Uncertainty Quantification and Sensitivity-based Optimization Methods for Robot Systems

Zur Erlangung des akademischen Grades einer

Doktorin der Ingenieurwissenschaften

von der KIT-Fakultät für Informatik
des Karlsruher Instituts für Technologie (KIT)

genehmigte

Dissertation

von

Woo-Jeong Baek

Tag der mündlichen Prüfung: 11. Juli 2024

1. Referent: Prof. Dr.-Ing. Torsten Kröger
2. Referent: Prof. Dr.-Ing. Tamim Asfour

This thesis would not have been possible without the following people to whom I would like to express my gratitude to.

First, I would like to thank Torsten Kröger for the opportunity to start this journey and for constantly encouraging me to follow my inner intuitions. I am additionally thankful for his advices that go far beyond the PhD and have contributed to my personal growth. Also, words cannot express how grateful I feel to Tamim Asfour for being the best on-site advisor I could have wished for; for his invaluable care, efforts, unconditional guidance during the entire PhD and everything he has done for IPR.

Furthermore, I am extremely thankful to Christoph Pohl for the fruitful collaboration; for showing up during the most challenging period of the PhD, for his encouragement, support and for being someone I can count on. Also, I am grateful to Lars Berscheid who provided me with highly constructive feedback on my papers. Generally, I would like to thank my colleagues Christoph Ledermann for helping me with my first steps at IPR, Michael Mende for making sure everything works at the institute and for providing me with countless amounts of high-quality sweets in the last years. Thanks also go to Jonas Kiemel for saving my life several times as well as Tom Huck, Xi Huang and Patrick Schlosser for their support in finding robotic use cases for my methods.

Moreover, I am deeply thankful to Noémie Jaquier who recently turned into both a personal career mentor and a precious friend whom I can share everything with. Also, I feel grateful to Jongseok Lee for being a role model in many regards and for making me disconnect from the thesis to enjoy life from time to time.

Outside KIT, I would like to thank DKG-BW – especially the board members, and the NJGDK that forced me to cut off from academia every now and then. Here, I would like to particularly mention Jonas Winter who was always open to proof-read German texts.

Within my private circle, I feel sincerely thankful for my incredibly valuable friends I have made during my time in Karlsruhe: Lutz Schimpf, Timothy Gebhard, Hendrik Seitz-Moskaliuk, Alex Mayer, Marie Weiel, Moritz Hackenjos, Moritz Machatschek, Will McGinley, Larisa Thorne, Ana Paula Vizcaya, Doyeong Kim, Sohye Park for their presence, warmth, availability in all kinds of situations and for their efforts to keep the close friendships despite distances.

My deepest gratitude is dedicated to my parents for their endless love, support and for inspiring me to approach challenges with curiosity, open-mindedness as well as positivity. Also, I am grateful to my brother In-Ho Baek for his strong belief in his sister and for simply existing in my life.

Abstract

In computer science, optimization methods have been studied across several subdomains like numerical optimization or program optimization. This thesis revolves around the development of uncertainty-aware optimization techniques for robot systems with a strong focus on quantifying and propagating uncertainties. Especially, the main goal is to study whether integrating the estimated uncertainty in the optimization pipeline can boost the performance of robot systems. To this end, we employ the uncertainty definition established in the field of *metrology*. This research domain provides definitions of measurement units (e.g., meter, kilogram,...) and calibration methods. Importantly, these studies are tightly linked to assessing the accuracy of measurements by evaluating whether the measured value of a quantity corresponds to its actual value.

In robotics literature, the majority of techniques for uncertainty quantification have been developed in an application- or task-specific manner. For example, methods for *handling* the uncertainties in robot navigation or manipulation tasks often propose to first model them heuristically to integrate these into the control architecture. So far, however, scarce attention has been devoted to accurately *quantifying* the uncertainty of entire robot applications by addressing the derivation of *one* unambiguous uncertainty measure. This might draw from the difficulty of unifying different uncertainty notations. Due to the fact that the uncertainty of hardware components (e.g., sensory devices, cameras,...) differs fundamentally from the uncertainty notation of software tools like neural network-based components, deriving one representation for the system uncertainty is challenging.

Nevertheless, the knowledge on the system uncertainty may be critical for assessing and optimizing robot applications. Also, being provided with the relationship between the system uncertainty, that reflects the probability for undesired fluctuations, and the system variables offers the possibility to adapt the system according to the minimum uncertainty. Especially, quantifying the uncertainty of a system becomes important when quantitative limits on system variables or parameters must be kept (e.g., for safety assurance). Obviously, the reliability on the compliance with limits directly depends on the probability for fluctuations.

Incentivized by the need for deriving one single uncertainty measure for an entire robot application to enable the development of uncertainty-aware optimization techniques, this thesis provides following key contributions: After introducing methods to quantify the uncertainty of single components and accumulating them, it is elaborated on how these can be accounted for in the optimization of robot systems. To be specific, three applications are considered by formalizing an optimization problem for each use case. Starting with a distinct binary optimization problem that is developed in the context of a humanoid robot grasp selection experiment, the advantage of integrating uncertainties is demonstrated. Next, the metrological uncertainty notation for technical components is unified with the robustness of neural network classifiers to one combined uncertainty representation. This unification is complemented by sensitivity analyses to explore the validity of the derived uncertainty notation for different system parameter regions. The experiments on an im-

age classification task show that the uncertainty of robot systems that consist of both neural networks and sensory components is effectively captured by the proposed notation. Third, continuous optimization problems are addressed by considering the quantified system uncertainty and the findings from the sensitivity analyses. By mapping the results on the probability for the occurrence of risks, uncertainty-aware safety evaluation and risk minimization methods are derived. The experiments conducted in both simulation environments and real-world robot applications convincingly demonstrate that the uncertainty calculated via the proposed quantification and propagation techniques not only allows the evaluation of quantitative safety limits (e.g., ISO standards) in online manner, but also provides the possibility to effectively minimize risks.

Overall, the results of this thesis show that the developed methods are suitable to quantify the uncertainty of robot systems. According to the findings of the performed experiments, the optimization performance is improved by more than $60\,\%$ on average by integrating the uncertainty. However, the notable discrepancy in the results of the three experiments indicate that the performance of the uncertainty-aware optimization techniques highly depends on the application.

Keywords: *Uncertainty Quantification, Uncertainty Propagation, Sensitivity Analysis, Robot Safety, Risk Analysis*

Zusammenfassung

In der Informatik kommen Optimierungsverfahren vielfältig zum Einsatz, wie zum Beispiel im Kontext numerischer Methoden oder der Optimierung von Programmlaufzeiten. Die vorliegende Dissertation beschäftigt sich mit der Entwicklung von Optimierungsmethoden für Robotersysteme unter Berücksichtigung von Unsicherheiten, wobei ein starker Fokus auf Unsicherheitsquantifizierungs- sowie propagationsmethoden gelegt wird.

So wird untersucht, wie die Integration der ermittelten Unsicherheiten zur Erhöhung der Performanz in Robotersystemen beiträgt. Hierbei wird die Definition der Unsicherheit aus dem Bereich der *Metrologie* verwendet. Dieses Forschungsfeld befasst sich mit der Definition von Messeinheiten (z.B. Meter, Kilogram, usw.) und der Entwicklung von Kalibrationsverfahren, die ebenfalls mit Methoden zur Bestimmung der Genauigkeit von Messungen stark verwandt sind. Ziel dieser ist es, zu ermitteln, inwiefern der gemessene Wert einer Größe mit dem tatsächlichen Wert übereinstimmt.

Existierende Ansätze zur Bestimmung von Unsicherheiten in der Robotik werden zumeist auf applikationsspezifische Art hergeleitet. Insbesondere werden Methoden mit dem Ziel, negative Konsequenzen im Systemverhalten durch das Auftreten von Unsicherheiten zu umgehen, für Navigations- und Manipulationsaufgaben entwickelt, indem zunächst Unsicherheitsmodelle aufgestellt und im Anschluss in der Roboterregelung berücksichtigt werden. Allerdings wurden bisher bei der Entwicklung von Verfahren zur Quantifizierung von Unsicherheiten, die neben der Berechnung des Umfangs der Unsicherheit auch Analysen von Parameterabhängigkeiten mit Systemvariablen beinhaltet, vergleichbar geringe Aufmerksamkeit gewidmet. Dies liegt möglicherweise darin begründet, dass die Berechnung der Gesamtunsicherheit eines Systems erfordert, verschiedene Unsicherheitsnotationen zusammenzuführen. Aufgrund von Diskrepanzen, wie beispielsweise zwischen der Unsicherheitsdefinition von Hardwarekomponenten (z.B. Sensoren, Kameras) und Software-Tools, die auf neuronalen Netzen basieren, ist die Bestimmung einer Gesamtunsicherheit nicht ohne Weiteres möglich.

Nichtsdestotrotz ist der Kenntnis über die Gesamtunsicherheit, die das mögliche Ausmaß von Abweichungen beschreibt, hohe Wichtigkeit im Hinblick auf die Beurteilung und Optimierung der Performanz in Robotersystemen beizumessen. Zudem erlaubt die Kenntnis über die Abhängigkeiten zwischen der Unsicherheit und den Systemgrößen, die Komponenten und Parameterbereiche entsprechend der niedrigsten Auftrittswahrscheinlichkeit für unerwünschte Abweichungen zu adaptieren. Vor allem in Fällen, in denen quantitative Anforderungen wie z.B. Beschränkungen für Systemgrößen erfüllt werden müssen, kommt der Unsicherheitsquantifizierung hohe Bedeutung zu, da die Zuverlässigkeit der Einhaltung von Anforderungen unmittelbar mit der Auftrittswahrscheinlichkeit von Fluktuationen zusammenhängt.

Die Kernbeiträge dieser Arbeit umfassen die Entwicklung von Verfahren zur Unsicherheitsquantifizierung und die anschließende Integration dieser in Roboteranwendungen. Die Herleitung der Methoden erfolgt anhand von drei Beispielen, für welche jeweils ein

Optimierungsproblem definiert wird. Zunächst wird ein diskretes binäres Optimierungsproblem für die Greifselektion bei einem humanoiden Roboter adressiert. Es wird demonstriert, dass die Berücksichtigung von Unsicherheiten zur Verbesserung der Greifselektion beiträgt. Anschließend wird eine Methode erarbeitet, die ermöglicht, die Unsicherheitsnotation technischer Komponenenten mit der Robustheit von neuronalen Netzen zu einer Unsicherheitsgröße zusammenzuführen. Diese Vereinheitlichung wird durch Sensitivitätsanalysen ergänzt, um die Gültigkeit der hergeleiteten Notation für verschiedene Parameterbereiche zu untersuchen. Mittels Experimenten, die sowohl Sensoren und Kameras als auch Klassifikatoren mit neuronalen Netzen verwenden, wird die Gültigkeit der Notation validiert. Im Anschluss werden die Unsicherheiten sowie Resultate aus den Sensitivitätsanalysen in kontinuierliche Optimierungsprobleme zur Evaluation der funktionalen Sicherheit und der Risikominimierung integriert. Es werden Experimente sowohl in der Simulation als auch in der realen Welt durchgeführt. Die Ergebnisse zeigen, dass die Integration der entwickelten Methoden zur Unsicherheitsquantifizierung und -propagation zusätzlich zur online Evaluierung quantitativer Sicherheitsvorschriften (z.B. ISO Standards) eine effektive Minimierung von Risiken bewirkt.

Die Resultate dieser Arbeit zeigen, dass die entwickelten Methoden zur Unsicherheitsquantifizierung für Robotersysteme geeignet sind. Gemäß der Ergebnisse der durchgeführten Experimente ermöglicht die Berücksichtigung der Unsicherheit eine Erhöhung der Performanz um mehr als 60 %. Allerdings lassen sich merkliche Diskrepanzen zwischen den Resultaten der drei Experimente erkennen. Diese weisen darauf hin, dass die Optimierung der Performanz von Robotersystemen unter Berücksichtigung von Unsicherheiten abhängig von der Anwendung ist.

Stichwörter: *Unsicherheitsquantifikation, Unsicherheitspropagation, Sensitivitätsanalyse, funktionale Sicherheit, Risikoanalyse*

Contents

1 Introduction

The advent of technological developments has led to a growing interest in research in the assistive and collaborative robotics domain. In a broader sense, the ultimate goal in engineering sciences is to provide applications and technologies that can assist humans, for example by contributing to more efficient work cycles and a higher productivity. From an application viewpoint, a wide range of promising approaches have been developed in robotics including control frameworks for robot manipulation and navigation tasks. While the performance of these developments is important, the most crucial prerequisite for facilitating the integration of robots in the everyday lives of humans (e.g., for collaboration or close interaction tasks) is safety assurance. Specifically, the term *safety* refers to the absence of unintended hazardous events that occur due to technical failures of the system. Thus, safety is clearly distinguished from *security*, which addresses specifically designed attacks on the software level of the target application.

This thesis draws upon the insight and motivation that estimating the occurrence probability for undesired events is possible via measurement uncertainty quantification techniques. These have been studied extensively in the domain of particle physics and may be transferred to robotics applications for safety evaluation purposes. In particular, the occurrence probability for undesired events in the majority of particle physics experiments is computed by the measurement uncertainty according to [26]. Especially, the goal in natural sciences is to detect unseen natural phenomena with high accuracy to assess whether the experimental results match the theoretical predictions, that are obtained on the basis of the scientific findings from the last centuries. Here, high emphasis is laid on the detection accuracy, that is often referred to by the *resolution*. Despite the theoretical expectations, the resolution in the observation of the novel phenomenon is considered as a representative measure for the reliability on the finding. Thus, achieving a sufficiently high resolution is one of the central goals in natural sciences. Technically, this resolution in the observation refers to the accuracy of the corresponding measurement. The higher the accuracy, the higher the resolution, and thus the reliability on the observation. Importantly, assessing the accuracy is performed inversely by quantifying the inaccuracies, that can be quantified via the *measurement uncertainties*.

In the context of robot applications, the incorporation of inaccuracies in observations or processes may contribute to enhanced performances. By monitoring the critical variables of a system and accounting for potential inaccuracies, a more precise representation of the actual state can be achieved. This representation, in turn, is likely to contribute to an improved situation understanding, thereby enhancing the reliability of action selections made by the robot. Evaluating whether considering these inaccuracies by employing measurement uncertainty techniques according to the Guide to the Expression of Uncertainties in Measurements (GUM) ([26], [13]) is beneficial for robot applications, a variety of experiments can be conducted. However, the goal of these applications must be clearly defined. For example, safety assurance can be formalized as a risk minimization problem. Essentially, the risk can be expressed via measurement inaccuracies that limit the knowledge of the possible undesired system behavior. In this case, it is explored by the

respective validation experiments whether the risks can be reduced when performing the safety assessment in an uncertainty-aware manner. Here, the risk of human-robot collisions in Human-Robot Collaboration is tightly bound on the measurement accuracy of the relative human-robot distance and velocity. In particular, inaccuracies in the measurement and monitoring process of these variables result in incorrect evaluations in risk minimization, which corresponds to an optimization problem. For instance, an underestimation of the relative velocity in human-robot collaboration would directly yield a wrong risk evaluation. Consequently, no measures to reduce the risk would be introduced; the robot would maintain its relatively high speed, potentially resulting in severe injuries to the human. In this context, the validation experiments performed in Section 5.3 aim to study how considering the measurement uncertainties in the estimation of human-robot distance and velocity contributes to a reduction of dangerous situations.

Uncertainty-aware Optimization of Robot Applications
From a broader perspective, the validation of uncertainty-aware optimization techniques can be easily extended to further robot applications. For instance, estimating the grasp selection performance in robot grasping with respect to the grasp success rate depends on the perception accuracy of objects, which can be described by the respective measurement uncertainties. Similarly, the rate of incorrect classifications in an image classification task may be reduced by accounting for the measurement uncertainties in the image recognition and processing step. Generally speaking, accounting for the measurement uncertainties provides a system state representation that corresponds, in the ideal case, to an accurate representation of the reality including the possible deviations. On the basis of this motivation, one core contribution of this thesis is given by the transfer and adaptation of well-established uncertainty quantification and propagation methods that are employed in other research domains, as chemistry, biology and physics to robotic applications. From a practical viewpoint, uncertainty-aware optimization techniques for robot systems are developed and designed to improve the achievement of the application-specific optimization goals. Specifically, this poses the challenge of formulating the goal in robot applications as optimization problems. The goal of selecting promising grasp candidates in a humanoid robot grasping application is defined as a maximization problem regarding the successful grasps. This, in turn, requires to identify an appropriate description of the grasp candidates and their uncertainties in first place. Similarly, the image classification task is referred to by the minimization problem of inaccurate classifications that necessitates the derivation of an uncertainty-aware system description. While these two examples correspond to discrete optimization problems and are thus formulated accordingly, evaluating safety by minimizing risks is treated by continuous optimization techniques, that are designed specifically for this use case.

Terminology of Measurement Uncertainty
The employed terminology of measurement uncertainties follows the guidelines that have been established in the field of *metrology*; a research domain concerned with the assessment and evaluation of measurements and mainly used in natural sciences. To be specific, this thesis builds upon the Guide to the Expression of Uncertainties in Measurement (GUM) in [26] and the respective supplement in [13]. Of course, this requires to describe and formalize the robot systems and the relevant system processes accordingly. At the same time, once these descriptions are derived, analogies between the optimization problems in physics experiments and robot application goals can be drawn, that in turn allow to apply the methods in a straightforward fashion.

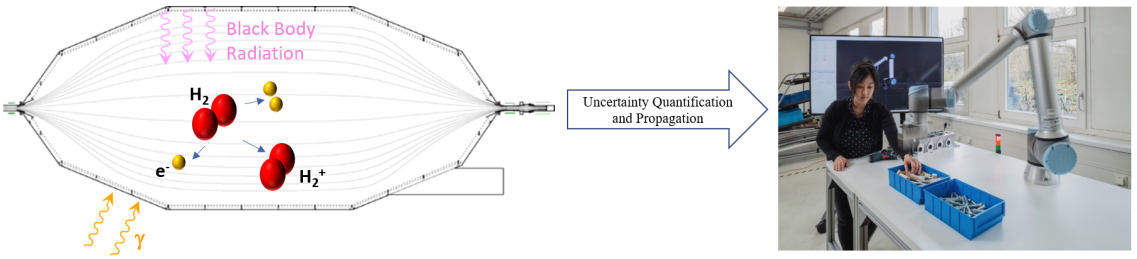


Figure 1.1: The underlying incentive of this thesis is to transfer and adapt measurement uncertainty quantification methods from the domain of particle physics to robotics. In doing so, it is aimed at enabling the straightforward calculation of the accumulated measurement uncertainty of entire robot applications. In particular, the knowledge on this uncertainty is suitable for evaluating safety requirements. Apart from that, the uncertainty quantification methods yield to enhanced performances in other robotic subdomains.

In addition, the discrepancy to physics experiments is also discussed in the face of the limited observability in robotic applications. While possible changes of the system states (e.g., the room pressure and temperature) are continuously captured by hundreds of sensors and maintained with a high accuracy in particle physics experiments, this usually can not be realized in common robot applications. Thus, it is not clear whether the techniques from the metrological domains are suitable for engineering applications, where an observation on microscopic scale is not provided. On the other hand, the majority of the system components are expected to be well-understood. All engineering applications have in common that they are invented and developed by humans, meaning that the entire system is expected to be understood in detail and can be described by accurate models. Hence, this prior knowledge on all system components of robot systems might equal out the missing possibility of the accurate observation, that may be a crucial prerequisite for an accurate uncertainty determination. Incentivized by these questions, establishing approaches that are suitable for robot systems will be addressed within the scope of this thesis, that, among above described challenges, requires to bridge the gap between the existing discrepancies in the uncertainty notations of different subdomains in robotics and to successfully combine them with the formalism in metrology.

1.1 Research Questions and Contributions

The contributions of this thesis are guided by the following three research questions:

1. **Which requirements must be met for the description of the robot system to enable the calculation of the accumulated uncertainty for an entire robot application?**
   After describing a robot system using the metrological formalism, a technique for quantifying the uncertainty of system parts and components with unknown uncertainties is developed. In doing so, the requirements that must be met to treat the robot system with common statistical tools are defined. On this basis, methods for propagating the uncertainties of single components along the system pipeline are introduced. In addition to transferring the techniques from the metrological domain, it will be explored how these can be adapted to robot applications.

2. **How can established uncertainty notations from the domain of robotics and learning be unified with the metrological uncertainty notation to enable the derivation of one single uncertainty measure for an entire robot system?**
   Existing uncertainty definitions that have been established in the robotics domain show differences to the metrological notation of uncertainties. At the same time, analogies can be found that can be exploited to bring the metrological uncertainties and those from the field of robotics together. Especially, the successful unification of these enables the derivation of one unambiguous uncertainty measure for an entire robotic system. This means that the formalization in above research question 1 should be complemented by a technique to combining different uncertainty notations. To this end, the second research question addresses the problem of calibrating the different uncertainty formalizations with each other and studying the analogies in the presence of existing discrepancies.
3. **How can the quantified uncertainties on component level be incorporated in the system to enhance the performance of robot applications?**
   The main goal of this thesis is to study how the incorporation of uncertainties impacts the performance of robot applications. To do so, a measure that is denoted by the term *optimization performance* is introduced. Shortly, this measure indicates the optimization quality – for example with regard to the minimization of risks in human-robot collaboration or the maximization of the number of successful grasps in a robot grasping application. In particular, the optimization performance facilitates the evaluation of whether considering the uncertainties in the system pipeline by calculating them online during run time is beneficial for robot systems.

1.2 Publications

In addressing above research questions, the developed methods and validation experiments have been published in the following works:

- **Woo-Jeong Baek*,** Christoph Pohl*, Philipp Pelcz, Tamim Asfour, and Torsten Kröger. "Improving Humanoid Grasp Success Rate based on Uncertainty-aware Metrics and Sensitivity Optimization". In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pp. 786-793. IEEE 2022 ([10])

  This publication demonstrates that considering the uncertainty of system variables yields the enhancement of the optimization performance on the example of a humanoid robot grasping application. Here, the goal lies in maximizing the grasp success rate by optimizing the grasp selection process in an uncertainty-aware manner. This is achieved by formulating a binary optimization problem. The key idea is to replace the scalar description of the grasp candidates by probabilistic models. By doing so, an uncertainty-aware approach to grasp selection is introduced. The experiments on the humanoid robot ARMAR-6 with more than 1000 real-world grasps show that incorporating the uncertainty in the grasp selection improves the grasp success rate significantly from 32.6 % to 73.8 %.

- **Woo-Jeong Baek,** Christoph Ledermann, and Torsten Kröger. "Uncertainty Estimation for Safe Human-Robot Collaboration Using Conservation Measures". In *Intelligent Autonomous Systems 17*, pp. 85-102, Cham, 2023. Springer Nature Switzerland ([12])

  This work suggests an approach to quantifying the uncertainty of black box tools. While the uncertainty of technical tools is usually provided by data sheets, this might not apply to all components, for example those that employ neural networks. This work proposes a method to determine the uncertainty by exploiting the existence of system characteristics that stay constant over a specified parameter space. In fact, so-called conservation equations are formulated that express that the system variables do not change. To calculate the uncertainty of a black box tool, data on the constant system variables is recorded. By estimating the violations on the conservation equations, the uncertainty of the black box component is calculated. The performed experiments show that the developed method delivers reliable uncertainty estimates.

- **Woo-Jeong Baek**, Christoph Ledermann, Tamim Asfour, and Torsten Kröger. "Combining Measurement Uncertainties with the Probabilistic Robustness for Safety Evaluation of Robot Systems". In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 473-480, 2023 ([11])

  The third paper addresses the discrepancy between the uncertainty notation of NNs and metrology. Especially, if the uncertainty of an entire system is of interest, successfully unifying different definitions is required. To this end, this work aims at combining the measurement uncertainty of technical tools with the robustness of neural network classifiers. To be specific, the analogy between the metrological uncertainty formalism and the robustness in the learning domain is exploited. By performing calibration experiments, sensitive parameter regions are identified. The validation experiments show that the derived uncertainty notation captures the system uncertainty.

- **Woo-Jeong Baek** and Torsten Kröger. "Safety Evaluation of Robot Systems via Uncertainty Quantification". In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10532-10538, 2023 ([9])

  In this contribution, a method to accumulating the uncertainties of single components along the system pipeline is presented. The goal is to determine the uncertainty of an entire robot application to map it on quantitative safety limits. Since these limits are expressed in terms of the probability for the occurrence of dangerous failures, the uncertainty of a system is a reasonable measure to evaluate safety. The correctness of the uncertainty propagation algorithm is verified on the basis of both simulated data and real-world experiments.

1.3 Structure of Thesis

This thesis is structured as follows: After motivating the transfer of metrological uncertainty notation into the field of robotics and engineering, Chapter 2 introduces the fundamentals of the statistical tools and definitions that serve as the basis for the development of the methods in Chapter 3. In addition, existing approaches and notations to calculating

uncertainties in robotics are described. In doing so, high emphasis is laid on the understanding of the term *uncertainty* that is tightly bound on the incentive behind developing uncertainty determination techniques.

Generally, the motivation in robotics is to enhance the performance of the respective application. In fact, the discrepancies to the uncertainty definition established in the metrological domain are pointed out. This enables to identify a discrepancy between these understandings that offers space and possibilities to exploring and deriving novel techniques that may be beneficial for both domains. Chapter 3 intends to introduce and adapt the established measurement uncertainty quantification methods from the metrological domain to robotics. By specifying how the measurement uncertainty is calculated for single components in Section 3.2.1, a technique that enables to determine the measurement uncertainties of black box tools is developed. Afterward, this technique is extended to a pipeline of components. In Section 3.2.2, it is explained how the uncertainties of different components can be combined. Here, the analogy between physics and robotics is emphasized. While accumulating the uncertainties in one representation can be performed in straightforward manner, unifying uncertainties of Neural Networks (NNs) with those of technical components is challenging due to the discrepancy in the uncertainty definitions. To address this issue, a reinterpretation of variables in the robustness definition of NNs is suggested and combined with a calibration.

By employing these techniques, methods to derive optimization methods in robot systems are introduced in Chapter 4. Briefly, the term *optimization* in this thesis describes the process of maximizing or minimizing a property in a robot application that is expressed via the system variables and respective constraints. The optimization techniques in this thesis are categorized in the binary optimization in Section 5.1.3 and Section 5.2.5 and continuous optimization techniques in Section 4.3. Importantly, these optimization methods are developed in an uncertainty-aware fashion by incorporating the techniques from Chapter 3.

In order to validate the proposed methods for the quantification and propagation of measurement uncertainties as well as the uncertainty-aware optimization methods, validation experiments are performed in Chapter 5. Specifically, an application for the selection of grasps in a humanoid robot grasping experiment serves as a use case for validating the binary uncertainty-aware optimization technique. In a similar manner, the correctness of the reinterpretation of the robustness of NNs with the metrological uncertainty notation is verified by an image classification experiment. Since the goal here lies in minimizing incorrect image classifications, where the image classes are categorized in distinct classes, this experiment is suitable for validating the optimization method introduced in Section 5.2.5. The third Section of Chapter 5 deals with the validation of the continuous uncertainty-aware optimization methods presented in Section 4.3 in the context of safety of robot applications. Since the central incentive of this thesis revolves around studying the safety assessment via measurement uncertainties, a variety of experiments in both real-world and simulation environments is performed. By employing the uncertainty quantification method for black-box components in Section 3.2.1, real-world Human-Robot Collaboration (HRC) experiments are performed. In fact, the uncertainty on the human position tracking is computed by applying the methods in Chapter 3. The optimization goal here lies in the minimization of risks.

In contrast, the second part of Section 5.3 deals with simulation studies, where the effect of the measurement uncertainties is explored for the severity of accidents. In doing so, it is aimed at providing a method that enables both the uncertainty-aware evaluation of

quantitative safety requirements and the uncertainty-aware minimization of risks of robot systems. Finally, Chapter 6 summarizes the findings of this thesis by referring to the research questions in Section 1.1. Apart from that, the limitations of the developed methods are clarified. The Section 6.2 intends to suggest possible directions for future research in the context of safety evaluation methods of robot systems. In doing so, it is outlined how further methods from the domain of natural sciences may be employed to robotics, thereby fostering the interdisciplinary transfer between different research domains. A schematic overview of the structure is shown in Figure 1.2.

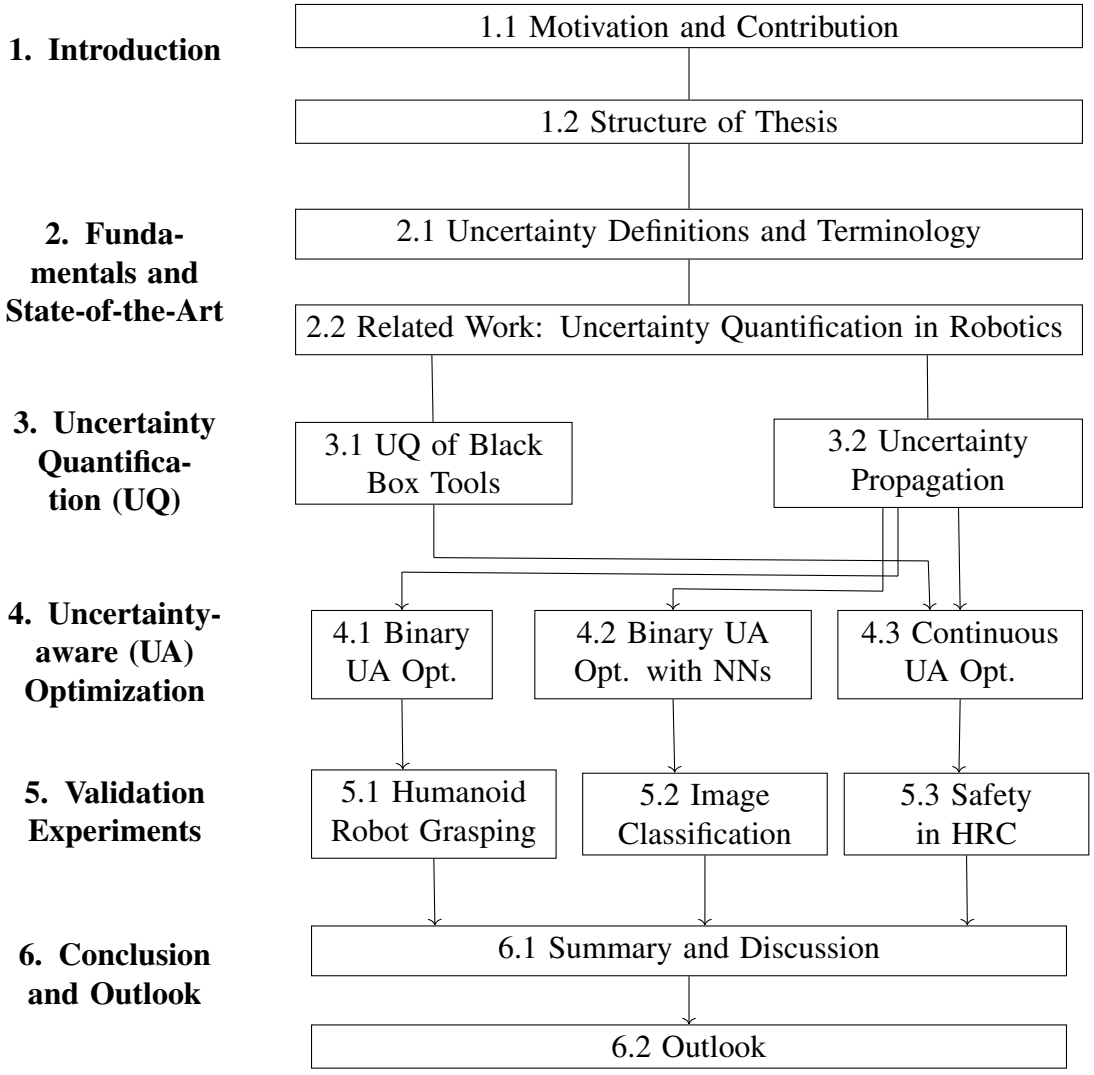


Figure 1.2: Structure of Thesis: After motivating the contribution of this thesis, the necessary fundamentals of uncertainty definitions and state-of-the-art literature on uncertainty quantification methods is provided in Chapter 2. Next, methods for the uncertainty quantification and propagation are introduced, that serve as the basis for the uncertainty optimization, that are validated afterward by means of three robotic applications. Finally, the contributions are summarized and discussed.

2 Fundamentals and Related Work

This Chapter introduces fundamental notions and definitions that will be referred to throughout this thesis. In addition, existing methods to uncertainty quantification in robotics are presented and discussed to highlight the novelty of this thesis. The underlying conjecture of this thesis is twofold:

1. any real-world system can be fully represented by continuously measuring system parameters with perfect (100%) accuracy
2. if one is provided with a perfectly accurate representation of the system, any optimization can be carried out with maximum performance

For example, a human-robot collaboration (HRC) system can be described by means of the human and robot positions and velocities at each time step, where these parameters must be obtained via measurements.

Therefore, the quality of the system representation depends on the perception accuracy of the parameters, that is restricted due to technical limitations of employed tools or the influence of environmental disturbances. These undesired influences cause inaccuracies in the measurement procedure that are denoted with the term *measurement uncertainties*. Apart from that, a perfectly accurate representation of a system would guarantee the performance of optimizations. This means that knowing all parameter relationships as well as their exact values would allow to perfectly adapt to minimizing or maximizing parameters to desired extent. To discuss these conjectures, we investigate the impact of uncertainties in three robot applications.

Prior to presenting our method, we introduce the notations that are used in this work. First, the metrological uncertainty is presented. By means of general terms that are needed for the basic comprehension of this thesis, existing uncertainty representations in robotics literature are outlined to highlight the differences. Second, the term *optimization* is specified with regard to robot applications, where the main objective of this work lies in contributing to the optimization performance in robot systems by incorporating measurement uncertainties. Furthermore, established approaches to uncertainty quantification in the domain of robotics are discussed. Related contributions from the subdomains safety in robotics, neural network based robot systems and humanoid robot grasping will be elaborated on in Chapter 5 by introducing the validation experiments.

2.1 Definitions, Terminology and Notation

This thesis revolves around assessing and evaluating the accuracy of measurements to gain an improved understanding of the system. Generally, the goal of measurements is to obtain a value for a well-defined system property, that are usually represented as parameters. Therefore, the process of a measurement involves the selection of tools and development of methods that enable to accurately determine the parameters. These parameter values

allow to draw conclusions regarding the system state. In the context of robot applications, the knowledge on the current state of a system is required to identify which actions a robot should take. Apart from that, collecting data on the system behavior by means of measurements offers the possibility to describe the application. However, the reliability of the obtained measurement data strongly depends on the employed tools and methods. Particularly, a perfect accuracy of 100 % cannot be achieved in real-world environments due to the limitations arising from the system.

Fundamentally, this thesis presents methods to quantifying these limitations in the measurement process, that are denoted with *measurement uncertainties*. In addition, the goal lies in studying how exploiting the knowledge on the measurement uncertainties can be integrated in reasonable manner in different robot applications. This thesis builds upon the metrological definition of uncertainties provided by the *Guide to the Expression of Uncertainties in Measurement (GUM)* in [26]. Accordingly, the measurement uncertainty reflects the possible dispersion around the measured quantity as will be specified in the following.

2.1.1 Measurement Uncertainties

The uncertainty of a measurement describes to which a quantity, denoted as *the measurand*, might deviate from the measured value. Generally, a measurement denotes the process of estimating the value of a physical property in a system. To achieve this, technical tools and algorithms are employed. However, these devices suffer from limitations such that a perfectly accurate measurement process can not be realized.

Error and Measurement Uncertainty

In addition, it is often not known how environmental disturbances like temperature fluctuations or sudden changes of the lightning conditions affect the quality of measurements. Basically, the measurement uncertainty captures possible deviations due to limitations and fluctuations that might contribute to inaccuracies in the measurement procedure. One important property of measurement uncertainties is that they cannot be entirely diminished, particularly in real-world systems. Any measurement process underlies uncertainties, either due to the lack of available data or technical limitations of employed tools. Practically, a perfect accuracy of $100\,\%$ corresponds to a measurement uncertainty of $u_{system} = 0$, which cannot be achieved in real-world systems. In contrast to *errors*, that are represented by scalar values and state to which extent an obtained quantity must be corrected, measurement uncertainties correspond to intervals that cover the possible values of the measurand. These intervals usually are assigned to probability density functions.

For the remainder of this thesis, the measurement uncertainty will be referred to by the term *uncertainty*. As will be elaborated in the following, uncertainties can be represented by scalars, analytical functions or distributions. Apart from that, the uncertainty is bound to a Confidence Level (C.L.) that is a measure for the significance of the obtained uncertainty estimate referring to the considered data set.

Type A and Type B Measurement Uncertainties

The Guide in [26] distinguishes between Type A and Type B uncertainties. While Type A uncertainties must be determined by applying analyses and combining experimental

results with the knowledge of the system, Type B uncertainties describe those that are known and modeled beforehand. For example, the latter ones are stated by manufacturer specifications of technical tools or the developers of algorithms. Type A uncertainties, in contrast, may originate from system-specific black box components or varying environmental conditions that must be determined individually for each application. As will be elaborated in Chapter 5 the type of uncertainty must be specified for the analyses. However, since both uncertainty types refer to single components, the same notation will be used.

The Propagated Measurement Uncertainty

In most cases, estimating the value for a property of interest requires employing several measurement devices. Specifically, the quantity of interest might depend on parameters that must be measured separately. This often applies to robot systems as for example in safety-critical applications, where the risk is one of the most crucial measures. However, assessing the risk of a robot system requires to specify and accurately measure the occurrence of hazardous events in first place. Therefore, the uncertainty of the property of interest often results from the uncertainty contributions of the respective tools. The *propagated uncertainty* denotes the resulting uncertainty of an all contributing system components with respect to the specified attribute.

2.1.2 Uncertainties and Robustness of Neural Networks

With the increasing employment of deep learning based methods in physical real-world applications as robotic systems, studies on the uncertainties of neural networks have gained high attention. Especially, the learning domain established specific uncertainty definitions given by the aleatoric (data) and epistemic (network) uncertainties that differ fundamentally from the metrological notation. However, the determination of the uncertainty of an entire application, the uncertainties of hardware and software tools must be effectively combined with each other. Therefore, the aleatoric and epistemic uncertainties will be briefly introduced in the following. In addition, the definition for the robustness of neural networks will be presented since this metric shows analogies to the metrological uncertainty definition as will be further elaborated on in Chapter 3.

Aleatoric and Epistemic Uncertainties

In accordance to the metrological understanding, the motivation behind uncertainties for neural networks (NNs) have been first introduced with the idea to provide a measure that indicates possible deviations in the NN output. To be specific, two types of uncertainties are introduced in the domain of NNs [43]. The aleatoric (data) uncertainties, that capture the stochasticity in the input data are referred to as the "irreducible" uncertainties while the epistemic (network) uncertainties that are inherent to the neural network architecture reflect such that can be reduced by observing more data and improving the model. Due to the growing demand of methods to assess the reliability of deep learning tools, a variety of methods to quantifying these uncertainties have been developed reaching from sampling techniques to Bayesian inference approaches. Details can be found in the survey of Gawlikowski et al. in [33]. Although a line of methods have been developed and discussed

within recent years, one critical limitation lies in the missing absolute notions. In fact, the distinction between epistemic and aleatoric uncertainties is only reasonable for specified network architectures and data sets. In particular, the definition of the *reducible*, that is, the epistemic uncertainty is not unambiguous. At first glance, increasing the training data may be one efficient possibility. However, this is only valid for the specified setting of the learning problem, that is, for an input space X, output space Y, hypothesis space H and the joint probability P on $X \times Y$. Particularly, if additional features are considered such that the input space X is replaced by X', the epistemic and aleatoric uncertainties change. Hence, these uncertainties are context-dependent. Especially, this context-dependency is highly case specific. Therefore, these uncertainty definitions are not considered as useful for practical robot applications. In order to combine the uncertainty of NNs with those of hardware components, this thesis refers to the robustness that shares similarities with measurement uncertainties as will be introduced in the following.

Robustness of Neural Networks

In contrast to aleatoric and epistemic uncertainties, the robustness describes the deviation of a NNs output relative to variations in the corresponding input samples. For an input sample $x_0 \in X$, the local robustness is satisfied if the condition

$$\forall x; \quad ||x_0 - x|| \leq \delta \Longrightarrow f(x_0) = f(x) \tag{2.1}$$

holds true, where $f(x_0)$ and $f(x)$ denote the NN output for x_0 and x, respectively and $||\cdot||$ stands for an arbitrary distance metric specified for the input space X. Basically, the global robustness requires the local robustness for all inputs in the input space, i.e.

$$\forall x, x'; \quad ||x - x'|| \leq \delta \Longrightarrow f(x) = f(x'). \tag{2.2}$$

Intuitively, the robustness can be imagined by a ball with radius δ, where all inputs x and x' yield the same outputs $f(x) = f(x')$. However, in real-world applications as robotic systems, meeting the global robustness requirement often is impossible. Due to this reason, Mangal et al. in [55] introduce the probabilistic robustness expressed as follows:

$$\Pr((\|f(x') - f(x)\| \leq k * \|x' - x\|) \mid \|x' - x\| \leq \delta) \geq 1 - \epsilon \tag{2.3}$$

for an input distribution D. Thus, the probabilistic robustness is defined via the Lipschitz equation and bounds the distance between the pair of outputs referring to the distance between the respective inputs instead of requiring the same distance for all input pairs. In addition, Equation (2.3) introduces a probability limit of $(1 - \epsilon)$, where ϵ is specified individually, for which the robustness property must be satisfied. This means that it is not required that all input samples must meet the condition such that the probabilistic robustness is more suitable for real-world physical systems. In fact, the robustness is required in certain regions such that violating the property for specific input points is acceptable.

2.1.3 Optimization Goals

The underlying incentive of this thesis is to study whether the optimization in robot applications can be enhanced by accounting for uncertainties. Generally, an optimization describes the process of minimizing or a maximizing a system property $\pi(\vec{x})$

$$\max / \min \pi(\vec{x}), \tag{2.4}$$

where $\pi : \mathbb{R}^n \rightarrow \mathbb{R}$ is the objective function that is supposed to be maximized or minimized and x denotes a vector of dimension n. In addition, there may exist additional constraints on the space for feasible solutions given by

$$\begin{aligned} g_i(x) \leq 0, \;\; i = 1, ..., m \\ h_j(x) = 0, \;\; j = 1, ..., l \end{aligned} \tag{2.5}$$

where $g_i(x) \leq 0$ and $h_j(x) = 0$ are referred to as inequality constraints and equality constraints, respectively for $m \geq 0$ and $l \geq 0$. In particular, $m = l = 0$ yields an unconstrained optimization problem. Practically, the goal of an optimization is to find the best solution for Equation (2.4) within the space of all feasible solutions. As will be elaborated on in detail in Chapter 5, improving the performance in robot applications can be formalized as optimization problems in many cases. However, identifying the optimization goals and specifying Equation (2.4) and the constraints highly depends on the system and is not always straightforward, especially in the presence of uncertainties. Apart from that, it is crucial to establish a measure that allows to compare how effective the optimization is carried out. Since the goal of this thesis is to analyze to which extent the incorporation of uncertainties influences the optimization in different robot applications, a metric that allows to directly assess the quality of the optimizations is required. For the remainder of this work, the so-called *optimization performance* $\eta : \mathbb{R}^+ \mapsto \mathbb{R}^+$ defined by

$$\eta := \pi_t(\vec{x}) / \pi_p(\vec{x}), \tag{2.6}$$

where $\pi_t(\vec{x})$ stands for the optimization goal $\pi_t(\vec{x}) = \max / \min \pi(\vec{x})$ and $\pi_p(\vec{x})$ the result obtained for the system property after carrying out the optimization. Thus, η captures how effective the optimization has been carried out. In the ideal case, the optimization performance corresponds to $\eta = 0$. Particularly, this measure enables to compare the optimization techniques of different applications. In robotic systems, reaching such an optimization goal often involves adapting the system constellation such as the robot movement or the selection of technical tools. However, especially in real-world robot systems, it is mostly impossible to guarantee that an optimization goal is met for any kind of circumstances. For example, assuring a success rate of $100\,\%$ in robot grasping applications is highly desired. While this might be possible for certain system constellations, designing the application in such a way that the maximum success rate is achieved for all real-world circumstances is almost impossible. Basically, this draws from the missing predictability and knowledge on the system behavior. Therefore, modifying the optimization techniques by accounting for further knowledge on relevant parameters, including their uncertainties, can contribute to improvements in the optimization. In order to evaluate to which extent the incorporation of uncertainties contributes to the optimization performance, different robot applications in both simulation and real-world environments are presented in Chapter 5.

2.1.4 Statistical Tools and Terms

As explained in Section 2.1.1, computing the propagated uncertainty of a system requires the knowledge on the system behavior. Specifically, to obtain the uncertainty arising from an entire system, it must be determined how the variables and parameters are related to

each other. While these relationships may be provided by the system representation in simple cases, the interplay between the parameters of interest must be usually estimated individually, in particular for complex (robot) applications. In order to explore how the relevant system variables and parameters are related to each other, the corresponding data must be thoroughly analyzed. This requires the employment of statistical tests and tools which will be briefly outlined in the following.

Covariance and Correlation Matrix

The variance in multiple dimensions can be represented by means of a covariance matrix that consists of the covariance values between pairs of elements. For a column vector $\vec{X} = (X_1, ...X_n)^T$, where the entries are random variables with variances and expected values, the entries (i, j) of the covariance matrix can be calculated via

$$K_{X_i,X_j} = \mathrm{cov}[X_i, X_j] = \mathbb{E}[(X_i - \mathbb{E}[X_i])(X_j - \mathbb{E}[X_j])]. \tag{2.7}$$

Here, $\mathbb{E}[\cdot]$ stands for the expectation value. Any covariance matrix is positive semi-definite and symmetric. The correlation matrix can be seen as an extension of the covariance matrix. Specifically, the correlation matrix contains the covariances of the standardized random variables $X_i/\sigma(X_i)$ for $i = 1, ..., n$, i.e.,

$$\begin{pmatrix} 1 & \frac{\mathbb{E}[(X_1-\mu_1)(X_2-\mu_2)]}{\sigma(X_1)\sigma(X_2)} & ... \\ \frac{\mathbb{E}[(X_2-\mu_2)(X_1-\mu_1)]}{\sigma(X_2)\sigma(X_1)} & 1 & ... \\ ... & ... & 1 \end{pmatrix}. \tag{2.8}$$

Since the correlation values are standardized, each off-diagonal element lies between -1 and 1. In contrast to the covariance, the correlation coefficient, often denoted with ρ, additionally captures the strength of the relationship between the element pairs. While a negative correlation coefficient indicates that the two variables move in opposite directions, $\rho \geq 0$ hints at a positive relationship. In case the two variables are independent from each other, i.e., one stays constant while the values of the other variables vary, the correlation coefficient amounts to $\rho = 0$. Importantly, the correlation does not provide any information on causal relationships. Thus, care is needed for the interpretation of ρ. While correlations between values can be found, the relationship may be due to other factors that are not considered in the determination of ρ. In order to study the cause and effects, additional analyses such as statistical tests or sensitivity analyses must be performed.

2.1.5 Sensitivity Analysis

Apart from exploring the relationships between system variables and parameters, analyzing how the uncertainty of a model or system can be allocated to the uncertainty sources of the system components is crucial. The so-called *sensitivity* is a measure that captures this phenomenon. Specifically, it reflects how variations of input variables affect the outcome of the considered model. Usually, sensitivity analyses are treated separately from the quantification and propagation of uncertainties, that are assigned to *uncertainty analysis* in literature. However, in this PhD thesis, the uncertainty quantification and propagation steps are combined with the sensitivity analysis depending on the application as will be

discussed in detail in Chapter 5. Here, the goal is to provide a short summary of the some representations of the sensitivity that are referred to in this thesis. Basically, the representations differ in the type of sensitivity measure. In the following, we consider a model $Y = f(X)$ for the input variables $X = X_1, ..., X_k$.

One-at-a-Time (OAT) Method

The One-at-at-time (OAT) method suggests varying one of the variables of interest while keeping the remaining ones constant. In doing so, it is observed how this variable affects the output of the considered system. This procedure is repeated for all variables and parameters of interest. Hence, the sensitivity is estimated by monitoring the variations in the output. The drawback of this technique is that simultaneous variations of variables cannot be considered such that possible interactions might be neglected. Therefore, the OAT method is not suitable for nonlinear models.

Derivative-based local Methods

Here, the sensitivity is measured by means of the partial derivative of the system output Y with respect to the input X_i, i.e.,

$$|\frac{\partial Y}{\partial X_i}|_{x^0}, \tag{2.9}$$

where x^0 denotes some fixed point in the input space. Thus, the sensitivity is evaluated at specified points, aiming at examining small perturbations instead of observing the entire input space. In particular, this derivative-based local methods allow to generate a matrix that contains the sensitivity values of relevant input variables which is impossible with global methods.

Variance-based Methods

Third, the sensitivity can be estimated by variance-based methods. To be specific, the variance of the model output is decomposed and attributed to the respective input variables as well as combinations of input variables. Practically, the sensitivity is expressed via conditional expectations:

$$\mathrm{Var}(\mathbb{E}_{X_{\sim i}}(Y|X_i)), \tag{2.10}$$

where $X_{\sim i}$ denotes the set of all input variables except i. In particular, above equation measures the contribution of X_i alone on the variance in the output Y and is usually referred to by the *first-order sensitivity index* or *main effect index*. It is worth noting that this measure does not capture the uncertainties due to interactions of the variables. To account for these, the calculation of the *total effect index* is necessary that returns the total variance in Y caused by X_i and interactions with the remaining variables. In addition, both the total and first-order effect index can be standardized by dividing with $\mathrm{Var}(Y)$. Importantly, variance-based methods enable the full exploration of the input space by considering for the interactions between variables and nonlinear responses. However, one disadvantage occurs in complex applications where the calculation might not be straightforward. In these cases, calculating the sensitivity involves the use of Monte-Carlo sampling methods.

2.1.6 Hypothesis Tests and Confidence Interval

One possibility to assess the output of measurements and the calculation of uncertainties is to calculate the respective confidence interval. This measure indicates the significance of the obtained result with respect to the underlying data set. Often, collecting a sufficient amount of data that allows to estimate the confidence with a reasonable accuracy is challenging in real-world applications.

To overcome this, the bootstrap technique can be applied. According to the Central Limit Theorem (CLT), the mean value of a data set consisting from random samples that are identically distributed can be approximated by a standard normal distribution. The bootstrapped distribution serves as the basis for hypothesis tests and the determination of the confidence interval. While the confidence interval can be directly deduced, hypothesis testing provides the possibility to study the validity of statements for the underlying data. In practice, a null hypothesis, an alternate hypothesis and a significance value are specified. The result of hypothesis tests indicate to which extent the null hypothesis is rejected. Therefore, hypothesis testing becomes relevant to explore whether parameter dependencies exist. In case the result hints at dependencies, deeper analyses methods are employed to study the type of relationships.

2.1.7 Monte-Carlo Sampling Methods

In general, the goal of Monte-Carlo (MC) methods [20] lies in approximating intractable solutions by applying random sampling techniques. In particular, there exist many problems where analytical solutions cannot be directly calculated – for example, the sum of discrete distributions or the integral of continuous distributions that are intractable to calculate. This intractability may be attributed to various reasons such as the large number of random variables, the stochastic nature of the domain or the lack of observations. Instead of using analytical tools, defining or estimating probability distributions for the respective random variables by means of theoretical models or computational simulations and sampling from these distributions is often more efficient.

In particular, sampling typically allows for a higher flexibility at reduced computational costs for the approximation of integrals. With regard to the contribution of this thesis, MC sampling is used, for example, to obtain approximates for the propagated uncertainty in Section 2.1.1. While in the following, a brief outline will be provided for the sampling methods employed in this thesis, details can be found in [20]. Generally, the key concept of a MC sampling run consists of drawing multiple random samples from a set of probability distributions that are provided beforehand. These distributions can be of any kind. Broadly, the three main steps are given by

1. Identifying all input components of the process of interest and defining the parameters of the distributions
2. Sampling from each of the distributions and integrating the results.
3. Repeating the process many times

Due to the effect of the Central Limit Theorem (CLT), the resultant parameter converges toward a normal distribution for any types of source distribution. This will be further discussed in the context of the use cases in Chapter 5. There exist different types of MC methods depending on the sampling procedure and the constraints imposed on these. One

challenge when applying MC methods is to attain a low variance in the approximation error (VAE). In fact, this measure enables to compare the different types of sampling methods and their performances in literature [41]. In this thesis, we apply following two techniques:

Direct Monte-Carlo Sampling

Direct MC Sampling describes the process of randomly drawing data samples from the input probability distributions. The key characteristic of this technique is that no prior information is taken into account. According to the law of large numbers, the distribution of interest can be approximated. While direct MC sampling methods are widely applied across different domains and are immensely popular due to their simplicity, particular attention must be paid to regions with sparse probability densities. Since MC sampling refers to randomly generating samples, only few samples are drawn in the areas where the probability density is low. As a consequence, the statistical significance of the corresponding approximation suffers. The quality of the approximation is measured by so-called variance of the approximation error (VAE) that is defined by

$$Var(X) = \mathbb{E}[X^2] - \mathbb{E}[X]^2. \tag{2.11}$$

In the ideal case, the VAE amounts to $Var(X) = 0$ that corresponds to a perfect approximation of the MC sampling procedure. However, this is often challenging to achieve. In the context of this thesis, one limitation occurs due to low probability densities in interesting parameter regions. Logically, randomly generating samples yields to lower sample sizes in regions with sparse probability densities. Consequently, the analyses and conclusions derived on the basis of these samples suffer from a low statistical significance.

Importance Sampling

One possibility to overcome the issue of high VAE values in regions with sparse probability densities is given by importance sampling. In particular, a second latent distribution is introduced to increase the number of samples in regions with low probability densities. Specifically, an additional weighting factor must be considered to perform approximations. For an input space $X : \Omega \mapsto \mathbb{R}$ with independent random samples $x_1, ..., x_n$ and some probability space (Ω, F, P), the expected value of a function $f(x) \mapsto \mathbb{R}^n$ with $x \propto p(x)$ for a probability density function $p(x)$ is calculated by

$$\mathbb{E}(f(x)) = \int f(x)p(x)dx. \tag{2.12}$$

While direct MC sampling provides accurate approximates for regions with reasonable probability densities, the VAE value becomes large in areas where $p(x)$ is difficult to sample from. Specifically, the generation of a sufficient amount of samples is challenging in regions where $p(x)$ is very small. Here, importance sampling suggests to introduce a so-called *proposal distribution* $q(x)$, i.e.

$$\mathbb{E}(f(x)) = \int f(x)p(x)dx = \int f(x)\frac{p(x)}{q(x)}q(x)dx, \tag{2.13}$$

where x is sampled from $q(x)$ and $q(x) \neq 0$. The core idea behind this is to simplify the sample generation by defining $q(x)$ in the region of interest. Logically, the calculation

of $\mathbb{E}(f(x))$ must be modified accordingly by the fraction $\frac{p(x)}{q(x)}$ as shown in above equation. Formally, sampling from $q(x)$ is justified by considering a random variable Y with independent random saples $y_1, ..., y_n$ such that

$$\mathbb{E}(f(y)) = \int f(y)\frac{p(y)}{q(y)}dy. \tag{2.14}$$

Practically, the sample generation can be now performed on the basis of the proposal distribution. Afterward, the so-called *importance weight* $\frac{p}{q}$ is applied to perform the resampling. In particular, the approximation accuracy of importance sampling depends on the definition of the proposal density q. Referring to above Equation (2.11), the mapping is given by $X \mapsto f(x)\frac{p(x)}{q(x)}$ such that the variance increases for large fractions $\frac{p(x)}{q(x)}$. Importantly, a low variance value can be achieved for $f(x)\frac{p(x)}{q(x)} = const.$ which leads to the thumb rule

$$q(x) \propto \|f(x)\|p(x). \tag{2.15}$$

Thus, it is desired to define the latent distribution $q(x)$ in regions, where the product between $f(x)$ and $p(x)$ is large. Since importance sampling enables to reduce the VAE, it is often referred to as a variance reduction technique. In the context of this thesis, a technique based on importance sampling will be derived to study whether focusing on the relevant parameter regions during the sample generation process yields improvements in the optimization of robot systems. Specifically, a grid-based approach will be presented in Section 4.3.6.

2.1.8 System Representation and Filtering Algorithms in Robotics

As will be specified in the application-specific Chapters 5, robot systems in this thesis are described by means of the notations used in [26] that rely on the metrological representation. However, in robotics, a system representation is formalized in a different manner. In addition, uncertainties are often referred to by the term *noise* in engineering sciences. In contrast to the definitions and distinction of uncertainties as presented in Chapter 3, the noise typically describes any types of undesired fluctuations. For completeness, and to allow the derivation of analogies, the terminology and notation used in engineering literature will be briefly summarized as follows.

System Description

Generally, the state of a system is represented by state variables such as the robot position or velocity in the respective parameter spaces. In most cases, this representation is given by a real-valued vector $\vec{x}_n$ for a discrete time step n. Particularly, the system state plays an essential role for developing control algorithms in localization, navigation or manipulation tasks. For example, a continuous time system can be formalized by a system and observation equation written as

$$\dot{x} := f(x(t), u(t), \omega(t)) \tag{2.16}$$

$$z(t) := h(x(t), v(t)). \tag{2.17}$$

Here, f describes the state transition function, $u(t)$ the control input, $\omega(t)$ the process noise, $z(t)$ the observation and $v(t)$ the observation noise. This set of equations is often referred to as the *state-space representation*. Importantly, it is assumed that the noises are additive by relying on Gaussian distributions for the process and the observation noise which forms one of the main discrepancies to the understanding of uncertainties of this thesis.

Filtering Algorithms

Filtering algorithms are employed across a wide range in robotics to avoid consequences due to the noise in the real-world environment. Relevant filters in the context of this thesis are given by the Kalman Filter (KFs) and Particle Filters (PFs) [75]. While both techniques predict and correct the state estimation by referring to observations, they rely on fundamentally different representations. The underlying principles of these two approximation methods are outlined in the following. Details can be found in [75].

Kalman Filters

Kalman Filters (KFs) are used as an essential component in many tracking or navigation applications. Basically, KFs operate in a "predict-correct" loop, where the system state and the corresponding uncertainty of the next step are predicted. In order to predict the future system state, that is described by a set of target parameters (e.g., position, velocity and acceleration), the KF is provided with the current state space model, the dynamic model and the corresponding noise models (measurement and process noise), the initial system state $\hat{x}_{00}$ and the corresponding variance p_{00}. This step is denoted as initialization. In addition, the measured system state z_n and the respective variance $p_{n,n-1}$ that are obtained via sensors, are considered. Based on this information, the state update process computes the subsequent system state via the state update equation

$$\hat{x}_{n,n} = \hat{x}_{n,n-1} + K_n(z_n - \hat{x}_{n,n-1}) = (1 - K_n)\hat{x}_{n,n-1} + K_n z_n \tag{2.18}$$

where K_n stands for the Kalman gain that weights the measurements. For a low measurement uncertainty compared to the estimated uncertainty, one would obtain a high Kalman Gain (close to 1) while a high measurement uncertainty relative to the estimated uncertainty would yield a low value for K_n (close to 0).
In analogy, a state vector must be provided in case of multivariate KFs. Specifically, a corresponding state space representation must be identified including the state transition and transition matrices that allow to solve the state space differential equations. Accordingly, the state update equation in matrix form results in

$$\hat{x}_{n,n} = \hat{x}_{n,n-1} + K_n(z_n - H\hat{x}_{n,n-1}). \tag{2.19}$$

Again, the estimated system state vectors are given by $\hat{x}_{n,n}$ and $\hat{x}_{n,n-1}$, respectively and the Kalman Gain by K_n. The observation matrix H transforms the state space representation into the measurement space. The corresponding covariance update is performed according to

$$P_{n,n} = (I - K_n H)P_{n,n-1}(I - K_n H)^T + K_n R_n K_n^T, \tag{2.20}$$

where $P_{n,n}$ and $P_{n,n-1}$ stand for the covariance matrix of the current state estimation and the prior estimate of the covariance matrix, respectively. The measurement noise covariance matrix is denoted by R_n.

While above equations are valid for linear systems with additive and independent noise models, slight adaptations must be made in case of nonlinearities. In the nonlinear version of the KF, denoted as the *Extended Kalman Filter (EKF)*, the state transition and observation models do not need to follow linear functions. In fact, the EKF linearizes about an estimate of the current mean and covariance by accounting for the first-order derivative of the Taylor expansion. After transforming the problem into the linear space with the Jacobian matrix, above equations can be applied.

Particle Filters

In accordance to the Kalman Filter, the goal of particle filters (PFs) is to approximate a posterior density function of hidden variables, such as state variables, by referring to observations. However, it is not assumed that the posterior can be modeled by Gaussian distributions. While the representation in KFs is given by mean and covariance values that parameterize the density, the approximation in particle filters is carried out by a weighted set of samples that are denoted as *particles*. Specifically, these particles are generated randomly in the first step according to an initial probability density function of the state. To estimate the hidden variables x_k by means of the provided observation variables $y_0, ..., y_k$ for any time step k, the particles are propagated according to the nonlinear system dynamics. This process is denoted as the resampling step.

Importantly, the system dynamics are provided by means of an analytical function or a probabilistic model to the PF. In contrast to KFs, PFs do not make use of local linearization methods. In addition, no restrictions are made on the noise distributions such that PFs are generally applicabile. However, the main disadvantage of PFs lies in the computational burden. Since the number of necessary evaluations grows with the amount of particles, the high computational power is required for highly nonlinear and complex systems. Thus, PFs are suitable for cases where Gaussian approximations are not valid for the posterior distribution and high computational costs are acceptable.

2.2 Related Work

2.2.1 Uncertainty Quantification and Propagation in Robotics

The incorporation of uncertainties in the field of robotics has been mainly addressed in the subdomain of navigation. In 1996, Kaelbling et al. provided one of the first works in [17] demonstrating that the consideration of uncertainties in the robot control architecture by means of Bayesian models leads to improvements in the navigation performance. Basically, the authors develop heuristic control strategies by referring to POMDPs (partially observable Markov decision processes). In contrast to MDPs, POMDPs do not rely on the assumption that the state is known with certainty. Instead, the probabilistic nature of the perception due to uncertainties is taken into account. By assuming Markovian states, the robot pose is represented via parametric probability distributions. Specifically, these probability distributions are modeled as Gaussian functions that are supposed to fully capture the uncertainties in the robot's observations and actions.

In addition, a topological map is generated by means of the robot pose to obtain a belief state. To update the belief distributions, the Bayes rule is applied. Here, the degree of

uncertainty is captured via the entropy that is calculated over all possible belief states and compared against a pre-defined threshold. Specifically, the MDP-based policies are followed while this threshold is not exceeded. At the same time, the authors aim to reduce this entropy while following these policies by selecting the action with the minimum expected entropy. In doing so, a system architecture based on two levels is introduced. First, a higher level navigator that uses the Bayesian representation of the environment to select reasonable actions is described.

Next, the authors develop a low-level software to bridge the gap between the actions and the abstract model built upon the robot observations. The validation is conducted in both simulation and real-world environments. Here, the authors distinguish between cases where the robot knows its starting point and those where the robot starts off without prior knowledge on its starting conditions. By comparing their two-level control architecture with further heuristic control strategies, it is demonstrated that the suggested method outperforms state-of-the-art approaches regarding the navigation performance. However, although the promising results in the experiments showcase the validity of the Gaussian assumptions for the uncertainties, it is not studied in detail whether or to which extent the uncertainty representations match the ground truth uncertainties.

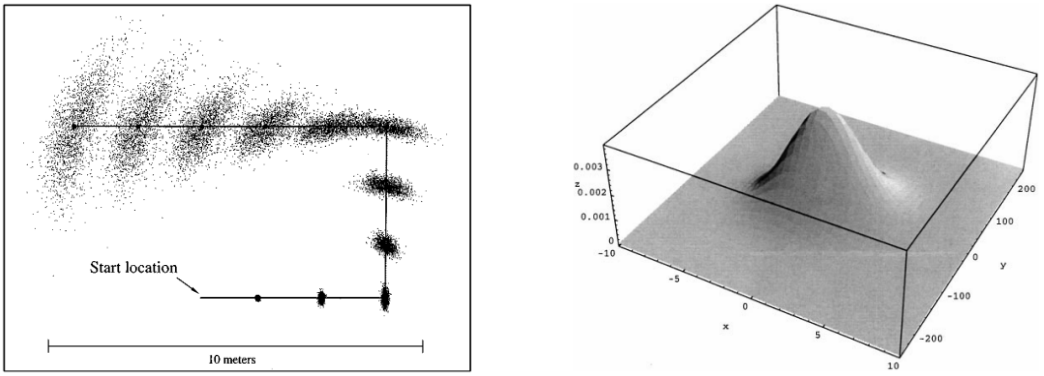
Probabilistic Approaches for Robot Localization and Mapping
This work was followed by Thrun et al. in [74] where different probabilistic approaches are suggested to account for uncertainties in localization and mapping tasks. Basically, the authors contend that modeling parameters as probability distributions improves the robot navigation performance compared to cases where only scalar parameter values are considered. This conjecture aligns with the underlying motivation of this thesis. However, one crucial difference lies in the uncertainty representation.

Generally, Thrun et al. introduce belief states that build upon the Markov assumption to generalize the kinematics of the corresponding robot in probabilistic manner. In the next step, the Bayes rule is applied to calculate the posterior distributions for localization and mapping purposes. To be specific, the probabilistic models for the robot perception and kinematics are modeled as an integral that allows to calculate the belief states for the next steps. In order to perform the localization, the authors introduce Monte-Carlo Localization (MCL) algorithms and derive the belief distributions by sampling uniformly from these. In addition, three algorithms are discussed regarding the generation of maps based on sensor measurements.

For instance, the use of particle filters is proposed, where the posterior is supposed to be represented by samples that are drawn in a specified manner (e.g., via importance sampling). In this context, the Kalman Filter (KF) as well as the Extended Kalman Filter (EKF) are introduced that provide the robot with measurement updates under the consideration of uncertainties. Apart from that, the authors present the EM algorithm that consists of an expectation and maximization step to achieve an enhanced mapping performance. Here, the expectation is calculated by means of the joint likelihood obtained via the measurement data and robot poses conditioned on the k-th map of the previous step. The maximization step then provides the most likely maps by considering these posterior estimates.

The third algorithm presented by Thrun et al. is given by the occupancy grid mapping. In contrast to the previous methods, the work space is modeled by means of grid cells. Based on the assumption that the robot poses are known, it is determined which grid cells are occupied to perform the localization and mapping procedure. In summary, Thrun

et al. develop approaches to model the perception in probabilistic manner, allowing the robots to handle ambiguity. In doing so, they distinguish between probabilistic perception and probabilistic control. While the performed experiments confirm that the navigation performance can be enhanced by modeling the robot's kinematics and the perception in a probabilistic fashion, the authors clearly state that the uncertainty models are crude. Therefore, while these works aim at considering the possibility of deviations, it is not specifically addressed how the exact amount and relationships with the system variables can be estimated. Similar to the above mentioned contribution in [17], the uncertainties are assumed to follow Gaussian distributions.



(a) Sampling-based calculation of robot belief states.

(b) Gaussian density representing the robot perception model.

Figure 2.1: Probabilistic robot localization approach presented by Thrun et al. in [74]. The figure on the left hand side illustrates the sample-based estimation of the robot position belief. It can be seen that the uncertainty arises due to the missing knowledge on the trajectory for higher distances from the starting point. In subfigure 2.1b, the Gaussian density for the robot perception model is shown.

Calculus-based Uncertainty Propagation for Robot State Estimation

More recent contributions build upon above techniques by adapting the sampling processes or the selection of particles. For example, Hill et al. provide a survey on the quantification and propagation of uncertainties in [42]. First, the authors introduce a general calculus-based approach to uncertainty propagation which is given by an approximation via Taylor series. This method is extended to the case of mutivariable functions in N dimensions by means of the covariance matrix. Afterward, the KF, EKF and the Unscented Kalman Filter (UKF) are discussed regarding the state estimation problem of robots.

Basically, these techniques build upon the aforementioned calculation method of uncertainties by relying on the Markov assumption. By highlighting that the EKF underestimates the uncertainties for general nonlinear systems, the authors suggest the use of the UKFs. Instead of linearized matrices, UKFs consider a fixed number of weighted sigma points for the uncertainty propagation. Here, the mapping is performed via nonlinear equations. However, KFs are limited to unimodal distributions such that propagating the uncertainties of more complex systems becomes practically impossible.

Therefore, the authors propose the use of particle filters (PFs) for systems where the approximation of more complex distributions is required. In fact, they suggest to combine

EKF approaches with PF techniques to efficiently avoid the computational burden due to the high number of particles which allows to deal with both nonlinearities as well as multimodal distributions. In the experiments, Hill et al. first validate the calculus-based method in simulation. Especially, they distinguish between full-mapping approaches, where the state space parameters and the corresponding uncertainties are directly transferred to the world frame, and the piecewise mapping algorithms, where the mapping of the uncertainties is carried out in an iterative fashion. In all cases, the propagated uncertainty provided by the Monte-Carlo (MC) sampling method is considered as the baseline. For the full mapping approaches, the authors show that the first-order term in the Taylor series yields reasonable approximations of the propagated uncertainty for negligible nonlinearities. In the case of significant nonlinearities, the UKF outperforms the Taylor series propagation.

However, the authors find that in some cases, the UKF overestimates the uncertainty. In contrast, the Jacobian mapping, which represents the Taylor series approximation for multivariable systems, mostly yields an underestimated uncertainty value. Comparing these values with the respective results obtained by the piecewise mapping approaches, no difference can be found in the case of the Jacobian approach. To be specific, applying the transformation in an iterative manner leads to the same results as the full mapping Jacobian method. In contrast, the unscented transform approach yields different results for the piecewise and the full mapping approach.

In particular, the number of sigma points is recalculated for each transformation, whereas these sigma points are required to properly estimate a distribution for a given mean value and covariance matrix. Due to the fact that the means and covariances of each step influence the subsequent transform, the results show significant discrepancies. In summary, the authors of this paper elaborate on the advantages and limitations of these uncertainty quantification techniques by focusing on robot localization problems. In doing so, they emphasize that all of these methods rely on the assumption of normally distributed data such that possible nonlinearities in the uncertainties are not captured.

Uncertainty Computation via Particle Filters

Another line of contributions as [35] and [30] develop adaptive particle filters. Generally, they aim for more accurate representations of the real-world environment. For example, the authors in [30] suggest to adapt the samples on the fly. To do so, the posterior probability density over the state space is conditioned on the measurement data. Specifically, it is estimated to which extent the sample data matches the sensor readings to derive importance weights. While low importance weights hint at unexpected features in the data and thus suggest to collect additional data points, high weight values indicate that the considered sample set is representative.

Here, the uncertainty arising from the sampling process is determined via the variance of the importance sampler. In order to derive an uncertainty bound by means of the Kullback-Leibler divergence to the posterior, the authors pose the assumption that the true posterior can be modeled as a discrete piecewise constant distribution, represented by a discrete density tree or multidimensional histograms. Importantly, Fox et al. demonstrate that estimating the number of bins which are supported during the sampling process is sufficient to approximate the posterior. In their experiments, the authors make use of sonar scan data and account for the odometry measurements of the robot. They show that the true posterior can be well-approximated by applying their method. However, one of the main limitations is that the implementation relies on discrete distributions with fixed bin sizes.

While this assumption might be efficient regarding the computation time, it is likely to result in significant inaccuracies in the uncertainty calculation.

A similar contribution to optimizing particle filters is presented by Grisetti et al. in [35]. The authors introduce so-called Rao-Blackwellized particle filters (RBPFs) to address SLAM problems. In their approach, each particle represents an individual map of the environment. To develop an efficient method, the authors deal with the question of how to reduce the number of particles while maintaining the necessary amount of data. The ultimate goal lies in estimating the joint posterior of the map and the robot trajectory. Specifically, two approaches are introduced. First, a proposal distribution that accounts for the accuracy of the robot's sensors is generated. This computation is based on the most likely robot pose that is obtained by combining the most recent measurement data with the robot odometry information. Secondly, an adaptive resampling strategy is suggested to keep the diversity of particles. Formally, the authors suggest a factorization denoted as Rao-Blackwellization to address the SLAM problem. To obtain the proposal distribution, an observation model and a motion model are considered. In order to draw samples, Gaussian approximations are made for the data points. After generating the sample points within the corresponding intervals, the mean and covariance matrices are determined to calculate importance weights. These importance weights are supposed to hint at meaningful areas of the observation function.

Grisetti et al. validate their approach in real-world environments and the respective datasets. Particularly, they demonstrate that the developed method leads to accurate maps. Their findings show that maps generated without the odometry data suffer from inaccuracies which emphasizes the relevance of the suggested Rao-Blackwellization. However, the evaluations are performed via visual inspection such that a quantitative assessment of the suggested approach is missing.

Uncertainty Estimation in Metrological Fashion

Apart from these works that focus on navigation tasks and the optimization of traditional methods as particle filters or Kalman filters, contributions that follow the uncertainty estimation methods from the field of metrology can be found. The paper [67] by NIST (National Institute of Standards and Technology) strongly motivates the consideration of measurement uncertainties in robot systems. Particularly, the authors agree with the basic conjecture of this thesis by arguing that the knowledge of measurement uncertainties is required to achieve high performances of robot applications. In this paper, the authors highlight the relevance for determining and incorporating the uncertainty in robot applications by referring to an assembly tasks. Here, they clearly demonstrate that the how rotational and translational uncertainties of the pose measurement affect the localization performance. Importantly, their findings show that the correlations between critical parameters must be considered to obtain accurate uncertainty estimates.

A separate work in [71] presented by Santolaria et al. underlines the relevance of incorporating uncertainties for robot calibration purposes. In this contribution, a technique to assess the uncertainties for a robot arm calibration process is developed. The authors rely on the so-called circle point analysis (CPA) method that suggests a robot calibration principle referring to the spatial localization of the joint axis for a specified robot configuration. By representing the robot joint axis in Pluecker coordinates, Santolaria et al. show that the kinematic parameters of the robot can be computed with the CPA method. In order to estimate the uncertainties, the authors follow the techniques presented in GUM. Hence, the fundamental idea to calculate the uncertainties in robot systems on the basis

of the metrological notations aligns with the underlying motivation of this thesis.

On the other hand, Frankhauser et al. present an uncertainty-aware elevation mapping method that efficiently handles drifts in the robot pose estimation in [67]. The main contribution of this paper lies in the formulation of a mapping algorithm that considers for the distance sensor uncertainties and the six-dimensional pose covariance of the robot. By performing measurements relative to the robot, the elevation map is updated with the robot motion. Specifically, regions ahead of the robot that are measured with higher frequencies are assigned to low uncertainties while areas apart from the sensor's field of view are supposed to underlie higher uncertainties. In order to propagate the uncertainties, the authors perform the fusion of subsequent maps based on the rule for combining standard deviations of multiple data sets. In analogy to the above works, this paper relies on the Gaussian assumption for the uncertainties.

Uncertainty-aware Methods for Robot Path Planning

A further publication of Van den Berg et al. in [77] proposes the optimization of robot path planning algorithm by accounting for both motion uncertainties and imperfect state information. To this end, the authors develop a linear-quadratic Gaussian Motion Planning (LQG-MP) to consider for all employed controllers and sensors during the execution of the path. Their motivation arises from the assumption of noisy sensors that limit the amount of information of the state. Specifically, Van den Berg et al. suggest a Kalman Filter that incorporates a stochastic observation model which in turn considers for measurement uncertainties of the sensors. In order to provide optimal control inputs during the path execution, the control policy refers to a quadratic cost function that is minimized. This method is validated by robot experiments, where the collision probability with obstacles is computed. The results show that the developed method outperforms the majority of uncertainty-aware planners in terms of the collision probability. According to the authors, this notable discrepancy originates from incorporating a priori information of the sensors in the stochastic observation model. However, it is noted that Van den Berg et al. focus merely on Gaussian measurement uncertainties. In fact, it is not described how the stochastic observation function, that relies on the mean and the variance, may be modified in case of multimodal distributions or generally for uncertainties that cannot be described by Gaussian functions.

In a similar sense, Agha-mohammadi et al. aim at developing a motion planning framework that addresses the imperfection in measurements in [2]. Here, they suggest the use of probabilistic roadmap methods on the basis of belief-spaces. In particular, the implementation of belief spaces enables to consider for uncertainties. Specifically, the roadmap is represented as a graph, where the nodes correspond to the beliefs and the edges to the local controllers in the belief space and operates on feedback-based information. By doing so, Agha-mohammadi et al. introduce a combined method of probabilistic roadmaps and tree-based approaches. In the context of uncertainties however, the authors neglect uncertainties due to environmental fluctuations and solely focus on the uncertainties in the robot motion and sensing model. While the computation of the sensor observation model and the belief state are performed on the basis of pre-collected data sets, the exact relationships between the system parameters are not studied thoroughly. Although the representation in the belief space accounts for the existence of deviations that are likely to occur according to the pre-collected history data, the authors state that studying parameter relationships may be beneficial to obtain more accurate uncertainty models.

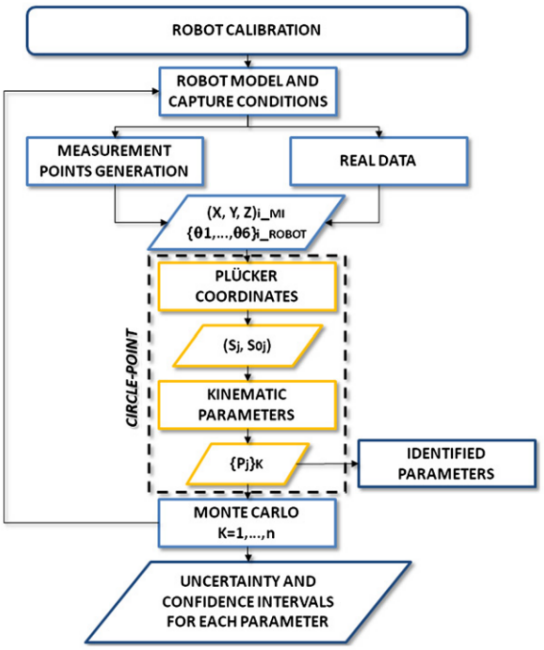


Figure 2.2: Uncertainty-aware robot arm calibration developed by Santolaria et al. in [71]: The joint axis configuration of the robot is expressed by Pluecker coordinates. On this basis, a circle point analysis is performed. For uncertainty calculation, the Santolaria et al. refer to the Monte-Carlo sampling technique suggested in the Guide [13].

Apart from the field of navigation, several works that derive uncertainty quantification techniques in application-specific manner exist in literature. For example, the contribution in [52] derive a safe exploration control algorithm that considers the uncertainties in the human position estimation. In particular, the human position is modeled as a Gaussian distribution instead of single points in the Euclidean space. Furthermore, time-varying parameters are incorporated by means of a covariance matrix. Since the suggested algorithm is an extension of the safe set algorithm that relies on the scalar position data of the human, the evaluation is carried out by comparing these two methods. In their simulation experiments, Liu et al. implement the control algorithm on an autonomous vehicle and perform human-robot interactions. Their findings show that the presented control architecture outperforms the safe set algorithm. While collisions between the human and the vehicle can be avoided in both cases, the uncertainty-aware control method yields smoother trajectories particularly for sudden environmental changes.

Uncertainties in Robot Manipulation Tasks

In the work of Nguyen et al. [60], a probabilistic framework that suggests the consideration of uncertainties in robot manipulation tasks is presented. First, the manipulation task is divided into a perception and a physical interaction part. While the perception uncertainties are assumed to occur mainly due to the calibration and camera pose uncertainties, the authors provide a detailed derivation for the physical interaction step. In fact, the perception uncertainty is calculated based on separate contributions that allow to estimate the uncertainty of the object pose with respect to the robot base by means of the covariance matrix. With the respective object distributions, a Bayesian framework is suggested for the planar grasping process, where a function is derived to encode the object motion in response to the motion of the gripper. Specifically, analytical state estimators are derived to

track the poses of objects based on gripper distances. The obtained Bayesian updates are then used to apply a particle filter to estimate the finger widths. In particular, the authors introduce a weighting step that assigns higher weighting factors to measurements with higher probabilities. Here, the measurements are assumed to underlie Gaussian noise. To perform the validation, Nguyen et al. carry out peg-in-hole experiments under real-world conditions with the presented uncertainty-aware manipulation framework. In doing so, they showcase that their method leads to higher performances compared to the traditional spiral search strategy in single and double pin insertion tasks. Furthermore, uncertainty-aware approaches have been suggested in the subdomain of robot motion planning.

For example, the work in [23] deals with developing a control algorithm that allows to avoid collisions in dynamic and uncertain environments. To address this challenge, Du Toit et al. build upon a probabilistic formulation that includes for disturbances in the object model and robot control actions. By framing the problem into the belief space, a probabilistic collision checking algorithm is derived based on so-called chance constraints that are linear constraints on Gaussian distributed states. The developed control algorithm is validated in simulation experiments, where simple and complex dynamic environments are considered. The results show that the uncertainty-aware control algorithm outperforms state-of-the-art methods particularly in uncertain locations. Although all of these contributions clearly demonstrate that the performance and efficiency of robot applications can be significantly improved by incorporating uncertainties, they rely on the Gaussian assumption. However, the Gaussian assumption for the uncertainty distribution is often not justified. Especially in real-world environments, the correlations between different state space parameters might result in more complex uncertainty models.

Learning-based Methods for Handling Uncertainties in Robot Systems

More recent approaches aim at handling consequences of uncertainties on the system behavior by means of learning algorithms. Petek et al. aim at developing a robust localization method for vehicles in dense urban scenarios in [63]. To be specific, they introduce an uncertainty-aware perception module that is responsible for the semantic segmentation and a bounding box detection. The uncertainty estimation is conducted by applying an evidential deep learning algorithm. Basically, the goal of this technique lies in estimating the priors over the network output distributions by collecting evidence. These prior distributions are used to determine the uncertainties.

Importantly, Petek et al. distinguish between aleatoric (data) and epistemic (network) uncertainties. In doing so, they clearly note that their focus is laid on epistemic uncertainties. Based on the results of the perception module, a differentiable cost map is generated to optimize the error metric of the vehicle pose. Together with a distance transform algorithm that delivers the Euclidean distance to the closest lane border, this cost map provides smooth transitions between the lane segments. For evaluation purposes, the Lyft data set is used. To evaluate the correctness of the computed uncertainty, the average values for the accuracy and the predicted confidence are calculated. According to the results, the proposed method significantly improves the localization accuracy as well as the reliability compared to approaches that are directly applied on the semantic outputs.

Another example is given by the contribution of Nava et al. in [59], where the goal lies in developing a self-supervised learning algorithm that efficiently handles uncertainties. More specifically, Nava et al. focus on problems, where the robots collect their own training data to execute application-specific tasks. The data is obtained by means of sensors

that are installed on the robot. From scientific perspective, the paper introduces modifications in the loss function of the learning algorithm and proposes to model the state estimates as Gaussian distributions. Here, the loss function is split into a task loss, that exploits the knowledge on the robot's odometry and ensures that the estimated prediction for an input matches the odometry data, and a state-consistency loss that is responsible for the consistency between the subsequent time steps.

In order to account for the uncertainty, the detected data, which equals the measurement data of this thesis, and the odometry data is considered. In fact, all poses of robot and objects are modeled as distributions in SE(3) group. Since the approach is formalized as a generally applicable fashion, the authors choose three experiments to showcase the correctness and efficacy of their method. By applying their algorithm on an object estimation task with a robot arm in a simulated environment, that is, a highly accurate detector and ground truth data of the object detection. Here, it is studied how the self-supervised learning algorithm deals with occlusions. Next, a robot heading estimation task is performed by utilizing infrared sensors. The authors corrupt the data with Gaussian noise and explore how these are handled by the suggested method. To perform a quantitative evaluation, the obtained predictions are compared with the exact odometry data for both static and dynamic environments.

In the third experiment, Nava et al. address an indoor localization problem of a ground robot, where the effect of the derived loss function is analyzed by comparing the positional and rotational components of the predicted pose with the robot's odometry for different state-consistency loss values. In all of these experiments, the authors demonstrate the ability of the self-supervised learning method to account and capture disturbances in the input data.

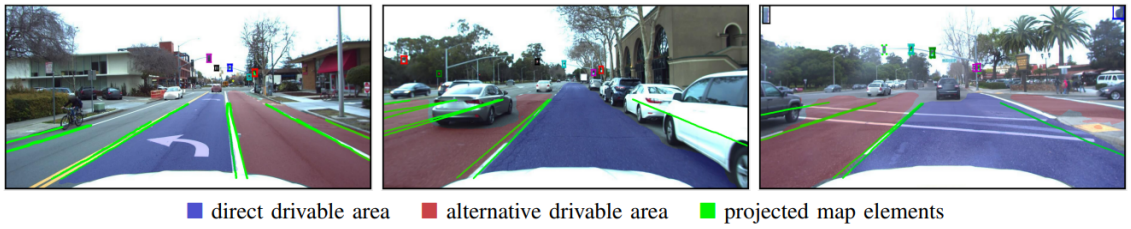


Figure 2.3: Localization method on the basis of the uncertainty-aware perception network developed by Petek et al. in [63]: The map is generated for three scenarios, to estimate the drivable and alternative areas. The width of the bounding boxes reflect the respective variances that are obtained via the uncertainty-aware component.

However, one limitation is that this approach requires ground truth data and information concerning the robot odometry. In applications where the respective data is not available, applying this algorithm might be challenging. Generally, employing learning-based methods are effective in *handling* uncertainties, that is, mitigating possible negative consequences due to their occurrence. Nevertheless, in systems where the knowledge of uncertainties is required to avoid severe consequences, applying learning algorithms might be debatable. The complexity and missing possibility to derive their uncertainty behavior may pose difficulties for calculating the uncertainty of an entire application.

Understanding of the term "Uncertainty"

Apart from this, the terms error, noise and uncertainty are not clearly distinguished in

above works. In fact, the majority of existing contributions in the field of robotics imply that noise and errors can be both covered by means of probability distributions. These probability distributions are denoted as uncertainties.

In the context of this thesis, however, the terms errors and uncertainties are clearly distinguished according to the Guide in [26]. While errors can be corrected, uncertainties can not be fully diminished. By referring to this guide, this thesis aims at developing a generic methodology to quantify the uncertainty of robot applications. In first place, this requires to clearly define the goal of the respective applications. Afterward, the uncertainties of single system components must be determined, including their relationships to each other. The overall goal is to estimate the accumulated uncertainty arising from the entire application. As can be recognized in above contributions from the robotics domain, the uncertainty calculation and incorporation pipeline is designed individually for each application. However, being provided with one generic method that is generally applicable for any type of application would enable the comparison of different setups or systems.

While such a method allowing the uncertainty quantification for a broader range of use cases has not been presented in the robotics domain yet, contributions from the field of physics demonstrate that following the suggestions in [26] and refining these according to the corresponding application yields promising results. As this thesis deals with transferring established uncertainty quantification techniques from physics to robotics, the following subsections will outline respective works to enable the identification of reasonable analogies between those two fields.

2.2.2 Uncertainty Quantification and Propagation in Physics

The main objective in natural sciences lies in discovering phenomena in nature to derive and validate existing theoretical models or even extend these. In a broader sense, research in these fields aims at accurately monitoring the environment or systems to explore and explain ongoing processes in nature. Instead of inventing and designing novel applications as in the domain of engineering, the focus is laid on providing a deeper understanding of the environment.

To this end, the perception accuracy plays a significant role for the respective experiments. Since the goal is to detect unexplored phenomena that may even contradict established theoretical models, exactly and accurately monitoring the ongoing processes is highly important. Specifically, the act of monitoring is realized by a continuous chain of *measurements*. However, perfectly monitoring the environment is impossible in real-world systems. Generally, the employed technical tools are limited regarding their precision. Therefore, one possibility to still provide an accurate interpretation of the observed phenomena, that usually exist in form of recorded data sets, is to consider for the technical limitations or in general, the probability of possible fluctuations in the entire perception and data acquisition process.

The Relevance of the Uncertainty for Measurements on Microscopic Scales
This especially becomes critical for measurements on microscopic scales. Here, validating the accuracy of observations becomes challenging. In contrast to robot applications, where the functionality of a robot task can be assessed in a straight-forward manner, the observations of particle interactions on molecular scale cannot be easily verified regarding their correctness. This particularly becomes difficult for entirely novel observations

where no corresponding theoretical model exists. One possibility to still allow for reliable interpretations on these observations is to thoroughly study the uncertainties of the employed hardware components as well as data processing tools prior to performing the measurements. In the ideal case, the knowledge on the amount and the behavior of the *measurement uncertainties* does not only provide a more thorough assessment of the obtained findings, but also the possibility to optimize the system. For example, it can be identified which system constellation yields to higher uncertainties in the output signal. Based on this, the components and system parameters can be adapted to reach a reduction of the uncertainty in the output. For example, the ATLAS collaboration aims at reconstructing particle jets in [8]. However, to derive such a reconstruction, the experimental data must be analyzed regarding possible uncertainty sources such as the proton-proton collisions. In fact, the reconstructed energy of the jets, that simply describe sprays of particles, are sensitive to proton-proton interactions. This means that neglecting their influence might result in invalid interpretations and therefore incorrect reconstruction models of the particle jets.

Uncertainty Modeling for Particle Interactions
To address this challenge, the ATLAS Collaboration derives a analytical function that enables to account for the uncertainty caused by proton-proton collisions. Particularly, the researchers rely on the techniques provided by [26] and demonstrate that incorporating their uncertainty models in the analysis yields results that match the corresponding theoretical models. Similarly, Acero et al. [1] develop a method to account for neutrino interaction models in the uncertainty determination for the NOVA detector. In analogy to above contribution, the physicists aim at detecting a natural phenomenon given by the neutrino oscillations. Due to the fact that neutrinos are the lightest massive particles according to the knowledge to date and only rarely interact with other particles, observing interactions is highly challenging. Therefore, Acero et al. perform simulation experiments to derive an uncertainty model. By performing experiments at energy ranges, they identify where the neutrino interactions become dominant for the uncertainty of detection performance. While the detection might be independent from the neutrino interactions in most cases, their influence on the detected signal becomes significant for certain energy ranges. Based on the obtained findings, this paper suggests adjustments of the experimental setup regarding the energy ranges. In doing so, the authors show that the occurrence of particle interactions that affect the detection performance can be circumvented successfully. Apart from that, the obtained uncertainty model is applied to assess the validity of the final result. A more recent work published by the KATRIN (KArlsruhe TRItium Neutrino) Collaboration in 2022 provides a comparison of different uncertainty propagation techniques.

Uncertainty and Resolution of Neutrino Mass Measurement
In analogy to above experiments, the main objective of the KATRIN experiment lies in the direct detection of neutrinos and the subsequent mass determination of these [3]. Here, an exact quantification of the measurement uncertainty is of crucial importance for an accurate evaluation of the measurement data of the neutrino mass. While the incorporation of the measurement uncertainty is a necessary prerequisite, the missing knowledge on its ground truth values causes a burden in identifying the most accurate uncertainty quantification technique. In particular, the desired resolution on the neutrino mass measurement is expressed via the light speed $c = 3{\cdot}10^8 \frac{m}{s}$ by $m_\nu \leq 0.3\, eV/c^2$. Obviously, achieving this

resolution is highly challenging. In fact, it requires a measurement accuracy that corresponds to at least this resolution. Reversely, this means that the measurement uncertainty must be accordingly small, and importantly the accuracy of the uncertainty calculation technique to be accordingly high. If the latter requirement is not met, the evaluation regarding the desired mass resolution is impossible. To this end, the researchers present four uncertainty propagation strategies in [4] given by the covariance matrix method, a fitting method based on a pull term, the Monte-Carlo (MC) propagation method and the calculation based on the Bayesian treatment of critical parameters. By applying these techniques on the same data set, the advantages and disadvantages of each method are discussed.

In summary, the authors find that the Bayesian method performs well in cases where constraints can be defined for the posterior distribution. This often applies for experiments in physics because the posterior is often naturally constrained to physically allowed values. The fitting method based on the pull term requires separate measurements for each of the parameters as the uncertainty is treated as a free parameter in the fit. Therefore, this method is hardly applicable for complex systems. The idea behind the covariance matrix is to use simulated spectra where the systematic parameters are varied within the corresponding probability density functions. However, this strategy does not account for the uncertainty propagation. In contrast, the MC propagation method distinguishes between statistical and systematic uncertainties and allows for their propagation. While the statistical uncertainty is obtained by randomizing MC spectra which are fit to a constant model, a statistically unfluctuated spectrum is simulated with varying systematic parameters (according to their PDFs) to estimate the systematic uncertainty. In the last step, the total uncertainty is computed by generating a histogram of the best fit parameters, where the best-fit value and the respective uncertainty are deduced from the resulting distribution.

3 Uncertainty Quantification and Propagation Techniques

The ultimate goal in this thesis is to examine how the incorporation of uncertainties helps to improve the optimization performance η in robotic applications. In the previous Chapter 2, the focus was laid on deriving the system description under considering the uncertainty from metrological viewpoint. In doing so, the definition of the term *uncertainty* was discussed in the context of different domains. This chapter now builds upon these elaborations and intends to introduce techniques to quantifying the uncertainty of single system parts and entire applications. Importantly, the purpose lies in presenting generally applicable methods that can be adapted to any type of robot application, where the uncertainty of one or several system properties is of interest. Therefore, the following chapter deals with the first core scientific contribution that will be followed by the development of uncertainty-aware optimization techniques in Chapter 4. While the optimization goals and the corresponding formalizations strongly depend on the system, this Chapter presents a technique to uncertainty quantification and propagation in a broader sense that serves as a generic part for the applications in Chapter 5. Prior to introducing this technique, it is demonstrated how the problem statement is split in a reasonable manner. As discussed in the previous Chapter, different methods to uncertainty calculation have been established depending on the subdomain: Kalman Filters and Particle Filters that were developed and are employed the field of robot control, the learning domain has established aleatoric and epistemic uncertainties. In order to obtain an uncertainty representation of an entire process that to incorporate it into the respective optimization pipeline, it is desirable to unify the different uncertainty representations to one metric.

In the following, the problem is split in three different parts given by the uncertainty quantification, the propagation along the system pipeline and the uncertainty-aware optimization. Especially, considering these parts as separate problems enables for system-specific adaptations. Importantly, the uncertainty propagation strongly depends on the employed tools and the complexity of the system. Apart from that, an essential prerequisite to enable the quantification of uncertainties is the appropriate system representation. As introduced in Section 2.1.8, the system description of robot applications usually follows the notation in the control domain. In order to apply the established techniques in the domain of metrology as presented in [26] however, a different form of system formalization is necessary. In particular, the remainder of this thesis deals with the quantification of the so-called *measurand* Y, that is defined by

$$Y(p_c) := f_S(X_1(p_c), ..., X_n(p_c)). \tag{3.1}$$

Here, Y can be any property, that is, either a system parameter or variable or any other measure that can be determined via the system variables X_1 via the function f_S. Particularly, it is important to first specify the critical parameter space p_c of Y (e.g., time, Euclidean space,...) to identify the relevant variables $X_1, ..., X_n$. The specification of

$Y(p_c)$ obviously depends on the complexity of the system. In case of highly complex applications, it is recommended to define several measurands Y and to apply the uncertainty propagation afterward. However, care must be taken in the consistency of the considered parameter space. In the following, it will be assumed that the system function f_S is available. Practically, this functional relationship can be obtained by approximating the system dynamics by means of theoretical knowledge or by conducting simulation studies. One possibility to obtain above system Equation (3.1) from the Equation (2.16) lies in performing simulations and monitoring the state variables $X_1, ..., X_n$ and Y over the course of the critical parameters. In the ideal case, but not often, the measurand Y equals the attribute a : Basically, this would mean that the uncertainty of Y is of interest. Often, however, it must be first estimated how the attribute a can be expressed via the measurand Y and the variables $X_1, ..., X_n$. As will be elaborated on in the following sections however, this usually is not the case such that the attribute a is defined via a separate functional relationship depending on f_S. In particular, this applies to cases where the attribute is not a system-inherent property.

3.1 Definitions and Problems

In order to facilitate the uncertainty-aware optimization for different types of applications, the problem is split in several parts. This enables to consider the application-specific characteristics in each of the three steps that is beneficial as will be demonstrated in Chapter 5.

3.1.1 The Attribute and Critical Parameters

So far, the uncertainty as well as the concepts concerning it were discussed in the context of applications or system properties. However, to allow methods for the quantification, it must be specified how these properties are formalized. To do so, the *attribute a* (p_{crit}) is introduced. Specifically, the attribute states the property where the uncertainty is of interest for while p_{crit} denotes the *critical parameter*. Often, yet not in all cases, the attribute a corresponds or can be expressed via the measurand as described above. The critical parameter specifies for which space the uncertainty of the corresponding attribute is of interest, i.e.,:

$$a(p_c) := f_a(Y_i(p_c), ..., Y_k(p_c). \tag{3.2}$$

Logically, it is possible to consider different parameter spaces: In case the attribute is defined in different spaces, it must be considered how these depend on each others. In particular, the knowledge on the dependency of the parameter spaces is crucial for computing the propagated uncertainty entire application as will be explained in Section 3.2.2. In order to facilitate an easily applicable uncertainty quantification and propagation mechanism, both the attribute and the critical parameters must be brought into the context of the metrological system in Equation (3.1). Especially, this is required for the uncertainty propagation, where the system equation serves as the basis. At the same time, it is desired to keep the quantification and formalisms as simple as possible. In the following, the uncertainty quantification and propagation mechanisms will be explained for both single components and the entire system. By starting with the uncertainty quantification on component level, the basis will be laid for more complex calculations in the second part of this Chapter.

3.1.2 Uncertainty Quantification in GUM

As elaborated in Section 2.1.1, the GUM suggests to distinguish Type A and Type B uncertainties. Apart from that, quantifying the uncertainty requires to formalize the system dynamics and to define the so-called *attribute*. In particular, the uncertainty regarding the attribute critically depends on the applied perception technique, that is, by the employed measurement devices and the environmental conditions.

Here, the attribute a can be any parameter, variable or characteristic of the system, where the uncertainty shall be determined for. For example, in detection tasks, a reasonable candidate for the attribute may be the position or orientation of the object of interest. The first step is to specify the attribute and to identify its functional relationship with the devices in the parameter space, where the measurement is performed in (e.g., measurement over time or space). According to this system equation, the respective uncertainty on $u_a(p_c)$ can be determined where p_c denotes the critical parameter. Logically, being provided with representations in different spaces offers more information, and thus a more precise uncertainty estimate. Especially, the uncertainty estimate is more precise since relationships between different parameters are likely to exist and can only be captured by considering these representations. This will be addressed experimentally in Chapter 5.

However, estimating the amount and behavior of the uncertainty arising from one entire system necessitates the identification of the uncertainty on component-level in first place. Ideally, these uncertainties are provided by manufacturer specifications (e.g., data sheets of sensory devices) or theoretical models. In case they are not available, the Guide in [26] suggests to determine them in statistical manner. Accordingly, measurements of the attribute a are conducted under non-changing environmental conditions. Afterward, the approximation of the expected value μ_a is calculated via

$$\mu_a = \frac{1}{n} \sum_{k=1}^{n} a_k. \tag{3.3}$$

The corresponding standard deviation σ^2 that captures the variability of the observed values a_k allows to calculate the standard deviation around the mean $s^2(\mu_a)$:

$$s^2(q_k) = \frac{1}{n-1} \sum_{j=1}^{n} (q_j - \mu_a)^2; \quad s^2(\mu_a) = \frac{s^2(a_k)}{n}. \tag{3.4}$$

Thus, the GUM proposes to compute the uncertainty $u_a(p_c)$ by means of n independent repeated measurements. While this might be possible for single hardware devices, collecting reasonable amounts of data in engineering applications often is more challenging. Particularly, system components in complex applications are compositions of single hardware devices. Apart from that, the obtained uncertainty may not be valid for all system constellations. Therefore, a technique that can evaluate the uncertainty of entire system parts of single system components is necessary. As will be elaborated on in 3, the uncertainty arising from system parts that consists of subcomponents can be treated as black-box tools, where the uncertainty of a black-box tool is denoted with $\tilde{u}_a(p_c)$ and ideally

$$\tilde{u}_a(p_c) \propto s^2(\mu_a). \tag{3.5}$$

Therefore, the first problem lies in deriving a method to quantifying $\tilde{u}_a(p_c)$ that approximates the uncertainty in Equation (3.4).

3.1.3 Uncertainty Propagation in GUM

In engineering applications, the uncertainty of entire processes or systems is usually of interest. This means that the uncertainties on component level must be accumulated according to the relationship stated in the system function equation in Equation (3.1). While the uncertainty for each component and space was determined separately in the previous step, the uncertainty propagation deals with bringing the results together. In doing so, the relationship between the uncertainties on component level must be analysed regarding the critical parameter space and possible dependencies between the components. The GUM [13] provides a general technique for calculating the propagated measurement uncertainty (PMU) $u_{a,prop}(p_c)$ by

$$u_{a,prop}(p_c) = \sqrt{\sum_{i=1}^{m} \left(\frac{\partial f_S}{\partial i} u_{a,i}(p_c) \right)^2 + \sum_{i=1}^{n-1} \sum_{q=i+1}^{n} \frac{\partial f_S}{\partial x_i} \frac{\partial f_S}{\partial x_q} u(x_i, x_q)} \tag{3.6}$$

for m system components that are characterized by i. Here, $u(x_i, x_q) = u(x_q, x_i)$ stands for the covariance between x_i and x_q. However, above equation is only valid for linear systems. In case the system function f_S is not provided or cannot be modeled analytically, GUM refers to the general MC sampling approach to infer $u_{a,prop}(p_c)$. Apart from that, the sampling approach must be refined individually for each application with respect to the number of samples and the region for the sampling process. In particular, MC algorithms yield homogeneously distributed samples within the considered space. Depending on the region of interest however, it may be desired to enforce the data generation in specified areas. For example, the robot may perform movements in certain areas more frequently such that evaluating the uncertainty and thus the generation of additional samples in this region is desired. In addition, a specific problem in robot applications is the combination of different uncertainty representations. As introduced in the previous Chapter, different uncertainty representations have been introduced in the subdomains: the notation for the metrological uncertainty of technical devices differs fundamentally from the epistemic and aleatoric uncertainties in the learning domain. To this end, the determination of $u_{a,prop}(p_c)$ involves the refinement of the propagation techniques, the identification of appropriate sampling techniques that provide sufficient samples in the regions of interest for nonlinear systems as well as the derivation of a mapping function $\Omega : \mathbb{R}^n \mapsto \mathbb{R}$ with $n \in \mathbb{N}$ that allows to estimate $u_{a,prop}(p_c)$:

$$\Omega : [u_{\omega_1}(p_c), u_{\omega_2}(p_c), ..., u_{\omega_m}(p_c)] \mapsto u_{a,prop}(p_c). \tag{3.7}$$

Especially, identifying Ω highly depends on the respective robot system and might require calibrations in case no prior information is available to determine the transfer function between the uncertainty spaces. Here, the term *calibration* refers to the process of exploring how the variables of a system influence each other by observing a pre-specified output and monitoring the variables of interest by considering the sensitivities. To be specific, it relies on the formal definition used in the metrological domain, provided by the International Bureau of Weights and Measures (BIPM)Apart from that, the refinement of traditional MC sampling methods may be desired. While classical MC approaches represent a powerful technique to generating homogeneously distributed samples, regions with sparse data occurrence may be especially interesting for the uncertainty quantification. One possibility to address this problem is given by the importance sampling technique that will be described in detail in Chapter 3.

3.1.4 Uncertainty-aware Optimization

Being provided with the uncertainty of the attribute arising from the entire system, the goal lies in exploring how this uncertainty contributes to the optimization performance η introduced in Equation (2.6), i.e.,

$$\eta(\vec{x}, u_{\vec{x}}) \geq \eta(\vec{x}), \tag{3.8}$$

where $\eta(\vec{x}, u_{\vec{x}})$ stands for the optimization performance achieved after accounting for the system uncertainty and $\eta(\vec{x})$ the optimization performance corresponding to the case, where the uncertainty is neglected. Depending on the complexity of the application, this step involves additional specifications regarding the optimization technique: While the consideration of the system uncertainty $u_{\vec{x}}$ may be sufficient, studying the influence of the system components' uncertainties on $u_{\vec{x}}$ or environmental factors may enable to identify further steps to reduce the system uncertainty. These uncertainty analyses can be conducted by calculating the sensitivity presented in Section 2.1.5. Importantly, accounting for the sensitivities can act as the basis for the successful selection of appropriate system components that are acceptable regarding their uncertainty and their functionality. Apart from that, findings from these analyses may help to adapt the environmental conditions to attain a higher optimization performance. In order to compute η, the optimization goal must be formulated under consideration of the system dynamics: Usually, the optimization goal is formalized via properties that are not unambiguously assigned to the parameters and variables that occur in the equations for the system dynamics. Therefore, the parameters and variables that are relevant for the optimization must be specified. Importantly, it is required to identify the critical parameter spaces. Based on this, the optimization function can be reformulated in an uncertainty-aware fashion. It is therefore aimed at deriving a function that determines the optimization efficiency on the basis of an uncertainty-aware formalization of the system, i.e.

$$\eta(\vec{x}) = \psi(\vec{x}(\gamma_i), u_{prop,a}(\gamma_i), ...), \tag{3.9}$$

where the vector $\vec{x}$ stands for the set of relevant parameters and variables and γ_i for the critical parameter spaces.

From practical viewpoint, this part may be the most important, yet highly challenging one in real-world robotic systems. While incorporating the results of the previous steps may sound straightforward at first glance, the challenge lies in identifying an appropriate uncertainty representation that does not only reflect the actual amount, but also can be incorporated efficiently in the optimization pipeline without much effort, where the efficiency becomes particularly relevant for online optimization. Here, the goal would be to develop a technique that does not entail a burdensome computation time. Obviously, the threshold for the acceptable computational load must be specified in an application-specific manner, as will be addressed in Chapter 5. Furthermore, evaluating Equation (3.8) must be accompanied by a confidence value that reflects how reliable the result is.

3.2 Methods

In this Section, techniques addressing above problems are derived in a generic fashion. As mentioned in the beginning of this Chapter, the development of the uncertainty-aware optimization techniques is addressed separately in Chapter 4. This Section presents following methods:

1. **Uncertainty Quantification of black box components**: First, a technique to calculating the uncertainty of black box components is developed. Here, a *black box component* stands for a tool where the uncertainty is not known.

2. **Uncertainty Propagation**: Second, it is elaborated how the uncertainty of an entire system can be obtained by referring to the uncertainties of single components. Especially, this contribution is further divided in two parts. After presenting how the accumulation of uncertainties is performed in the metrological space, it is introduced how the metrological uncertainty notation can be combined with the robustness of NNs.

In the context of robot applications, the uncertainties of technical components like sensors or cameras are mostly stated by the manufacturers' specifications. However, this does not necessarily apply to software tools. For example, the uncertainty of NN-based software components is generally challenging to estimate. While, as mentioned above, black box tools stand for any component with unknown uncertainty, it can also describe entire system parts where the functionality is not fully tractable or not understood regarding the behavior of the subcomponents. Logically, a missing understanding of a systems' inner components impedes the quantification of the uncertainty since estimating the amount of possible deviations requires the knowledge on the sources for undesired fluctuations.

In order to derive a technique to estimating the uncertainty of black box components, the knowledge on the invariants on system characteristics is exploited. These invariants are referred to as *conservation equations*. Briefly, it is suggested to identify measures of the system that do not change with respect to the critical parameter space. For example, static objects can be described with a conservation equation with regard to the time and the non-changing position of the object. The uncertainty in the respective position measurement is estimated by evaluating these conservation equations. While this method is developed in generic fashion in this Chapter, the validation experiments are performed in the context of safety evaluation as will be explained in Section 5.3.3.

Next, the uncertainty propagation is addressed with the goal to calculate the total uncertainty of the system by accumulating the uncertainties of the system components. Importantly, the presented technique is applicable to any type of uncertainties. In contrast to existing methods in literature, where the uncertainty of the system components follow Gaussian distributions, the introduced uncertainty propagation method enables to deal with uncertainties of any type. In accordance to the method to quantifying the uncertainty of black box components, the propagation technique is also validated for robot systems in Section 5.3.3.

In addition, techniques that enable to unify different uncertainty representations is developed. As mentioned in Chapter 2, the robustness of neural networks shows analogies to the metrological uncertainty formulation. To this end, an approach to combining these two notations to one uncertainty measure will be addressed by pointing out to the similarities. In fact, analogies between the Lipschitz function and the metrological sensitivity definition are drawn. In contrast to the uncertainty quantification and propagation method, this technique to unifying the uncertainty representations is studied in the context of an image classification experiment in Section 5.2.2.

Finally, it is motivated how these techniques can be employed for the purpose of optimizing the performance of robot systems that is addressed in the next Chapter.

3.2.1 Uncertainty Quantification of Black Box Components

As explained above, the GUM presents techniques to calculating the uncertainty on the basis of measurement data. Specifically, this guide focuses on the uncertainty determination in physical systems, where the uncertainty of the components are assumed to be known: The authors state that statistically estimating the uncertainties of single components can be carried out in straightforward manner. To be specific, uncertainties can be distinguished in *statistical uncertainties* and *systematic uncertainties*. While the latter ones describe those that arise from the technical component itself, statistical uncertainties reflect the missing knowledge due to the limited data. Obviously, being provided with an infinite amount of data that represents all possible situations to all time steps is unrealistic and impossible. The statistical uncertainty u_{stat} aims at capturing the lack of information due to this limitation by assuming that all data points are equally relevant for the data collection, i.e.,

$$u_{stat} = \frac{1}{\sqrt{N}}, \tag{3.10}$$

where N denotes the number of data considered in the measurement.

However, in complex robotic systems, the uncertainty of single components on small scales may not be of interest. Instead, focusing on the uncertainty of entire system parts or tools is usually more reasonable. In addition, the uncertainty of these black box tools may vary over time or other system parameters. The key motivation behind the following technique is that exploiting certain characteristics of an application and quantifying the uncertainty of entire system parts may be more efficient than attempting to investigate the uncertainty of single system components and their relationship with each other on small scales. In order to estimate $\tilde{u}_a(p_c)$, it is suggested to identify system properties that are constant in a with respect to the critical parameter p_c and can be measured by the tool of interest. Ideally, these system invariants are known due to physical or technical constraints and can be formalized as equations, i.e.,

$$\nabla\vec{x}(\gamma_i) = \text{const.} \tag{3.11}$$

For example, if the objects are known to be static and not move during operation, the respective equation may be stated by $\nabla\vec{r}(t) = 0$ for static objects, where $\vec{r}(t)$ provides the object position at time t. Such equations that reflect the invariant character are denoted as *conservation equations*. In the next step, the data of the system invariant properties are monitored over the course of γ_i with the black box tool. These measurements must be conducted under the same environmental conditions. By analyzing the resulting data collections, violations on the conservation equation(s) in (3.11) are determined. Importantly, since these violations are known to actually not occur due to technical limits or physics laws, they can be directly attributed to the measurement uncertainty of the employed measurement tool. As a logical consequence, the uncertainty calculation can be performed by determining the violations on the system-specific conservation equations. Specifically, the violations on the conservation equation(s) are evaluated by means of the bootstrapping technique. The steps can be summarized as follows:

1. Formulate the conservation equations according to Equation (3.11). Ideally, the invariant is defined with respect to the critical parameter space p_c of the attribute a:

   $$\nabla\vec{x}_a(p_c) = z_i \tag{3.12}$$

   In addition, the value of z_i must be specified.

2. Record the data for evaluating the conservation equation. In addition, an appropriate time period for the data monitoring must be defined. Specifically, if the tolerated statistical uncertainty $u_{stat,tol}$ is known, the minimum data recording time can be calculated:

$$u_{stat} \leq u_{stat,tol} \rightarrow N_{crit} \tag{3.13}$$

   By referring to the data monitoring frequency f_d that is usually provided with the frame rate of the employed data acquisition tool, the required data monitoring time is determined. In all cases, the environmental conditions within one data set must be constant to maximize the accuracy in the uncertainty quantification.

3. The conservation equation are evaluated on the basis of the collected data sets. Importantly, possible dependencies must be considered at this stage. For example, if the critical parameter is captured over the time, the accuracy in the data recording frequency must be maintained. It is crucial to distinguish violations due to monitoring inaccuracies from the uncertainty originating from the measurement device. To be specific, it is expected that the conservation equation, and thus the amount of z_i is determined constantly during the data acquisition and stored in a vector $\vec{z}_i$. Logically, the number of elements in this vector amounts to the collected data N that ideally equals the number of evaluations of above equation. Basically, this evaluation step involves generating a bootstrap distribution for each of the sampled sets. Next, the mean values of all bootstrapped distributions are calculated. As described in Chapter 2, these mean values are distributed in a Gaussian fashion, where the mean value corresponds to the uncertainty of the measurement tool:

$$\mu_{viol} \pm \sigma_{viol} \mapsto \tilde{u}_a(\gamma_i), \tag{3.14}$$

   where the confidence level α is computed via the bootstrapping technique. a

4. The uncertainty of the measurement tool of interest is determined with respect to the parameter relationships by referring to the Gaussian distribution. Particular, hypothesis tests can be conducted. In online applications, the calculation can be carried out during operation such that the user is notified with the actual uncertainty on an attribute of interest. Here, a reduced statistical uncertainty u_{stat} and a higher confidence is expected to be achieved since the amount of considered data increases with time.

As explained in Chapter 2, this method enables to calculate the mean of the occurred violation and the corresponding confidence interval. Depending on the application, this procedure can be performed online. The calculation can be repeated several times during operation after specifying the required data amount for one evaluation run. Furthermore, observing how the uncertainty on the component level behaves over time is necessary to identify dependencies with additional system state variables. Deriving the corresponding relationships by means of data fitting techniques might therefore facilitate even the prediction of uncertainties, especially in systems with repetitive tasks. Especially, this would mean that the existing relationships are accurately modeled which might enables the development of uncertainty minimization techniques.

Apart from that, conservation equations can be formulated for different parameter spaces. For example, temporal as well as spatial constraints may exist. An object may remain its position during the entire process. Also, it may be known that its size does not vary over time. These characteristics can be expressed by two conservation equations: one

for the temporal and one for the spatial parameter space. Logically, the accuracy of the uncertainty quantification is enhanced by accounting for more parameter spaces. This particularly applies to cases where conservation equations can be formulated for correlated parameter spaces. Indeed, considering for several conservation equations in parallel enhances the accuracy of the uncertainty calculation, and thus the obtained uncertainty. However, to verify the amount of correlation or further relationships, additional statistical tests can be performed. For instance, these tests may enable to identify correlations as

$$\mathrm{corr}(\tilde{u}_a(\gamma_i), \tilde{u}_a(\gamma_k)) \quad \text{for } i \neq k \tag{3.15}$$

In addition, tendencies or dependencies between the state variables, environmental conditions and the uncertainty behavior can be further explored by statistical hypothesis testing. Based on the findings, the state variables or application can be adapted to enable an appropriate system constellation or to achieve the optimization goal.

3.2.2 Uncertainty Propagation and Analogy to Physics

As described above, the basic approach to accumulating uncertainties is provided by the Guide in [26]. However, the presented methods rely on the assumption that the uncertainty on component level and the system equation are provided. Unfortunately, this usually does not apply to robot applications. In addition to the unknown relationship among the components, the uncertainties are usually not stated based on the metrological notation. While the uncertainties of hardware devices as sensors or cameras are specified by following the formalism used in metrology, the uncertainties of algorithms usually do not stand in accordance with the metrological notation. Instead, depending on the software tool, the understanding of the uncertainty may be domain-specific. This was described briefly in Chapter 2. For example, the uncertainties of neural networks are defined in a different manner. Especially, the measure that shows the highest similarity to the metrological uncertainty is given by the robustness.

Since one of the central aims in this thesis is to enable the determination of the system's uncertainty that unifies different representations, one main challenge that must be addressed lies in bringing together the different uncertainty formalisms. Importantly, finding an approach to unify these formalisms involves identifying existing analogies and considering the differences in the definition thoroughly. This analysis must be performed prior to accumulating the uncertainties in the respective spaces. Afterward, it is necessary to explore how the attributes depend on each other to allow for the calculation of the propagated uncertainty in the desired parameter space.

In the first part of the uncertainty propagation, the simplified case of propagating uncertainties in one representation is introduced. This part is mainly concerned with the identification of relationships between the components (sources of single uncertainty contributions) regarding the critical attributes and the parameter spaces. Next, an approach to combine the uncertainties of NN-based tools with the metrological uncertainty notation is developed. Here, it is explained how the definition of the robustness of NNs can be interpreted from the metrological perspective. Finally, it is elaborated how existing analogies can be exploited to facilitate the unification of the uncertainty of hardware components with the robustness stated for NN-based components to one propagated uncertainty measure.

Uncertainty Propagation in one Representation

For example, in case it is of interest to compute the uncertainty on the movement of the robot, it is crucial to be provided with the spatial uncertainty as well as the temporal uncertainty to correctly assess the uncertainty regarding the robot velocity. Depending on whether the movement depends on environmental fluctuations or other restrictions, it must additionally be considered to which extent the restrictions can be hold and whether the conditions affect the temporal and spatial uncertainties. Here, the attribute defines the relevant spaces for which the uncertainties must be determined.

In particular, the relationships between the parameter spaces and the environmental state variables may enable to even predict the uncertainty. Assuming that sufficient data is available for performing analyses, functional relationships between the parameters and variables can be derived by means of fitting techniques. In fact, these relationships are incorporated in the uncertainty propagation pipeline. Since the accuracy of the uncertainty calculation is directly tied to the accuracy of the system model, accounting for all dependencies between the critical parameter p_c and the remaining variables is essential. The propagated uncertainty $u_{prop}(p_c)$ is essential for the uncertainty-aware optimization as stated by Equation (3.9). To be specific, $u_{prop,a}$ that stands for the propagated uncertainty with respect to the attribute a can be expressed by the relationship $G : \mathbb{R}^n \mapsto \mathbb{R}$

$$u_{prop,a}(p_c) := G(f_a(p_c), f_S(p_c)), \tag{3.16}$$

where f_a and f_S in turn depend on the measurands $Y_i, ..., Y_k$. Importantly, $u_{prop,a}(p_c)$ is highly dependent on the uncertainty of the measurands $Y_i, ..., Y_k$, that not necessarily correspond to scalar values. In this case, the Equation (3.6) cannot be applied in straightforward manner, but must be extended by MC sampling.

The underlying idea is to consider the distributional character of the uncertainties $u_{Y_i}, ..., u_{Y_k}$ in the propagation as explained in the Supplement of the GUM [13]. In fact, the uncertainty distributions are propagated according to the system equation $f_S(p_c)$by generating MC samples with following steps (see [13] for details):

1. Set number M of MC trials.
2. Create M vectors from the n probability density functions assigned to measurands Y_i.
3. Evaluate value for attribute a according to the model function $f_S(p_c)$ for each of the M vectors.
4. Sort the M values of a into increasing order. This enables the derivation of the distribution G (see [13], Chapter 5.3 for details).
5. As state in above Equation (3.16), G serves as an approximation for $u_{prop}(a)$.

In addition, if several critical parameter spaces must be considered, it is recommended to account for the correlation

$$\mathrm{corr}(u_a(p_{c,i}), u_a(p_{c,j})) \tag{3.17}$$

with $i \neq j$.

Next, the obtained propagated uncertainty distribution is assigned to a confidence measure that reflects the reliability of the result. In the next step, the propagated uncertainty in different spaces are combined to compute the uncertainty on the target attribute. To do so, fitting techniques can be applied. Especially, data fitting becomes relevant in cases where

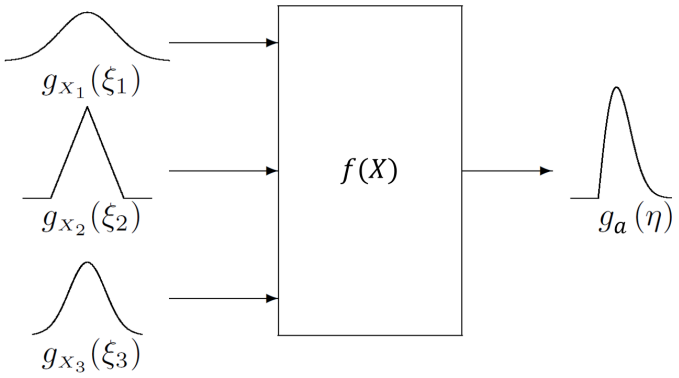


Figure 3.1: Illustration of the uncertainty propagation for nonlinear input uncertainties taken and adapted from [26]. In case the equation for linear and scalar uncertainties is not applicable, the propagated uncertainty $u_{prop,a}(p_c)$ is obtained via MC sampling.

the uncertainty estimates suffer from low confidence values due to the limited amount of data samples in the previous step. In the context of fitting techniques, the goodness of fit is one critical metric indicating the trustworthiness of the final distribution. Therefore, the first step serves to explore parameter relationships and correlations to compute the propagated uncertainty in different parameter spaces. Importantly, it is required that the metrological representation of the uncertainties are available. However, this is not always the case as will be discussed in the following.

Uncertainty Propagation with NN based tools

So far, calculating the uncertainty $u_a(p_{c1}, p_{c_2})$ for an attribute a with the critical parameters $p_{c,i}$ was addressed for the metrological representation. In fact, above described approaches assume that the uncertainties on component level are provided in the metrological representation, that is, by means of probability distributions indicating the possible deviations. Unfortunately, a robot application may consist of tools, where the uncertainty follows a different representation. For example, neural network classifiers or methods based on deep learning are often employed in robot systems.

As indicated in Chapter 2, the aleatoric and epistemic uncertainties are commonly used within the learning domain. The missing absolute notations and the context-dependency however impede the derivation of a technique for accumulating the uncertainties correctly. Especially, in real-world robot applications, the aleatoric uncertainty crucially depends on the metrological uncertainties of the measurement devices that are used to capture the input data for the learning algorithms. This dependency however has not been addressed in existing literature. To be specific, the robustness of NN classifiers is defined by the Lipschitz equation

$$\mathcal{L}_\epsilon(\delta) := Pr(\|f_{NN}L(x_0) - f_{NN}(x')\| \leq k \cdot \|x_0 - x'\|) \geq [1 - \epsilon] \tag{3.18}$$

for an l_p-ball around an input data point x_0 and the output $f_{NN}(x_0)$. This equation will be explained in more detail in Section 5.2.5. In fact, this notation of the robustness serves as the basis to incorporate the possible deviations of learning algorithms into the uncertainty propagation pipeline: In accordance to the metrological uncertainty representation, the robustness captures the deviation of the output data by with regard to the corresponding

input data fluctuation. Here, the input data is constrained by an l_p-ball that can be specified by the user. The robustness therefore measures the sensitivity of the output $f_{NN}(x)$ with regard to the input.

As will be described in Section 5.2.4, the works in the learning domain focus on the development of methods to preventing adversarial attacks. In doing so, several contributions as ([80], [55]) have demonstrated that the output of NN classifier that can formally be described by a nonlinear, non-convex and usually very complex function $f_{NN}(\cdot)$ with general activation functions can be bounded by two linear functions $l_t^L(x) : \mathbb{R}^{n_0} \mapsto \mathbb{R}$ and $l_t^U(x) : \mathbb{R}^{n_0} \mapsto \mathbb{R}$ such that

$$l_l^L(x) \leq l_t(x) \leq l_t^U(x). \tag{3.19}$$

These linear functions are formalized in terms of the input x as follows:

$$l_t^L(x) = A_{t,:}^L \vec{x} + d^L \; ; \quad l_t^U(x) = A_{t,:}^U \vec{x} + d^U, \tag{3.20}$$

where $A_{t,:}^L, A_{t,:}^U \in \mathbb{R}^{1xn_0}$ represent constant row vectors and d_L, d_U for the weights and biases, respectively. Therefore, for NNs that can be linearized, the robustness definition is valid in the sense that possible deviations as well as the sensitivity of the NN output with respect to the input data is accurately captured. Especially, a probabilistic representation of the robustness can be introduced to obtain probability density functions that carry mutual information as the metrological uncertainties presented above as illustrated in Figure 3.2.

Although it requires to gather a considerable amount of data to determine a probabilistic representation with reasonable confidence, this would allow to relax the robustness requirement in certain cases. This simplifies the practicality for real-world robot applications since the robustness is not supposed to be satisfied in all cases. Apart from that, the uncertainties inherent to the input data space depend on the measurement tools as sensory devices or cameras that are denoted as the *perception uncertainty*.

In both representations (the probabilistic robustness and metrological uncertainty), the possible deviations in the output are measured regarding the desired ground truth result. In addition, the sensitivity with respect to the input deviation is considered. While the aleatoric and epistemic uncertainty notations do not include reference measures, the metrological uncertainty formalism and the robustness definition are directly related to the corresponding input data. Formally, combining the robustness of NNs with the metrological notations of uncertainties refers to the identification of a transfer function introduced in Equation (3.9). In the given case, the transfer function is specified by

$$\psi_{NN}(\mathcal{L}_\epsilon(\delta), u_{prop,a}(p_c), ...). \tag{3.21}$$

However, the derivation of ψ_{NN} is not straightforward. Despite the fact that an analogy can be identified between $\mathcal{L}_\epsilon(\delta)$ and $u_{prop,a}(p_c)$, determining the exact relationship between the parameters in the Lipschitz equation, the critical parameter p_c and the attribute a in the metrological formulation is challenging. Importantly, these relationships between the parameters and variables do not necessarily require the derivation of an analytical function. Instead, it is sufficient to establish a calibration procedure that inherently identifies the relationships and accounts for these.

In practical robot systems, this means that a reasonable amount of calibration data must be recorded prior to system commissioning. This step can be performed in a simulation environment if it is desired to consider for different system constellations due to the difficulty of accurately monitoring all parameters under real world conditions. Afterward,

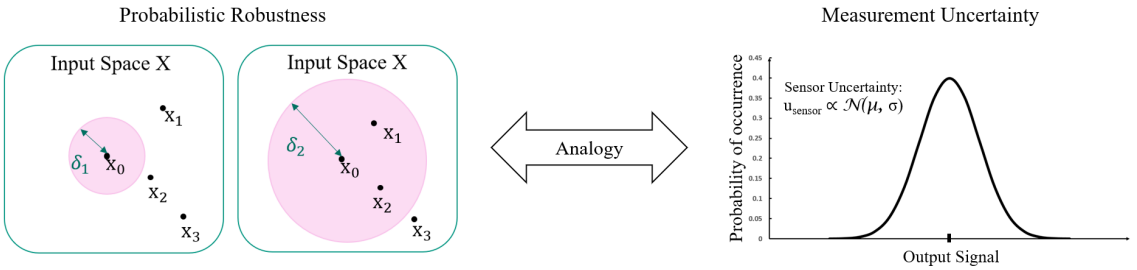


Figure 3.2: Analogy between the probabilistic robustness and the measurement uncertainties: In both cases, the purpose is to capture how sensitive the output of a measurement tool or neural network reacts on input data. **Left**: The input space is represented by an l_p-ball in the Lipschitz equation, that defines the robustness. The Lipschitz-ball is symmetric. **Right**: The measurement uncertainty is represented by a probability distribution. For technical tools, these are provided by the manufacturer specifications and are modeled as Gaussians.

the relevant regions of the assigned parameters and variables in the robustness equation must be identified. This can be achieved by referring to a measurement tool, where the metrological uncertainty is clearly known. Recording the calibration data with it enables to explore the critical areas within the parameter space in the robustness equation. In order to accomplish the propagation, possible relationships between the uncertainties of the remaining technical tools and the NN input data must be considered in the calculation. For example in vision tasks, the input deviations are directly affected by the resolution of the employed cameras. Furthermore, the resolution of the cameras may depend on environmental fluctuations that must be considered. In Chapter 5, the validity of the presented technique to unifying the metrological uncertainty with the NN robustness will be demonstrated on an image classification and 3D human joint detection.

3.3 Summary of Chapter

The focus of this Chapter was laid on providing uncertainty quantification and propagation techniques. To this end, it was first described how a robot system can be described by the metrological notation. By doing so, the necessary terms and definitions for the development of uncertainty-aware optimization methods were introduced. Especially, the optimization performance η was defined that serves as the basis for evaluating the performance of the robot applications.

Afterward, a method to quantifying the uncertainty of black box components was derived. As will be further elaborated on in Section 5.3.3, this technique enables to calculate the uncertainty of components or system parts, where no information is provided by the manufacturer. One challenge here however lies in the identification and formulation of the conservation equations. Often, these cannot be identified in straightforward manner. The limitations of this method will be discussed in detail in Section 6.1.1.

Third, it was shown how the uncertainty of entire applications can be calculated by introducing the uncertainty propagation method. After presenting the Monte-Carlo-based propagation algorithm that enables to accumulate the uncertainties of the system components in the metrological notation, the second part was concerned with combining differ-

ent notations. To be specific, it was suggested to leverage the analogy in the definition of the robustness of NN-based classifiers with the metrological uncertainty definition. Here, it was proposed how the calibration procedure of the parameters can be performed. However, it was not clearly specified which relationships must be exploited to enable a successful unification. As will be explained in the following Chapter, the exact interpretation of the parameters depends highly on the considered application.

The uncertainty quantification and propagation methods presented in this Chapter provide the basis for the uncertainty-aware optimization techniques. In the context of robot systems, the uncertainties ideally reflect the amount of possible deviations on critical system variables in the parameter space of interest. The knowledge of the possible amount of undesired fluctuations may allow the introduction of countermeasures as will be discussed in the context of safety evaluation in Section 5.3.3.

4 Uncertainty-aware Optimization Techniques

The central goal of this thesis is to explore how the consideration of the uncertainties of a system contribute to higher optimization performances in robot systems. As introduced in Section 3.1.4, the optimization performance generally measures to which extent the the desired application-specific goal has been achieved. Practically, in the context of robot systems, the robot is often confronted with a set of possible actions and would ideally select those with the most promising outcome. This Chapter intends to present methods that allow to consider for the uncertainties of single components as well as the accumulated uncertainty in the system pipeline. As will be elaborated in detail, the incorporation and uncertainty-optimization is designed individually for each optimization type. While optimization problems and approaches to solve these are studied extensively in the domain of mathematics, this thesis aims at framing common goals in robot applications as optimization problems and to derive uncertainty-aware optimization techniques for these.

In this Chapter, the thesis studies whether conducting the optimization in an uncertainty-aware fashion yields a higher performance. Specifically, it is distinguished between discrete and continuous optimization problems. While improving the decision-making process of the robot in cases, where the goal is clearly known, is crucial, the ideal constellation of the robot is often not known in more complex applications where additional system parameters play an important role.

Especially, *discrete* optimization problems refer to those where the optimization is performed with respect to one or several classes. Image classification tasks fall into this category since the goal lies in assigning the correct class to the respective data. In contrast, the optimization in *continuous* optimization problems is performed on the basis of a continuous function that ideally accounts for the relationships between the relevant system parameters. In fact, many optimization problems in complex real-world applications, where the goal lies in optimizing an overall measure as risk minimization, cost reduction or efficiency maximization can be seen as continuous problems. Specifically, solving these problems require parameter adaptations on a larger scale as will be further elaborated on in Section 4.3. In the following, these two types of optimization problems will be further specified by defining the assumptions that are made in this thesis. Afterward, approaches to solving these will be derived in three steps: While the first two parts address settings that fall into discrete optimization problems, that can be distinguished in fully explainable systems and those employing NN-based tools, the third section focuses on the derivation of uncertainty-aware continuous optimization methods for more complex applications.

4.1 Uncertainty-aware Binary Optimization

First, a method to binary optimization is developed. Practically, this method addresses problems where the robot is confronted with two distinct choices, that can be expressed via binary variables. To obtain the optimization performance η, it is estimated to which degree the optimization goal is achieved, and especially whether the incorporation of uncertainties yields a higher η value. Here, the optimization is defined via a vector $\vec{x}_B \in \mathbb{B}^n$, with $\mathbb{B} = \{0, 1\}$. Obviously, the size of the vector $\vec{x}$ must be specified individually according to the use case. The goal is to derive a cost function $z_Q : \mathbb{B}^n \mapsto \mathbb{R}$ given by

$$z_Q(\vec{x}) = \vec{x}^T Q \vec{x} = \sum_{i=1}^{n} \sum_{j=1}^{n} Q_{i,j} x_i x_j, \tag{4.1}$$

where Q stands for a triangular matrix $Q \in \mathbb{R}^{n \times n}$ that assigns weights for the entries x_i and x_j of the binary vector. Then, the optimization goal is to find a vector $\vec{x}^*$ with

$$\vec{x}^* = \arg\min z_Q(\vec{x}). \tag{4.2}$$

While the domain of binary optimization problems have been studied extensively in theoretical manner in the field of mathematics, framing optimization problems in robotics into this formalization comes with following challenges.

Most problems need to be framed accordingly in first place which often requires a mapping from the input data space to the binary space $\mathbb{B}^n$ prior to deriving the cost function $z_Q(\vec{x})$. In particular, the aim of this thesis is to consider the uncertainties in both parts the specification of the problem and the definition of the cost function. Here, the objective function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is assumed to be convex, but not necessarily smooth on a convex set Λ and $\{-1, 1\}^n \cap \Lambda \neq \emptyset$. Particularly, non-convexities are only assumed to occur due to binary constraints on the optimization problem. To derive a technique that addresses this problem in an uncertainty-aware fashion, following additional assumptions are made:

- Assumption 1: A baseline data collection that has been generated under the same environmental conditions exists. Essentially, this data set acts as a basis for the mapping between the system variables and parameters on the binary space $\mathbb{B}^n$.
- Assumption 2: The actual system state such as state space variables and parameters are provided and can be monitored.
- Assumption 3: The uncertainty models on the system variables and parameters are known or can be modeled. Otherwise, it is assumed that the requirements for applying the uncertainty quantification technique presented in Chapter 3 (presence of conversation equations) are met.

In case the vector $\vec{x}_B$, that contains the input data, corresponds to scalar values, one established method to solving this problem lies in framing it as a quadratic unconstrained binary optimization problem (QUBO) [47]. Generally, this method enables to calculate the distance between the classes to perform an assignment of data points. The main incentive in this thesis, however, is the derivation of an uncertainty-aware technique that results in replacing the scalar parameter values by probability distributions. These probability distributions reflect the uncertainty models of the system variables. Since applying methods for QUBO in a straightforward manner is challenging in this case, the thesis suggests to develop an approach that considers the divergence between two distributions:

To be specific, the data collection is characterized in two probability density distributions P_b and Q_b, where the data corresponding to the desired class of outcomes are distinguished from those with the class of undesired outputs for each parameter or variable, respectively. This means that two probability distributions are assigned to each of the input variables $x_{B,i}, ..., x_{B,n}$ of the vector $\vec{x}_B$. In particular, these two probability distributions result from the addition of the single probability distributions representing the data points plus their uncertainty models. Since the goal of the optimization is to select robot actions that yield desired outputs, a two-stage scoring technique is introduced. Here, the goal is to enable the rating in analogy to the cost function in Equation (4.1). Specifically, each grasp candidate is attributed to a total score $f_Q(x)$ that is calculated by introducing a global score $f_{glob}(x)$ and a local score $f_{loc}(x)$ as follows.

4.1.1 Global Score

The global score is computed via the Kullback-Leibler divergence D_{KL} between P_b and Q_b. As defined in Section 2.1, the Kullback-Leibler divergence, referred to by D_{KL} in the following, measures how two distributions differ from each other. Importantly, the D_{KL} is asymmetric and does not satisfy the triangle inequality such that it does not fall into the class of metrics, but considered as a statistical distance measure. This asymmetry is exploited for the score function that assigns higher scores to candidates that are closer to the desired outputs. Assuming that a range of options for the selection are offered to the robot, the rating is performed via D_{KL}, where the Kullback-Leibler divergence intuitively captures how influential a system parameter or state variable behaves regarding the outcome, that is

$$f_{glob}(x) := D_{KL}(P_b||Q_b). \tag{4.3}$$

4.1.2 Local Score

In addition to the global score $f_{glob}(x)$, the probability of the each candidate to fall into the data collection P is considered. In contrast to the global score, the local score $f_{loc}(x)$ computes an individual score for each grasp candidate. As a result, candidates that are likelier to belong to the group of the desired outcome will be assigned to higher score values. Specifically, each candidate $\vec{c} \in \mathbb{R}^m$ is assumed to be defined via m features, where m is specified individually for each application. The local score is given by the probability fraction

$$f_{loc}(x) = \frac{p_{k,s}}{p_{k,s} + p_{k,f}}, \tag{4.4}$$

where $p_{k,s}$ and $p_{k,f}$ denote the probability for the candidate to belong to the distribution P of desired outcomes and Q the undesired outcomes, respectively.

4.1.3 Total Score

The score for each candidate is then calculated by multiplying the global and the local score with each other for all metrics m, i.e,:

$$f_Q(x) = \sum_{k=1}^{m} f_{glob}(x) \cdot f_{loc}(x) = \sum_{k=1}^{m} D_{KL} \cdot \frac{p_{k,s}}{p_{k,s} + p_{k,f}}. \tag{4.5}$$

It is worth noting that the goal is to find the candidate that maximizes this score, that is

$$\vec{x}_{BO} = \arg\max f_Q(x) \tag{4.6}$$

instead of the minimization as defined by the original QUBO. Logically, this technique requires that the baseline data set is representative regarding the data amount and the conditions of the application. Practically, this places the assumption on the data set that the interesting regions are monitored as mentioned above. Specifically, the confidence of the score depends on the statistical uncertainty u_{stat}. As will be demonstrated in Section 5.1, one advantage of this technique lies in its online applicability. While the computational burden increases with the number of critical parameters, calculating the probability fractions and the Kullback-Leibler divergence is straightforward. Therefore, this technique is appropriate for settings, where the robot is provided with different possibilities and supposed to perform a binary optimization regarding the selection. However, estimates for the uncertainty models for all parameters that influence the optimization must be available. While this applies to most technical components, the employment of NN based tools that are not fully understood regarding their uncertainty behavior places a burden to framing the optimization problem in the above presented fashion. To be specific, the difficulty lies in estimating uncertainty models for the NN and to estimate the probability distributions for the intrinsic NN parameters. Hence, this method is limited to use cases, where no NN based component exists and where heuristics or theoretical uncertainty estimates are available. In the next section, the research question of combining the definition of the robustness of NNs with the metrological uncertainty notation will be addressed.

4.2 Uncertainty-aware Binary Optimization with linearizable NNs

Apart from systems, where heuristics of uncertainty estimates are available, the employment of NN based tools has become attractive in robot applications. As will be elaborated in the following, the NN robustness can be combined with the metrological uncertainty notation to one measure. One research question in this thesis is how this unified uncertainty measure contributes to the improvement in the optimization process. The key idea behind this type of optimization lies in the uncertainty-aware parameter adaptation. In the first step, computing the uncertainty regarding one attribute requires to combine the uncertainty propagation for one representation with the propagation technique for different representations. Practically, this means that after unifying the different uncertainty representations, that is, the NN robustness with the metrological uncertainty notation, the method for uncertainty propagation in one representation is applied. This requires to account for the correlations between the system components. To be specific, the input data set for the NN is limited by the measurement uncertainty of the considered component that is used for the data perception (e.g., cameras, recorders,...) and provides the data. For example in image recognition, the accuracy in the input data directly depends on the number of the employed cameras and their resolution limits. Specifically, the Lipschitz equation for the robustness should be modified by specifying the parameters in a practical context. This means that essentially, the terms in the robustness equation

$$\Pr((\|f(x') - f(x)\| \leq k * \|x' - x\|) \mid \|x' - x\| \leq \delta) \geq 1 - \epsilon \tag{4.7}$$

are specified individually for each application. However, to remain the quantification of the sensitivity and possible deviations, the terms $\|x' - x\|$ and $\|f(x') - f(x)\|$ are assigned to the input data and output data, respectively. From the perspective of uncertainty analysis, it becomes relevant whether x represents a ground truth data point, that is, free from uncertainties or already contains the uncertainty due to the data taking procedure. The following Section will discuss the reinterpretation of the parameters depending on the data quality and the available information on the uncertainty.

4.2.1 Case 1: Input Data x as Ground Truth Data

In this case, it is assumed that x is free from uncertainties and thus reflects the actual value of the input data point, but x' is affected by the perception uncertainty. For example, x may be provided by a model (e.g., simulation). In this case, following assignments to the terms in Equation (4.7) are suggested:

$\|x - x'\|$**:** This term would be subject to the uncertainty in x'. Here, it must be considered how the uncertainty is related to the l_p ball. Importantly, if the uncertainty is not symmetric with respect to the l_p ball measure, the direction between the points x and x' must be taken into account. Apart from that, a pointwise evaluation of the Lipschitz equation would address the question of how the uncertainty inherent to x' (and thus the difference between x and x') affects the constraint on the respective output term $\|f(x) - f(x')\|$.

δ**:** In this case, increasing or reducing δ simply corresponds to allowing a higher uncertainty in the perception of x'.

k : Logically, k hints at how sensitive the output of the NN responds to the uncertainty in x'.

$\|f(x) - f(x')\|$**:** Evaluating this distance between the outputs provides insights in the uncertainty in x'. In particular, it may be analyzed whether the uncertainty is symmetric by comparing $\|f(x) - f(x')\|$ of two x' values that lie in different directions from the ground truth data point x'.

From practical perspective, identifying the appropriate values for the control parameter k and the radius of the Lipschitz ball δ would hint at replacing the components that are responsible for the data acquisition. Specifically, an acceptable limit may be defined via $\|f(x)-f(x')\|$ that corresponds to the maximal radius δ. As δ reflects the tolerated spread around the clean data point x, from which x' is sampled, meeting this requirement would hint at reducing the uncertainties of the components that process the input data points accordingly.

4.2.2 Case 2: Missing Ground Truth Data

In most real-world applications, the knowledge on x is limited because the actual value of the input data point does not necessarily correspond to the theoretical model or simulation. In particular, the value of x may differ depending on the environmental state, where the dependencies between x and possible undesired disturbances (e.g., temperature changes etc.) are usually not known. In addition, the uncertainty on x is not provided such that the ground truth value for x cannot be easily determined. As a consequence, the terms in the

Lipschitz equation are affected by this unknown amount of uncertainty which necessitates the modification of the parameter interpretation accordingly:

$\|x - x'\|$: While this term still measures the distance between the input data points, it does not provide any succinct information on the uncertainty. Due to the fact that the uncertainty on x is not available, the distance to x' captures how far these two input points are separated from each other, including their uncertainties. Especially, if both input data points are subject to the same uncertainty, the obtained value would correspond to the ground truth distance. Furthermore, the symmetry of the uncertainty plays a critical role. This means that the same distance values may yield different values for $\|f(x) - f(x')\|$, depending on the direction of x'.

δ : Analogously, δ now specifies solely the allowed distance between x and x'. In particular, the tolerated uncertainty becomes challenging to determine for dynamic uncertainty behavior, that is, for constellations, where the accuracy in the data processing of x may differ from that of x'.

k : In contrast to above case, where the radius of the l_p ball δ places the limit on the acceptable uncertainty, the control parameter k becomes more relevant for deriving a threshold on the absolute uncertainty. While in accordance to the previous case, k reflects how the NN responds to the input data points x and x', monitoring the effect of different k-values in addition to the l_p-ball radius is necessary for studying the uncertainty threshold.

$\|f(x) - f(x')\|$: In analogy to the first case, this term allows to verify whether the desired limit on variations in the output is met by the corresponding input data points.

Due to the fact that the influence of the uncertainty cannot be directly determined in this case, the collection of offline data is necessary to facilitate the development of uncertainty-aware optimization techniques. This is especially important in real-world (robot) applications, where usually no theoretical uncertainty model exist. To be specific, it is desired that the offline data set contains information on the system variables or parameters that influence the uncertainty in the input data. It is preferable to conduct the data recording under fixed conditions.

Furthermore, following cases must be distinguished:

- **Symmetric Uncertainty**: In case the uncertainty is symmetric, that is, that the uncertainty amounts to the same value for all x' with the same $\|x - x'\|$-value regardless of their orientation from x, it is sufficient to evaluate one output term $\|f(x) - f(x')\|$ for each $\|x - x'\|$. Assuming that the environmental conditions are constant during the data recording process, monitoring how the fraction of input data points that satisfy the Lipschitz condition develops with increasing $\|x - x'\|$-values is required for analyzing the uncertainty. Based on the results, statistical tests and correlation studies must be applied to gain further knowledge on the relationships between the uncertainty and the system characteristics. Finally, bringing together these findings may enable to derive a basic estimate of the uncertainty and its dependencies.
- **Non-symmetric Uncertainty**: In the given context, a non-symmetric uncertainty behavior can be concluded once two x' that are equal in $\|x - x'\|$ yield different values for $\|f(x) - f(x')\|$. Importantly, this conclusion is valid since evaluating the robustness with the Equation (4.7) requires that the input data points can be represented by an l_p ball, which in turn necessitates the assumption of a symmetrical

input space. If different output terms occur for same input point distances, additional studies on the dependency of the uncertainty on the orientation of x' must be performed to enable the development of an uncertainty-aware optimization method. At the same time, generating input data points within the entire l_p ball is tied to high computational costs. Therefore, in the context of practical applications, specifying the interesting directions of x' within the l_p ball is recommended. In particular, the direction of input data points that are most likely to occur in the real-world application can be identified to define a vector between x and x'. By generating data points along this vector and observing the variations of $\|f(x) - f(x')\|$ enables to derive estimates for the uncertainty inherent in x.

Apart from that, the determination of the input data uncertainty can be performed by applying the method based on conservation equations introduced in Section 3.2.1. One prerequisite however of this technique is that system invariants must be identified and formulated via equations beforehand. In order to carry out the uncertainty-aware optimization, the dependencies between the critical parameters of the estimated uncertainties as well as the knowledge on the system equation f_s is leveraged. Practically, this means that the results from the uncertainty analyses, like the relationships between system parameters (e.g., lightning or temperature conditions) and the uncertainty in the input data points allow to identify the constellation that lead to the highest fraction of the desired output, that is, the values of $X_1, ..., X_n$ from Equation (3.1) that correspond to

$$\eta(f_S) = \eta_{max}. \tag{4.8}$$

Specifically, in the case of robot applications, the dependency of the uncertainty on the direction in the l_p ball may enable to derive regions for the robot parameters and movement that correspond to lower probabilities of undesired outcomes. In addition, this technique can be complemented by the data collected online as will be described in Section 5.2.2. Consequently, since the data amount for the optimization increases with the time, the statistical significance of the uncertainty-aware optimization becomes higher. This is a crucial advantage, especially for repetitive tasks, since the optimization becomes more reliable and accurate. Logically, higher optimization performances η_{cont} are expected to be achieved.

4.3 Continuous Uncertainty-aware Optimization

In many real-world robotic applications, an optimization with respect to the system efficiency or safety is desired. While this may appear straightforward at first glance, such an optimization requires more detailed knowledge on the system. In contrast to the discrete optimization methods, continuous optimization addresses applications, where the objective lies in maximizing or minimizing one specified property as described in Section 2.1.3, where the dependency with the system variables and parameters are expressed via continuous functions. In the ideal case, the relationship between the system parameters and the optimization goal is known or available in form of an analytical function. This would allow to identify the optimal system constellation and to adapt the state of the system with its dynamics accordingly. Unfortunately, this is usually not the case, in particular for complex real-world environments. In addition to the relationship between the system parameters and the optimization, it is often not known how the environmental conditions influence the optimization performance. While rough intuitions may be available on how

the system components are related with the system's efficiency or safety, incorrect approximations may result not only in a decreased optimization performance, but also in an undesired system behavior. In this case, considering models of the parameters' and variables' uncertainties are one possibility to account for the limited knowledge and possible deviations from the approximated system states. Ideally, these considerations would enable a more accurate representation of the actual state. Specifically, estimating how much and in which sense the system state may deviate from the anticipated values due to possible inaccuracies in the modeled relationship between the system dynamics and the optimization goal is of interest. In doing so, it is aimed at capturing the probability for shifts in the output of a system due to deviations in the input data or employed technical tools. Thus, an ideal uncertainty-aware optimization would build upon an online uncertainty quantification technique: the optimization would account for the actual uncertainty of the system with the influence of each component and, based on this, adapt the system dynamics to reach the optimization goal. Here, the components with dominant uncertainty contributions must be monitored accurately to adapt the system once necessary. An additional challenge occurs in tasks, that are not repetitive or are subject to environmental changes: Logically, the statistical uncertainty in unexplored regions arises due to the missing data. In these cases, simulation studies can be performed beforehand. Apart from that, employing importance sampling instead of traditional MC sampling (see Section 2.1.7) can help to enhance the generation of samples in interesting regions. To this end, following steps are suggested for a successful uncertainty-aware optimization:

1. The relationship between the system state variables and parameters and the optimization goal must be formalized or modeled in a simulation. Specifically, it is desired to be provided with an analytical function that describes or at least approximates the system.
2. The uncertainties of the system variables and parameters must be estimated regarding the attribute. Here, it is required to specify the relevant parameter spaces. In case the optimization goal is of interest for different spaces, the dependencies between these must be considered. It is assumed, that the propagated uncertainty is obtained according to the previous Section 3.2.2.
3. The uncertainties must be incorporated in the optimization pipeline. Here, the main challenge lies in identifying an appropriate technique: Since the goal is to improve the optimization by considering the uncertainty, that is, by adapting the system constellation (the parameter and technical components) that corresponds to the highest η, the sensitivity is monitored in addition to the uncertainty. In doing so, two aspects are observed: The relationship between the uncertainty and the optimization performance as well as the influence of each system component on the optimization performance η.
4. The optimization is carried out by referring to the sensitivities. Practically, this means that the uncertainty for the parameters and variables that are attributed to high sensitivity values for η is reduced or aimed to be kept at a certain level. Depending on the application, this may be achieved by replacing technical tools or limiting the fluctuations of the environmental disturbances by introducing certain restrictions. This will be discussed more thoroughly in Chapter 5.

In particular, step 3 highly depends on the application and the formalization of the optimization goal. In addition to the evaluation of η that may be challenging in cases where many parameters dependencies exist, certain parameter regions might be of higher relevance. However, if not sufficient data is generated in these areas, the reliability of both

the uncertainty quantification and the optimization will suffer from it. This chapter is dedicated to addressing above challenges. After drawing the analogy to particle physics and proposing the incorporation of the sensitivity as a reasonable measure for enhancing the optimization performance in robot applications, particular emphasis will be placed on the step 3 and step 4 to discuss the evaluation of η for robot systems.

4.3.1 Sensitivity-based Uncertainty-aware Optimization

While the entire scope of this thesis is concerned with transferring established techniques in the domain of particle physics into robotics applications, the uncertainty-ware optimization with continuous functions offers a notable set of possibilities for drawing useful analogies. Specifically, improving the detection efficiency of particles on microscopic scales can be expressed via optimization functions. Usually, these functions are defined with respect to the resolution of the employed detectors. Despite the fact that the system dynamics in robot systems is mostly described on macroscopic scales and generally formalized in terms of equations from the robot control domain, the experimental setups in both particle physics and robotics share the goal of maximizing the optimization performance η.

In the following, the similarities and discrepancies will be outlined to point out that compromises between the transfer and the individual adaptation in robot applications are equally important. In addition, finding these differences and analogies will help to derive appropriate optimization techniques as will be elaborated on in Chapter 5. In this context, the *sensitivity* introduced in Section 2.1.5 is considered in the majority of physics experiments to monitor the development of the uncertainty. This allows to derive countermeasures to limiting the uncertainty to a tolerable amount or to adapt the system according to the actual constellation.

Interestingly, although uncertainty-aware methods have been developed in the field of robotics regarding various application scenarios, monitoring the sensitivity while analyzing and interpreting the system data online has not been suggested in the context of robotic applications. By addressing this gap, the thesis aims at exploring whether incorporating the sensitivities in analogy to physics experiments can improve the optimization in robot systems. In the following, the steps for establishing an uncertainty-aware optimization technique by accounting for sensitivities are described.

4.3.2 Formulation of System Function and Optimization Goal

To perform a continuous optimization, a function describing the system in the context of the optimization goal must be formulated. This formalization should ideally describe how the parameters and variables are related to the optimization goal. In contrast to the discrete optimization, continuous optimization addresses such problems where the optimization is not carried out with respect to one desired class of outcomes, but the maximization or minimization of one property. Therefore, the challenge lies in modeling the optimization problem by integrating the system dynamics in a reasonable manner. First, the attribute where the uncertainty is of interest for and the system dynamics must be specified. Importantly, to enable the consideration of uncertainties, the system equations must be formalized accordingly with respect to the attribute and the critical parameter space, that is, the terms introduced in Equation (2.16) must be reformulated such that the process noise

$\omega(t)$ and the observation noise $v(t)$ are defined with respect to the measurement. This step stands in accordance with the formalization for the uncertainty determination of single components as well as the propagated uncertainty in Chapter 3. Next, the optimization goal, that is, the continuous function describing the relationship between the optimization measure and the system parameters and variables must be defined. Usually, this formalization is given by a model or must be derived. For example, if the minimization of the system's risk is desired, the critical parameters and variables for defining risk situations must be specified. Importantly, it is assumed that the relevant correlations between the parameter spaces for the determination of the accumulated uncertainty u_{prop} have been completed in the previous steps and are thus available.

4.3.3 Assumptions and Data Monitoring

To enable the uncertainty-aware optimization, following assumptions are made for the remainder of this thesis:

1. The data monitoring is performed in real-time. This means that the latency in the data acquisition process is assumed to diminish, that is, $t_{latency} = 0\,s$.
2. The data flow can be monitored for all parameters and variables. As a consequence, the state of the system can be fully described and characterized by means of the measurement data.
3. Unless stated otherwise, all input data points are assumed to carry equal amount of information. As a logical consequence, the absence of each data point affects u_{stat} equally such that the statistical uncertainty is assumed to depend solely on the number of data points N.
4. The recorded data is available at all time steps. Particularly, unlimited data storage is assumed.
5. Lastly, application-specific assumptions are made, which will be defined at the beginning of each experiment. Particularly, these enable to justify the parameter regions where the input data is sampled from as will be elaborated in Chapter 5.

Therefore, the optimization techniques that consider uncertainties are derived based on these assumptions. While these cannot be maintained to full extent under real-world conditions, the consequences of limitations will be discussed in the context of the validation experiments.

4.3.4 Uncertainty-aware Optimization Function

The derivation of the uncertainty-aware optimization function is performed on the basis of the equations for the system dynamics and the formalization of the optimization goal. In particular, the first step lies in identifying the critical parameters. Usually, the optimization is defined via measures that do not occur in the system function. Hence, it must be first identified which parameters of the system function f_S are related to the measures in the optimization goal, and thus influence the optimization. In doing so, the respective uncertainties and existing parameter dependencies must be considered. In addition, it is crucial to define the relevant parameter spaces for evaluating η.

While the optimization goal may be stated by one equation that builds upon a limited number of measures, monitoring them and evaluating the optimization performance might

require the consideration of several parameter spaces. In particular, the parameter spaces might depend on each other according to the regions that are relevant for the uncertainty incorporation. Hence, the main challenge at this stage is the successful incorporation of the uncertainties of the parameters and variables in the system equation into the optimization function. This is essential to assess whether performing the optimization in an uncertainty-aware fashion yields more promising results. To this end, studies that allow to deduce the critical parameters and variables for the optimization and how they exactly influence the optimization performance.

The main steps for deriving the uncertainty-aware optimization function can be summarized as follows:

1. **Identification of critical parameters and variables:** First, the critical system parameters and variables for the optimization function

$$\eta_c(\vec{x}_c) = \psi\vec{x}_c(\gamma_i, u_{prop}(\vec{x}_c), ...) \tag{4.9}$$

must be specified. Specifically, this corresponds to the specification of the critical parameters, the measurands and the propagated uncertainty $u_{prop,a}(p_c)$:

$$\eta_c := f_{co}(Y_1(p_c), ..., Y_n(p_c), u_{prop,a}(p_c)). \tag{4.10}$$

Logically, the optimization function must be provided or at least approximated by referring to the measurands $Y_1(p_c), ..., Y_n(p_c)$. Practically, this means that it must be specified which system characteristics must be measured to evaluate the optimization goal of the respective robot application.

2. **Reformulation:** Based on the assignment in the previous step, the optimization function is reformulated by using the metrological notation. To be specific, the measurand is defined according to Equation (3.1). In doing so, the parameter spaces γ_i must be considered:

$$f_{opt}(a, s_{o,1}, ..., s_{o,n}) \; ; f_{eng}(x, u, \omega), z \; \mapsto f_{metrol,system}(Y_1(\gamma_i), ..., Y_n(\gamma_i)). \tag{4.11}$$

It is worth noting that the reformulation of the optimization does not necessitate to unify all measurands into one equation. Indeed, as described in the introduction of this Chapter, the system equation may be split into several parts, depending on the parameter regions and their relationships to each other. Therefore, the goal in this step is to derive an optimization function that is defined using the metrological formalism. In addition, the optimization performance is introduced by referring to the desired outcome of the optimization $p_t(\vec{x})$.

3. **Uncertainty Specification:** Logically, the uncertainties of the state variables and parameters must be considered:

$$Y_1(\gamma_i) \mapsto Y_1(\gamma_i) \pm u_{Y_1}(\gamma, i), \tag{4.12}$$

such that the measurand with its uncertainty $u_{Y_1}(\gamma_i)$ on the right hand side is modeled via a probability distribution. In particular, it may be required to apply propagation techniques to obtain $u_{Y_1}(\gamma_i)$ depending on the complexity of the system. It is assumed that $u_{Y_1}(\gamma_i)$ has been determined via the uncertainty quantification techniques explained in the previous Section 3.2.1 and Section 3.2.2.

4. **Critical Monitoring Step Size (Sensitivity):** While calculating the optimization performance η by accounting for the uncertainty, that is, observing the development of η, the uncertainty values are monitored. This helps to identify dependencies. In particular, these data must be stored and made available for the analysis in the next step. Selecting an appropriate time window for the monitoring process is crucial since it directly affects the accuracy of the sensitivity calculation. Furthermore, it is highly dependent on the application and parameter space: In regions with high robot velocities, higher monitoring rates may be desired while in regions with sparse occupancies, large time windows are sufficient. The monitoring time is denoted with t_m. In particular, the optimization performance is monitored with the sensitivity regarding the system components:

$$\eta \left(\frac{\partial f_{sys}(\gamma_i)}{\partial Y_j(\gamma_i)} \right). \tag{4.13}$$

5. **Data Evaluation/Confidence:** To derive the measures to improve the performance η, the data collected in the previous step is investigated regarding the influence of each variable and its uncertainty. The main question is, for which set of variables the maximum of η is achieved. The corresponding analysis is performed by sensitivity analyses and statistical studies. In particular, when monitoring the state variables and their fluctuations, statistical fitting methods can be applied. Thus, it is aimed to identify following set of variables

$$\mathcal{V}_{opt} = \{Y_1, ..., Y_n\} \quad \text{for} \quad \eta \rightarrow \eta_{max} \tag{4.14}$$

   Based on the amount of data considered for the analyses and the monitoring step-size, the confidence of the derived parameter relationships is assessed. These confidence values are then used to determine the optimization measures in the following step.

6. **Sensitivity-based Optimization of η :** The final step of the optimization is highly dependent on the application- specific characteristics of the robot system. Logically, a setting with stable and minimum uncertainty is preferred. Although these states and settings may be derived in a straightforward manner on the basis of the results obtained in the previous step, the optimal setting may be difficult to be realized under real-world conditions. For example in human-involved applications, the human may be required to keep the walking velocity below a certain limit or the optimization efficiency η might suffer from small disturbances in the lightning conditions. Practically, one optimization technique here is to monitor the sensitivities simultaneously – that is, observing the development of the optimization performance η for changing human velocities. Apart from that, possible steps to boost η might be given by the adaptation of the environmental parameters or the replacement of one technical component by such with smaller resolution uncertainties. However, deriving generally valid constellations for any type of application is impossible assuring the maximum η. In this thesis, the derivation of optimization methods are discussed in the context of three experiments as will be detailed in Chapter 5.

4.3.5 Uncertainty-aware Evaluation of Limits

So far, the main goal for calculating the uncertainty was the incorporation in the optimization pipeline. However, the knowledge on the uncertainty of an entire system, that is, the

propagated uncertainty, can help to verify quantitative requirements that are placed on the application. This becomes particularly relevant for safety-critical systems since official regulations exist. For example, in order to facilitate real-world human-robot collaboration, these limits must be kept throughout run time. Ideally, the monitoring must be conducted continuously to notify the system once essential limits are violated. While safety limits correspond to a suitable use case where the uncertainty becomes relevant, the mapping on threshold is not limited. Logically, the verification of threshold can be performed for any types of limits placed on certain parameters of the system. In the following, the uncertainty is directly mapped on thresholds that are defined for robot applications. In fact, these thresholds are expressed via probabilistic measures.

Here, the mapping focuses on three types that occur in safety standard literature. Accordingly, the validation experiments in Chapter 5 demonstrate that the system's safety can be assessed via the uncertainty under certain assumptions. Formally, by referring to a limit λ that stands for the probability of occurrence for an undesired outcome, the evaluation is performed on the basis that the probability of a critical variable to falling below (or exceeding) a critical limit is not larger than λ, i.e.:

$$\Pr(a \leq a_{crit}) \leq \lambda. \tag{4.15}$$

For a real-time system, where all parameters and system variables are obtained, evaluating this equation is straightforward. However, uncertainties in the monitoring or measurement process of the attribute a can cause to inaccurate evaluations. In particular, above equation must be modified to

$$\Pr((a \pm u_a) \leq a_{crit}) \leq \lambda. \tag{4.16}$$

In contrast to evaluating Equation (4.15), verifying whether λ holds for Equation (4.16) is challenging since u_a must be known. Especially, it is very likely that u_a must be obtained via propagating the uncertainties of all relevant components as elaborated in Section 3.2.2.

Specifically, this means that countermeasures can be directly introduced once Equation (4.15) is violated. Therefore the probability for the occurrence of undesired happenings is solely attributed to the uncertainties, that is

$$\int_T u_a dt \mapsto \lambda, \tag{4.17}$$

where u_a is integrated over the run time T. The exact mapping depends on the definition of the dangerous situations. For example, in Human-Robot Collaboration, dangerous events are characterized by the human-robot distance d_{HR} and the relative velocity v_{HR}. Thus, these variables would be treated as attributes. Above Equation (4.17) however is valid for any type of robot system, where the critical variables may differ from d_{HR} and v_{HR}. As will be specified in Chapter 5, the safety assessment via the mapping on the probability limit defined by λ is suggested by

$$\Pr\left(a \pm \int_T u_a dt\right) \leq a_{crit} \leq \lambda. \tag{4.18}$$

In particular, the safety evaluation is conducted on the basis of the actual value of a: In certain regions (e.g., in areas where the human is far away from the robot), a high uncertainty in the attribute (e.g., the human-robot distance) may be acceptable. In these cases, a large uncertainty would not yield a violation on λ. On the other hand, small uncertainties u_a may be highly influential in other cases. In the context of practical applications,

this mapping technique should be reliable. Especially, introducing countermeasures in the application, like the adaptation of environmental conditions or the selection of technical tools is bound to additional efforts. This means that the reliability on these suggestions should be sufficiently high. This reliability is given by the confidence. For example in above mentioned use cases such as safety-critical applications or real-world systems in general, reasonable confidence values are highly desired. Quantifying the confidence can be performed by means of statistical tests. In addition to the confidence, specific hypotheses are tested that are defined individually for each robot system in Chapter 5.

Secondly, the confidence directly depends on the amount of considered data. In addition, the aleatoric and epistemic uncertainty from the learning domain are suggested to be treated as confidence values. Practically, this means that the aleatoric and epistemic uncertainty, if available for the respective system, are included for the calculation of the confidence value. While these can be generated with simulations, artificial data may be not represent certain cases to sufficient extent. For instance, the occurrence of risk situations in real-world often originates from a lacking accuracy in the simulation. Apart from that, theoretical models may not cover all possible situations in the real world. Especially, constellations may be not considered that may occur due to correlation effects that were not taken into account theoretically.

In the context of risk analyses, worst case situations are of particular interest. In addition to their origins and the process of their evolvement, their occurrence probability is necessary to develop risk minimization techniques. However, studies on worst case situations is strongly limited due to the sparse availability of data collected under real-world conditions. In addition to the variety of possible constellations leading to dangerous outcomes, the obtained analysis results underlie low confidence values. While the first problem cannot be solved in straightforward manner by applying statistical methods, the confidence for the results of worst case analyses can be improved by means of importance sampling. As will be explored in more detail in the next Sections, importance sampling enables for dedicated generation of data points in interesting parameter regions. Although this technique has been developed and studied extensively in the domain of statistics and computer vision, not much attention has been attributed to applying it to robot systems in existing literature. By addressing this gap, this thesis aims at employing importance sampling for deriving risk minimization methods with the goal of enhancing the confidence of the uncertainty-aware optimization.

4.3.6 Interesting Regions and Importance Sampling

One assumption that must be met for enabling a reliable the uncertainty quantification, propagation and the uncertainty-aware optimization was that the samples collected for the calculations represent the corresponding regions with sufficient accuracy. However, especially in real-world robot applications, this assumption may be not met. In fact, it may be even difficult to meet this requirement as the interesting parameter regions might exactly be those where only a sparse amount of data is generated. For example, it may be unnecessary to generate samples in regions that will not be occupied by the robot due to possible limitations in the configuration. On the other hand, specific areas may be of higher interest for the optimization although they may be not occupied often: In order to derive an effective optimization algorithm, enhancing the sampling process for the regions of interest is recommended. Particularly, the necessity of a refined sampling algorithm might draw from a tolerated threshold for the approximation. For example, a

high accuracy is highly desired for the optimization techniques in safety-critical systems, as also elaborated in the work by Gandhi et al. in [31]. In the following, we show how this can be achieved by an algorithm based on importance sampling. As introduced in Section 2.1.7, importance sampling differs from classical Monte-Carlo sampling in the sample generation stage: Particularly, importance sampling enables to focus on specified regions of interest. This helps to reduce the Variation in the Approximation Error (VAE) in cases with low probability densities, that is, those cases that contribute to low occurrence probabilities like risk events. In the following, a defensive version of importance sampling that is employed to simplify the sample generation in areas with low probability densities.

Defensive Importance Sampling

In the context of this thesis, extending the uncertainty-aware optimization by means of importance sampling is reasonable for the risk reduction. In contrast to the image classification and humanoid robot grasping experiments, high approximation errors in the optimization algorithms for safety-critical environments can result in severe injuries on human beings. Therefore, integrating a method in the optimization pipeline that reduces the variance in the approximation error (VAE) is of particular interest. Formally, this is achieved by considering a different latent probability density function q that is referred to as the *proposal density* $q(x)$. The idea behind this proposal density is to assign higher sampling probabilities to the regions of interest. For instance, if the goal lies in the risk minimization, the occurrence of hazardous events is very rare. Here, the introducing q yields to enforcing the sample generation in the parameter regions corresponding to safety-critical events or even accidents.

According to the safety regulations, the danger level in Human-Robot collaboration directly depends on the relative distance between humans and robots. Hence, q would assign high probability occurrence values in low distance regions. Importantly, introducing $q(x)$ requires to weight the calculation of the VAE accordingly: Shortly, to enable the consideration of q instead of the actual probability density function p necessitates the modification in the equation for the VAE according to Equation (2.15) . Basically, following this thumb rule enables the identification of a reasonable q as explained and mathematically formalized in [61]. Obviously, the main challenge lies in the choice of the proposal density $q(x)$. As explained in Section 2.1.7, this can be achieved by the fraction $\frac{f(x)p(x)}{q(x)}$. However, even if the requirement is matched such that $q(x) \neq 0$, the fact that the proposal distribution stands in the denominator may lead to errors. Especially when it converges faster to 0 than the product $f(x) \cdot p(x)$, undesired errors are amplified. To prevent this, the concept introduced in [14] suggests to generate a third probability density function by mixing the values of $p(x)$ and $q(x)$. Following this concept, an importance sampling technique based on grid partitioning is derived in the next section.

Grid-based Importance Sampling

Usually, estimating the continuous importance sampling distribution is difficult and ineffective in terms of the computation time. Instead, as was convincingly proven and verified via experiments in the publications ([14], [48], [53]), grid-based importance sampling is sufficient for most robot applications. In particular for the uncertainty-aware optimization algorithm that was derived in the previous Sections, it is of interest to facilitate online application. Apart from that, it has not been studied to date whether importance sampling is

suitable for the goal of improving the analysis of cases with low occurrence probabilities in robot systems. To this end, this thesis provides the first step by developing a grid-based importance sampling algorithm enabling the integration of this statistical technique in the optimization of robot applications.

Essentially, grid-based importance sampling corresponds to a simplified version of importance sampling, where the work space is partitioned in 2D cells. Indeed, the grid partitioning must be specified in terms of the number of grids and the edge size. Still, the grid-based approach simplifies the estimation of a reasonable proposal density. In fact, the densities are calculated separately for each cell under the assumption that the cells are independent from each other. To be specific, a probability density $q_G(x)$ is defined for each of the grid partitions in G, where G formally is represented as a matrix. Here, the grid partitions correspond to 2D cells of equal size. The probability density $q_G(x)$ for any point in the work space is calculated via

$$q_g(x) = \frac{\sum_{x \in G} p(x)}{\sum_{i=1}^{n} p(x_i)}, \tag{4.19}$$

where the nominator equals the sum of the probability densities of the grid cells and the denominator the total density. Therefore, the proposal distribution for any point in the work space is obtained by dividing the density of critical samples in one grid cell by the total density of the entire work space. In order to estimate the number of cells for the work space, an algorithm of two stages is developed. Inspired by the contribution in [14], a learning phase is conducted prior to the importance sampling. In this phase, traditional MC is applied to find the clusters with high probability densities. Here, finding the appropriate amount of samples that are needed to identify these high probability density clusters is challenging. For this thesis, it is assumed that the work space is rectangular and thus can be split in rectangular cells. To be specific, the suggested approach is split in two stages: The learning stage and the importance sampling phase. While the goal of the learning stage lies in the identification of appropriate parameter settings, the importance sampling itself is conducted afterward. In the following, these two phases will be explained.

Learning Stage:Parameter Specification

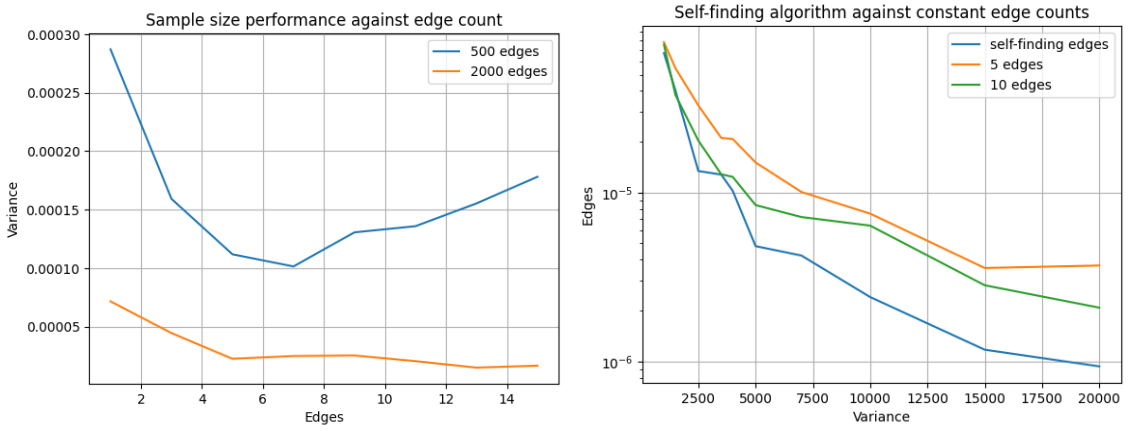
First, the parameters for the grid-based importance sampling such as the number of grids, must be found. From formal viewpoint, this step corresponds to an abstracted version of finding an the proposal distribution $q(x)$. First of all, the space $\mathcal{D}$ must be isomorphic. This assumption must be met to ensure that the work space can be partitioned into rectangular grids. Usually, the model of the work space is obtained by recording data points. However, as mentioned above, an essential limitation comes from the limited amount of data. In the first part of this step, traditional MC sampling as introduced in Section 2.1.7 is performed along the entire work space. Here, the goal lies in finding all clusters with high densities. Obviously, it is difficult to set the required number of samples that are required to draw conclusions. Despite from the minimum number of samples, specifying the parameters for the grids and importance sampling is difficult due to complex relationships between the search algorithm. As will be introduced in Chapter 5, the scenarios where importance sampling is applied to within this thesis will be limited to simulation experiments with reasonable complexities. Particularly, the interesting regions for the risk occurrence are not highly challenging such that an approximated version of the parameter specification approach is sufficient. To this end, the dangerous space is assumed as a 2D

circle. In order to perform the learning, that is, the identification of the equivalent to the proposal distribution $q(x)$, a fraction β of the total amount of generated samples is used. Intuitively, this fraction grows with the sparseness of interesting regions. The need for applying importance sampling increases with the relative sparseness of generated samples in areas of interest. In addition, the sample number in the learning distribution must be sufficiently high to find at least one sample per cluster.

Although these issues in importance sampling have been studied extensively in the domain of mathematics like in [41] and [61], a generic technique to estimating the optimal parameter setting and the number of required samples does not exist to date. Thus, the necessary parameter relationships must be estimated and considered individually. To this end, the following parameters are studied and specified:

1. Sample fraction β considered for the learning stage.
2. Number of edges e.
3. Amount of mixture for defensive importance sampling.

First, an approach to estimate the number of samples and grids is introduced. By referring to the example with the circle, the variance in the approximation error was estimated for three different values of β in Figure 4.1. Also, the dependency on the edge number is considered as shown in Figure 4.2. From these plots, it can be recognized that the choice of the fraction β does not affect the VAE significantly. In addition, it seems that the number of initial samples grows with the fraction β for importance sampling.



(a) The VAE is shown vs. the sample number for 5 edges and 10 edges. The blue line corresponds to the results obtained by the method for the self finding edges.

(b) The VAE is estimated for different fractions β. The figure shows that the VAE decreases with the number of edges, that matches the expectation.

Figure 4.1: Above subfigures depict the influence of the parameter β on the VAE.

The second parameter that must be specified is the number of edges e, where the number of grid cells can be calculated by e^n with the dimension n of the domain $\mathcal{D}$. As demonstrated in Figure 4.2, the VAE behaves differently for the number of samples and edges. Obviously, an optimal number of edges cannot be estimated in straightforward manner. However, the existing dependency can be leveraged to approximate the ideal number of edges. To do so, a density-based edge filtering is suggested. Specifically, it is aimed at using the learning samples that were used to initialize the grid partition G for finding the

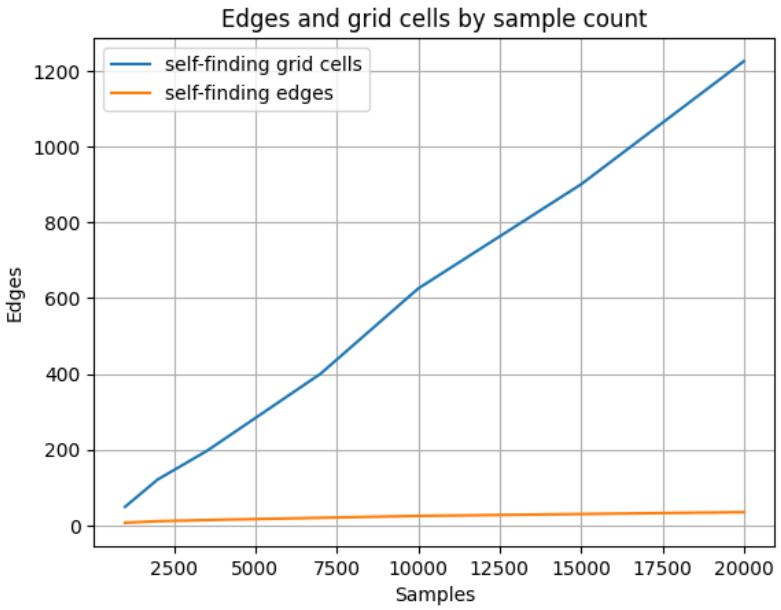


Figure 4.2: The relationship between the edge number and the amount of samples is shown.

number of edges. Intuitively, the number of edges are expected to be proportional to the amount of samples. At the same time, it is known the the number of cells is proportional to the n-th power of the edges e^n such that the edges can be computed by the equation

$$edges \propto \sqrt[n]{density}. \tag{4.20}$$

Therefore, the determination of the edge number can be performed by including the fraction $|s|/\beta$ for the determination of the edge number, as well as the area d of the entire work space:

$$e = \sqrt[n]{\frac{|s|}{\beta} \cdot \frac{1}{d}}. \tag{4.21}$$

Figure 4.2 shows a logarithmic increase of the number of edges when the amount of grid cells is increased linearly. Importantly, it was compared whether applying Equation (4.21) yields lower VAEs than fixed grid resolution approaches. As can be seen in Figure 4.1, estimating the edge number by above equation is leads to a smaller VAE values.
The third parameter that must be fixed is the amount of mixture. According to the contribution of Owen et al. in [61], combining the importance distribution with a certain fraction with the original probability density $p(x)$. While Owen et al. suggest a mixture between $10\,\%$ and $50\,\%$ based on theoretical derivations in [61], the exact amount of fraction must be specified individually. This means that the mixture must be chosen according to the robot application and its parameter regions of interest. Of course, the amount of grids and edges influences the value of the fraction as well. For the experiments in this thesis that are described in Chapter 5, the mixture is set to $25\,\%$.

Importance Sampling Stage

In the second phase, the importance sampling algorithm is designed according to above parameter constellation. The remaining step that must be taken at this stage include the

reweighting. Following Equation (2.14), the samples must be reweighted to equal out the effect of the proposal density. Logically, this adjustment also depends on the application. In terms of the grid-based importance sampling algorithm that is applied in this thesis, the reweighting is performed by the area of the work space A_G. Specifically, the fraction is given by

$$|C| \cdot \frac{A_G}{A_g}, \tag{4.22}$$

where $|C|$ denotes the value of the critical samples and thus corresponds to $q(x)$.
The algorithm for the suggested importance sampling is shown in Algorithm 1.

4.4 Summary of Chapter

Building upon the uncertainty quantification and propagation methods in Chapter 3, uncertainty-aware optimization techniques were presented. To be specific, the approaches to distinct optimization were split in a binary quadratic unconstrained optimization problem and the binary optimization with linearizable neural networks. While the first method performs the optimization on the basis of a pre-defined cost function, the optimization with linearizable neural networks refers to the analogy between the robustness and the metrological uncertainty notation introduced in Section 3.2.2. In particular, it was suggested to assign meanings to the parameters in the Lipschitz equation by referring to the availability of the ground truth data points. This draws from the fact that the sensitivity analysis as well as the accuracy in the uncertainty quantification, and thus the quality of the uncertainty-aware optimization highly depends on the amount of available ground truth data. Importantly, in the case of neural network classifiers, the quality of the training data plays a decisive role. By considering this in the unification, it was aimed at accounting for the data quality and amount.

In addition, continuous uncertainty-aware optimization techniques were derived in Section 4.3. Here, high emphasis was laid on estimating and incorporating the sensitivities. To this end, it was elaborated on the system description and the formulation of optimization goal. Essentially, this corresponded to an extension of the basic notation provided in the beginning of Chapter 3. In order to facilitate the application to robot systems, it was explained how the uncertainty-aware optimization function can be described by identifying the critical system parameters, reformulating the system equation and integrating both the uncertainties and the sensitivities. Furthermore, the importance of the confidence that basically reflects the trustworthiness of the optimization process was highlighted.

Apart from that, a technique to evaluate quantitative limits on the basis of the uncertainty was developed in Section 4.3.5. As mentioned in the previous Chapters, this thesis aims at providing a method to evaluate safety limits of robot systems. With regard to this goal, a technique to mapping the uncertainty to probability limits was introduced. In fact, this is motivated by the definition of the standards provided by the International Organization for Standardization (ISO), that are expressed via the probability for the occurrence of risks. Thus, the presented mapping technique showed that the derived uncertainty can be directly employed to assess safety of robot systems.

Here, one possible challenge arises due to the low amount of data on risk events. While assessing safety is highly important for real-world robot applications, the accuracy of the

evaluation technique directly depends on the amount of data that reflect dangerous events. In order to overcome this issue, and to still enable the development of a reliable safety assessment method, an algorithm based on grid-based importance sampling was introduced. In contrast to traditional Monte-Carlo sampling, where samples are generated in random manner, importance sampling aims at enforcing the data generation in interesting parameter regions. In the case of safety evaluation, more data is created in regions where dangerous events are likely to occur. Since the focus of this thesis does not lie in studying the quality of importance sampling techniques, a simplified grid-based version is suggested.

In the next Chapter, the above developed uncertainty-aware optimization methods will be applied in real-world robot experiments. By doing so, the optimization functions will be evaluated by means of the optimization performance η. Specifically, the experiments will explore whether incorporating the uncertainty yields improvements of the performance in robot systems.

Algorithm 1 Grid-based Importance Sampling

procedure IMPORTANCESAMPLE
    $n \leftarrow$ *number of total samples*
    $\beta \leftarrow$ *fraction of samples for learning phase s*
    $e \leftarrow$ *number of edges per side*
    Gaussian $\leftarrow$ *function to create Gaussian noise*

    $G \leftarrow$ Grid partition (sample space)
    $S \leftarrow$ sample($n * \beta$)
    $C \leftarrow$ Critical samples in S
    Sort critical samples into partition G:
    for $c \in C$ **do**
        $i \leftarrow$ Index of cell that contains c in G
        G[i] $\leftarrow$ *G[i]* $+ 1$
    end for
    $t \leftarrow 0$ ▷ Helper variable (density of all grid cells)

    Calculate total density (all cells):
    for $g \in G$ **do**
        $g \leftarrow \frac{g}{|C|}$
        $g \leftarrow g +$ Gaussian()
        $t \leftarrow (t + g)$
    end for
    Adjust density (account for added Gaussian noise):
    for $g \in G$ **do**
        $g \leftarrow \frac{g}{t}$
    end for

    $r \leftarrow n * (1 - \beta)$
    output $\leftarrow 0$
    for $g \in G$ **do**
        $S \leftarrow sample_g(r * g)$ ▷ samples taken inside g
        $C \leftarrow$ critical samples in S
        $A \leftarrow$ Area of g / Area of sample space
        output $\leftarrow |C| * \frac{A}{g}$
    end for
    output $\leftarrow \frac{\text{output}}{r}$
    return output
end procedure

5 Validation Experiments

This Chapter intends to explore and validate the applicability of the uncertainty-aware optimization methods presented in the previous Chapter to robot systems. Basically, the developed techniques can be applied to any types of robot systems, where a metrological representation is provided. In addition, the possibility to record data on the critical system variables must be given.

As elaborated in Chapter 1, the applicability of these methods to use cases from the domain of natural sciences has been proven successful. However, the question arises whether these notations and methods can be directly applied to robot systems. While the formalization may be carried out in analogous manner, studying the applicability and benefits of uncertainty-aware approaches in robot applications may motivate a stronger use of uncertainty-aware techniques in robotics. Therefore, three robot applications are introduced that serve as examples to validate the uncertainty-aware optimization algorithms developed in Chapter 4. In doing so, state-of-the-art literature is briefly summarized at the beginning of each application example to point out the lack of uncertainty-aware optimization techniques in the respective subdomains.

1. **Maximization of Successful Robot Grasps:** The first application falls into the category of robot grasping. Specifically, Section 5.1 explores how modeling the characteristics of robot grasps in an uncertainty-aware manner influences the optimization performance. Specifically, the optimization corresponds to the maximization of the rate of succeeded grasps. A humanoid robot grasping application is introduced, where heuristic uncertainty estimates are integrated into the grasp selection pipeline. Since these heuristic estimates are provided, the uncertainty quantification and propagation steps that were developed in Chapter 3 are not applied here. In fact, this application intends to point out that replacing scalar representations by uncertainty-aware models can enhance the optimization performance significantly.
2. **Minimization of Inaccurate Image Classifications:** Section 5.2 strongly focuses on the determination of a system's uncertainty by bringing together notations from different domains. By referring to the optimization method for systems with linearizable NNs in Section 5.2.5, an industrial robot application for image classification is presented. While the goal lies in minimizing the rate of incorrect image classifications in an uncertainty-aware fashion, this Section elaborates on the reinterpretation of the parameters in the Lipschitz equation in Equation (4.7).

   By demonstrating that integrating the combined uncertainty into the optimization pipeline yields an enhanced optimization performance η in the context minimizing incorrect image classifications, the validity of the proposed unification is discussed. Next, the experiments are extended to 3D human position detection. In particular, the practicability of this uncertainty metric for real-world human-robot collaboration is discussed.
3. **Safety Evaluation and Risk Minimization in HRC:** Section 5.3 is concerned with validating the majority of the methods developed in this thesis. Specifically, the

experiments are performed in the context of safety evaluation and risk minimization of robot systems. First, the uncertainty quantification technique for black box tools derived in Section 3.2.1 is validated on real-world data in Section 5.3.3. Second, the correctness of the uncertainty propagation algorithm introduced in Section 3.2.2 is verified by experiments performed in a PyBullet simulation environment. Afterward, the real-world applicability of the uncertainty propagation algorithm for evaluating safety limits (e.g., standard ISO 13849) is studied in Human-Robot Collaboration (HRC). To this end, the mapping technique described in Section 4.3.5 is validated.

In order to explore whether the grid-based importance sampling technique introduced in Section 4.3.6 contributes to a higher accuracy in the risk analysis, simulation experiments are performed in Section 5.3.4. Particularly, these experiments explore whether the introduced grid-based importance sampling approach is suitable for studying risk regions by accounting for both the uncertainty and severity of accidents.

The findings on the above applications are summarized in Section 5.4.

5.1 Binary Uncertainty-aware Optimization for Humanoid Robot Grasp Selection

Robot Grasping is one well-studied domain in robotics. Especially, developing techniques to robot grasping would allow to employ robots in a variety of applications reaching from industrial systems to daily tasks. The grasping process of a robot can thereby be divided in different steps. Also, the characterization and formal description of a robot grasp highly depends on the application and the employed robot. Apart from the specifications of the system, the robot is generally faced with the challenge of deciding on an action to execute. In fact, studying this decision making step is fundamental for robot manipulation tasks.

In the context of robot grasping, possible grasp candidates are commonly generated in first place. To do so, the robot is provided with information on the scene, for example with point clouds. Based on this, the robot selects the most promising grasp candidate for execution. Here, one major challenge arises from the incomplete knowledge on the scene. Existing methods in literature focus on methods that build upon neural networks. Essentially, the goal lies in predicting grasp poses or the classification of promising grasp candidates. While these approaches have proven efficient , one drawback arises from their black-box character. Especially, the missing understanding on the decision-making process of neural-network based methods places a burden on deriving approaches that may further enhance the grasp success rate. Particularly, it is difficult to identify the relevant characteristics of the grasp candidates that affect the grasp success.

However, gaining knowledge on how the grasp, or more specifically the metrics that define the grasp, influence the grasp success may be beneficial on a broader scale. In particular, a detailed knowledge would enable to eliminate factors that influence the success negatively. Following the basic motivation of this thesis, that lies in the exploration of uncertainty-aware optimization techniques, this Section will focus on the incorporation of uncertainties into the grasp selection process and the afterward evaluation with the grasp success rate. In fact, the grasp will be described by so-called *grasp metrics* that characterize each grasp candidate. While several approaches in literature like [64] and [18] propose

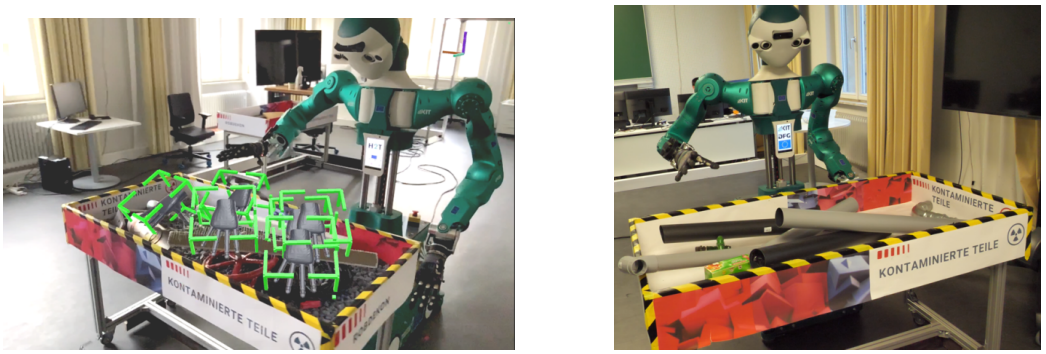


Figure 5.1: Experimental Setup with the humanoid robot ARMAR-6 for grasp selection.

the consideration of uncertainties for the grasp selection, the authors do not analyze the effects on the grasp success.

In the context of the uncertainty-aware optimization techniques introduced in the previous Chapter 4, robot grasping falls into the category of a binary optimization. Specifically, each grasp are distinguished in either succeeded or failed, while the optimization goal lies in attaining a maximum rate of succeeded grasps. After expressing the optimization problem in terms of the grasp metrics and its uncertainties as suggested in Section 5.1.3, the experimental setup of the humanoid robot grasping application will be presented. In doing so, the incorporation of heuristic uncertainty estimates will be discussed. Based on real-world experiments with the humanoid robot ARMAR-6 [7], the effects on the grasp success are analyzed statistically. In a broader sense, this section intends to highlight the effects of considering the uncertainties in a robot application, where the uncertainties are assumed to be provided.

5.1.1 Related Work: Uncertainty-aware Robot Grasp Selection

First, a brief outline of state-of-the-art literature on uncertainty-aware robot grasp selection is provided. While development of an uncertainty-aware grasp selection technique serves merely as an example to validate the applicability of the binary optimization, the explicit consideration of uncertainties in the grasp selection process complements existing methods in the domain of robot grasping. Logically, all contributions share the goal of maximizing the rate of successful grasps. Here, optimizing the grasp selection on the basis of generated grasps plays a crucial role.

Here, several contributions suggest learning-based methods to derive ranking methods of grasp candidates or their prediction. For example, the paper of Erkan et al in [24] introduces an extension of a Kernel logistic regression training algorithm, thereby bringing together semi-supervised and active learning methods. With that, labeled grasps, even if extremely limited in their availability, can be referred to for calculating success probabilities for unseen grasp candidates. The core idea lies in leveraging that similar grasp configurations yield similar success and failure probabilities. In the context of uncertainties, an uncertainty-based active learning approach is applied. Here, the uncertainty describes the entropy: the algorithm queries grasps with the highest entropy to assign these with the correct labels. Similarly, Goins et al. develop a Gaussian Process based classifier with 12 different grasp metrics in [34]. In fact, a Gaussian Process based classifier is trained and validated with data generated in simulation. Thus, the grasp quality is predicted prior

to the execution of a grasp: Specifically, a continuous value between 0 and 1 reflects the success probability of the respective grasp.

While these works as well as Kappler et al in [45] present methods to predicting the grasp success based on black box tools such as machine learning algorithms that are not fully understood to date, Chen et al. introduce a more transparent approach in [18]. Here, four hand-crafted metrics are suggested to describe a grasp. By combining these metrics with a probabilistic surface representation, the likelihood for succeeding the grasp is computed. Particularly, the surface representation is provided in an uncertainty-aware fashion, while the authors rely on fixed values of heuristic estimates. Essentially, the uncertainties are not updated during operation, which is a discrepancy to our technique. Apart from that, Rubert et al. in [69] elaborate on existing machine learning methods by comparing their performance in predicting the success probability. However, the authors refer to known objects in simulation such that the comparison with the grasping experiments conducted in this thesis is not reasonable. Importantly, the binary optimization presented in Section 5.1.3 is extended by heuristic uncertainty estimates of grasp metrics that define grasp candidates. Thereby, a technique to selecting grasp candidates with the goal of enhancing the rate of successful grasps is developed. Importantly, it is assumed that the grasp generation is completed such that the method focuses solely on the selection.

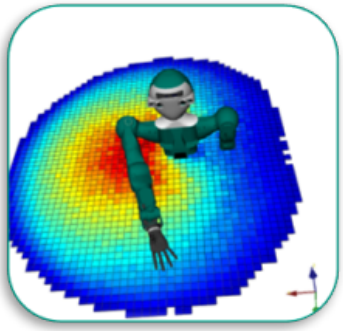
5.1.2 Grasp Metrics and Uncertainties

To apply the binary uncertainty-aware optimization introduced in Section 5.1.3, each grasp candidate must be expressed by the system's variables or metrics. Furthermore, following assumptions will be made for the remainder of this section:

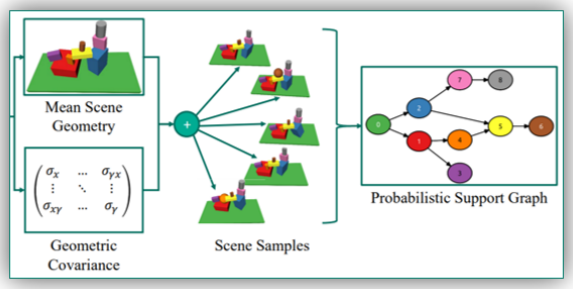
1. The robot is provided with visual information of the objects that are offered for grasp selection.
2. Each grasp candidate g is fully described by four grasp metrics $m_1, ..., m_4$.
3. The grasp metrics $m_1, ..., m_4$ are independent.
4. All uncertainties of the grasp metrics are modeled as Gaussian distributions: $u_{m_i} \propto N(\mu_i, \sigma_i^2)$.
5. The data set collected with the humanoid robot ARMAR-6 serves as the population for the statistical studies. This means that this data set is assumed to contain all possible grasp candidates.

Here, the grasp metrics are obtained from visual perception. To be specific, a method for Probabilistic Action Extraction and Fusion (PAEF) published in [64] is applied. Briefly, PAEF is provided with point cloud data of the scene and yields clusters that consists of points. These clusters are denoted as Supervoxels. By referring to these Supervoxels, a coordinate frame is established. This coordinate in turn enables to track action hypothesis of the scene by referring to multiple observations. Apart from the PAEF, an unscented Kalman Filter (UKF) is employed to determine the covariance of the action's executions pose. In addition a Hidden Markov Model tracks the number of scene observations, that allow to estimate action hypotheses. Based on these, a further measure that is included in the characterization is the existence certainty ϵ. Essentially, this value denotes the probability for existence of this action.

The four grasp metrics that characterize each grasp candidate g are defined in the following. In addition, it is specified how the respective uncertainties are derived.



(a) Manipulability



(b) Support Graph: The support relations are obtained by applying the PAEF.

Figure 5.2: Illustration of the manipulability and the support relations.

Grasp Height (h): The height of a grasp candidate h reflects its shortest distance to the floor. Logically, the value of h is larger for objects that lie on the top of the clutter. The corresponding Gaussian distribution that is defined by the mean value and the standard deviation of the height that reflects the uncertainty is directly obtained by the PAEF.

Distance to Object Center (d): The distance amounts to the distance between the grasp candidate to the center of the bounding box of the object closest to the grasp pose. For the calculation, a segmentation of the scene is combined with above described PAEF. Here, the mean value μ_d corresponds to the distance while the variance σ_d^2 is approximated by $10\,\%$ of the bounding box.

Support Relations (s): This metric indicates the number of objects that are supported by the point cloud segment closest to the grasping pose. It is computed by means of the PAEF and the segmented scene. The parameters for the Gaussian distribution μ_s and σ_s^2 are determined via the probabilistic support graph presented in [6]. Briefly, a support graph provides the information on the support relations between the objects in the scene. Specifically, a probabilistic support graph provides probabilities that can be used to specify the uncertainty parameters in this thesis.

*Manipulability (a):*The manipulability denotes how freely the robot end-effector can move at a given position. Details on the derivation of this quality metric are provided in [76]. Especially, a is determined by referring to the grasp pose from the *PAEF*. In order to calculate the uncertainty parameters μ_a and σ_a^2, random configurations of the robot joints are sampled. The manipulability is defined for a voxelized workspace and is updated according to the sampled parameters. Therefore, this metric assigns higher scores to grasps that can be easily reached by the robot.

5.1.3 Optimization Function and Uncertainty Incorporation

To this end, above defined grasp metrics are expressed via Gaussian distributions according to the respective uncertainty parameters μ_i and σ_i^2, where i indicates the metric. The goal in the grasp experiments lies in maximizing the rate of succeeded grasps. Here, grasp attempts that are grasped by the humanoid robot ARMAR-6 are categorized as succeeded while those that are not successfully completed are classified as failed grasps. Following the optimization method presented in Section , this problem is addressed by framing it

as an quadratic unconstrained binary optimization problem (QUBO). Due to the binary distinction between successful and failed grasps, where the goal lies in improving the rate of succeeded grasps, employing this optimization method is reasonable. Specifically, it is desired to identify the most promising grasp candidate among the suggested ones. Accordingly, the cost function is defined as

$$z_{Q,g}(\alpha, m_i, \beta) := \alpha \cdot \sum_{i=1}^{n} f_{tot,i} \tag{5.1}$$

where the optimization goal lies in finding the grasp g_{opt} with

$$\vec{g}_{opt} = \arg\max z_{Q,g}(\vec{g}). \tag{5.2}$$

Above equation corresponds to the specified form of Equation (4.1). Particularly, $z_{Q,g}$ differs in the constants α: these are introduced for the selection of the grasp candidate due to application-specific reasons: As described above, the existence certainty ϵ is not modeled as a Gaussian distribution, but remains as a scalar instead. However, it states the probability for the existence of a grasp candidate at a given position: If this probability amounts to $\epsilon = 0$, the grasp candidate is assumed to not exist and the calculation of the cost function is not performed. In the given application, the cost function is referred to by the term *score* since higher scores are linked to grasp candidates that are preferred, while those with low scores should ideally be neglected by the robot. In order to perform the experiments, a reference data set is collected under real-world conditions. This data set serves as the basis to perform the optimization. In particular, above defined grasp metrics *h,d,s* and *a*, that is, the respective Gaussian distributions, and the scalar values for the existence certainty ϵ are monitored during run time and saved. Afterward, the data set is divided in two subsets: One data set that contains all succeeded grasps and the remaining one with all failed grasp attempts. As a result, each of the grasp metrics can be characterized by two probability distributions (succeeded and failed grasps). These distributions serve as the basis for the uncertainty-aware optimization.

5.1.4 Experimental Setup: Humanoid Robot Grasping with ARMAR-6

In order to study the validity of the above suggested optimization function, grasping experiments are performed with the humanoid robot ARMAR-6 [7]. As specified by assumption 4.1 in Section 5.1.3, a data baseline data set that acts as a basis for the optimization must be collected.

To this end, more than 1100 grasp attempts on unknown objects are conducted under real-world conditions by employing ARMAR-6. Specifically, the experiment corresponds to a box emptying experiment, where the unknown objects are given by five plastic pipes, four boxes and two metal pipes. These objects are randomly distributed in the box as shown in Figure 5.1. The grasp candidate extraction is performed by the PAEF method introduced above. In addition, the reachability of the grasp candidates and its inverse kinematics are checked prior to performing the grasp. Importantly, all of these grasps are selected randomly. After executing the grasp, the robot placed the object is placed back into the same box, thereby introducing random changes to the scene. In case the scene was not changed for several attempts, for example because no object was grasped successfully

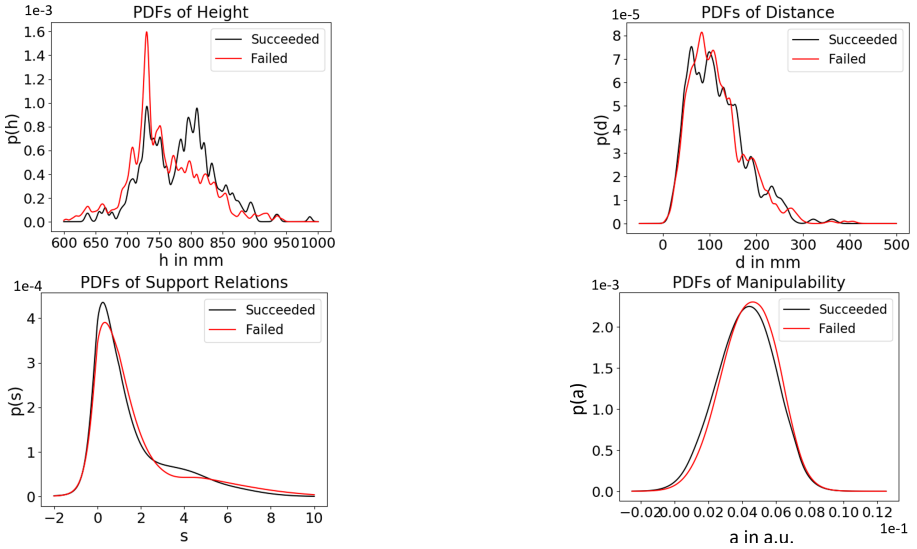


Figure 5.3: Probability density functions (PDFs) for the succeeded (black) and failed (red) grasp attempts performed in the random grasp selection experiments.

or the same object was grasped repeatedly, the objects were randomly rearranged by a human operator.

For the executed grasps, above defined grasp metrics m_1 to m_4 as well as the existence certainty ϵ are determined. Thus, each grasp is characterized by four Gaussian distributions and one scalar value for the existence certainty. Most importantly, it is stored whether the grasp attempt succeeded (s) or failed (f). As mentioned above, two curves are obtained for each grasp metric (one corresponding to the succeeded and the second one for the failed attempts) as shown in Figure 5.3. Specifically, the curves in the Figures are obtained by building the sum of the Gaussian distributions that represent the executed grasps and normalizing the result to obtain the probability density functions. From 932 randomly executed grasp attempts, 304 were successful while 628 failed. Hence, the success rate in the random grasp selection amounts to $32.6\,\%$.

5.1.5 Uncertainty-aware Sensitivity Optimization

To validate whether the rate of successful grasp attempts increases by incorporating the uncertainty-aware sensitivity optimization function, a second round of experiments is conducted. Importantly, the experiments are performed with the same experimental setup under the same conditions. The only difference lies in the grasp selection: Instead of randomly choosing them, the robot executes the grasp with the the highest score evaluated by Equation (4.5). To evaluate this equation, the Kullback-Leibler divergence between the curve for the succeeded and failed curves is determined. In fact, the Kullback-Leibler divergence reflects the sensitivity. The influence of one metric on the success of a grasp candidate is modeled by the difference in the curves, that is captured by the KL divergence.

However, in contrast to the experiments in Section 5.2, the sensitivity measure is static. This means that it is computed one time by referring to the baseline data set such that all grasp candidates are assigned to the same KL divergence values. Since it is desired to consider the individual characteristics of the grasp candidates as well, the likelihood of

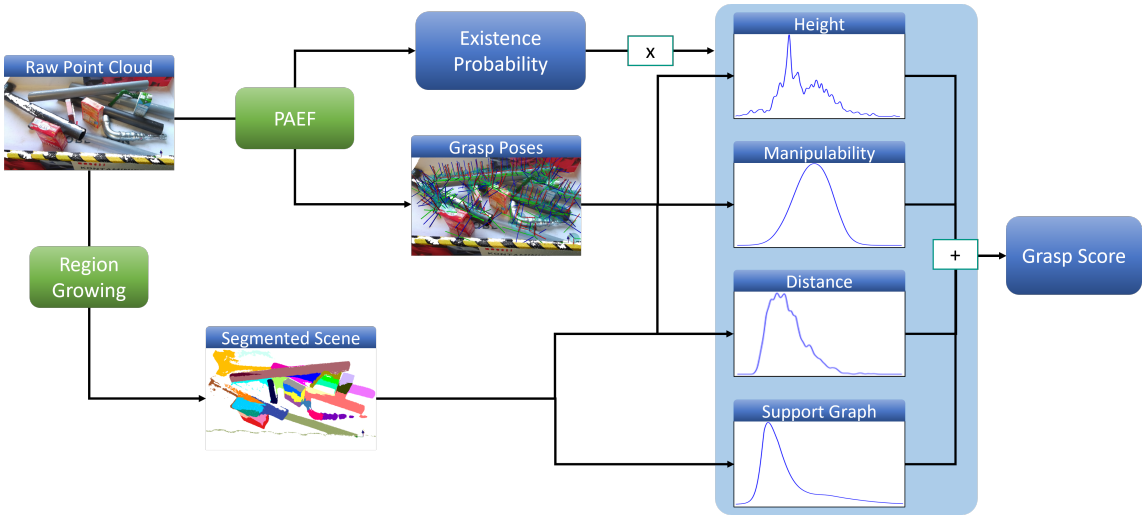


Figure 5.4: The uncertainty-aware sensitivity optimization is performed on the basis of a raw point cloud. After applying a region growing algorithm and the probabilistic action extraction, above described grasp metrics are computed based on the grasp poses and the segmented scene. In particular, these metrics are modeled as Gaussian distributions that yield the PDFs on the right hand side. Finally, the grasp score z is calculated. Figure taken from [10].

each grasp candidate to belong to the succeeded and failed grasps is incorporated as was specified in Equation (5.1). This means that each grasp candidate is assigned to its z that acts as a ranking score. The robot executes the grasp with the highest z-value.

5.1.6 Results and Evaluation

Apart from the baseline data collection, additional 187 grasps are performed by selecting the grasps according to the uncertainty-aware sensitivity optimization in Equation (5.2). In this setting, 138 successful grasps were carried out that corresponds to a success rate of $73.8\,\%$. Hence, an enhancement of $43.8\,\%$ was reached by selecting the grasps with the suggested uncertainty-aware optimization function. As can be seen in Figure 5.3, the height metric is by far most influential: Here, the discrepancy between the succeeded and failed grasps, and thus the KL divergence is dominant which can be seen in Table 5.1. In particular, the influence of the remaining metrics seems negligible on first sight. However, in specific regions, the distance becomes influential. While this can be roughly recognized in the plots, dedicated statistical analyses are performed to gain a deeper understanding on the relationship between the metrics.

5.1.7 Evaluation and Results

Statistical Analyses: Influence of Metrics

From statistical viewpoint, the data collected by conducting 187 grasp attempts corresponds to a rather low number of samples. Thus, stating the confidence of the following analyses is reasonable. In first place, it is investigated whether the metrics are correlated with each other. Since it is assumed that the metrics are independent, it is studied how

Table 5.1: Kullback-Leibler divergences of the metrics obtained from random grasping.

Metric	Kullback-Leibler Divergence
Height h	0.460
Distance d	0.034
Support Relations s	0.014
Manipulability a	0.010

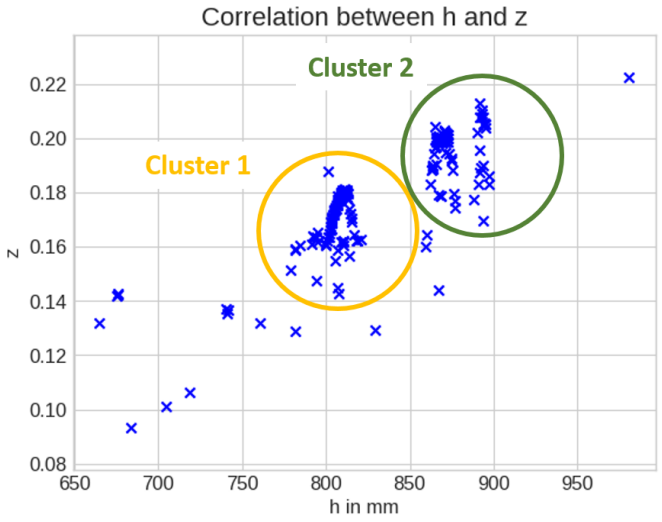


Figure 5.5: The correlation between the Height metric h and the grasp score z for the grasps performed via the uncertainty-aware binary optimization.

the each single metric is related to z. First, it is of interest how the metrics correlate with the score z. This is quantified by means of the *intercorrelaction value (IC)*. Specifically, the IC is calculated for all four metrics. As shown in Table 5.2, the correlation between the height and the ranking score z is dominant. In fact, the IC value amount to more than twice of the correlation with the distance d.

In Figure 5.5, two clusters can be recognized. Especially, the IC values between the score and the height are not linearly correlated in these clusters. Thus, the IC values are determined for these clusters separately. According to the findings in Table 5.2, the correlation between d and z as well as the IC value between a and z become negative. In particular, the amount of the IC value with the distance increases significantly. This indicates that for small differences of the height in the regions of Cluster 1 and Cluster 2, the distance to the object center d also plays a relevant role for the grasp selection. Interestingly, the global IC value of the distance reflects that objects with larger distances to the object center are preferred. However, the sign obviously changes in Cluster 1.

Generally, these results are not sufficient to derive conclusive statements concerning the preferences of the metrics. This draws from the missing knowledge on the dependency between the metrics: For example, the significant dominance of the height metric might automatically lead to the selection of objects with larger distances to the object's center. In addition, the confidence of the obtained findings must be accompanied by the statistical uncertainty since 187 grasps correspond to a relatively small data amount.

Table 5.2: Correlation and intra-cluster correlation (IC) values between the final score and the metrics.

Metric	Corr. with z	IC Cluster 1	IC Cluster 2
Height h	0.8267	0.3592	0.2236
Distance d	0.2998	**-0.2967**	0.1387
Support Rel. s	-0.0982	-0.1424	**-0.1902**
Manipulability a	0.0345	**-0.0749**	-0.0166

Influence of the Uncertainties and the Optimization Performance

Apart from correlation studies to explore the dependency between the metrics and the score, the explainable nature of the optimization method enables to analyze also the influence of the uncertainties. As described above, the uncertainties of all metrics are assumed to follow the Gaussian distribution. Now, the goal is to analyze the influence of uncertainties on the success rate. Of course, these analyses are limited since experiments only were conducted by considering one uncertainty constellation. Nevertheless, rough conclusions can be derived by applying statistical tests on the baseline data set, where the grasps were selected randomly.

In the first step, the bootstrapping technique introduced in Section 2.1 is applied to the set of succeeded grasp attempts. By relying on the Central Limit Theorem, 10.000 samples are drawn from the distribution of the succeeded grasps 10.000 times. This process is performed for each metric. As a result, 10.000 histograms are obtained with 10.000 data points each for each metric. The resulting Gaussian distribution of the mean values provides information on the mean value μ_s for the respective metric. Importantly, these Gaussian distributions serve as the basis for the statistical tests to find out whether one grasp candidate would fall into the group of succeeded or failed grasp attempts.

In the second step, the Gaussian distribution describing the single grasp candidates are modified by varying the standard deviation σ. Logically, this leads to a different PDF representing all grasps since this distribution is obtained by summing up and normalizing the distributions of all single grasps as shown in Figure 5.6.

In the third step, samples are drawn from the modified distribution. By applying the p-test explained in Section 2.1 with respect to the metrics' distributions estimated in the previous step, it is explored whether the grasp candidate belongs to the succeeded or failed attempts. In doing so, the approximated success rate is determined.

Obviously, the obtained results merely correspond to approximations. Among other factors, the amount of data collected for the baseline data set affects the reliability of the obtained result. These steps are performed for three values of σ for the height h and the distance d since these two metrics influence the grasp selection in a dominant manner. The results are presented in Table 5.3.

On first glance, it might seem surprising that lower uncertainty values seem to result in lower success rates: in contrast to the obtained grasp success rate of $73.80\,\%$, the success rate decreases to $57.63\,\%$ when reducing the variance to $0.25\,\sigma$. However, it must be considered that these results must be interpreted in the context of the baseline data set. Importantly, the table does not provide any statements on the expected success rates for a certain uncertainty constellation in the metrics. Instead, the values for the success rate indicate which σ-value approximates the uncertainty best: Logically, a perfectly correct

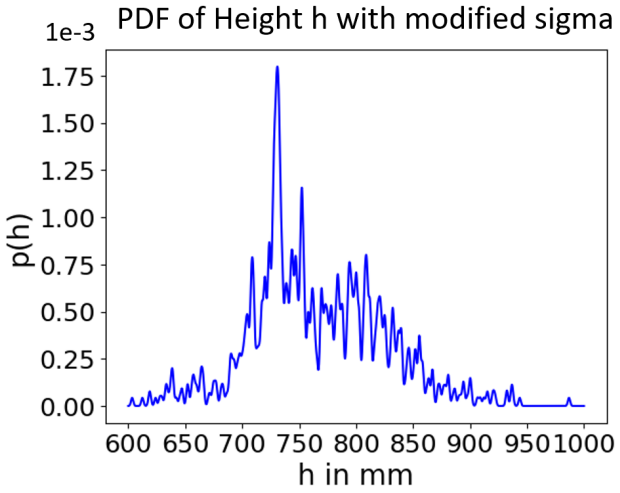


Figure 5.6: The distribution of the height metric h for a lower uncertainty σ.

uncertainty model, regardless of whether it is Gaussian with a σ-value of $0.10\,\sigma$ or a more complex behavior, would yield the maximum success rate of $100\,\%$.

To this end, the findings in Table 5.3 do not only support the uncertainty-aware optimization, but also the heuristic uncertainty estimates. According to the second line of this table that contains the expected success rates for difference distance values, the expected σ value may have been slightly overestimated. Especially, the success rate for $0.25\,\sigma$ is higher than the success of $73.80\,\%$ that was obtained by the experiments.

Apart from the approximation of the grasp metrics' uncertainties, it may be interesting to study the correlations between the metrics. In particular, the knowledge on these might simplify to study and interpret the causality between the metrics and potential disturbances among them in more detail. Thus, it can be concluded that performing the grasp selection in an uncertainty-aware manner by treating the grasp metrics as probability distributions has led to a notable improvement in the grasp success rate. However, it is worth noting that the validation, and thus the evaluation was carried out merely for one experimental setting such that applying this optimization technique is recommended for further robot systems with binary optimization goals.

Table 5.3: Expected success rates for modified σ.

Metric	σ	0.50σ	0.25σ	0.10σ
Height h	73.80 %	64.05 %	57.63 %	60.19 %
Distance d	73.80 %	74.01 %	76.04 %	70.96 %

5.2 Uncertainty-aware Optimization with NN Classifiers

This Section addresses the uncertainty quantification of a robot system consisting of both hardware tools with known metrological uncertainties and NN classifiers. To this end, an experimental setting with an industrial robot arm is considered. Here, the goal of the robot is to correctly classify image data while performing movements. Apart from

the NN that performs the classification, uncertainties arise from the robot's velocity and hardware components as for example the employed cameras. Hence, the determination of the uncertainty of the entire system necessitates to bring together the metrological uncertainty and such of the NN classifier(s).

Specifically, the technique explained in Section 3.2.2 is refined with respect to the application. As will be elaborated below, the idea lies in combining the probabilistic robustness of NN classifiers with the metrological uncertainty formalism by exploiting the analogies. Due to the missing generality of the uncertainty definition of NNs in existing literature, the quantification of a unified uncertainty measure is performed via the sensitivity: In fact, the sensitivity is tightly related to the metrological uncertainty. At the same time, the probabilistic robustness suggested by Mangal et al. in [55], shows meaningful similarities with the sensitivity.

In the following, this analogy is drawn by referring to the definition of the robustness in Equation (2.3) in Section 2.1.2 and the metrological uncertainty definition that was formalized in a generic manner in Section 3.2.2 can be refined for a robot application. As will be explained in the following, drawing this analogy is also bound to limitations. Practically, this means that the determination of the combined uncertainty with the proposed technique places restrictions on the application as will be specified in the next subsections. The validation experiments are performed on two data collections MNIST [21], that contains digits and GTSRB [72] with traffic signs. Specifically, it is of interest whether the rate of correct classification results of images in a robot application can be enhanced when adding the uncertainties into the optimization pipeline.

To this end, the optimization goal lies in the minimization of incorrect classifications

$$\min r_v := \frac{N_v}{N_{tot}}, \tag{5.3}$$

where N_v stands for the total number of samples and N_v the number of incorrect classifications. In actual fact, N_v can be expressed via the correct classifications N_c by $N_v = 1 - N_C$. However, this relationship does not contribute to the online detection of incorrect classifications. Especially, the problem falls into the the category of binary optimization since it is distinguished between the correct and incorrect classifications.

Prior to specifying the assumptions and formalism, a brief overview on existing uncertainty notations is provided. In particular, combining the probabilistic robustness of NNs with the metrological uncertainty is motivated by pointing out the discrepancy to the epistemic and aleatoric uncertainty introduced in Section 2.1.2. Afterward, the experimental setting of the robot system is introduced. By specifying the optimization goal and describing the incorporation of the uncertainty as explained in Section 5.2.5, the calibration step is described in detail. Finally, the correctness of the obtained and the applicability of the presented method to robot applications is discussed on the basis of two data sets.

5.2.1 Related Work: NN Uncertainties for Robot Systems

Within recent years, employing NNs in robot systems has become attractive, especially for data-driven applications. Due to their ability to process huge amount of data sets efficiently, the use of NNs often leads to an enhanced productivity. To date, the employment of NNs in safety-critical environments is not recommended due to the missing understanding on their uncertainties: In contrast to hardware components, manufacturer specification

that provide information on the uncertainties of NNs do not exist. However, a measure that indicates the possible deviation of the expected outcome is crucial to facilitate the use of NN-based tools in real-world applications, especially in those that may underlie risks. To this end, the aleatoric and epistemic uncertainties are introduced in the learning domain as described in Section 2.1.2. Essentially, the aleatoric uncertainty is supposed to capture the uncertainty due to the input data while the intrinsic uncertainty of the NN is covered by the epistemic uncertainty. However, as pointed out by Huellermeier et al. in [43], absolute notations as equations to capture these uncertainties do not exist to date. This makes it challenging to derive a technique to quantifying the uncertainty of an entire system, that employ NNs. Specifically, it is difficult to study how the epistemic and aleatoric uncertainties influence each other since formal descriptions do not exist.

The survey paper published by Gawlikowski et al in 2019 [33] presents a variety of uncertainty determination techniques for DNNs reaching from ensemble methods as published by Martinez et al. and Reich et.al ([15], [68], [56], [40]) that suggest the generation of multiple models and the subsequent combination of predictions into one output to Bayesian methods that represent the weights of NNs as distributions to model the epistemic uncertainty as in the contributions ([15], [44], [46], [25], [66], [22], [73]). Briefly, the presented uncertainty quantification methods in the survey are classified in four categories: single network deterministic methods, Bayesian methods, ensemble methods and test-time augmentation methods. In addition, each of these categories are further divided in subgroups as shown in Figure 5.7.

In the case of single deterministic methods, the uncertainty is quantified by performing one single forward pass in a deterministic network and applying external techniques for the uncertainty calculation. In contrast, Bayesian methods refer to all types of stochastic Deep NNs, where conducting several forward passes on one sample yield different outputs. On the other hand, ensemble methods suggest to combine the predictions of a variety of deterministic networks at inference. Lastly, test-time augmentation methods address to augment the input data during test time. In doing so, several predictions are generated. These are used to assess the prediction's certainty, and reversely the uncertainty. While these techniques enable to propagate the uncertainties to the NN output, they do not provide information on how these correlate with the aleatoric uncertainties in the input data. Thus, it is not clear how distortions in the input data may affect the epistemic uncertainty, and how possibly existing dependencies between these uncertainty types influence the NN output.

On the other hand, the robustness of NNs has been defined and introduced by several works as Hein et. al. in [39], Pauli et al. in [62] or Cisse et al. in [19]. In contrast to the epistemic and aleatoric uncertainty, the robustness addresses the determination of the metrological sensitivity. By relying on the assumption that the space of the input data can be modeled by an l_p-ball, the maximum distortion in the input data to still obtain an unaltered NN output is estimated. This definition equals the formalization of the metrological sensitivity. In fact, the robustness can be seen as one specific case of the sensitivity: While the sensitivity in metrology does not place limitations on the input data space, the robustness specifically addresses NNs. Here, several contributions as [80], [62] provide methods to quantify the lower bound on this distortion in the context of adversarial attacks. In particular, these works focus on the development of methods to effectively detect designed security attacks.

One central goal of this thesis is to develop techniques to quantifying the uncertainty of entire robot applications that requires to unify the uncertainty notations of employed

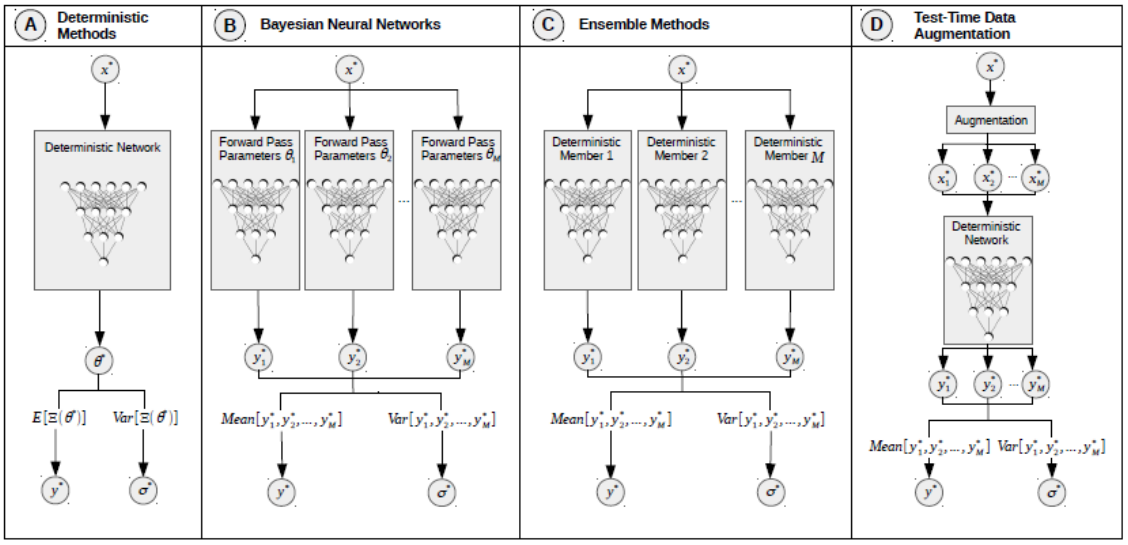


Figure 5.7: Overview of uncertainty quantification methods presented and discussed in the survey of Gawlikowski et al. in [33].

system components. As emphasized so far, this thesis builds upon the metrological understanding of uncertainties of the Guide [26]. Since the robustness shows similarities to the metrological uncertainty, it is suggested to exploit this analogy. In doing so, the findings from the above mentioned works, namely the conclusions drawn by Lyu et al. in [54], Weng et al. in [81] and Weng et al. in [80] will be used to state requirements and limits on the applicability of the method.

5.2.2 Experimental Setup: Combining Measurement Uncertainties with NN Robustness

The validation of the reinterpretation of the parameters in the definition of the probabilistic robustness explained in Section 5.2.5 is performed in a real-world experiment: An industrial robot arm is equipped with an intel RealSense D435 camera device. The robot's task lies in performing point to point movements while detecting image data and classifying them. Once the classification uncertainty exceeds a pre-defined threshold, the robot must reduce its speed.

Importantly, the classification uncertainty does not depend solely on the NN classifier, but also on the employed camera. Apart from that, the uncertainty is likely to depend on the lightning conditions and additional environmental disturbances. It will be thus aimed at combining the metrological uncertainty with the probabilistic robustness of NNs to finally quantify the classification uncertainty. After monitoring how this uncertainty develops during system run time, a parameter setting that minimizes the rate of incorrect classifications is derived. Especially, the experiments are performed in two settings: First, it is assumed that ground truth data points exist. In fact, this is possible for the experiments performed in this thesis since the test and training data sets are provided for both MNIST and GTSRB.

In the second step, this assumption is dropped such that the validation is conducted on the basis of disturbed input images. For the remainder of this Section, x_0 is referred to by the term *reference data*. In contrast, the *input sample* corresponds to the image data that is

actually recorded during run time. In particular, x' is disturbed by the uncertainties of the system components such as the camera or possible robot movements. In both experiments, the rate of incorrect classifications given by

$$r_v = \frac{N_v}{N_{tot}} \tag{5.4}$$

is tracked for varying k-values and radii δ. These measurements are performed at three speed settings of the robot and a frame rate of $f_p = 30\,fps$ of the intel RealSense D435 camera. In particular, the calibration in each experiment was performed on 40 input data points each. Figure 5.9 illustrates the obtained results. Especially, conclusions can be drawn on the uncertainties of the employed tools and the influence of environmental disturbances.

5.2.3 Assignment of Parameters in the Lipschitz Function

This Section refers to the binary optimization with linearizable NNs introduced in Section 5.2.5. In order to facilitate the applicability to real-world systems, a proper calibration must be performed. In fact, the goal of this calibration lies in identifying a reasonable constellation of the reinterpretated parameters. In general, it is aimed to develop an uncertainty-aware optimization technique, that logically places the requirement on the technique to be sensitive to uncertainties. As presented in Section 5.2.5, the dependencies of the parameters in the Lipschitz equation on the application places a challenge in identifying the ideal parameter constellation. To this end, a two-step calibration is proposed: In the first step, the ground truth data point must be provided or given by the corresponding data sets. In actual fact, this does not apply to the majority of applications.

However, ground truth data points may be at least partly available, which is assumed in this Section. In the first calibration step, the Lipschitz equation is evaluated by referring to the ground truth data points. In contrast, the ground truth data is replaced by static recordings. Again, the evaluation of the Lipschitz equation is conducted as elaborated below. Briefly, comparing the results of these two calibration experiments enables to infer the uncertainties and to finally consider them in the optimization procedure. In the following, the reinterpreted parameters will be specified for the above described application scenario. Afterward, the two-step calibration and the derivation of the uncertainty will be explained in detail by referring to the respective findings. Generally, the optimization goal in this Section lies in minimizing the rate of incorrect classifications by identifying the optimal parameter constellation in the reinterpreted Lipschitz equation

$$\Pr((\|f(x') - f(x)\| \leq k * \|x' - x\|) \mid \|x' - x\| \leq \delta) \geq 1 - \epsilon \tag{5.5}$$

While the generic reinterpretation was provided in Section 5.2.5, it was emphasized that the parameter specification must be performed individually for each system. For the setting of the validation experiments, the reinterpretation is given as follows:

$\|x - x'\|$: By following the description in Section 5.2.5, this term is assigned to the distance in the input data points, where it is assumed that the input data can be modeled by an l_p-ball. Here, the input data corresponds to the images that are captured by the intel RealSense D435 camera device. Particularly, the assignment of x depends on the case whether ground truth data is available or not. In the case where the ground truth data points are available, x describes the ground truth

data while x' corresponds to the data points subject to uncertainties. In case the training data of the NN is available, these can be considered as x. If no ground truth data points exist, the data for x must be generated. In the given experiment, static recordings of the images are a reasonable candidate for x: Especially, the uncertainty of the camera is provided in the manufacturer specifications such that subtracting it from static recordings enables to approximate the ground truth.

δ: This parameter is modifiable as explained in 5.2.5 and places a threshold for the neighborhood of x. In this example, δ is estimated via the uncertainty of the camera device and the latency due to the limited frame rate that especially becomes relevant for high robot speeds. The exact value of δ is determined in the calibration step.

k : Since the uncertainty of those components, where the uncertainty is provided in data sheets or modeled in a simulation, is covered by δ, k logically captures how the NN output reacts on the uncertainty inherent to x'. Briefly, k reflects how sensitive the NN is with respect to the distortion in the corresponding input data point.

$\|f(x) - f(x')\|$: In the application example, this this term is computed via the distance of the NN output vectors. In the ideal case, the classifier would perform perfectly such that the probability for a resulting top-1 score would amount to 100%. Since this cannot be realized in real-world systems, $\|f(x) - f(x')\|$ introduces a tolerated amount of deviation. For example, failure limits can be expressed by this term. After each classification, evaluating the Lipschitz equation would then correspond to checking whether a pre-defined failure limit is violated. Especially, the robot movement can be adapted according to the obtained result.

ϵ: In the domain of adversarial attacks, ϵ is introduced to relax the requirement on the robustness. Practically, it means that the robustness may not be achieved for all data points within the l_p-ball. Therefore, it can be used as a confidence measure for the reinterpretation: The confidence intuitively captures the reliability of the findings.

The system pipeline that builds upon this reinterpretation is shown in Figure 5.8. The respective algorithm is shown in Algorithm 2.

Case 1: x_0 as Ground Truth Data

In the first case, the Lipschitz equation is computed by selecting the reference data x_0 from the set of the test data set. These samples are treated as undisturbed ground truth data. Practically, this means that the uncertainty in x' can be determined in straightforward manner. It can be recognized in Figure 5.9, that the curve representing the rate of incorrect classifications, specified by the Lipschitz equation, drops with increasing k-values. This matches the expectation: since the requirement on the tolerated violations of the Lipschitz equation becomes weaker with larger k-values, the rate of incorrect classifications automatically decrease, especially for a small radius $\delta = 0,1$. Since the value of k does not depend on δ, it is plausible that the curve of r_v shows a steep drop at small k-values.

In particular, increasing the radii of the l_p-ball, where x' is drawn from, yields to a plateau of r_v, but only for the MNIST data set. For the traffic signs, the increase of δ from $\delta = 0.1$ to $\delta = 0.2$ does not seem to influence the development of r_v. This observation can be explained with the uncertainty that due to the camera, environmental disturbances and the NN: Despite the increase of k, the rate of violations cannot be reduced for a certain

Algorithm 2 Quantification of System Uncertainty

Input: measurement data x_i; reference input x_0; NN $f(\cdot)$;
sensor uncertainties u_{sensor}; system function u_{system};
confidence level σ = 1-ϵ; safety limit λ; time steps t.
Output: probabilistic robustness check(bool); system uncertainty u_{system}; safety limit check (bool).

```
for i ≤ t do
    dev[i] ← ‖x_0 − x_i‖
    if dev[i] ≤ δ then
        Check L_ε for ‖x_0 − x_i‖
        if L_ε == 1 then
            Calculate u_system
            print u_system
            map u_system on λ
        else
            Stop system
        end if
    end if
end for
```

area. Since x_0 stands for the ground truth data, it is not clear whether it is merely due to the uncertainty in the employed technical components. For example, the discrepancy $|||x_0 - x'|||$ may be large due to the different lightning conditions during the recording of x_0.

Hence, the only valid conclusion for this plateau is that the area, where N_v is constant, draws from a constant uncertainty value or error in the data perception and the processing. Furthermore, comparing the curves of N_v for both data sets demonstrates that the rate of violations can be decreased at smaller k-values for the GTSRB data set than for MNIST. This may be attributed to the conditions, under which the test data was collected in the case of GTSRB: Since traffic signs are supposed to be classified correctly at high robot speeds, the classifier was likely tested against more disturbed data. Thus, the MNIST data set is likely to behave more sensitive to disturbances.

Case 2: x_0 as Static Recordings

For the second part of experiments, static recordings of the 40 image data samples serve as x_0. Importantly, the recording process is conducted with the same camera intel RealSense D435. In addition, the recording must be completed prior to the experiments. In fact, the Lipschitz equation is evaluated in analogy to the previous case and, as described above, by computing the pixelwise difference to the data points collected during run time. By doing so, the violations are determined with respect to the static recordings. Again, the robot is stopped once the rate of violations exceeds the specified value as in the previous Section. As can be seen in Figure 5.9, using static recordings as x_0 leads to significantly wider plateaus. Practically, this means that the violations cannot be reduced by weakening the requirement on the Lipschitz condition. This hints at the constant uncertainty of the camera intel RealSense D435 with respect to k: in contrast to the above case, where x_0 was represented by the static recordings, the distance $|||x' - x_0|||$ is smaller in this

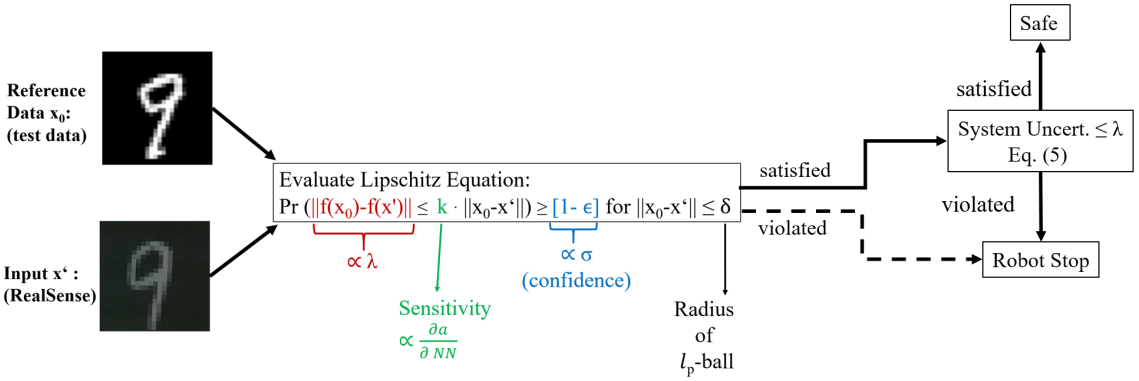


Figure 5.8: The reinterpreted Lipschitz equation is validated in two settings: First, x_0 corresponds to the ground truth data. In the second case, static recordings captured with the intel RealSense D435 camera are treated as x_0. To validate the reinterpretation of the variables, it is studied whether the rate of incorrect classifications is reduced via considering this equation. Specifically, a robot movement stop is initiated once the rate of violations on the probabilistic Lipschitz equation exceeds the critical limit.

case. Since x_0 was collected with the same device, the difference between x' and x_0 is significantly smaller.

Reversely, this means that comparing the plateaus of both results with each other enables to estimate the camera uncertainty: In the first case, the uncertainty of the camera leads to the fact that the distance between x_0 and x' are that large that the rate of incorrect classifications is reduced at a small k. Because the main goal of this calibration lies in identifying an ideal parameter constellation, the region sensitive to the camera uncertainty becomes interesting. Obviously, this region is given in the area of $k \in [4.2; 10.3]$. In this region, the plateau is larger such that a higher value of k is necessary to satisfy the Lipschitz requirement. By assuming a symmetrical camera uncertainty that is linear with respect to k, it is concluded that the difference in the plateau width enables to compute the uncertainty of the intel RealSense D435, and that above specified range for k is a reasonable region. In fact, this region does not only reflect the sensitive region for the camera, but also the remaining uncertainty sources that may have been different in the first and the second calibration step. Also, the plots demonstrate that this effect is clearly visible for both data sets.

However, the speed in the robot movement seems to influence the classifications more significantly for the data set MNIST while the traffic signs classifications do not seem to underlie dependencies on the velocity. Again, this may be attributed to the fact that the traffic signs are supposed to be stable against velocity disturbances since these should be ideally recognized at high velocities of a vehicle. Thus, it is likely that the respective test data test contains the corresponding images, leading to no significant differences in the curves of N_v at different robot speeds.

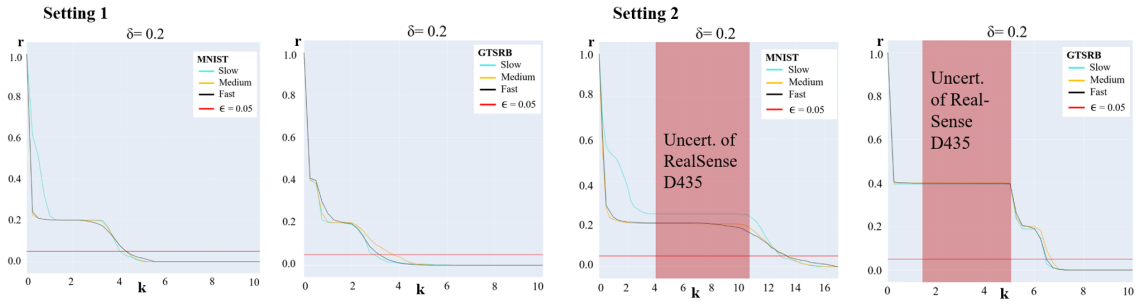


Figure 5.9: The calibration experiments are performed for two settings: In the first case, samples from the training data are treated as x_0 while in the second setting, static recordings with the intel RealSense D435 serve as the reference data x_0. In all settings, the l_p-ball radius was set to $\delta = 0.2$. In both cases, the left plot refers to the MNIST data set while the plot on the right hand side shows the result on the GTSRB traffic signs.

5.2.4 Validation Experiments: Classification with the Reinterpreted Parameters

With the reinterpreted variables and the estimated region for the parameter k, it is now aimed at studying how this suggested evaluation of the Lipschitz equation contributes to the reduction of incorrect classifications. Therefore, a third round of experiments is performed, where the radius of the l_p-ball is set to $\delta = 0.2$ and the k-value to $k = 11.8$. Specifically, this region does not underlie the plateau-effect due to the camera uncertainty. Thus, it is sensitive for capturing additional undesired disturbances such that it is considered as suitable for uncertainty-aware image classification. Basically, the experimental setting and the environmental conditions remain as in the calibration phase. One difference is that the robot must detect and classify five images during its horizontal point-to-point movement. In the ideal case, the threshold of $||f(x) - f(x')||$ can be held such that the robot now does not stop unless external disturbances or unexpected behaviors of the classifier occur. To explore whether the classifier reacts appropriately, occlusions are introduced randomly: The image data is partly covered by objects. In addition, the lightning conditions are varied. The results are shown in Table 5.4.

5.2.5 Results and Evaluation of the Binary Optimization with NN Classifiers

In contrast to the binary optimization in Section 5.1, a technique addressing the NN classifiers was derived by reinterpretating the variables and parameters in the Lipschitz equation. According to the findings, the reinterpretation is valid such that an uncertainty-aware optimization can be performed effectively by the derived method, especially for systems that consist of NN classifiers. In fact, this technique can be extended to any kind of NNs that are linearizable as elaborated by the contribution of Zhang et al. in [82]: Given that the above stated assumptions can be held, the proofs on the validity of the Lipschitz equation for the detection of adversarial attacks can be directly transferred to the above described use cases. Practically, this means that the developed reinterpretation, and thus the uncertainty-aware optimization with the respective Lipschitz condition can be applied to robot systems that employ linearizable NNs.

In the context of the optimization performance, the obtained results convincingly demonstrate that the rate of incorrect classifications can be reduced: According to Table 5.4, the detection rate of incorrect classifications can be enhanced from $50.0\,\%$ to $83.3\,\%$ by reinterpreting the variables in the Lipschitz equation, that corresponds to an improvement of $33.3\,\%$. However, this is only valid for applications where the environmental conditions are constant: Once small unexpected changes occur, the system is stopped. Especially, since the Lipschitz equation is very sensitive to the radius of the l_p-ball δ, small fluctuations in the input data may cause undesired system stops.

Due to the fact that the Lipschitz condition is evaluated online, this might yield to inefficient system behavior. Thus, although both the correctness and validity of the reinterpretation was successfully demonstrated by experiments under real-world conditions, it is recommended to apply this technique particularly for systems with repetitive tasks, that only underlie minor variations in the environmental conditions. In order to apply this method to safety-critical systems, it must be accompanied by more detailed information on the uncertainty. While the reinterpretation enables to check whether a pre-specified limit that is defined in the output space is achieved, it may happen that dangerous situations or hazards remain undetected: Importantly, the parameter δ merely refers to the distance to x_0 and not the direction. However, as explained in Section , the asymmetric uncertainties might exist. In this case, the representation with the l_p-ball causes limitations: despite the fact that the probabilistic formulation of the Lipschitz equation relaxes the robustness requirement, the dependency on the direction is not considered. Therefore, to expand the practicability to safety-critical robot systems, for example human-involved applications, studies on incorporating details of the uncertainties are required.

Table 5.4: Comparison of the detection rate of incorrect classifications with the combined uncertainty (reinterpreted) Lipschitz equation (left) and by neglecting the uncertainty (right).

Setting	Detection rate (with reinterpretation)	Detection rate (w/o uncertainties)
No occlusions	$83.30\,\%$	$83.30\,\%$
Occlusions	$83.30\,\%$	$50.0\,\%$

5.3 Continuous Uncertainty-Aware Optimization for Safe Robot Systems

In this Section, the optimization problem will be extended to continuous cases. Instead of distinguishing between distinct classes, like successful and failed grasps or correct and incorrect images, it is aimed at minimizing or maximizing a continuous function. Again, the central incentive is to study how the integration of the quantified uncertainties contribute to the improvement in the optimization performance.

While the methods developed in Section 4.3 are generally applicable, the validation experiments focus on risk minimization for safety-critical robot applications. Apart from that, approaches to risk minimization have gained attention in the robotics domain. However, methods for online risk minimization and safety assessment that can be directly interpreted in the context of pre-specified thresholds (e.g., ISO standards) do not exist to

date. To this end, risk detection serves as a reasonable use case to validate the continuous uncertainty-aware optimization presented in Section 4.3. Especially, the term *uncertainty* is tightly related to the definition of *risks* in the safety domain. Ideally, the uncertainty captures the possible deviation of a quantity of interest that can be directly interpreted to estimate the risk, that occurs due to unexpected, but possible system behavior. Here, the analogy between robot applications and physics experiments is emphasized. Apart from the domain, the measurement uncertainty generally serves as a measure to estimate the amount of undesired events due to inaccuracies in the system.

After clarifying the definition of the term *safety* and specifying how risks and uncertainties are related to each other, a brief overview of the state-of-the-art literature on safety evaluation methods of robot systems is provided. Next, the optimization goal is formulated by referring to the risk definition. Particularly, the optimization is split in two parts. First the severity of events is approximated by a constant value. On this basis, it is studied how the uncertainty affects the evolvement of risks such that the quantitative assessment of safety requirements is conducted by means of the propagated uncertainty. In doing so, the correctness of the uncertainty propagation method explained in Section 3.2.2 is verified. Prior to applying the technique on a real-world robot system, the uncertainty propagation validated in a simulation scenario, where the ground truth uncertainty is provided. This enables to study the accuracy of the uncertainty propagation algorithm introduced in Section 3.2.2.

Afterward, the applicability of the uncertainty quantification and propagation methods are demonstrated in real-world Human-Robot Collaboration. Specifically, the uncertainty in the detection of the human's pose is computed to directly map it on quantitative safety limits. While this allows to detect hazardous events online during run time, and perform an optimization by monitoring the sensitivity, the origins of accidents is challenging to identify. In actual fact, the occurrence probability of hazards is too low such that the statistical significance of the analyses on them is highly limited. To overcome this, the second part of the optimization employs the importance sampling algorithm explained in Section 4.3.6. With this technique, the sample generation is artificially enforced in parameter regions where risk events occur. By referring to the statistical analysis performed on the increased number of samples, risk minimization methods are derived. This validation is conducted on the basis of three simulated human-robot collaboration scenarios. Finally, it is discussed how the pipeline for uncertainty-aware optimization of robot systems can be extended or further refined to simplify the practical applicability to more complex real-world robot systems.

5.3.1 Safety and Security

The exact definition of the term *safety* is domain-specific. In the context of robot systems, it is important to clearly distinguish *safety* and *security*: While the latter one aims at mitigating undesired and dangerous consequences due to specifically designed attacks on the system (e.g., due to hackers), safety addresses on unintended and unexpected failures. Therefore, security is a necessary prerequisite for safety: Especially, research on safety assurance becomes relevant for robot systems under the assumption that security lacks do not exist. By relying on this assumption, methods to safety assurance and the risk minimization focus on avoiding and reducing the rate of undesired happenings that originate from inaccuracies and unexpected deviations of the system behavior. In order

to validate the uncertainty propagation and optimization techniques in the previous Chapter, the methods are employed for reducing the probability for risks, thereby evaluating quantitative safety requirements. This means that the optimization goal is formulated with respect to the definition of the *risk*.

Since the goal is to enable the practical applicability of the derived techniques in this thesis, it is referred to standards and definitions stated in official regulations provided by the International Organization for Standardization (ISO). At the same time, it is highly desired to maintain the generic character of the presented uncertainty-aware optimization algorithm such that the risk minimization is formulated with respect to critical parameters p_c in general, that must be specified to the respective application. According to the regulations of the ISO, the risk of an incident i depends on its severity and the occurrence probability $Pr(i)$, i.e.

$$risk(i) \propto f_{risk}(severity(i(p_c)), Pr(i(p_c))), \tag{5.6}$$

where p_c stands for the critical parameter(s). Especially, this relationship can be exploited for assessing safety since

$$\text{safety} \propto 1/\text{risk} \tag{5.7}$$

For example, an incident may be described in the temporal space or Euclidean coordinates to specify where the risk has occurred. Here, the probability for the occurrence can be approximated with the measurement uncertainty: essentially, the uncertainty introduces a probabilistic representation of the system variables. From practical viewpoint, it thus assigns a probability to the area of values instead of relying on a distinct representation.

Therefore, safety-critical robot applications, where the risk minimization play a decisive role, serve as suitable use cases for the validation of continuous uncertainty optimization techniques in Section 4.3. Of course, the probability must be further specified depending on the application: For human-involved robot systems, the distance between humans and robots and the relative velocities are of particular interest. While this is stated by the technical specification of robot systems in ISO 10218 [28], these variables are reasonable for assessing the safety state in human-robot collaboration. Therefore, the optimization function is formulated with respect to the distance and velocity for the following experiments. Especially, it is studied how the uncertainty of the distance and the velocity between humans and robots is quantified and integrated in the system function to enable an online safety evaluation during run time.

Quantitative Safety Requirements

The safety evaluation is performed by referring to quantitative safety requirements, that is, by aiming at developing a method that allows to

1. Estimate the risk originating from a robot system
2. map it directly on the tolerated failure rate specified by pre-defined threshold (e.g., ISO standards)

In particular, it is assumed that the qualitative requirements, such as the definition of safety zones are fulfilled. As indicated above, high emphasis is devoted to probability thresholds. While threshold values are specified for the safety-critical variables as the distance or the velocity between humans and robots, that are also discussed in the context of the

validation experiments, evaluating the probability limits in robot systems is of particular interest: In contrast to physics experiments, where the uncertainty can be computed on the basis of large data sets gathered during measurement periods of several years, the operating time of robot applications prior to system commissioning is rather short. Therefore, assessing safety in robot systems must be accompanied by additional steps as will be elaborated below.

5.3.2 Related Work: Safety Evaluation of Robot Systems

Existing literature on safety evaluation and risk mitigation methods in robotics can be broadly classified in two categories. Since this thesis argues for the consideration of uncertainties, risk detection and safety evaluation methods will be briefly presented in this Section. Fault tree analysis (FTA) and Reliability Block Diagrams (RBD) represent two categories of classical risk assessment approaches in robotics. Here, a tree graph serves as the basis for the FTA, where the undesired events (e.g., dangerous failures as accidents) are modeled by the root and the leafs reflect the contributing factors (e.g., failures of system components). Especially, probability values are specified to the events that are accumulated over logical gates that connect the leafs with each other [70]. With these values, the propagation of single events is modeled within a fixed time window.

Similarly, RBDs are block diagrams, where each block stands for a critical system function. While redundancy is represented via parallel blocks, dependent ones are connected as series [38]. The analysis for RBDs is conducted via Boolean logic. Due to the simplicity in the models, FTA and RBD can be converted into each other which is practically beneficial. However, one major limitation is given by the binary representation of possible outcomes and the assignment of fixed probability values to all events. Moreover the FTA and RBD analyses are both only applicable to events that are statistically independent from each other. In practical real-world systems however, it is often difficult to verify whether this requirement is met: Especially in the context of risks, it is likely that causal relationships exist among the events which impedes the employment of FTA and RBD analyses. Furthermore, the possibility to consider temporal aspects, such as the order of happenings and dangerous events, is not provided. This poses a challenge to apply these methods for real-world systems, where the temporal dependencies play crucial roles to estimate the current risk.

Due to these limitations, these approaches were extended to dynamic fault trees (DFTs) in [58], that account for the time dependency in failure scenarios. Furthermore, the survey of Ruijters and Stoelinga in [70] outline how traditional FTAs are converted to Bayesian Networks (BN). In particular, this extension allows for a more informative failure analysis: In contrast to the basic event failure analysis, the Bayesian representation enables to identify the root causes for the occurred risk events. Practically, this means that for the top leaf, that presents the dangerous outcome, the contribution of the previous leafs can be estimated in straightforward manner. As elaborated by Weber et al. in [79], BNs enable to deal with statistically dependent events and are not restricted to binary outcomes. However, they also rely on graph-based representations. Also, these models consider fixed time horizons during which each component can fail only once. Furthermore, Markov models also offer one possibility for probabilistic risk assessment. Here, the system description is performed by probabilistic state transitions.

A second line of safety evaluation methods deals with developing control algorithm that inherently assess safety requirements. For example, the paper of Hadaddin et al. in [37]

derive the relationship between the severity of injuries on pig samples with the robot velocity, thereby studying the consequence of possible collisions for varying impact masses. Next, the obtained findings are adopted in the robot control architecture by developing a motion supervising component for collision avoidance purposes. Similarly, Zanchettin et al. develop a controller that interprets the sensor signals directly and avoids unsafe actions. However, this interpretation requires the exact specification of safety constraints beforehand such that the distinction of safe and dangerous actions highly depends on the prior knowledge on the respective application. One further publication that falls into the control domain is given by the paper of Lasota et al. in [51], where a real-time safety system is introduced. Especially, the robot speed and position are adapted by referring to the human pose, that is obtained via a PhaseSpace motion capture system. However, uncertainties are generally considered as negligible in this approach.

In addition, a long line of works that use learning methods to derive belief spaces. Briefly, the purpose of these belief spaces is to provide the robot with a probabilistic representation of the environment to improve the selection of safe actions. However, since the focus of this thesis lies in developing tractable techniques, these will be not presented in detail. A third, relevant subfield in robot safety addresses the development of risk mitigation techniques and safety assessment techniques by directly referring to the critical variables in the official regulations. Here, simulation-based approaches that are suitable at the system design stage as the contributions in [5],[78], [36] proposed.

In the context of real-world safety evaluation techniques, that are more relevant for this thesis, the literature in the robotics domain address the requirement in ISO 10218 [28], that defines the speed and separation monitoring (SSM) mode. The goal of this mode is to maintain a certain safety distance and speed between humans and robots. In order to facilitate flexible human-robot collaboration by adapting the safety-critical parameters according to SSM, Lacevic and Polverini et al. introduce the computation of kinestostatic danger fields in [49] and [65]. In their first publication, the basic idea of capturing possible candidates for collision objects by computing and adapting the robot velocity in the respective direction is presented. Afterward, the authors extend their approach by generalizing their algorithm to dynamic scenarios.

In 2017, Marvel et al. published a paper by providing a more thorough formalization of the SSM, that is, elaborating on the quantitative requirements in [57]. Importantly, the authors emphasize that the evaluation of the SSM criterion and the safety assessment of robot systems necessitates not only the knowledge on the severity of possible risks, but also the measurement uncertainty in all relevant robot tasks. Similarly, the survey on methods for facilitating safe HRC systems provided by [50] argues for a stronger probabilistic representation of critical variables that include possible deviations of outcomes. In particular, Lasota et al. emphasize the need for a risk measure that can be directly interpreted correctly by humans.

To address this gap by validating the continuous uncertainty-aware optimization techniques introduced in the previous Chapter at the same time, it is suggested to perform the safety evalution of robot systems by referring to the system's uncertainty. To this end, the correctness of the developed techniques is first validated in simulation environments, where the ground truth uncertainties can be estimated in straight-forward manner. Afterward, it is demonstrated how the quantified uncertainty can be directly used to assess the risk and therefore safety of a robot application in online fashion.

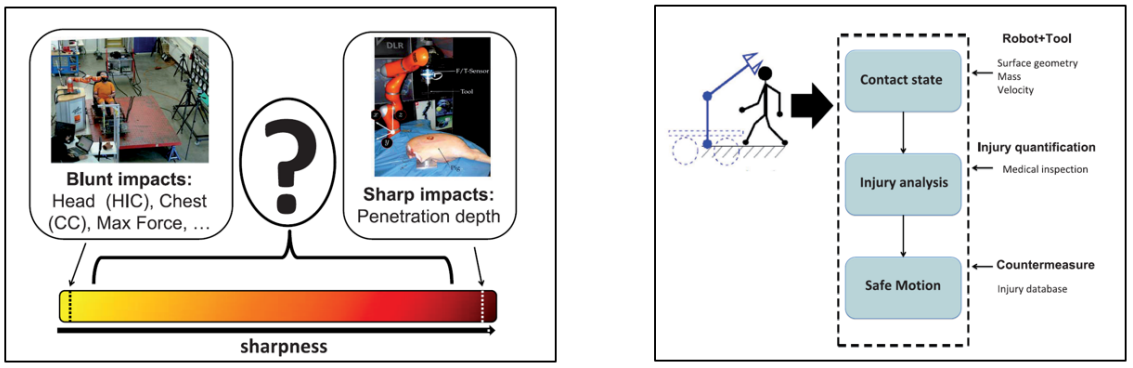


Figure 5.10: In the paper of Haddadin et al. [37], the authors study the severity of injuries on pig samples. Here, the relationship between the robot force, velocity and the penetration depth are analyzed. The obtained findings are integrated in the robot control framework to circumvent the occurrence of dangerous collisions between humans and robots. Figures taken from [37].

5.3.3 Validation Experiments: Uncertainty Quantification and Propagation

Now, the uncertainty quantification for BB tools in Section 3.2.1 and the uncertainty propagation algorithm presented in Section 3.2.2 will be validated. Next, the proportionality in Equation (5.6) is leveraged to directly map the uncertainty of a HRC system on the probability for the risk occurrence Since the critical system variables and parameter spaces for the risk determination highly depend on the application, these will be specified individually for each experiment. Apart from that, following assumptions are made for the experiments that will be introduced below:

1. Assumption 1: Real-Time Capability: The measurement data is directly accessible.
2. Assumption 2: All variables can be measured. Practically, this means that each fluctuation in the environment and all system variables are clearly defined and can be monitored.
3. Assumption 3: In the HRC experiments, the robot is confronted with only one human worker. This means that it is excluded that two or more human workers will enter the safety-critical space around the robot.

Uncertainty Quantification of Black Box Components

As motivated in Section 3.2.1, most technical components are accompanied by data sheets that provide information on its uncertainty behavior. However, BB tools may be employed in robot applications: For example, NN-based components have gained much attention in recent years. At the same time, a thorough understanding on the uncertainties of NNs is missing to date as was elaborated in the previous Sections.

While the analogies between the probabilistic robustness and the metrological uncertainty have been exploited to calculate a unified uncertainty value, several requirements as the linearizablity of the NN architecture must be theoretically validated. In addition, the identified parameters are only reasonable if the environmental conditions stay constant during system commissioning. In case these requirements cannot be met, the uncertainty

of BB tools can be estimated via leveraging the knowledge on the conserved properties of a system as presented in Section 3.2.1. Especially, this method can be performed online such that variations due to possible environmental disturbances are considered in the quantification.

Experimental Setup

The uncertainty quantification of BB tools can be estimated in straightforward manner. Basically, the validation is performed by applying the technique explained in Section 3.2.1 on a measurement tool with known uncertainty and comparing the results with the available manufacturer specification. Here, the SICK S3000 safety scanner serves as the reference tool. Specifically, the 2D position (x-y plane) of static objects during the measurement period is estimated via the laser scanner data. Accordingly, the conservation equation is given by

$$\vec{\nabla}\vec{r}_{BB}(t) = 0, \tag{5.8}$$

where $\vec{r}_{o,BB}(t)$ denotes the position of the respective object at time t. As described in Section 3.2.1, above Equation is evaluated during run time, that is, with the same data frequency as the laser scanner. Importantly, violations on the above equation are stored. After collecting a data set, the uncertainty quantification method for BB tools is applied. Specifically, the distributions are bootstrapped to obtain a reasonable amount of significance. The corresponding pseudocode is given in Algorithm 3.

Algorithm 3 Conservation based Uncertainty Estimation

Input: measurement data $x_{a,i}$; conservation Eq. $f_C(\cdot)$;
relationship betw. attribute and data $a(x_{a,i})$; confidence level σ; parameter for correl. analysis ξ; safety limit λ; time steps t.
Output: total combined uncertainty $u_C(a)$;
safety limit check (bool)
for $i \leq t$ **do**
  $dev[i] \leftarrow f_C(x_{a,i}, ..., x_{n,i})$
end for
for $z \leq 10.000$ **do**
  $b[z] \leftarrow$ bootstrap dev[]
end for
From b[] compute $u_C(a)$ for user-defined σ
Test H_0 for given p-value and parameter ξ
if H_0 rejected **then**
  $cov(u_C, \xi)$
  print $cov(u_C, \xi)$
end if
$r \leftarrow u_C \cdot l_{bio}$
if $r \leq \lambda$ **then**
  return 1
else
  return 0
end if

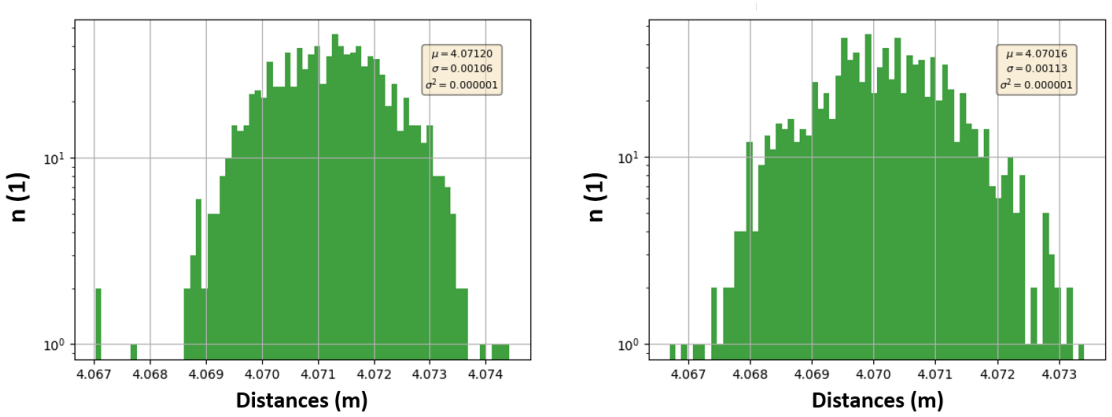


Figure 5.11: The object position of static obstacles was recorded under non-changing environmental conditions with the safety scanner SICK S3000. This figure shows the results for the uncertainty in the distances to these objects calculated by Algorithm 3 by considering 100 recordings.

Results

The results are then saved and evaluated according to above pseudocode in Algorithm 3. The systematic uncertainty obtained by evaluating the conservation equation amounts to $u_{LS,exp} = 0.0038\,\mathrm{m}$ for the SICK S3000 laser scanner at a scanning range of $4.0\,\mathrm{m}$. This corresponds to a relative uncertainty of

$$u_{LS,exp} = 0.095\,\% \tag{5.9}$$

on a confidence level of C.L.=$95\,\%$. Comparing this result with the manufacturer specification given by

$$u_{LS,theo} = 0.091\,\% \tag{5.10}$$

yields to a discrepancy of $4.0\,\%$. This fluctuation in the measurement data can be explained by the statistical uncertainty that is computed via the amount of data considered for this calculation by means of Equation (5.8).

Correctness of the Uncertainty Propagation

Prior to performing an uncertainty-aware optimization for the purpose of minimizing risks and assessing safety, it must be first validated whether the developed algorithm yields correct results. Logically, the uncertainty itself depends on the employed technical components and the corresponding resolutions which poses a challenge to verifying the correctness in real-world systems. In addition, the ground truth uncertainty is not accessible in real-world applications which makes it challenging to assess the accuracy of the proposed uncertainty propagation method in Section 3.2.2. Instead, the accuracy is studied on the basis of a simulation environment in PyBullet. The main advantage of a simulation environment is that the environmental parameters and the impact of these on the employed measurement tool as well as the robot parameters can be adapted and controlled with high accuracy.

Experimental Setup

First, proof-of-concept experiments are carried out in a PyBullet simulation environment. Here, an industrial robot arm performs point-to-point movements between two specified points in 2D plane with a constant velocity. The goal lies in tracking the 2D position the robot end-effector by using a camera with a frame rate of $f_s = 30\,\mathrm{fps}$. Here, the attribute a and its uncertainty u_a are given by

$$a \mapsto \vec{r}_{s,R}(t) = (x_{s,R}, y_{s,R}) \quad \text{and} \quad u_a \mapsto u_{\vec{r}_{s,r}}(t), \tag{5.11}$$

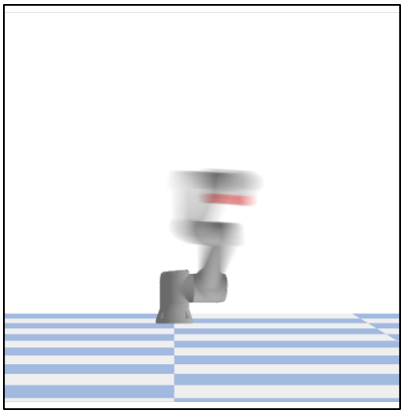
where the critical parameter p_c corresponds to the time t and x_s, y_s denote the x, y-coordinates of the robot end-effector in Euclidean space, respectively. Specifically, 6000 frames for each robot velocity settings are considered, where $v_1 < v_2 < v_3$. The uncertainties are considered to follow Gaussian distributions for the camera and the robot velocity. While a resolution uncertainty of $0.02\,\%$ is approximated for the camera, the robot velocity is assumed to underlie an uncertainty of $10\,\%$. The number of MC trials is set to $M = 10^5$. It is expected that the uncertainty increases with the robot velocity since the frame rate remains at $f_{p,p} = 30\,fps$. The experimental setup in PyBullet is shown in Figure 5.12.

In the second step, the real-world applicability of the uncertainty propagation algorithm in one representation introduced in Section 3.2.2 is studied. Thus, the position tracking of the UR10e robot end-effector is performed under real-world conditions. In analogy to the PyBullet simulation environment, the intel RealSense D435 camera is employed. By using the Real-Time Data Exchange (RTDE) interface provided by Universal Robots, a point to point trajectory for the robot is implemented in horizontal direction. Thus, the ground truth position of the robot end-effector joint is easily accessible by RTDE.

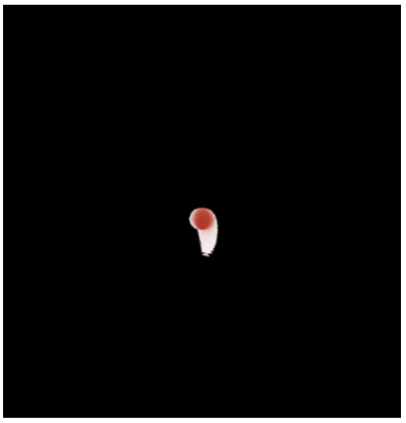
To explore the uncertainty propagation algorithm, an ARUCO marker [32] is attached on the robot end-effector. Therefore, the position tracking of the robot end-effector is performed by the detecting the ARUCO marker. As in the previous case, the tracking process is subject to uncertainties due to the robot velocity and the limited performance of the camera. After completing the calibration of the camera intel RealSense D435 into the world frame, the actual end-effector position provided by the RTDE framework is compared with the coordinates obtained from the ARUCO marker tracking. This enables to verify whether the ground truth value lies in the estimated range including the uncertainty, i.e.,

$$\vec{r}_A(t) \pm u_{prop,A}(t) \geq \vec{r}_{GT}. \tag{5.12}$$

Here, $\vec{r}_A(t)$ stands for the end-effector position obtained by tracking the ARUCO marker, $u_{prop,A}(t)$ the corresponding PMU and $\vec{r}_{GT}$ the end-effector ground truth position provided by RTDE. Again, the experiments are carried out at three robot velocity settings $v_1 < v_2 < v_3$. For each setting, 900 frames are collected. The number of MC trials amounts to $M = 10^5$. To obtain the ground truth uncertainty, it is referred to manufacturer specifications in the data sheet of the intel RealSense D435 camera, that states an uncertainty of $0.02\,\%$.

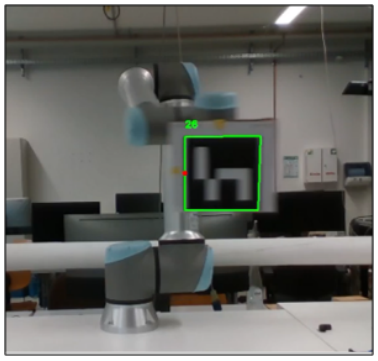


(a) The robot end-effector position is represented by a red mask.

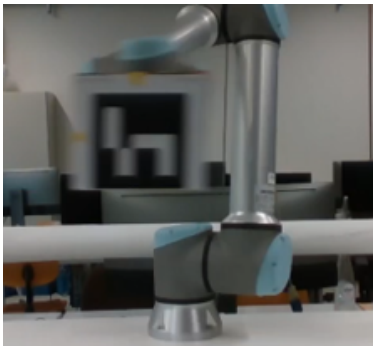


(b) To estimate the robot end-effector position, the red mask is tracked.

Figure 5.12: The MC sampling based uncertainty propagation technique in Section 3.2.2 is validated on robot end-effector tracking experiments in a PyBullet simulation environment. Particularly, the ground truth values for the robot end-effector position and the uncertainties are known.



(a) The ARUCO marker is attached on the robot end-effector to enable a 2D tracking.



(b) In case the robot speed is too high, the marker tracking cannot be performed.

Figure 5.13: The uncertainty propagation technique in Section 3.2.2 is applied on a real-world 2D tracking experiment.The tracking performance depends on the velocity of the robot.

Results

The results are summarized in Table 5.5. As the results of the simulation experiments seem promising, a similar experiment is conducted under real-world conditions: Here, the industrial robot arm Universal Robot (UR10e) is equipped with an intel RealSense D435 camera at its end-effector. The task of the robot lies in detecting ARUCO markers [32]. Importantly, the uncertainty in the detection is computed online by applying the algorithm developed in Section 3.2.2, where the attributes and critical parameters are defined by the position of the robot end-effector. However, one limitation of real-world systems is that the ground truth uncertainty is not known. Especially, the knowledge on how the environmental parameters, as lightning conditions, influence the detection, is not available.

Therefore, the experiments in the second round serve to studying the efficacy and applicability for the evaluation safety limits. To do so, the lightning conditions are varied during the experiments. Specifically, an external light source is employed to introduce sudden variations. Although the ground truth value of the uncertainty that is caused by these variations is not known, it is obvious that the uncertainty in the detection becomes larger. Ideally, the robot would thus stop or reduce its velocity according to the sudden increase in the detection uncertainty that is estimated by the uncertainty propagation algorithm.

Importantly, this experiment offers the possibility to observe whether the computation of the propagated uncertainty can be used to evaluate threshold online. In fact, the tolerated uncertainty in the detection of the ARUCO marker is specified prior to performing the experiments. By referring to the accumulated uncertainty calculated by the method in Section 3.2.2, it is checked whether the threshold is exceeded by employing the mapping on the safety limit according to Section 4.3.5. The result of this evaluation is then forwarded to the robot control algorithm to stop the robot movement or reduce its velocity.

Table 5.5: Discrepancy between the ground truth uncertainty and u_{prop}.

Robot Velocity	v_1	v_2	v_3
PyBullet	$6.30\,\%$	$9.55\,\%$	$8.70\,\%$
Real World (ARUCO)	$9.05\,\%$	$10.32\,\%$	$11.03\,\%$

Safety Evaluation in Human-Robot Collaboration

Based on the findings in the above experiments, where the uncertainty propagation algorithm was studied regarding its correctness and applicability to real-world environments, the use case is now extended to evaluating safety in HRC.

The experimental setup consists of an industrial robot arm Universal Robot (UR10e) that is installed on a table and performs point-to-point movements between two boxes that are filled with random objects. The task of a human is to empty the boxes with the objects that requires to enter the work space of the robot. In order to estimate the position of the human, the NN-based human pose estimator OpenPose 3D presented in [16] is employed. Briefly, this NN-based tool provides the 3D joints of the human body as shown in Figure 5.14. Here, intel RealSense D435 cameras with a frame rate of $f_p = 30\,fps$ are used to capture the image data of the human body. The human-robot distance $d_{HR}(t)$ is calculated by referring to the robot end-effector position that is obtained by the robot control and the closest human joint position among the 25 joint positions yielded by OpenPose. The optimization goal in this experiment lies in minimizing the probability for risk by

$$\min risk(t) \propto \min Pr(i(t)) \propto \min u_{prop}(d_{HR}(t)), \tag{5.13}$$

where the severity of accidents is approximated to be constant: In the case of HRC, the severity of collisions with the robots is measured by the human-robot collision force. The force tolerance on the human's body parts are specified by distinct values in the context of ISO standards [27]. Hence, the human-robot distance $d_{HR}(t)$ serves as the critical variable for the safety evaluation. Especially, the uncertainty on the human-robot distance $u_{prop}(d_{HR}(t))$ resulting from all system components is directly proportional to the risk. Following uncertainty sources are considered for the uncertainty propagation algorithm:

1. intel RealSense D435 with $u_{camera}(t) = 0.091\,\%$

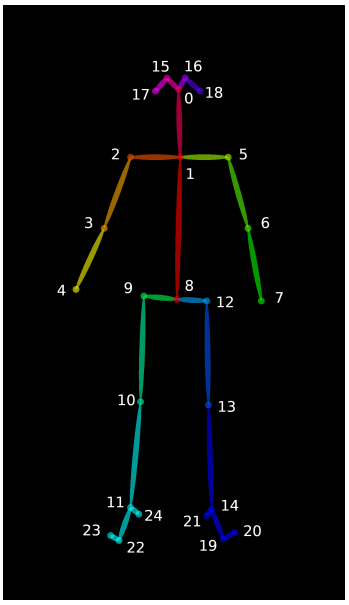


Figure 5.14: The NN-based human pose tracking tool OpenPose provides the positions of the 25 human joints

.

2. uncertainty in the human detection

The uncertainty on the robot end-effector position is neglected such that

$$u_{r_R}(t) = 0. \tag{5.14}$$

While the camera uncertainty is taken from the manufacturer specification of the intel RealSense D435 data sheet, the uncertainty in the human joint positions is obtained by applying the uncertainty quantification method based on conservation equations in Section 3.2.1. Here, the conservation equation is defined via the constant length of the human's body parts:

$$||\vec{r}_{H,i} - \vec{r}_{H,i}|| = const., \tag{5.15}$$

where $\vec{r}_{H,i}$ and $\vec{r}_{H,j}$ stand for the 3D position of the human joints i and$j$, respectively with $i \neq j$. Practically, Equation (5.15) reflects that the length between the human's body parts do not change during run time. Reversely, the violations on this Equation are directly assigned to the uncertainty of the human pose estimator. The violations of this conservation equation are evaluated and monitored during runtime with a frame rate of $f_p = 30\,fps$. Thus, the statistical significance of the uncertainty, and thus the confidence grows with the run time. However, since the data amount is limited during the first minutest, the bootstrapping technique is applied with N=10.000 for the uncertainty estimation.

As explained in Section 5.3.1, the human-robot distance is one critical variable for safety evaluation. To this end, the assignment of the attributes given by

$$a \mapsto \{d_{HR} \mapsto \vec{r}_R, \vec{r}_H\}. \tag{5.16}$$

In particular, the critical parameter space corresponds to the time: While the exact location of the robot and the human may be important in some applications, the focus of the risk detection is based on the time in the performed experiments, such that

$$p_c \mapsto t \tag{5.17}$$

Figure 5.15: The uncertainty-aware safety evaluation is applied to real-world Human-Robot Collaboration. Here, the distance between the human worker and the robot end effector is calculated to evaluate the occurrence probability for dangerous situations by referring to the probability threshold in ISO 13849 [29].

Mapping on Safety Limits

To perform the mapping on the safety limit, it is referred to the definitions provided by the ISO. As described in Section 5.3.1, the risk is defined via the probability for the occurrence of dangerous failures. In the performed experiments, these dangerous failures correspond to the violations of a specified Human-Robot distance. On the other hand, the SSM mode specifies the tolerated distance values and a relative velocity of $v_{HR} \leq 1.5\,\mathrm{m/s}$. Regarding the tolerated failure rate, ISO 13849 states a limit of $\mathrm{Pr} \leq 1/10^6$, that is, one accidents in 10^6 hours.

However, for practical reasons, the limit is lowered to $\mathrm{Pr}_{\mathrm{ev}} \leq 1/10^3\mathrm{h}$. By applying the mapping technique in Section 4.3.5, the uncertainty is interpreted on ISO 13849. Specifically, the safety evaluation is conducted online such that the human worker is notified once the safety requirement is violated. At the same time, this information is forwarded to the robot control framework to initiate a system stop once d_{HR} falls beyond the critical limit.

Results: Detection of Collisions

To validate the applicability of the uncertainty-aware safety evaluation technique in HRC, the above described task was performed for a run time of 6 minutes, resulting in 10800 frames for the human detection, and thus the evaluation of the human-robot distance. It was observed whether the situations with $d_{HR}(t) \leq d_{crit}$ can be detected. The results are shown in Table 5.6.

Table 5.6: Averaged u_{prop} values for different body segments

Body Segment	Averaged u_{prop}
Spine	20.2 %
Collarbones	4.2 %
Thighs	23.6 %
Arms	15.7 %

5.3.4 Uncertainty-aware Risk Reduction via Importance Sampling

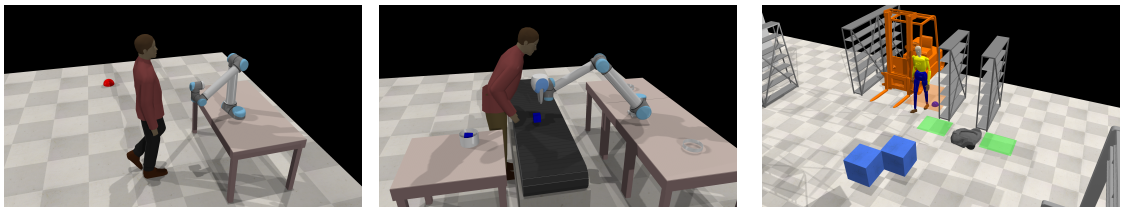
One crucial limitation in the previous Sections is that the severity of accidents was assumed to be constant. While approximating the severity with the collision force, and thus the tolerated collision force on the human body is reasonable, the severity may depend on the uncertainty or additional factors. Here, the main challenge arises to the missing knowledge on the possible hazardous events. Especially, their occurrence and their process of development is highly dependent on the application. This means, that studying the severity necessitates a sufficient amount of data on the accidents. However, this requirement often cannot be met, especially in real-world applications. Ideally, the occurrence probability of dangerous events is held low since the main goal lies in assuring a safe system. On the other hand, data on the occurrence of hazardous events is highly desired to derive safety measures.

One approach to address this dilemma lies in refining the traditional MC sampling to Importance Sampling, that was introduced in Section 2.1.7. In the following, the grid-based importance sampling approach developed in Section 4.3.6 is applied in three HRC scenarios modeled in simulation. In contrast to the above experiments where it was aimed to validate the approaches, the main incentive lies in studying the efficacy for applying IS to enhance the identification of hazardous events. Due to the existing risks of causing injuries on human-beings or severe accidents, the experiments are conducted merely in simulation environments. Especially, temporal and spatial uncertainties are both considered in the simulation studies. The obtained findings enable to address the question how these uncertainties contribute to the occurrence of risks.

Simulation Scenarios

The grid-based IS algorithm in Section 2.1.7 is applied to three scenarios in simulation. In all experiments, the human-robot distance is measured.

1. **Scenario 1**: In the first scenario, the human worker approaches an industrial robot arm UR10e that is installed on a table. The robot arm performs a movement along a clearly defined trajectory. This scenarios serves to explore whether the grid-based representation is appropriate to capture the hazardous events. Here, the space of possible hazardous events is modeled by referring to a sphere around the human joints. Specifically, the radius of this sphere is computed via the relative velocity between the robot and the human worker. The robot performs evasive movements once the human-robot distance falls below a certain limits.
2. **Scenario 2**: Second, the experiment with the UR10e is extended to a more complex use case: Here, the tasks of the robot and the human worker lie in detecting the color of boxes that move with a conveyer belt. As depicted in Figure 5.16, collisions may occur once the robot and the human reach for the same object at the same time. In analogy to Scenario 1, the robot moves back once the distance threshold falls below the limit.
3. **Scenario 3**: Finally, a basic experiment with a mobile robot is conducted: While the mobile robot navigates in a warehouse environment, the human worker performs independent tasks as shown in Figure 5.16. A system stop is initiated for the mobile robot once the distance limit cannot be assured.



(a) Scenario A **(b)** Scenario B **(c)** Scenario C

Figure 5.16: The grid-based importance sampling technique is validated by referring to three simulated HRC scenarios. In all scenarios, the robot performs evasive movements once the human-robot distance falls under a specified limit.

Parameter and Uncertainty Specification

For all scenarios, spatial and temporal uncertainties u_s and u_t are considered. However, in contrast to the above experiments, the focus is laid on studying the efficacy of the importance sampling technique. Thus, the uncertainties are modeled in a simplified manner by adding constant, distinct values to the measured human-robot distance. In fact, two critical parameters are considered: In order to account for possible time delays to the robot's safety reaction, like the evasive movements or the safety stop in above scenarios, a constant value is sampled from a uniform distribution. Specifically, the temporal uncertainty u_t for one simulation run is computed via

$$u_t = N_{sim} \cdot T_{sim}, \tag{5.18}$$

where T_{sim} is the simulation step and N_{sim} sampled from a uniform distribution $\mathcal{U}0;9$ with $\mathcal{U} : \mathbb{N} \mapsto \mathbb{R}$. The spatial uncertainties reflect spatial deviations in measuring the human-robot distance d_{HR}:

$$\tilde{d}_{HR} = d_{HR} + u_s, \tag{5.19}$$

where d_{HR} and $\tilde{d}_{HR}$ specify the ground-truth distance yielded by the simulator and the measured value provided to the robot, respectively. The distance uncertainty u_s is obtained via

$$u_s = \Delta d_0 + c \cdot v_H. \tag{5.20}$$

Here, Δd_0 represents a constant value sampled from a Gaussian curve, v_H the human velocity and c a scalar value sampled from a uniform distribution.
These two uncertainties are modeled to account for possible sensor uncertainties. Of course, these can be extended by additional parameter dependencies or detailed knowledge on environmental disturbances.

Apart from that, it is not distinguished between the uncertainty arising from the robot position $\vec{r}_R$ and the human position $\vec{r}_H$ which is argued as follows: Obviously, both uncertainties are linearly related to the distance $d_{HR} = ||\vec{r}_H - \vec{r}_R||$. This means that the amount of u_d does not depend on the uncertainty source: the uncertainties from the position measurement of the robot end-effector have the same effect on u_d as the uncertainty in the human position measurement.

Implementation of the Experiments and Results

As described in Section 4.3.6, evaluating the IS technique is performed via the VAE value. To this end, this value is referred to for comparing the grid-based IS algorithm

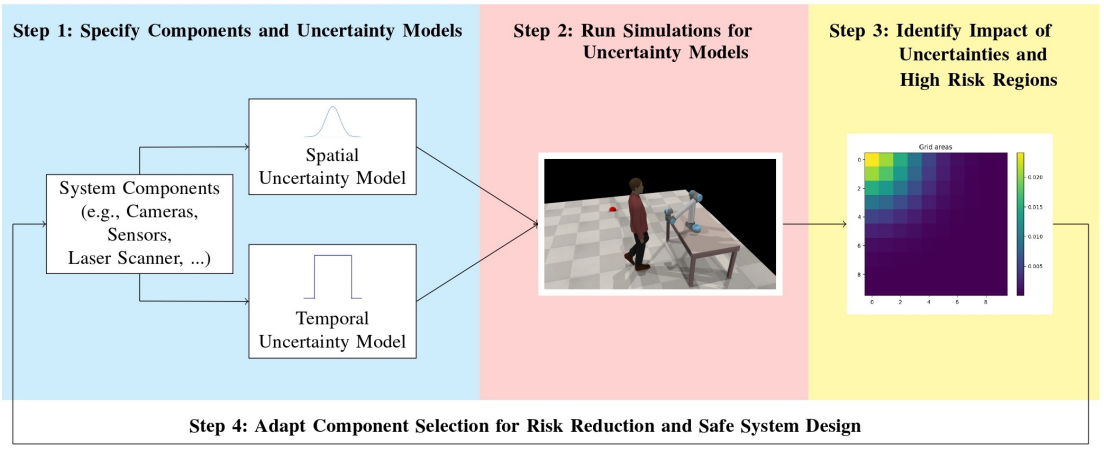


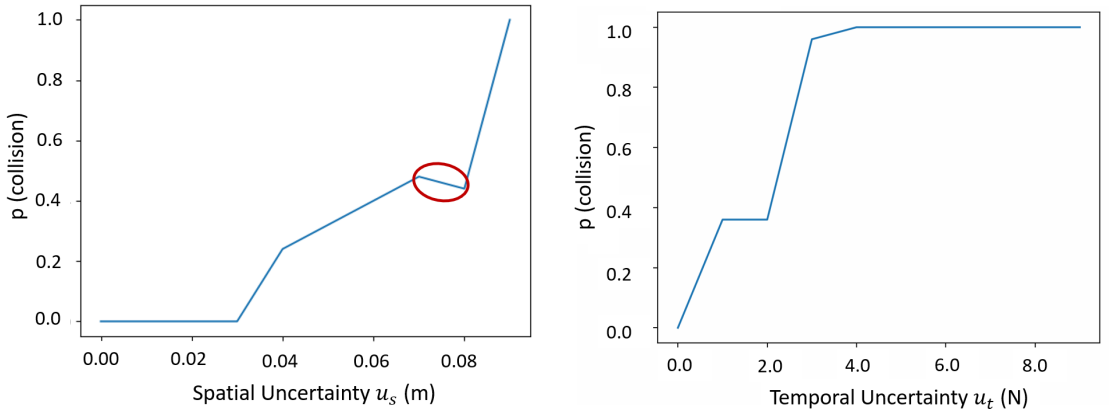
Figure 5.17: IS-based approach for uncertainty aware risk minimization. In order to explore whether a risk reduction and a safe system design can be achieved by employing the importance sampling algorithm introduced in Section 4.3.6, spatial and temporal uncertainties are sampled from pre-defined distributions (Step 1). In particular, the simulation scenarios described in Section 5.3.4 are run by considering the uncertainty samples (Step 2). The obtained results logically enable to study the impact of the uncertainties on the occurrence of risks (Step 3). In the last step, the grid-based importance sampling technique in Algorithm 0 is applied to study whether the findings contribute to the reduction of risks.

with traditional MC sampling for the three simulation scenarios described above. Here, CoppeliaSim robotics [1] is employed as the simulator. To parameterize the variables, ten samples are drawn for the time delay T, ten samples for c and 25 samples for d_0 from the corresponding distributions that yield 2500 parameter combinations for each of the above scenarios. In addition, the severity is computed by the fraction between the actual collision force F_c and the tolerated collision force $F_{c,max}$ for the affected human body part collision provided in ISO TS 15066:2016 [28].

Results: Uncertainty-aware Safety Evaluation based on IS

On the basis of the simulation results, the probability for collisions was evaluated for the spatial and the temporal uncertainties as demonstrated in Figure 5.16. At first glance, these findings support the theoretical expectation that that probability for collisions increases with higher uncertainties. However, it can be seen that the collision probability for $u_s = 0.3\,\mathrm{m}$ and $u_t = 0.09\,\mathrm{s}$ is lower than for $u_s = 0.1\,\mathrm{m}$ and $u_t = 0.08\,\mathrm{s}$. This effect is likely to be attributed to the sampling process. Specifically, it is likely that a rather low uncertainty value was drawn during the sampling procedure. In order to analyze the relationship between the risk probability and the uncertainty. The curves in Figure 5.18 do not hint at causal relationships between the uncertainties and the collision probability. Hence, higher collision probabilities may not necessarily originate from the amount of uncertainty, but additional system characteristics that were not considered in the above analyses.

[1] https://www.coppeliarobotics.com/



(a) Collision probability vs. u_s

(b) Collision probability vs. u_t

Figure 5.18: By referring to the experiments described in Section 5.3.4, the collision probabilities p(collision) are plotted against the spatial and temporal uncertainty $u_s(m)$ and $u_t(N)$, respectively. While the general expected tendency of an increasing collision probability for higher uncertainties is verified, the collision probability obviously decreases for certain regions, as for example in the area within the red circle. This implies that correlations with additional system variables that were not considered in the performed studies may exist.

VAE Reduction via Importance Sampling

In addition to the safety evaluation, the aim of this Section was to assess the suitability of IS for the analysis of hazardous events. To this end, the VAE is computed for the experiments performed in these three simulation scenarios. By doing so, traditional MC sampling is compared with the grid-based IS approach developed in Section 4.3.6. The obtained findings in Table 5.7 support that IS yields more promising results for enhancing the statistical significance in interesting regions with low probability densities. However, this seems to be only valid for scenario 1 and scenarios 3. In the case of scenario 2, the traditional MC sampling yields lower VAE values that is justified as follows: In contrast to scenario 1 and scenario 3, the dangerous region is relatively large for the occurrence of collisions. In the algorithm for the grid-based IS however, assumptions are made on the size of the dangerous zone such that the dangerous zone is underestimated during the learning phase. Since the learning phase serves as the basis for the grid-based IS, this significantly affects the result of the VAE.

Table 5.7: VAE values for MC Sampling and grid-based IS

Sampling Method	Scenario A	Scenario B	Scenario C
MC Sampling	3.4 e-05	1.3 e-04	1.6 e-03
Grid-based IS	2.8 e-05	3.8 e-04	7.4 e-04

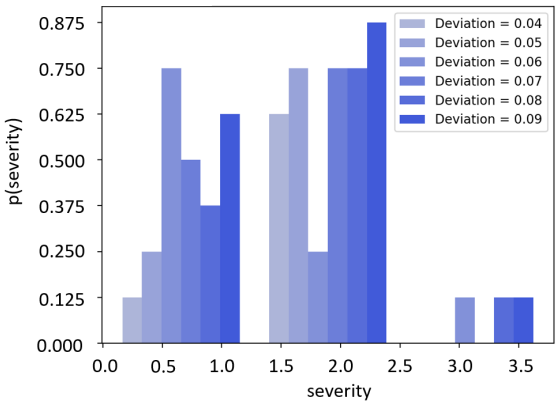


Figure 5.19: Histogram of severity values of hazardous events. On the y-axis, the occurrence probability for the severity values is shown. The deviations correspond to the spread of the Gaussian distributions from which the samples of the spatial uncertainties u_s are generated. Therefore, this figure depicts the relationship between the occurrence probability for severity values for different spatial uncertainty constellations.

5.4 Summary of Chapter

The contribution of this Chapter was twofold. In addition to the validation of the the uncertainty quantification and propagation techniques, their applicability to robot systems was explored in both simulation and real-world environments. To be specific, the performed studies yielded following findings.

1. **Maximization of Successful Robot Grasps:** The grasp selection experiments with the humanoid robot ARMAR-6 showed that modeling the grasp metrics as Gaussian distributions yields to a significant increase of succeeded grasps (from 32.6 % to 73.8 %). Hence, these results motivate to replace scalar representations by probabilistic models to obtain higher optimization performances in binary optimization problems.

2. **Minimization of Inaccurate Image Classifications:** The goal of the experiments in Section 5.2 was laid on the calibration of the NN robustness defined by Equation (4.7) and the metrological uncertainty notation. To this end, the assignment of safety-critical measures to the parameters and the sensitive parameter regions were thoroughly discussed. The results of the image classification experiments showed that considering the proposed notation for unifying the uncertainties yields to the reduction of incorrect classifications.

3. **Safety Evaluation and Risk Minimization in HRC:** The first part of this Section addressed the validation of the uncertainty quantification method of black box tools. To this end, the method introduced in Section 3.2.1 was applied to a the SICK S3000 safety scanner. The comparison of the obtained results with the manufacturer specifications led to a discrepancy of $4.0\,\%$. This deviation is attributed to the limited amount of data.

   Furthermore, the correctness of the uncertainty propagation algorithm was successfully verified in simulation experiments, where a 2D tracking was performed. Afterward, the applicability in real-world conditions was tested in an ARUCO marker

tracking experiment. Here, the findings showed that the discrepancy between the ground truth uncertainty and the result of the propagation algorithm are larger in real-world settings (10.32 %).

Next, the probability for the occurrence of dangerous situations, that is defined as a critical measure for evaluating safety by standard ISO 13849 was estimated by referring to the uncertainty mapping technique in Section 4.3.5. In particular, the online safety assessment was performed in Human-Robot Collaboration experiments, where the neural network based tool Open Pose 3D was employed for human tracking. The results demonstrated that the occurrence rate of dangerous events can be reduced by calculating the uncertainty online by means of the methods derived in this thesis.

Lastly, simulation experiments with CoppeliaSim were performed by accounting for both the severity and uncertainty. The goal of these studies was to explore the suitability of the grid-based importance sampling algorithm for risk detection purposes. The obtained results indicate that no significant improvement in the VAE is achieved compared to traditional MC sampling.

6 Discussion, Conclusion and Outlook

In this Chapter, the main findings of this thesis, including the presented methods for quantifying and propagating the measurement uncertainties in robot systems and the uncertainty-aware optimization techniques will be summarized. With the motivation to enable the evaluation of robot systems in the context of safety by describing robot systems accordingly in the state space, methods to quantifying the uncertainties of black box tools and accumulating them along the system pipeline were presented. In particular, the thesis analysed how integrating these uncertainties contributes to the performance of the robot applications. In the following, the results from the validation experiments will be shortly discussed and referred to for both deriving the limitations and emphasizing the contributions. In this context, the main findings on the questions introduced in Chapter 1 will be provided. Finally, potential approaches to transferring methods from the physics to the robotics domain are discussed by suggesting directions for future research.

6.1 Discussion

Prior to summarizing the main findings of this thesis, the results on the experiments of the uncertainty quantification, propagation and uncertainty-aware optimization methods will be shortly discussed. In doing so, the effects of the consideration of uncertainties on the optimization performance and the applicability of the uncertainty-aware optimization methods to robot systems will be addressed.

6.1.1 Uncertainty Quantification and Propagation

The main incentive of this work was to study whether applying the notation of measurement uncertainties according to the Guide to the Expression of Uncertainty in Measurement [26] to robot systems yields higher performances. After specifying the requirements for applying the techniques stated in this Guide, a method for quantifying uncertainties of black box tools was derived in Section 3.2.1. The key idea behind the developed technique is to leverage the knowledge of conserved properties of a system, express them in the state-space and to evaluate them online during run time. After specifying system variables that are known to stay constant during operation, recording data with the black box tool regarding these constant properties enables to deduce on the measurement uncertainty. Especially, the uncertainty of the respective black box tool can be computed directly and online by referring to the violations on the conservation equations.

The validation of this technique was performed by means of the SICK S3000 laserscanner. After applying the presented method on the measurement data of this tool, that is, by treating it as a black box tool, the obtained results were compared with the manufacturer

specifications of the laserscanners' data sheet. This comparison shows a discrepancy of merely $0.4\,\%$ between the estimated uncertainty and the ground truth value. Since this difference can be explained by the statistical uncertainty, arising from the limited availability of data, one can conclude that this technique yields reliable results. At the same time, it is worth noting that the identification of conserved properties, that can be expressed in terms of the measurement data from a measurement tool of interest is not always straightforward. In addition, it is likely that evaluating the conservation equations during run time can cause high computational burden. Also, most importantly, the uncertainty quantification with conservation equations is only reasonable for systems that do not underlie many variations. Since these equations may be sensitive to sudden changes, it is recommended to limit its employment in repetitive robot tasks with constant environmental conditions. Thus, while the correctness of the uncertainty quantification technique of black boxes was successfully validated, the practical applicability apart from safety-critical systems is suggested to be studied for different use cases individually to derive more generic conclusions.

Next, approaches for uncertainty propagation were developed. In addition to transferring the sampling-based uncertainty propagation algorithm, that is applicable for systems with non-Gaussian distributed input uncertainties, a technique to bringing the probabilistic robustness of NNs with the metrological uncertainty notation was derived. In particular, the variables in the Lipschitz equation were reinterepreted, thereby identifying analogies to the formalism of measurement uncertainties. Although the validity of this reinterpretation was studied by referring to an image classification experiment with NNs, the practicability of this approach to more complex robot systems is highly arguable. In fact, the identification of appropriate parameter ranges, which is essential to employ the technique, is bound to long system runs where the experimental prerequisites must be met and held during system run time.

Apart from that, the specification of the relevant parameter ranges necessitates a reference data set for different system constellations. While this requirement was met by the validation experiments in this thesis, it might be challenging to gather these data sets for more complex applications. Furthermore, the reinterpreted Lipschitz equation is highly sensitive to changes of parameter ranges. This makes it difficult to apply the suggested method to common robot applications. On the other hand, it was also shown that monitoring the sensitivity can contribute to the practicability: In fact, if the computation power of the robot system allows to constantly compute and control the sensitivity, it may relax the strictness on the parameter ranges in the Lipschitz equation. Nevertheless, the validation experiments clearly demonstrated that the proposed reinterpretation of the Lipschitz equation does not only enable to accumulate uncertainties of technical devices with those of NNs, but also provides one step towards evaluating NNs with respect to safety limits. In particular, it was shown that specifying a tolerance threshold on the incorrect classifications was met by computing the system uncertainty on the basis of the suggested reinterpretation.

6.1.2 Uncertainty-aware Optimization

In addition to quantifying the uncertainty of black box tools and accumulating them along the system pipeline, one of the central questions of this thesis was to evaluate how the uncertainty contributes to the performance of robot applications. For this purpose, the so-called *Optimization Performance* was introduced as a measure to quantitatively assess whether the consideration of uncertainties yields an enhancement in the achievement of the optimization goal. By deriving uncertainty-aware optimization techniques, distinct and continuous optimization methods were distinguished from each other. To be specific, approaches to binary optimization, minimization of inaccurate classification and continuous optimization for an analytical function were presented.

In Chapter 5, each of these techniques was validated in the context of a robot application.

Binary Optimization on Humanoid Robot Grasping
First, the binary optimization was implemented and tested on a humanoid robot grasping application with the goal of maximizing the rate of successful grasp selections. Here, the critical metrics for the grasp selection were modeled in a probabilistic manner by introducing Gaussian-distributed uncertainties. Specifically, these uncertainty models were derived on heuristic estimates. In order to perform the grasp selection in an uncertainty-aware fashion, a baseline data set was generated under real-world conditions. In fact, this baseline data set served as the basis to form a cost function that enables to characterize grasp candidates according to their success probability. The obtained results showed that the rate of succeeded grasps could be indeed improved from 32.1 % to more than 70 %. While this enhancement underlines the benefits of considering uncertainties, or more specifically the possibility of fluctuations, it is worth noting that the incorporated uncertainty models were not validated beforehand. Thus, it is recommended to first study the exact uncertainty models and possible correlations between the considered metrics to incorporate these in the optimization function. Apart from that, one limitation occurs due to the limited amount of data in the baseline data set. In the performed experiments, 1000 real-world grasps were performed and used as the baseline during four days. Thus, while compared to existing real-world humanoid grasp database, this collection contains a high amount of data, it corresponds to a rather small data set from statistical viewpoint. Consequently, this means that the findings on the statistical analyses that were provided in Section 5.1.7 may be complemented by additional studies performed on a larger data collection. In addition, varying environmental conditions or a variety on objects might be considered.

Distinct Optimization with NN Classifier
Next, the distinct optimization problem was formulated as the minimization of inaccurate classifications on an image classifier under real-world conditions. Importantly, the conclusions drawn on the results of these experiments are twofold: In addition to the correctness of the quantified uncertainties, that were obtained by applying the methods suggested in Chapter 3, the optimization performance η was enhanced significantly in both applications, thereby motivating the consideration of uncertainties in the robot applications. By performing the experiments, the reinterpreted version of the Lipschitz equation was employed. Therefore, the respective parameter ranges were considered. This parameter setting and the considered uncertainty notation enabled the reduction of inaccurate classification. However, one main limitation here is given by the missing generalizability.

Due to the fact that the parameter calibration is highly dependent on the application, this step must be performed individually for a system. Particularly in the case of the considered data bases MNIST and GTSRB, a sufficient amount of training and test data was provided. This however does not apply in the majority of applications such that the developed method to minimize the classification inaccuracies requires efforts to collect data on the application prior to operation. Furthermore, studying the parameter dependencies might be bound to computational burden in more complex applications such that the online uncertainty calculation may suffer from it. Here, it may be beneficial to analyse how the computation time increases with the parameter dependencies to consider this for online uncertainty calculation purposes.

Continuous Optimization for Risk Minimization in HRC
Finally, the continuous uncertainty-aware optimization was validated in safety-critical robot applications with the goal of minimizing risks. Here, it was referred to the risk definition provided by the International Organization of Standardization, that was considered as the basis for the safety assessment of industrial robot applications. According to this definition, the risk depends on the severity of accidents and the probability for the occurrence of the dangerous events. By estimating the latter one by means of the system uncertainty, that is obtained by the uncertainty quantification and propagation algorithms in Chapter 3, it was demonstrated how the compliance of the probability thresholds regarding hazardous events is evaluated. Specifically, the focus was laid on studying the uncertainty in the human position tracking by employing OpenPose 3D. This tool was treated as a black box with respect to its uncertainty since methods to accurately quantifying neural network uncertainties do not exist to date. Thus, after quantifying the uncertainty of OpenPose 3D by referring to the conservation equations, the obtained uncertainty was incorporated in the uncertainty propagation pipeline together with the uncertainties of the intel RealSense D435 cameras. Importantly, the rate of dangerous situations in real-world Human-Robot Collaboration could be effectively reduced by computing the system uncertainty online, monitoring the respective sensitivity values and mapping these results on the probability threshold (e.g., ISO 13849).

However, it is worth noting that only one robotic application was considered for the evaluation. Although the experiments were conducted with different humans to study how the accuracy of the uncertainty quantification depends on the length of the human joints, the industrial robot arm UR10e was employed in all experiments. In addition, the environmental conditions did not vary significantly during operation. Therefore, sudden changes were not considered for the uncertainty calculation. Furthermore, more complex scenes could be studied in future work. For example, several humans might interact with the robot or introduce more variability and occlusions in the human tracking performance. This, in turn, would yield higher uncertainties. In order to explore the performance and generalizability of the uncertainty propagation method, it is recommended to consider a higher variety of applications.

Studies on the Relationship between the Severity and Uncertainty in HRC
Next, the focus was laid on studying the dependency between the severity and the probability for the occurrence of dangerous events. As mentioned above, this probability was obtained by referring to the propagated uncertainty. The goal here was laid on the identification of risks. In this regard, three industrial Human-Robot Collaboration scenarios were modeled in the simulation environment CoppeliaSim. In order to overcome the problem

of sparse data on dangerous events that occurs due to their low occurrence probability, a grid-based importance sampling algorithm was introduced, thereby artificially enhancing the rate of hazardous events. On this basis, the question of how temporal and spatial uncertainties of sensors and technical components affect the occurrence of dangerous events was addressed. Overall, the results indicated that no analytical relationship can be derived between these uncertainties and the probability of risks. While applying the importance sampling algorithm yielded higher statistical significance than classical Monte-Carlo sampling in the context of risk analysis, the results also showed that the advantages of applying the importance sampling technique highly depends on the application. This leads to the conclusion that evaluating quantitative safety requirements by means of the system uncertainty is recommended. However, the benefits of the importance sampling technique for risk detection purposes in robot systems are challenging to assess due to the high sensitivity on the application-specific characteristics. Particularly, the considered scenarios were generated by the same simulation framework and correspond to relatively simple use cases. In all three scenarios, the distance between the human and the robot was observed for constant velocities. The large variations in the results obtained by applying importance sampling indicate the limited suitability of the proposed grid-based importance sampling algorithm. To be specific, the suggested grid-based importance sampling method corresponds to a simplification of the classical continuous version, where the derivation of the latent distribution q requires more thorough studies. Thus, although the efficacy of the introduced grid-based importance sampling was validated in the above simulation scenarios, improving the suitability for risk reduction necessitates both the refinement of the grid-based approach and further experimental studies in more complex robot applications.

Overall, the averaged improvement in the optimization performance η that is intended to measure how the incorporation of uncertainties contributes to the achievement of the optimization goal in the considered robot applications, amounts to $\Delta\eta = 64.04\,\%$. To this end, it is argued that quantifying and integrating the uncertainties along with the sensitivity monitoring yields to desirable effects in reaching the optimization goal.

6.2 Scientific Findings and Outlook

Finally, the research questions introduced at the beginning of this thesis in Section 1.1 will be discussed by referring to the obtained findings. Afterward, ideas for future research directions are briefly outlined.

6.2.1 Research Questions and Scientific Findings

1. **Which requirements must be met for the description of the robot system to enable the calculation of the accumulated uncertainty for an entire robot application?**
   According to the findings in this thesis, applying the technique for quantifying uncertainties and propagating the uncertainties on the basis of the system description in a metrological fashion yielded promising results. In particular, the considered use cases showed that the uncertainty propagation techniques are suitable for evaluating safety requirements online in an uncertainty-aware manner. Thus, in addition to the correctness, the presented techniques could be successfully applied for real-world robot systems.

2. **How can established uncertainty notations from the domain of robotics and learning be unified with the metrological uncertainty notation to enable the derivation of one single uncertainty measure for an entire robot system?**
   By developing techniques to accumulating the uncertainty of different components, a method to bringing together the robustness of neural networks with the metrological uncertainty notation was derived. Specifically, it was found that analogies can be identified between these two formalisms. However, unifying them to one measure was not performed explicitly. Instead, a reinterpretation of system variables and the subsequent calibration of the application was conducted to quantify the accumulated uncertainty. The respective validation experiments showed that the uncertainty of the system can be successfully captured intrinsically by this technique.
3. **How can the quantified uncertainties on component level be incorporated in the system to enhance the performance of robot applications?**
   For each of the experiments performed within the frame of this thesis, a different approach was applied: While the uncertainties for the binary uncertainty-aware optimization were directly considered in the definition of the variables (grasp metrics), an implicit incorporation was proposed for the NN classifiers. In the context of the risk minimization techniques, sampling based methods have led to promising results. In addition, the studies with the importance sampling based algorithm showed that general relationships between the uncertainties and the optimization performance of robot systems may be challenging to estimate. Thus, it is recommended to first study the impact of the uncertainties of single components in simulations or by applying further model-based techniques and to afterward monitor the sensitivities on the most influential components experimentally under real-world conditions. Based on the obtained results, the system components can be replaced or adapted to boost the performance of robot systems accordingly.

6.2.2 Outlook

The central incentive of this PhD thesis was to study how established quantification methods of measurement uncertainties can be employed and adapted to robot systems. In doing so, useful analogies could be identified, while at the same time, discrepancies must be coped with. However, in summary, it was shown that transferring techniques can be helpful and simplify the achievement of certain goals: Most importantly, the evaluation of robot systems with respect to quantitative safety limits, that are usually defined in the metrological space was enabled by describing and quantifying the uncertainties of the applications' components by referring to the metrological notations.

In the context of safety and risk mitigation, introducing the formalism of symmetries may yield to more efficient, yet well-understood risk minimization methods. Especially, the discovery of symmetries in natural phenomena is considered as highly valuable: Once verified, they allow to derive conclusions in several regards by referring to only small sets of data. Logically, this becomes highly useful for cases, where the availability of the interesting data is sparse or the data acquisition is challenging. In terms of robot applications, symmetries may exist for accidents between humans and robots (e.g., collisions) that could be exploited to derive the risk regions of additional accidents on the basis of small data sets. Practically, this would mean that being provided with the data on a small set on real-world accidents would allow to explore a wider area in the space of risk events, thereby enabling a more detailed analysis on the origins of risks.

However, the derivation and identification of these symmetries is bound on intuitively studying the possible spaces, that underlie these symmetries, in first place. Due to the fact that, according to the existing literature in robot safety, the risk analyses have not been performed with regard to the search of symmetries to date, it remains an open issue whether research in this direction may yield promising results in the long run. Apart from this, however, it may be one approach to fostering the transfer of established methods from the natural to the engineering sciences domain.

Bibliography

[1] M. A. Acero, P. Adamson, G. Agam, L. Aliaga, T. Alion, V. Allakhverdian, N. Anfimov, A. Antoshkin, L. Asquith, A. Aurisano, et al. Adjusting Neutrino Interaction Models and Evaluating Uncertainties Using NOvA near Detector Data. *The European Physical Journal C*, pages 1–19, vol. 80, 2020.

[2] A. Agha-mohammadi, S. Chakravorty, and N. M. Amato. FIRM: Sampling-based feedback motion-planning under motion uncertainty and imperfect measurements. *The International Journal of Robotics Research*, pages 268 –304, vol. 33, 2014.

[3] M. Aker, K. Altenmüller, A. Beglarian, J. Behrens, A. Berlev, U. Besserer, B. Bieringer, K. Blaum, F. Block, B. Bornschein, et al. Analysis Methods for the First KATRIN Neutrino-Mass Measurement. *Physical review D*, 104(1):012005, 2021.

[4] M. Aker, M. Balzer, D. Batzler, A. Beglarian, J. Behrens, A. Berlev, U. Besserer, M. Biassoni, B. Bieringer, F. Block, et al. KATRIN: Status and Prospects for the Neutrino Mass and Beyond. *Journal of Physics G: Nuclear and Particle Physics*, 49 (10):100501, 2022.

[5] D. Araiza-Illan, D. Western, A. G. Pipe, and K. Eder. Systematic and Realistic Testing in Simulation of Control Code for Robots in Collaborative Human-Robot Interactions. In *Annual Conference Towards Autonomous Robotic Systems*. Springer, 2016.

[6] T. Asfour and F. Paus. Probabilistic Representation of Objects and Their Support Relations. In *International Symposium on Experimental Robotics (ISER)*, pages 510–519, 2020.

[7] T. Asfour, M. Wächter, L. Kaul, S. Rader, P. Weiner, S. Ottenhaus, et al. ARMAR-6: A High-Performance Humanoid for Human-Robot Collaboration in Real World Scenarios. *IEEE Robotics and Automation Magazine*, pages 108–121, vol. 26 (4), 2019.

[8] Atlas Collaboration ATLAS, G. Aad, T. Abajyan, B. Abbott, J. Abdallah, S. Abdel Khalek, O. Abdinov, R. Aben, B. Abi, M. Abolins, et al. Jet Energy Measurement and its Systematic Uncertainty in Proton–Proton Collisions at s= 7 s= 7 TeV with the ATLAS Detector. *The European Physical Journal C*, pages 1–101, vol. 75, 2015.

[9] W-J. Baek and T. Kröger. Safety Evaluation of Robot Systems via Uncertainty Quantification. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10532–10538, 2023. doi: 10.1109/ICRA48891.2023.10160598.

[10] W-J. Baek, C. Pohl, P. Pelcz, T. Kröger, and T. Asfour. Improving Humanoid Grasp Success Rate based on Uncertainty-aware Metrics and Sensitivity Optimization. In *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, pages 786–793. IEEE, 2022.

[11] W-J. Baek, C. Ledermann, T. Asfour, and T. Kröger. Combining Measurement Uncertainties with the Probabilistic Robustness for Safety Evaluation of Robot Systems. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 473–480, 2023. doi: 10.1109/IROS55552.2023.10342112.

[12] W-J. Baek, C. Ledermann, and T. Kröger. Uncertainty Estimation for Safe Human-Robot Collaboration Using Conservation Measures. In Ivan Petrovic, Emanuele Menegatti, and Ivan Marković, editors, *Intelligent Autonomous Systems 17*, pages 85–102, Cham, 2023. Springer Nature Switzerland.

[13] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP, and OIML. In *Evaluation of Measurement Data – Supplement 1 to the Expression of Uncertainty in Measurement, JGCM 101:2008*, 2008.

[14] S. Brechtel, T. Gindele, and R. Dillmann. Recursive Importance Sampling for Efficient Grid-based Occupancy Filtering in Dynamic Environments. In *2010 IEEE International Conference on Robotics and Automation*, pages 3932–3938. IEEE, 2010.

[15] W. L. Buntine and A. S. Weigend. Bayesian Back-Propagation. *Complex Systems*, pages 604 – 643, vol. 5, 1991. URL `https://api.semanticscholar.org/CorpusID:14814125`.

[16] Z. Cao, G. Hidalgo Martinez, T. Simon, S. Wei, and Y. A. Sheikh. OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[17] A.R. Cassandra, L.P. Kaelbling, and J.A. Kurien. Acting under Uncertainty: Discrete Bayesian Models for Mobile-Robot Navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 963–972, vol.2, 1996.

[18] D. Chen, V. Dietrich, Z. Liu, and G. von Wichert. A Probabilistic Framework for Uncertainty-Aware High-Accuracy Precision Grasping of Unknown Objects. *Journal of Intelligent & Robotic Systems*, pages 19–43, vol. 90 (1–2), 2018.

[19] M. Cisse, P. Bojanowski, E. Grave, Y. Dauphin, and N. Usunier. Parseval networks: improving robustness to adversarial examples. In *Proceedings of the 34th International Conference on Machine Learning*, ICML'17, pages 854–863, vol.70, 2017.

[20] G. Cowan. *Statistical Data Analysis*. Oxford science publications. Clarendon Press, 1998. ISBN 9780198501558. URL `https://books.google.de/books?id=ff8ZyW0nlJAC`.

[21] L. Deng. The MNIST Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Processing Magazine*, pages 141–142, vol. 29(6), 2012. doi: 10.1109/MSP.2012.2211477.

[22] S. Depeweg, J. M. Hernandez-Lobato, F. Doshi-Velez, and S. Udluft. Decomposition of Uncertainty in Bayesian Deep Learning for Efficient and Risk-sensitive Learning. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, 2018.

[23] N. E. Du Toit and J. W. Burdick. Robot Motion Planning in Dynamic, Uncertain Environments. *IEEE Transactions on Robotics*, pages 101–115, vol. 28 (1), 2011.

[24] A. N. Erkan, O. Kroemer, R. Detry, Y. Altun, J. Piater, and J. Peters. Learning probabilistic discriminative models of grasp affordances under limited supervision. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1586–1591, 2010. doi: 10.1109/IROS.2010.5650088.

[25] S. Farquhar, L. Smith, and Y. Gal. Try Depth Instead of Weight Correlations: Mean Field is a Less Restrictive Assumption for Variational Inference in Deep Networks. In *Bayesian Deep Learning Workshop At NeurIPS*, 2020.

[26] Joint Committee for Guides in Metrology (JCGM). Evaluation of Measurement Data — Guide to the Expression of Uncertainty in Measurement (JGCM 100:2008(E)). Technical report, JCGM, 2008.

[27] International Organization for Standardization. *ISO 12100:2010 Safety of Machinery - General Principles for Design - Risk Assessment and Risk Reduction*. International Standard. International Organization for Standardization, 2010.

[28] International Organization for Standardization. *ISO 10218: Robots and Robotic Devices : Safety Requirements for Industrial Robots*. International Standard. International Organization for Standardization, 2011.

[29] International Organization for Standardization. *ISO 13849-2:2015 Safety of Machinery — Safety-related Parts of Control Systems — Part 1: General Principles for Design*. International Standard. International Organization for Standardization, 2015.

[30] D. Fox. KLD-Sampling: Adaptive Particle Filters. In *Advances in Neural Information Processing Systems*, volume 14. MIT Press, 2001.

[31] M. Gandhi, H. Almubarak, Y. Aoyama, and E. Theodorou. Safety in Augmented Importance Sampling: Performance Bounds for Robust MPPI, 2022.

[32] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. Automatic Generation and Detection of Highly Reliable Fiducial Markers Under Occlusion. *Pattern Recognition*, pages 2280–2292, vol. 47 (6), 2014.

[33] J. Gawlikowski, C. R. N. Tassi, M. Ali, J. Lee, M. Humt, J. Feng, A. Kruspe, R. Triebel, P. Jung, R. Roscher, M. Shahzad, W. Yang, R. Bamler, and X. Zhu. A Survey of Uncertainty in Deep Neural Networks. *Artificial Intelligence Review*, pages 1513–1589, vol.56, 2023. doi: 10.1007/s10462-023-10562-9.

[34] A. K. Goins, R. Carpenter, W-K. Wong, and R. Balasubramanian. Evaluating the Efficacy of Grasp Metrics for Utilization in a Gaussian Process-based Grasp Predictor. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3353–3360, 2014.

[35] G. Grisetti, C. Stachniss, and W. Burgard. Improved Techniques for Grid Mapping with Rao-Blackwellized Particle Filters. *IEEE Transactions on Robotics*, pages 34–46, vol. 23 (1), 2007.

[36] J. Guiochet, Q. A. D. Hoang, M. Kaaniche, and D. Powell. Model-based Safety Analysis of Human-Robot Interactions: The MIRAS Walking Assistance Robot. In *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*, pages 1–7, 2013. doi: 10.1109/ICORR.2013.6650433.

[37] S. Haddadin, S. Haddadin, A. Khoury, T. Rokahr, S. Parusel, R. Burgkart, A. Bicchi, and A. Albu-Schäffer. On Making Robots Understand Safety: Embedding Injury Knowledge Into Control. In *The International Journal of Robotics Research*, pages 1578–1602, vol. 31(13), 2012.

[38] O. Hasan, W. Ahmed, S. Tahar, and M. S. Hamdi. Reliability Block Diagrams Based Analysis: A Survey. volume 1648, 2015. doi: 10.1063/1.4913184.

[39] M. Hein and M. Andriushchenko. Formal Guarantees on the Robustness of a Classifier Against Adversarial Manipulation. *Advances in Neural Information Processing Systems*, 30, 2017.

[40] E. J. Herron, S. R. Young, and T. E. Potok. Ensembles of Networks Produced from Neural Architecture Search. In *International Conference on High Performance Computing*. Springer, 2020.

[41] T. Hesterberg. Weighted Average Importance Sampling and Defensive Mixture Distributions. *Technometrics*, 37(2):185–194, 1995.

[42] J. Hill and D. C. Conner. Revisiting Uncertainty Propagation in Robotics. In *2020 SoutheastCon*, pages 1–8. IEEE, 2020.

[43] E. Hüllermeier and W. Waegeman. Aleatoric and Epistemic Uncertainty in Machine Learning: An Introduction to Concepts and Methods. *Machine Learning*, 110:457–506, 2021.

[44] Y. Ito, C. Srinivasan, and H. Izumi. Bayesian Learning of Neural Networks Adapted to Changes of Prior Probabilities. In *International Conference on Artificial Neural Networks*. Springer, 2005.

[45] D. Kappler, J. Bohg, and S. Schaal. Leveraging Big Data for Grasp Planning. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4304–4311, 2015. doi: 10.1109/ICRA.2015.7139793.

[46] A. Kendall and Y. Gal. What Uncertainties Do We Need In Bayesian Deep Learning For Computer Vision? In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 5580—-5590, Red Hook, NY, USA, 2017. Curran Associates Inc.

[47] G. Kochenberger, J-K. Hao, F. Glover, M. Lewis, Z. Lü, H. Wang, and Y. Wang. The Unconstrained Binary Quadratic Programming Problem: A Survey. *Journal of Combinatorial Optimization*, pages 58–81, vol. 28, 2014.

[48] H. Kurniawati and D. Hsu. Workspace Importance Sampling for Probabilistic Roadmap Planning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1618–1623, vol. 2. IEEE, 2004.

[49] B. Lacevic and P. Rocco. Kinetostatic Danger Field – A Novel Safety Assessment for Human-Robot Interaction. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2169–2174. IEEE, 2010.

[50] P. A. Lasota, T. Fong, and J. A. Shah. *A Survey of Methods for Safe Human-Robot Interaction*. Foundations and Trends in Robotics, 2014.

[51] P. A. Lasota, G. F. Rossano, and J. A. Shah. Toward Safe Close-Proximity Human-Robot Interaction with Standard Industrial Robots. In *IEEE International Conference on Automation Science and Engineering, (CASE)*, pages 339–344. IEEE, 2014.

[52] C. Liu and M. Tomizuka. Safe Exploration: Addressing Various Uncertainty Levels in Human Robot Interactions. In *2015 American Control Conference (ACC)*, pages 465–470. IEEE, 2015.

[53] Y. Luo, H. Bai, D. Hsu, and W. S. Lee. Importance Sampling for Online Planning Under Uncertainty. *The International Journal of Robotics Research*, pages 162–181, vol. 38, 2019.

[54] Z. Lyu, C-Y. Ko, Z. Kong, N. Wong, D. Lin, and L. Daniel. Fastened Crown: Tightened Neural Network Robustness Certificates. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 5037–5044, vol. 34, 2020.

[55] R. Mangal, A. V. Nori, and A. Orso. Robustness of Neural Networks: A Probabilistic and Practical Approach. In *2019 IEEE/ACM 41st International Conference on Software Engineering: New Ideas and Emerging Results (ICSE-NIER)*, 2019.

[56] E. E. Marushko and Alexander A. Doudkin. Methods of Using Ensembles of Heterogeneous Models to Identify Remote Sensing Objects. *Pattern Recognition and Image Analysis*, pages 211 – 216, vol. 30, 2020.

[57] J. A. Marvel and R. Norcross. Implementing Speed and Separation Monitoring in Collaborative Robot Workcells. *Robotics and Computer-Integrated Manufacturing*, pages 144–155, vol. 44, 2017.

[58] FTA NASA. Fault Tree Handbook With Aerospace Applications. *Prepared for NASA Office of Safety and Mission Assurance NASA Headquarters Washington, DC*, vol. 20546, 2002.

[59] M. Nava, A. Paolillo, J. Guzzi, L. M. Gambardella, and A. Giusti. Uncertainty-aware Self-Supervised Learning of Spatial Perception Tasks. *IEEE Robotics and Automation Letters*, pages 6693–6700, vol. 6(4), 2021.

[60] H. Nguyen and Q-C. Pham. A Probabilistic Framework for Tracking Uncertainties in Robotic Manipulation. *arXiv preprint arXiv:1901.00969*, 2019.

[61] A. Owen and Y. Zhou. Safe and Effective Importance Sampling. *Journal of the American Statistical Association*, pages 135–143, vol. 95 (449), 2000.

[62] P. Pauli, A. Koch, J. Berberich, P. Kohler, and F. Allgöwer. Training Robust Neural Networks Using Lipschitz Bounds. *IEEE Control Systems Letters*, pages 121–126, vol. 6, 2020.

[63] K. Petek, K. Sirohi, D. Büscher, and W. Burgard. Robust Monocular Localization in Sparse HD Maps Leveraging Multi-Task Uncertainty Estimation. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4163–4169. IEEE, 2022.

[64] C. Pohl and T. Asfour. Probabilistic Spatio-Temporal Fusion of Affordances for Grasping and Manipulation. *IEEE Robotics and Automation Letters 7*, pages 3226–3233, vol. 7(2), 2022.

[65] M. P. Polverini, A. M. Zanchettin, and P. Rocco. Real-time Collision Avoidance in Human-Robot Interaction based on Kinetostatic Safety Field. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4136–4141. IEEE, 2014.

[66] J. Postels, F. Ferroni, H. Coskun, and et al. Sampling-free Epistemic Uncertainty Estimation Using Approximated Variance Propagation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019.

[67] F. Proctor, M. Franaszek, and J. Michaloski. Tolerances and Uncertainty in Robotic Systems. In *ASME International Mechanical Engineering Congress and Exposition*, volume 58356, page V002T02A075. American Society of Mechanical Engineers, 2017.

[68] S. Reich, D. Mueller, and D. Andrews. Ensemble Distillation for Structured Prediction: Calibrated, Accurate, Fast - Choose Three. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP*, pages 5583–5595. Association for Computational Linguistics, 2020.

[69] C. Rubert, D. Kappler, A. Morales, S. Schaal, and J. Bohg. On the Relevance of Grasp Metrics for Predicting Grasp Success. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 265–272, 2017.

[70] E. Ruijters and M. Stoelinga. Fault Tree Analysis: A Survey of the State-Of-The-Art in Modeling, Analysis and Tools. *Computer Science Review*, pages 29–62 vol. 15, 2015.

[71] J. Santolaria and M. Ginés. Uncertainty Estimation in Robot Kinematic Calibration. *Robotics and Computer-Integrated Manufacturing*, pages 370–384, vol. 29, 2013.

[72] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel. The German Traffic Sign Recognition Benchmark: A Multi-Class Classification Competition. In *The 2011 International Joint Conference on Neural Networks*, pages 1453–1460. IEEE, 2011.

[73] S. Sun, C. Chen, and L. Carin. Learning Structured Weight Uncertainty in Bayesian Neural Networks. In *Artificial Intelligence and Statistics*. PMLR, 2017.

[74] S. Thrun. Probabilistic Algorithms in Robotics. *AI Magazine*, pages 93–93, vol. 21(4), 2000.

[75] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. Intelligent Robotics and Autonomous Agents. MIT Press, 2005. ISBN 978-0-262-20162-9.

[76] N. Vahrenkamp, T. Asfour, G. Metta, G. Sandini, and R. Dillmann. Manipulability Analysis. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 568–573, 2012.

[77] J. van den Berg, P. Abbeel, and K. Goldberg. LQG-MP: Optimized Path Planning for Robots with Motion Uncertainty and Imperfect State Information. *The International Journal of Robotics Research*, pages 895–913, vol.30, 2011.

[78] P. Wadekar, V. Gopinath, and K. Johansen. Safe Layout Design and Evaluation of a Human-Robot Collaborative Application Cell through Risk Assessment – A Computer Aided Approach. *Procedia Manufacturing*, 2018.

[79] P. Weber, G. M-O., C. Simon, and B. Iung. Overview on Bayesian Networks Applications for Dependability, Risk Analysis and Maintenance Areas. *Engineering Applications of Artificial Intelligence*, pages 671–682, vol. 25, 2012.

[80] L. Weng, H. Zhang, H. Chen, et al. Towards Fast Computation of Certified Robustness For Relu Networks. In *International Conference on Machine Learning*. PMLR, 2018.

[81] L. Weng, P-Y. Chen, L. Nguyen, M. Squillante, A. Boopathy, I. Oseledets, and L. Daniel. PROVEN: Verifying Robustness of Neural Networks with a Probabilistic Approach. In *International Conference on Machine Learning*, pages 6727–6736. PMLR, 2019.

[82] H. Zhang, T-W. Weng, P-Y. Chen, C-J. Hsieh, and L. Daniel. Efficient Neural Network Robustness Certification with General Activation Functions. *Advances in Neural Information Processing Systems*, vol. 31, 2018.

Acronyms

GUM Guide to the Expression of Uncertainty in Measurement. This guide serves as a reference to evaluating the accuracy of measurements..

HRC Human-Robot Collaboration. Mode in which humans and robot collaborate with each other. In contrast to Human-Robot cooperation, where humans and robots share their space, HRC refers to operation modes where both parties interact with each other by often sharing the same goal..

IS Importance Sampling. A specification of Monte Carlo sampling with the aim of generating samples in certain regions..

ISO International Organization for Standardization. The ISO defines standards (e.g., ISO 12100, ISO 13849,...) that state limits on safety-critical parameters in robot applications..

MC Monte Carlo. Describes a random sampling method..

NN Neural Network. Refers to all types of artificial neural networks from the learning domain..

VAE Variance in the Approximation Error. Enables to compare the accuracy of sampling algorithms by measuring the amount of error between the result of the approximation and the ground truth..

List of Figures

List of Tables

List of Algorithms