncertainty Quantification in Chapter 4, and intensively discusses the numerical results there. One uses the Bayesian inference and related methods to calculate the distributions of the inputs and parameters. A contribution of this work lies in the combination of sensitivity analysis based on the Sobol' indices of the surrogate model with the modeling of the Likelihood function. Especially with time-dependent QoIs, it is possible to select the relevant information of the measurements required for the investigation and to speed up the calculation. By using the measurement data of the test bench, one can show that the MCMC method used is well suited to approximate the desired distributions. This result includes recorded measurements from the test bench hardware and the complex simulation model of an electric drive. One can show that the surrogate model accelerates the calculation of distributions by a factor of six to twenty. Despite faster calculation, one can maintain the quality of the evaluation. This statement is valid, although one can only give approximate estimates for the distributions of the individual hyperpriors. In summary, the particular steps result in a sequence that one can directly transfer to different application examples. Another contribution of this thesis is the use of the No

U-Turn Sampler (NUTS) method in combination with a PC surrogate model. Coupling the analytical solution of the surrogate model derivatives with the NUTS algorithm significantly increases the efficiency. This characteristic is particularly reflected in very fast convergence. Nevertheless, the easy applicability of the method remains very simple for the end-user. It could also be shown that even with little information, a good posterior distribution of the parameters can be approximated.

Considering the simulation model and the test bench, one can derive several enhancements. A possible extension is to add further uncertain quantities to the structure to extend the considered parameter space and to examine the scalability of the algorithms more closely. The measurement data used is recorded under constant measurement conditions. This concerns above all the ambient temperature, but also the temperatures of the individual elements of the electric drive. Depending on the operating time, the temperatures can vary greatly and thus also have a significant influence on the measured variables. A possible extension is the modeling of these uncertain quantities to reduce the requirements for recording measurements and thus increase the applicability of the methodology also for measurement data from fleets or products from a practical operation.

As long as detailed investigations or measurement data are available, it is possible to evaluate the quality of the surrogate model. To increase the applicability of such surrogate models in industry, error estimators are needed. A possible extension is an algorithm which, starting from a given error limit, increases the evaluations on the original model until the desired approximation quality is achieved.

The backward Uncertainty Quantification described applies to the assumption of a Gaussian distribution and is suitable for industrial applications. A conceivable extension is the generalization of this approach so that less prior knowledge must be brought into the selection of suitable hyperparameters. A further challenge in the presented procedure is the selection and modeling of a suitable Likelihood function for the MCMC procedure. At this point, one should mention further methods, such as the Approximate Bayesian Computation, which bypasses the evaluation of the Likelihood function. Finally, the extension of the parameter space would also be very interesting to obtain further information about the scalability of the methods.

Bibliography

[1] Arbib, Michael A.: *The handbook of brain theory and neural networks.* 2nd ed. Cambridge, Mass. and London : MIT, 2003

[2] Banks, H. T. ; Kunisch, K.: *Estimation Techniques for Distributed Parameter Systems.* Birkhäuser Boston, 1989

[3] Barthelmann, V. ; Novak, E. ; Ritter, K.: High dimensional polynomial interpolation on sparse grids. In: *Advances in Computational Mathematics* 12 (2000), Nr. 4, S. 273–288

[4] Bauer, Heinz: *Wahrscheinlichkeitstheorie.* 5th ed. De Gruyter, 2001

[5] Behrends, Ehrhard: *Introduction to Markov Chains.* Vieweg+Teubner Verlag, 2000

[6] Berg, Bernd A.: *Introduction to Markov Chain Monte Carlo Simulations and their Statistical Analysis.* 2004

[7] Besag, Julian ; Green, Peter ; Higdon, David: Bayesian computation and stochastic systems. In: *Statistical Science* 10 (1995), Nr. 1, S. 3–41

[8] Blatman, Géraud ; Sudret, Bruno: An adaptive algorithm to build up sparse polynomial chaos expansions for stochastic finite element analysis. In: *Probabilistic Engineering Mechanics* 25 (2010), Nr. 2, S. 183–197

[9] Box, George E. P. ; Draper, Norman R.: *Response surfaces, mixtures, and ridge analyses.* 2nd ed. Hoboken, N.J. : Wiley and Chichester : John Wiley [distributor], 2007

[10] Brigham, E. O.: *The fast Fourier transform and its applications.* Englewood Cliffs, N.J. : Prentice Hall, 1988 (Prentice-Hall signal processing series)

[11] Brooks, S. ; Gelman, A. ; Jones, G. ; Meng, X.L.: *Handbook of Markov Chain Monte Carlo*. CRC Press, 2011 (Chapman & Hall/CRC Handbooks of Modern Statistical Methods)

[12] Butterworth, Stephen: On the theory of filter amplifiers. In: *Experimental Wireless and the Wireless Engineer* 7 (1930), S. 536–541

[13] Caflisch, Russel E.: Monte Carlo and quasi-Monte Carlo methods. In: *Acta Numerica* (1998), S. 1–49

[14] Cameron, R. H. ; Martin, W. T.: The Orthogonal Development of Non-Linear Functionals in Series of Fourier-Hermite Functionals. In: *The Annals of Mathematics* 48 (1947), Nr. 2, S. 385

[15] Coleman, Thomas F. ; Li, Yuying: On the convergence of interior-reflective Newton methods for nonlinear minimization subject to bounds. In: *Mathematical Programming* 67 (1994), Nr. 1-3, S. 189–224

[16] Coleman, Thomas F. ; Li, Yuying: An Interior Trust Region Approach for Nonlinear Minimization Subject to Bounds. In: *SIAM Journal on Optimization* 6 (1996), Nr. 2, S. 418–445

[17] Deb, Manas K. ; Babuska, Ivo M. ; Oden, J. T.: Solution of stochastic partial differential equations using Galerkin finite element techniques. In: *Computer Methods in Applied Mechanics and Engineering* 190 (2001), S. 6359–6372

[18] Debusschere, Bert J. ; Najm, Habib N. ; Pébay, Philippe P. ; Knio, Omar M. ; Ghanem, Roger G. ; Le Maitre, Olivier P.: Numerical Challenges in the Use of Polynomial Chaos Representations for Stochastic Processes. In: *SIAM Journal on Scientific Computing* 26 (2004), Nr. 2, S. 698–719

[19] dSpace GmbH: *ControlDesk*. 2016

[20] Duane, Simon ; Kennedy, A.D. ; Pendleton, Brian J. ; Roweth, Duncan: Hybrid Monte Carlo. In: *Physics Letters B* 195 (1987), Nr. 2, S. 216 – 222

[21] Eldred, M. S. ; Webster, C. G. ; Constantine, P. G.: Evaluation of Non-Intrusive Approaches for Wiener-Askey Generalized Polynomial Chaos. In: *49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference* 2008 (2008), Nr. April, S. 1–22

[22] Eldred, Michael ; John, Burkardt: Comparison of non-intrusive polynomial chaos and stochastic collocation methods for uncertainty quantification. 47th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition (2009)

[23] Fishman, George S.: *Monte Carlo.* 1999

[24] Fishman, George S.: *A first course in Monte Carlo.* 1st ed. Belmont CA : Thomson Brooks/cole, 2006 (Duxbury advanced series)

[25] Gelfand, Alan E. ; Smith, Adrian F. M.: Sampling-Based Approaches to Calculating Marginal Densities. In: *Journal of the American Statistical Association* 85 (1990), Nr. 410, S. 398–409

[26] Gelman, Andrew ; Hill, Jennifer: *Data analysis using regression and multilevel.* Cambridge : Cambridge University Press, 2007 (Analytical methods for social research)

[27] Geman, S. ; Geman, D.: Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. In: *IEEE transactions on pattern analysis and machine intelligence* 6 (1984), Nr. 6, S. 721–741

[28] Gerstner, Thomas ; Griebel, Michael: Numerical Integration using Sparse Grids. In: *Numerical Algorithms* 18 (1998), S. 209–232

[29] Ghanem, Roger: *Handbook of uncertainty quantification.* New York NY : Springer Berlin Heidelberg, 2017

[30] Ghanem, Roger G. ; Spanos, Pol D.: *Stochastic Finite Elements: A Spectral Approach.* Springer, 1991. – ISBN 978–1–4612–7795–8

[31] Giraldi, Loic ; Litvinenko, Alexander ; Liu, Dishi ; Matthies, Hermann G. ; Nouy, Anthony: To be or not to be intrusive ? The solution of parametric and stochastic equations — the " plain vanilla " Galerkin case. (2013), S. 1–24

[32] Glaser, P. ; Schick, M. ; Petridis, K. ; Heuveline, V.: Comparison between a polynomial chaos surrogate model and Markov Chain Monte Carlo for inverse uncertainty quantification based on an electric drive test bench. In: *ECCOMAS Congress 2016 - Proceedings of the 7th European Congress on Computational Methods in Applied Sciences and Engineering* 4 (2016)

[33] Hasting, W. K.: Monte Carlo sampling methods using Markov chains and their applications. In: *Biometrica* 57 (1970), Nr. 1, S. 97

[34] Helton, J. C. ; Davis, F. J.: Latin hypercube sampling and the propagation of uncertainty in analyses of complex systems. In: *Reliability Engineering & System Safety* 81 (2003), Nr. 1, S. 23–69

[35] Hoffman, Matthew ; Gelman, Andrew: The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. In: *Journal of Machine Learning Research* 15 (2011), 11

[36] Iman, R. L. ; Davenport, J. M. ; Zeigler, D. K.: *Latin hypercube sampling (program user's guide). [LHC, in FORTRAN].* 1980

[37] Isermann, Rolf: *Identifikation dynamischer Systeme 1.* Berlin, Heidelberg : Springer Berlin Heidelberg, 1992

[38] Isermann, Rolf ; Münchhof, Marco: *Identification of dynamic systems: An introduction with applications / Rolf Isermann, Marco Münchhof.* Berlin : Springer, 2011. – ISBN 9783540788799

[39] Jacod, Jean ; Protter, Philip E.: *Probability essentials.* 2nd ed. New York : Springer, 2003 (Universitext)

[40] Kaipio, Jari ; Somersalo, E.: *Statistical and Computational Inverse Problems.* Bd. 160. New York : Springer-Verlag, 2005

[41] Kolmogorov, Andrej N.: *Ergebnisse der Mathematik und ihrer Grenzgebiete.* Bd. Bd. 2, 3: *Grundbegriffe der Wahrscheinlichkeitsrechnung*. Repr. [d. Ausg.] Berlin, Springer, 1933. Berlin, Heidelberg, New York : Springer, 1973

[42] Le Maîitre, O. P. ; Knio, Omar M.: *Spectral Methods for Uncertainty Quantification.* Springer, 2010

[43] Le Maître, O. P. ; Najm, H. N. ; Ghanem, R. G. ; Knio, O. M.: Multiresolution analysis of Wiener-type uncertainty propagation schemes. In: *Journal of Computational Physics* 197 (2004), Nr. 2, S. 502–531

[44] Liu, Jun S.: *Monte Carlo strategies in scientific computing.* New York : Springer, 2008 (Springer series in statistics)

[45] Ljung, Lennart: *System identification: Theory for the user / Lennart Ljung.* 2nd ed. Upper Saddle River, N.J. : Prentice Hall, 1999

[46] LUNZE, Jan: *Regelungstechnik.* 8., neu bearbeitete Aufl. Berlin : Springer, 2010 (Springer-Lehrbuch)

[47] MASSEY, Frank J.: The Kolmogorov-Smirnov Test for Goodness of Fit. In: *Journal of the American Statistical Association* 46 (1951), Nr. 253, S. 68

[48] METROPOLIS, Nicholas ; ROSENBLUTH, Arianna W. ; ROSENBLUTH, Marshall N. ; TELLER, Augusta H. ; TELLER, Edward: Equation of State Calculations by Fast Computing Machines. In: *The Journal of Chemical Physics* 21 (1953), Nr. 6, S. 1087–1092

[49] MÜLLER-GRONBACH, Thomas ; NOVAK, Erich ; RITTER, Klaus: *Monte Carlo-Algorithmen.* 2012

[50] NOBILE, F. ; TEMPONE, R. ; WEBSTER, C. G.: A Sparse Grid Stochastic Collocation Method for Partial Differential Equations with Random Input Data. In: *SIAM Journal on Numerical Analysis* 46 (2008), Nr. 5, S. 2309–2345

[51] NOVAK, Erich ; RITTER, Klaus: High dimensional integration of smooth functions over cubes. In: *Numerische Mathematik* 75 (1996), Nr. 1, S. 79–97

[52] OPPENHEIM, Alan V. ; SCHAFER, Ronald W.: *Zeitdiskrete Signalverarbeitung.* 3., durchgesehene Auflage. reprint 2015. Berlin/Boston : De Gruyter Oldenbourg, 1998 (Grundlagen der Schaltungstechnik)

[53] OWEN, Art B.: Controlling Correlations in Latin Hypercube Samples. In: *Journal of the American Statistical Association* 89 (1994), Nr. 428, S. 1517

[54] RASMUSSEN, Carl E. ; WILLIAMS, Christopher K. I.: *Gaussian processes for machine learning.* Cambridge, Mass. and London : MIT, 2006 (Adaptive computation and machine learning)

[55] RIPLEY, Brian D.: *Stochastic simulation.* 2006. Hoboken : Wiley-Interscience, 2006 (Wiley series in probability and statistics)

[56] SALTELLI, A. ; TARANTOLA, S. ; CAMPOLONGO, F.: Sensitivity Analysis as an Ingredient of Modeling. In: *Statistical Science* 15 (2000), Nr. 4, S. 377–395

[57] SINGHEE, Amith ; RUTENBAR, Rob a.: Why quasi-Monte Carlo is better than Monte Carlo or Latin hypercube sampling for statistical circuit analysis. In: *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 29 (2010), Nr. 11, S. 1763–1776

[58] Sinha, N. K. ; Rao, G. P.: *Identification of Continuous-Time Systems.* Dordrecht : Springer Netherlands, 1991

[59] Smith, Ralph C.: *Uncertainty quantification.* Philadelphia, Pa. : SIAM, Soc. for Industrial and Applied Math, 2014 (theory, implementation, and applications)

[60] Smolyak, S.: Quadrature and interpolation formulas for tensor products of certain classes of functions. In: *Soviet Math. Dokl.* 4 (1963), S. 240–243

[61] Steinhilper, Waldemar ; Sauer, Bernd: *Konstruktionselemente des Maschinenbaus 2.* Berlin, Heidelberg : Springer Berlin Heidelberg, 2012

[62] Sudret, Bruno: Global sensitivity analysis using polynomial chaos expansions. In: *Reliability Engineering & System Safety* 93 (2008), Nr. 7, S. 964–979

[63] Tanner, Martin A. ; Wong, Wing H.: The Calculation of Posterior Distributions by Data Augmentation: Rejoinder. In: *Journal of the American Statistical Association* 82 (1987), Nr. 398, S. 548–550

[64] Tarantola, Albert: *Inverse problem theory and methods for model parameter estimation.* Philadelphia, PA : Society for Industrial and Applied Mathematics, 2005

[65] The MathWorks Inc. , Natick, Massachusetts, United States: *MATLAB and Optimization Toolbox.* 2012

[66] Tierney, Luke: *Markov Chains for Exploring Posterior Distributions.* 1994

[67] Tikhonov, A. N. ; Arsenin, V. I.: *Solutions of ill-posed problems.* Washington, D.C. : V.H.Winston, 1977 (Scripta series in mathematics)

[68] Toliyat, Hamid A. ; Kliman, G. B.: *Handbook of electric motors.* 2nd ed., rev. and exp. New York : Marcel Dekker, 2004

[69] Tuzlukov, V. P.: *Signal processing noise.* Boca Raton : CRC Press, 2002 (The electrical engineering and applied signal processing series)

[70] van Valkenburg, M. E.: *Analog filter design.* New York and London : Holt, 1982 (HRW series in electrical and computer engineering)

[71] Wang, Jingbo ; Zabaras, Nicholas: A Bayesian inference approach to the inverse heat conduction problem. In: *International Journal of Heat and Mass Transfer* 47 (2004), Nr. 17-18, S. 3927–3941

[72] Wiener, Norbert: The Homogeneous Chaos. In: *American Journal of Mathematics* 60 (1938), Nr. 4, S. 897–936

[73] Xiu, Dongbin ; Hesthaven, Jan S.: *High-Order Collocation Methods for Differential Equations with Random Inputs.* 2005

[74] Xiu, Dongbin ; Karniadakis, George E.: *The Wiener–Askey Polynomial Chaos for Stochastic Differential Equations.* 2002