POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

EXECUTIVE SUMMARY OF THE THESIS

Verification, Validation, and Calibration under Uncertainty for a Scaled Experimental Rotor Model

LAUREA MAGISTRALE IN AERONAUTICAL ENGINEERING, INGEGNERIA AERONAUTICA

Author: GIUSEPPE ABATE

Advisor: PROF. GIUSEPPE QUARANTA

Academic year: 2020 – 2021

---

1. INTRODUCTION

Rotorcraft development is a highly complex process, joining different technical fields for a heterogeneous description. Apart from design and manufacturing stages, helicopters must satisfy specific certification requirements in order to operate in safe and controlled conditions, minimizing the failure occurrences and providing specific maneuvers in case of emergency. Compliance with certifications relies on both test and model analysis to cover all the possible design conditions, making simulations and computer modeling useful tools to derive quantities of interest. In addition, constructing reliable models allow to reduce the number of experiments, and to test different emergency cases without incurring particular danger situations for both pilots and vehicles. The ROtorcraft Certification by Simulation (ROCS) project tries to identify the possibilities and limitations of using simulations as a Means of Compliance (MOC) for helicopter certifications [1]. Model predictions must lead to accurate results and be compared with test data in order to be accepted as certification tools. This presumes an exact knowledge of the model and its input data, which is something difficult to accomplish due to both systematic and random measurement and model errors. In this view, uncertainties must be taken into account to not incur misleading conceptions and erroneous results. This thesis aims on establishing a complete uncertainty quantification (UQ) for a scaled experimental rotor model. Different numerical methods are exploited to derive statistical descriptions on input variables starting from test data (Inverse UQ) and vice versa (Forward UQ). Their implementation is possible thanks to a Dakota-MBDyn interface. Dakota is an open-source software that provides a vast variety of engineering-type problems, such as optimization, calibration, sensitivity analysis, and uncertainty quantification. Its main feature is the possibility of connecting with external dynamic solvers without concerning about their internal structure. It only provides iteratively input data and collects

responses to infer the desired analysis. MBDyn is a free-general purposes multibody dynamics analysis software developed at Politecnico di Milano, used to build the main rotor model which is able of reproducing the scaled experimental helicopter. Figure 1.1 shows a loop schematization of the Dakota-MBDyn interface.

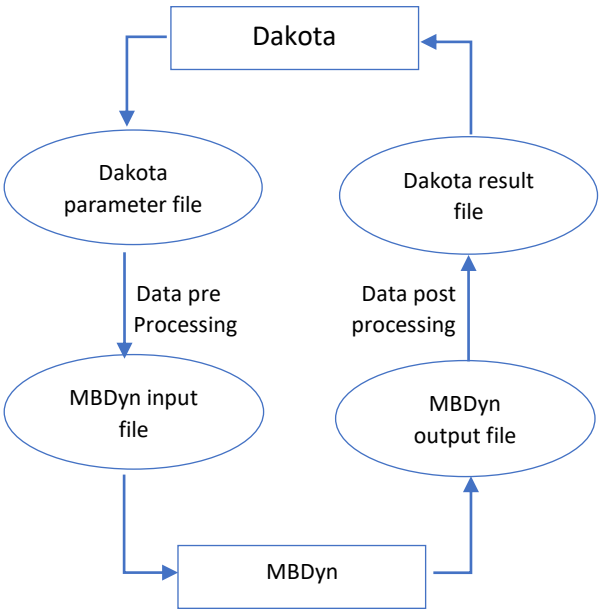

Figure 1.1: Interface loop Dakota-MBDyn

2. UQ METHODS

Uncertainty quantification is the process of identifying input uncertainties, propagating them through the computer code, and estimating statistical contents of quantities of interest. This thesis is mainly focused on *Inverse Propagating UQ* techniques which are used to infer calibration type problems in a statistical sense. From the knowledge of output values from experiments, in fact, a statistical assessment of unknown input parameters is performed. Two different methods, implemented in Dakota, will be performed for this purpose in a consecutive approach.

In *Bayes Calibration* (BC), prior distributions of input variables are updated through experimental data in order to derive posterior distributions which are more likely to reproduce the observed data. Given input parameters θ, a model M used to make predictions of the process and a set of experimental data D, the problem can be formalized using Bayes Theorem [2]:

$$P(\theta) = \frac{\pi(\theta)L(\theta)}{Z} \qquad (2.1)$$

in which $\pi(\theta)$ is the prior distribution, $L(\theta)$ the Likelihood function, $Z = \int \pi(\theta)L(\theta)d\theta$ the model evidence, and $P(\theta)$ the posterior distribution. More than computing posterior distribution itself, Bayes Calibration's main scope is to derive probability moments and expected values of generic functions $f(\theta)$. To estimate integrals and move into the parameter space, Bayes Calibration makes use of the Markov Chain Montecarlo (MCMC) technique. The initial probability information derived through BC can then be used and processed by a *Calibration Under Uncertainty* (CUU) algorithm, which identifies model parameters resulting in a "best fit" between experimental data and simulation results in a non-deterministic context. Specifically, input variable statistical moments can be calibrated according to output ones. Considering standard deviations as quantities indicating parameter uncertainty levels, CUU toolkits allow the identification of the accepted uncertainties on input variables in order to obtain a pre-set margin on response functions. The latter can be set as multiples of the output standard deviations. In Dakota, CUUs are implemented as nested models, in which an internal method exchanges information with an external one until convergence. The inner one is usually an uncertainty quantification method that derives statistical measures given input probability distributions, while, the outer one, is a least squares algorithm that tries to minimize the residual between model results and test data. *Mean Value Method* will be used as uncertainty quantification internal algorithm, while, NL2SOL [3] as least squares external one. The former computes all the statistical measures of interest from a single evaluation of response functions and their derivatives on input mean values, while the latter is a secant-based least squares algorithm that iteratively chooses between the Gauss-Newton Hessian approximation and this approximation augmented by a correction secant term.

3. VALIDATION

Before running the described methods on real conditions, a validation analysis is mandatory to establish the level of accuracy that can be addressed to the used toolkits. According to the guideline published by the American Society of Mechanical Engineers (ASME), "*validation is the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model"* [4]. Usually the predictive capability of the models is inferred by comparing the simulation results with real experimental data. Since this thesis concerns uncertainty quantifications, test data are directly created assuming input probability distributions. This ensures a direct comparison between the defined distributions and the calibrated ones, giving a direct insight into the confidence level of the simulations. Validation analysis is performed using a multibody model of Bo105 main rotor on MBDyn. Input variables are chosen from both aerodynamic and structural fields after performing a sensitivity analysis to derive the most influential ones. The latter is carried out by computing sensitivity indices through the Variance-based Decomposition (VBD) technique, which summarizes how uncertainties in model outputs can be linked to uncertainties of single input variables. Lift slope coefficients $C_{L_\alpha}s$ at Mach numbers $M = [0.3; 0.5; 0.6]$ and flap hinge stiffness value K_f are chosen as characteristic input variables. The aerodynamic related coefficients have been obtained after linear interpolation on the c81 data file of NACA 23012, used as aerodynamic profile to describe rotor blades. Test data consist of 100 values of mean thrust $\bar{T}$ and torque $\bar{Q}$ obtained through *Latin Hypercube Sampling* (LHS) forward uncertainty technique assuming Gaussian distributions, of known means and standard deviations, on input variables. These are given back to the Bayes Calibration method in order to derive input distributions and compare results. Bayes Calibration is performed using a total chain length of 100.000 samples and assuming uniform prior distributions for all the variables. The large number of iterations required to derive reliable statistical posterior moments makes Bayes Calibration impractical if performed with real model evaluations. For this reason, the usage of surrogate models to approximate responses is recommended especially for time-consuming single running models. In this case, a *Gaussian process* surrogate model collects 500 load evaluations to derive response surfaces used as approximation values. Tables 3.1 and 3.2 report the first two statistical moments of the real distributions and derived posterior ones. Mean values are accurately reproduced by the Bayes Calibration algorithm, with a relative error of less than 1% with respect to the nominal values, while standard deviations are overestimated and underestimated respectively for aerodynamic coefficients and flap hinge stiffness. A comparison between all probability distribution types for C_{L_α} at Mach 0.3 is shown in figure 3.1, where the posterior distribution is approximated as Gaussian type using mean and standard deviation derived from Bayes Calibration. The obtained results are then passed to a Calibration Under Uncertainty CUU nested model to derive the allowable uncertainties given constructed test data. The latter consist of thrust and torque standard deviations obtained again through the LHS technique, assuming input Gaussian distributions with means deriving from Bayes Calibration and arbitrarily chosen standard deviations. Comparison between real standard deviations and calibrated ones are shown in table 3.3. As can be seen, only aerodynamic lift coefficients are taken into consideration when performing CUU analysis. This is because the method shows correct operational behavior only when variables of the same sensitivity indices are considered. This ensures that only one values combination leads to the desired result, with no parameters of higher influence overcoming lower ones.

Variable	Mean	Std Deviation
$C_{L_\alpha}(0.3)$	0.1075	$5.375e-3$
$C_{L_\alpha}(0.5)$	0.1172	$5.860e-3$
$C_{L_\alpha}(0.6)$	0.1135	$5.675e-3$
$K_f\ [N/m]$	15035.467	751.774

Table 3.1: Real distribution moments

Variable	Mean	Std Deviation
$C_{L_\alpha}(0.3)$	0.1082	$7.844e-3$
$C_{L_\alpha}(0.5)$	0.1170	$1.298e-2$
$C_{L_\alpha}(0.6)$	0.1154	$1.034e-2$
$K_f\ [N/m]$	14995.33	269.75

Table 3.2: Posterior distribution moments

Variable	Real Std Dev	Cal Std Dev
$C_{L_\alpha}(0.3)$	$3.608e-3$	$3.223e-3$
$C_{L_\alpha}(0.5)$	$3.900e-3$	$4.356e-3$
$C_{L_\alpha}(0.6)$	$3.867e-3$	$3.803e-3$

Table 3.3: Standard deviation comparison

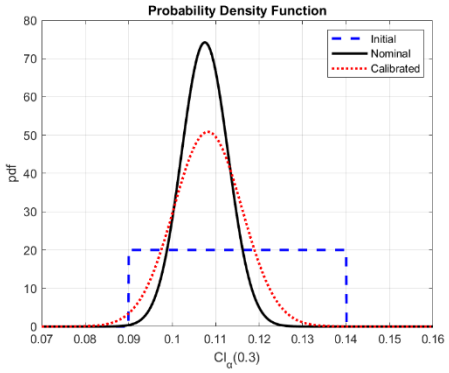

Figure 3.1: $C_{L_\alpha}(0.3)$ distribution comparison
(-- initial ; -- nominal ; -- calibrated)

4. EXPERIMENT

After deriving the confidence level that can be entrusted to uncertainty quantification and calibration methods, the explained techniques can be used in a real engineering application, where no complete prior knowledge is assumed and where the computational model is covered by uncertainty. The helicopter experiment was conducted in the large test chamber of GVPM (Galleria del Vento Politecnico di Milano) with 13.84 m wide, 3.84 m high, and length of 35 m. The helicopter model is composed of a fuselage and a rotor with four untapered and untwisted rectangular blades made of carbon-fiber composite materials. The rotor has a diameter of 970 mm and includes a complete swashplate mechanism, allowing collective and cyclic commands to be applied to the blades to obtain trim conditions at specific forces and moments. The fuselage is mounted on an internal metallic structure with a six-component strain gauge balance which measures forces and moments in all six degrees of freedom. A driving motor system, consisting of a brushless motor of 3.3kW continuous power, is installed inside the fuselage as well. The rotorcraft model was attached to a horizontal strut connected to a system of two orthogonal sliding guides, which allow the translation of the helicopter in both vertical and longitudinal directions as shown in figure 4.1. The helicopter is considered in hover at 4 different altitudes, respectively of four, three, two, and one time the rotor radius, in order to simulate both OGE (Out of Ground Effect) and IGE (In Ground Effect) conditions. Experimental data consist of averaged values of thrust, roll, pitch, and power (obtained multiplying the mean torque moment with the rotor rotational speed) after 10 seconds of flight test. These have been used to derive probability distributions of unknown parameters through the combination of Bayes Calibration and CUU algorithm.

Figure 4.1: Experiment setup

5. CALIBRATION AND UQ

MBDyn model of helicopter main rotor is used to simulate all the experimental conditions. Bayes Calibration is performed using the loads described

above at an altitude of four times the rotor radius ($H = 4R$) as test data. Input parameters have been selected from different fields, and after sensitivity analysis, in order to be representative of the uncertainty related to the model in conditions as general as possible. They consist of lift slope coefficients $C_{L_\alpha}s$ and zero lift drag coefficients $C_{D_0}s$ at Mach $M = [0.2; 0.3]$, pitch link stiffness K_p, bending stiffness EI_y and collective and cyclic commands. Aerodynamic coefficients have been extrapolated through linear and quadratic interpolation from the c81 data file of NACA 0012 used as aerodynamic profile for the rotor blades. As for the validation case, Bayes Calibration makes use of 100.000 chain samples and uniform prior distributions to derive posterior quantities. A Gaussian process surrogate model approximates response functions after collecting 300 real model load evaluations. The obtained parameter means and standard deviations, characterizing Normal distributions, are then passed to a Forward Uncertainty LHS analysis to derive load statistical contents and confirm results. Table 5.1 shows the obtained means and standard deviations for the loads at 4R altitude. Good agreement with experimental data is obtained since all the values differ from their respective means less than one standard deviation. Figure 5.1 reports the obtained Probability Density Function (PDF) for thrust at the same altitude. Calibration under Uncertainty (CUU) is performed considering only command controls due to the sensitivity requirement. Command allowable uncertainties, in form of standard deviations, are obtained considering arbitrarily chosen load standard deviations and maintaining all the other variables at their respective mean values obtained from Bayes Calibration. To confirm the obtained results, LHS computes load standard deviations when input commands follow Normal distributions with the new obtained standard deviations. Table 5.2 compares such results showing algorithm success. Since operating near the ground introduces special features in flight dynamic behavior, crucial point is to assess if the calibrated variables, and their distributions, are optimal estimations even in IGE conditions. For this reason, a forward uncertainty quantification is run characterized by the same features explained above, and with input parameters following Normal distributions with Bayes results. The model is considered operating at reduced altitudes. Particularly heights of two and one time the rotor radius are chosen as done in laboratory. The 3R case is not reproduced since it is still in OGE condition and no variations from the previous results are expected. No particular differences of Roll and Pitch moments are obtained with respect to the OGE case independently on the altitude considered. On the other hand, an increase in thrust and power is observed especially in the 1R case, since the ground effect acquires greater prominence at altitudes equal to or less than the rotor radius. Nevertheless, the great number of variables and uncertainty considered, allows to keep the differences between experimental data and response means below the one standard deviation threshold. Since model predictions of thrust and power (and equivalently torque) in IGE conditions underestimate experimental data, the last step of the thesis concerns the identification of a possible cause for such behavior. The idea is to address the difference between model computed loads and data, in the uncertainty related to the pitch link value, that can change in ground proximity due to the swashplate flexibility and its non-linear behavior. To set up the problem, only pitch link stiffness is assumed as an uncertainty variable following the Normal probability distribution constructed from Bayes Calibration results. A statistical assessment of thrust and torque coefficients is performed through LHS. Figure 5.2 shows the obtained results in terms of thrust coefficient normalized with respect to a nominal value obtained at 4R altitude. The vertical bars are constructed considering 3 times the obtained standard deviations. Also, the Fradenburgh equation is shown, which is usually used as reference result for rigid rotor test cases.

Load	Mean	Std Deviation
Thrust T $[N]$	26.1148	3.5505
Roll R $[Nm]$	-0.0069	0.5604
Pitch L $[Nm]$	-0.2501	0.5661
Power P $[W]$	353.4290	47.0523

Table 5.1: Load statistics at 4R altitude

Load	Cal. Std Dev	Exp Std Dev
$Thrust\ T\ [N]$	1.1638	0.8600
$Roll\ R\ [Nm]$	1.6650	0.1670
$Pitch\ L\ [Nm]$	1.6693	0.1670
$Power\ P\ [W]$	11.6875	11.6600

Table 5.2: Load standard deviations

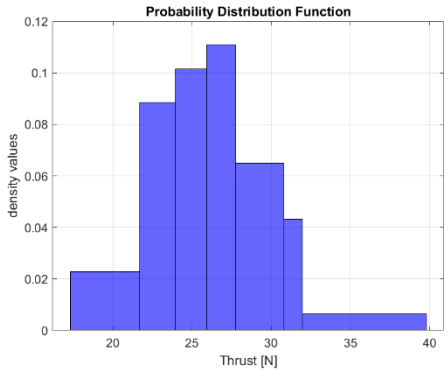

Figure 5.1: Thrust PDF

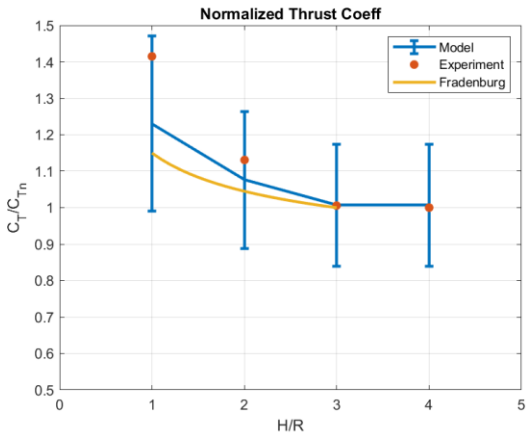

Figure 5.2: Normalized thrust coefficient
(° Experiment; -- Model; -- Fradenburg)

6. CONCLUSIONS

Dakota-MBDyn interaction has provided a comprehensive investigation tool to validate and assess different types of analyses on helicopter multi-body models. When no precise variable information is available, Bayesian Calibration is capable of deriving input parameter main statistical moments from experimental data, allowing the possibility of reproducing the test conditions within an uncertainty perspective. This initial information can be augmented by a Calibration under Uncertainty nested model, which in return provides the limit uncertainty conditions that can be accepted in order to obtain a pre-set margin on response functions. The cases taken into consideration in this work are of simple nature, where the rotorcraft, in hover, changes its altitude. More complete and difficult conditions, as for certification requirements, are left for future researches.

7. BIBLIOGRAPHY

[1] Quaranta, Giuseppe & Hoff, Stefan & Jones, Michael & Lu, Linghai & White, Mark. (2021). Challenges and Opportunities Offered by Flight Certification of Rotorcraft by Simulation.

[2] E. T. Jaynes and G. Larry. Bretthorst. Probability theory: the logic of science. Cambridge University Press, Cambridge, UK; New York, NY, 2003.

[3] Dennis, J. E., Gay, D. M., and Welsch, R. E.,ALGORITHM 573: NL2SOL–an adaptive nonlinear least squaresalgorithm, ACM Trans. Math. Software, 1981.

[4] White, M. D., Cameron, N., Lumsden, B., and Padfield, G. D., "Virtual Engineering in Skills Acquisition and Development in the Career of the Rotorcraft Engineer," Conference on Rotorcraft Virtual Engineering, Liverpool, UK, November 2016.

POLITECNICO DI MILANO
Corso di laurea Magistrale in Ingegneria Aeronautica
Dipartimento di Scienze e Tecnologia Aerospaziali

POLITECNICO
MILANO 1863

Verification, Validation, and Calibration under Uncertainty for a Scaled Experimental Rotor Model

Relatore: Prof. Giuseppe Quaranta

Giuseppe Abate, matricola: 915360

Anno Accademico 2020-2021

To my family

Abstract

Nowadays flight simulations and computer modeling are well-established instruments that run in parallel with the development of fixed and rotary-wing aircraft. Their primary advantage is the possibility of reproducing different environmental and flight conditions, reducing the time and costs of experimental campaigns and real flight tests. Furthermore, if consolidated and validated, they can help simulate emergency case maneuvers increasing pilots' safety. To obtain such benefits, a validation analysis is mandatory to ensure the correct operation of the tools, through the comparison between simulation results and experimental data. Both of them can be characterized by uncertainties and errors that must be taken into consideration to estimate the accuracy and the credibility of the processes. This thesis aims to infer a complete uncertainty quantification (UQ) analysis considering a multi-body model. This is constructed to reproduce experimental data obtained through a scaled helicopter. A Dakota-MBDyn interaction allows the execution of different methods, chosen according to the type of information available and the time required for a single run. Particular interest is posed on Bayes Calibration used to derive parameter probability distributions given test data. Previously, a validation analysis is executed to gain a higher understanding of the algorithms and to determine the level of confidence of the results.

Riassunto in italiano

I simulatori di volo e la modellazione sono strumenti che stanno acquisendo una maggiore importanza e rilevanza nello sviluppo di velivoli ad ala fissa e rotante. La possibilità di riprodurre un gran numero di condizioni ambientali e di volo, senza ricorrere a lunghe ed eccessive campagne sperimentali, li rendono vantaggiosi sotto molti punti di vista. Inoltre, possono essere utilizzati per simulare manovre di emergenza senza la possibilità che il pilota incorra in qualche rischio. Prima di poterli utilizzare in condizioni operative, è necessario che questi vengano validati confrontando i dati simulati con quelli sperimentali. Entrambi possono essere caratterizzati da incertezze ed errori che devono essere considerati e valutati per assicurare e stimare la credibilità dei risultati. Questa tesi ha come scopo la completa valutazione statistica di un modello multicorpo tramite quantificazione di incertezza. Il modello è stato concepito per simulare dati sperimentali ottenuti tramite un elicottero in scala. L'interazione tra Dakota e MBDyn permette l'esecuzione di vari metodi numerici per l'analisi statistica, scelti in base alle informazioni e alle risorse disponibili. Particolare attenzione è posta sulla Calibrazione di Bayes, che permette di ottenere le funzioni di distribuzione delle variabili in input grazie ai dati sperimentali. Prima di ciò, viene eseguita un'analisi di validazione dei modelli per ottenere il livello di confidenza attribuibile ai risultati.

Acknowledgments

First, I want to thank my thesis supervisor Prof. Quaranta for giving me the opportunity to work on this thesis and introducing me to the helicopter world. The Dakota-MBDyn implementation, with all its difficulties, made me grow from an engineering point of view.
Then, I want to thank Matteo and Neda for constantly helping me in the work development. Despite my constant questions, you have always been courteous and of help.
Thanks to my friends, from the most recent ones to the longest ones, for making all these years unforgettable and for supporting me regardless of the distance. We still have pages of history to write together.
Last but not least, a special thank you to my family, especially to mom and dad. All of this would never have been possible without you. It was tough, but we made it. I owe you everything.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Rotorcraft design is a highly complex process, joining different application fields for a heterogeneous technical description [1]. The design of aeronautical vehicles is usually divided into three different stages: conceptual design, preliminary design, and detailed design [2]. The conceptual design stage lies the foundation and the basic description of the machine according to the given specifications. These are subsequently controlled and increased in complexity within the preliminary design, while the detailed stage establishes the final concept that has to be passed to the manufacturing process. These early stages have a huge impact on rotorcraft development cost and time involving, therefore, particular attention. In addition rotorcraft vehicles must satisfy specific certification requirements in order to operate in safe and controlled conditions, minimize the likelihood of failure occurrence, and provide emergency recovery procedures in case the latter happen [3,4]. The first quantitative safety and reliability analyses are mainly provided at the final stages of the development cycle, which can be too late. Providing results at earlier stages can influence in a stronger way design decisions, leading to a more robust and dependable product [5]. To this aim, modern project cycles have become shorter and more parallelized, with the development of methods used to infer quantitative dependability predictions at early stages. Simulations and computer modeling are the main tools used to access results and to simulate helicopter behavior at different flight and environmental conditions. These are especially useful when little experimentally derived information is available and advantageous in terms of economy and safety. Constructing a reliable numerical model allows to reduce the number of experiments conducted in flight, and to replicate different emergency maneuvers without incurring risky conditions. Compliance with certifications usually relies on both test and model analysis in order to cover all the critical design conditions [6]. The compliance demonstration is the most time demanding and expensive part of the certification requirements due to the amount of test data needed in both ground and flight conditions. From this view, the ROtorcraft Certification by Simulation (ROCS) project is trying to identify the possibilities of applying flight simulations to obtain compliance with respect to the certification standards for helicopters and tiltrotors [7]. Specifically, in which conditions the simulations can be used as a substitute for flight testing as a Means of Compliance (MOC) for helicopter certifications (CS-29). These include risk conditions or environmental and aircraft ones that are difficult to obtain. Model predictions must lead to accurate results and be compared with test data in order to be accepted as certification tools. No clear foundation is present about simulation fidelity metrics that can encompass all the requirements giving rational tolerances. For these reasons, the acceptable tolerances are directly linked to the demonstration requirement. A verification and validation process is mandatory to assess the model's credibility. More precisely,

verification is the process of determining that the computational model accurately represents the mathematical one, while validation is used to determine if the model is a reliable representation of the real world by comparison with test data. All the steps involved presume the exact knowledge of the model and its input data, which is something difficult to accomplish due to both systematic and random errors on measurements and model representation. Uncertainties and related errors must be taken into account to not incur misleading conceptions or complete erroneous results. In addition, comparing model responses to test data in a non-deterministic sense requires the use of appropriate methods and valuation models, since no precise knowledge of the test conditions is available. The problem can be addressed either through an uncertainty quantification, which evaluates output statistical behavior given the input statistical characterization, or with a more deterministic approach relying on safety factors. The latter are in general not derived directly by probabilistic considerations, but are based on experience and used to increase the conservativeness of the design process. In the present work, various types of uncertainty-related methods are described and exploited in conjunction with multibody helicopter models. Particular attention is posed to Bayes Calibration techniques used to derive input parameter uncertainty distributions starting from experimental data. All the methods considered are not intended directly as certification tools, since much simpler operating flight and environmental conditions will be taken into account. They are instead exploited to verify the level of accuracy and the operative capabilities of Bayes calibration, even in conditions as general as possible. The proposed approaches could be used as guidelines to derive the possible applications when joined with more sophisticated and complete rotorcraft aerodynamic and structural models. All the mathematical techniques that will be presented are implemented in the software Dakota, while rotorcraft models are described using Mbdyn multibody solver. The interaction of the two allows a complete description and assessment of different rotorcraft problem types, including uncertain ones.

1.1 MBDyn

MBDyn is a free general-purposes multi-body dynamics analysis software developed at Politecnico di Milano. MBDyn features the integrated multidisciplinary simulation of multi-body systems, including nonlinear mechanics of rigid and flexible bodies subjected to kinematic constraints, along with smart materials, electric and hydraulic networks, active control, and essential elements of fixed-wings and rotorcraft aerodynamics [8]. Mbdyn is a command-line tool, since, to run it, executables have to be started from the terminal. All the indications and information needed for the simulation, such as problem type, numerical solver, and mechanical model geometry are passed through input files. The constrained nonlinear dynamics of both rigid and flexible bodies are formulated as sets of Differential-Algebraic Equations. Nodes are the basic structure of an MBDyn model, configuring kinematic degrees of freedom and the corresponding equilibrium equations [9]. Elements connect nodes, thus forming the main components of the multi-

body model and writing connective and constitutive contributions to equations. Maximum flexibility is left to the user concerning the identification and description of elements' orientations through local reference frames. It is also possible to exchange kinematic and dynamic information from external software or to describe in a completely arbitrary way functions representing loads distributions or input applied time histories.

1.2 Dakota

Dakota is an open-source software that provides a flexible interface between simulation codes and different analysis types. It can be used to cover a vast variety of engineering-type problems, such as optimization, calibration, sensitivity analysis, and uncertainty quantification. All the numerical methods and criteria are already implemented in specific libraries, without requiring a user-defined algorithm. Dakota can work either by internal dynamic solvers or by combination with external ones. Dakota's internal interface structure (called direct mode) is limited by the small number of solvers implemented in Dakota itself. The external structure (called fork mode) instead, allows Dakota to interface with a user-defined software and to implement all the analysis listed above. The main advantage is that Dakota does not affect the internal structure of the dynamic solver, considering it as a "*black box*". It only provides iteratively new input data and collects responses to infer the analysis taken into consideration. This exchange of information is possible thanks to text files that are passed from Dakota to the simulation code and vice-versa. A visual schematization of the loop running during operating conditions between Dakota and MBDyn is visualized in figure 1.1. A more detailed and complete description of the steps involved when interfacing the two will be presented in the following chapters. More information is available in the Dakota users' manual [10].

1.3 Overview

This thesis is structured as follows: in chapter 2 an introduction and explanation of uncertainty-related methods are presented with particular attention on those that will be used to infer the study cases. It has been chosen to give a complete overview, even for the methods not considered in this specific application, to provide a general understanding of the different ways in which uncertainty problems can be described. Chapter 3 is about validation analysis, in which computer model results will be compared with known quantities, to test methods' reliability when working in unknown conditions. After this, the real test case will be presented in chapter 4, where uncertainty calibration and quantification techniques are used on the helicopter multi-body model constructed to reproduce experimental data. Finally, all the considerations and limitations of the obtained results will be discussed in chapter 5.

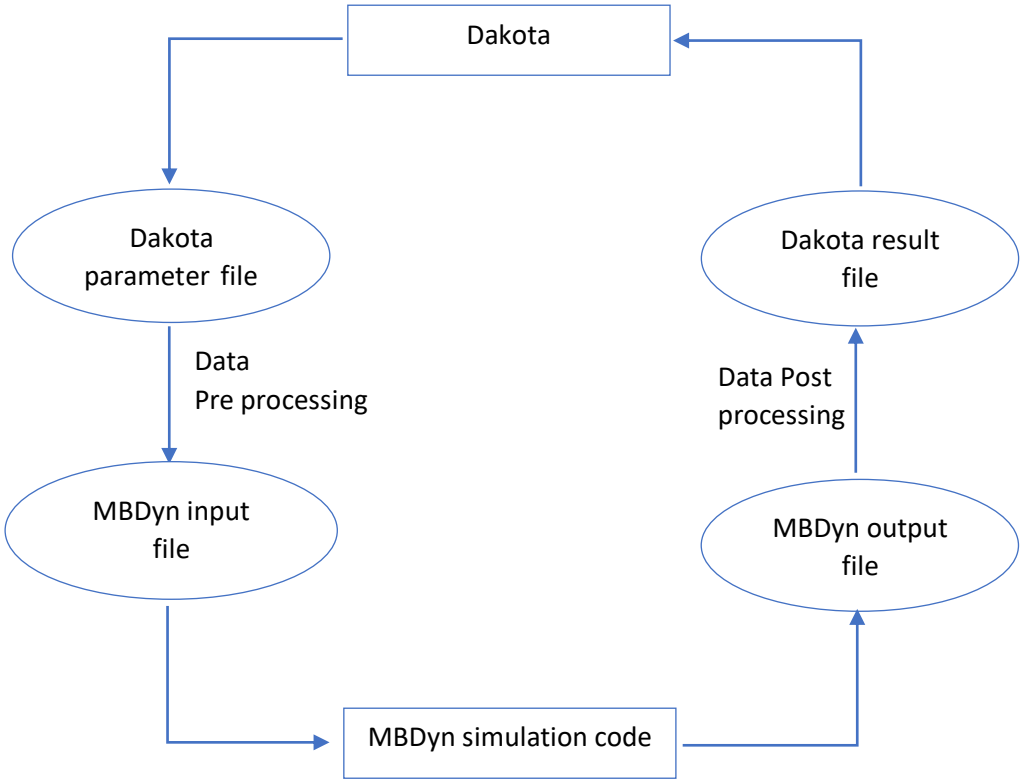

Figure 1.1: Interface loop Dakota-MBDyn

Chapter 2

Numerical Methods

2.1 Uncertainty Types

Computer modeling is nowadays a crucial part of studying and predicting physical and engineering system behaviors. The implemented mathematical model can be challenging and complex, leading to simulation codes that are very time-consuming even for a single run. Nevertheless, their usage is still highly recommended instead of direct reproductions and measurements of the real process. Moreover, experimental data can be achieved only for a limited range of parameters and environment values, depending on setting conditions and on the time/cost required for every evaluation. This limitation can be overcome using codes that in principle can reproduce the required analysis for the entire space of variables and conditions of interest, replicating the experiments and giving new results, leading to a deeper understanding of the phenomenon involved.

Simulation codes are deterministic, i.e., replicating the analysis when the same input values are provided. This is in contrast with the common conception of a physical process. Especially in engineering applications and experiments, reproducing the same outcome is not an easy task. A slight modification of configuration or environmental conditions can change the measurements, forcing one to repeat the process and average the results to obtain valuable measures with a sufficient level of confidence. In addition, various sources of uncertainties are embedded in real engineering problems, such as those coming out from production processes and material variability, initial conditions, and system surroundings. The model itself can be known up to a certain level of uncertainty due to physical simplifications, lack of knowledge, and not-so-detailed assumptions of the real process.
Considering and measuring these uncertainties has become a priority to investigate their effect on responses and ensure a more reliable and robust understanding of the problem under investigation. Various methods of uncertainty quantification (UQ) have been developed to achieve this task, and some of them will be reviewed in the next paragraphs.

An outline of the various types of uncertainties is presented [11], with a specific focus on those characterizing computer models.

1. Parameter uncertainties: Uncertainty related to variable values of computer code input. They specify particular features or applications of the simulation, driving

the dynamical behavior of the system. This type of uncertainty can be classified as either aleatory or epistemic [12]:

Aleatory uncertainties are irreducible variabilities. The randomness nature of the variable of interest is generally characterized by a probability distribution function (PDF) or by a cumulative distribution function (CDF), which describes the possible variable values and their probability of occurrence.

Epistemic uncertainties arise from a lack of knowledge of the variables under consideration. No probability or cumulative distributions can be assigned to them. The uncertainty is typically addressed through intervals where no values are more likely to occur than others. Epistemic uncertainties can be reduced with a deeper understanding and knowledge of the parameters and of the problem itself.

2. Model inadequacy: Running the model with the same input variables as the real process will not give the same result. Even if no parameter uncertainty is present, and the variables are characterized by deterministic values, there will be a discrepancy between model predictions and actual responses. This difference is model inadequacy.
3. Residual variability: The model is supposed to reproduce the real process results under the same conditions written as input variables. In practice, the process may not exhibit the same behavior every time due to intrinsic randomness. The residual variability is the deviation of results when all conditions are determined.
4. Observation error: When performing calibration analysis, experimental data are used to derive unknown model parameter values. The data are usually subjected to measurement errors which should be taken into account.

2.2 Uncertainty Quantification (UQ)

All kinds of uncertainties presented so far can influence the model and its capability of assessing the real dynamic behavior of the system. Nevertheless, the majority of the uncertainty quantification and analysis tools rely on the estimation of randomness via input variables. More precisely, Uncertainty Quantification (UQ) is the process of identifying input uncertainties, propagating them through the computer code, and performing statistical estimations of the response functions.

The way the propagation part of the analysis is handled by the method divides the Uncertainty Quantification into two main families [13]:

- *Forward Propagating UQs* are designed to evaluate the way input parameter uncertainties affect the result functions. They are the most used methodologies for statistical analysis and require the knowledge

and the characterization of input uncertainties through probability distributions.

- *Inverse Propagating UQs* work in the opposite way. From the knowledge of the output values or uncertainties, a statistical assessment of unknown input variables is performed. These methods are mostly used in calibration analyses since model responses must be known in advance in some manners (mainly by experimental campaigns)

The employment and implementation of such methods vary depending on the means available and the final purpose of the study.

An overview of the methods belonging to both classes of UQ is presented, with a particular focus on Inverse Propagation methodologies such as Bayes Calibration. This will serve as clarification on the choice of the methods used later on.

2.3 Forward Propagating UQ

This section provides a review of the main methods for forward propagating uncertainty quantification. Many other mathematical techniques, different from the ones listed below, have been implemented and are present in the literature.

The principal classes are:

- *Sampling Techniques*
- *Reliability Methods*
- *Stochastic Expansion Methods*
- *Interval Analysis Methods*

The description of uncertainties related to input variables through distributions (PDFs) is a common feature of all the methodologies presented. The differences rely on how these distributions are treated to propagate uncertainties and obtain statistical measures of responses. Interval Analysis Methods concern epistemic type input variables.

2.3.1 Sampling Techniques

Sampling techniques rely on the identification of samples. These are iteratively used as input values for the model to simulate system response functions. Statistical quantities are then constructed by simple operations on the results obtained. Sampling methods are the most effective algorithms for stochastic simulations thanks to their simplicity and convergence behavior assessed independently from the number of input variables. The major drawback is the time needed for running an uncertainty quantification. The amount of samples required to obtain reliable statistical results depends on the number of input

variables under investigation and their respective distribution functions, making these types of simulation often impractical for large problems. On the other hand sampling techniques are easy to implement since external to the simulation codes used to assess the dynamic behavior of the system.

MonteCarlo method (MC) is the most popular sampling algorithm. Given a vector of random variables $\boldsymbol{x} = \{x_i\}$ with $i = 1, \dots n$ and assigned probability distribution functions $\boldsymbol{P}(x_i)$, MC computes statistics on desired functions $\boldsymbol{f} = f_j(x_i)$ with $j = 1, \dots m$, by randomly sampling among $\boldsymbol{x}$ according to $\boldsymbol{P}$, and computing iteratively the response functions. Given N samples [14], the expected values and variances of $\boldsymbol{f}$ are respectively

$$E(\boldsymbol{f}) = \frac{1}{N}\sum_{i=1}^{N} \boldsymbol{f}_i \tag{2.1}$$

$$Var(\boldsymbol{f}) = \frac{1}{N^2}\sum_{i=1}^{N} Var(\boldsymbol{f}) = \frac{Var(\boldsymbol{f})}{N} \tag{2.2}$$

i.e., the sample means and variances.

MonteCarlo sampling is the simplest algorithm but also the more time-demanding. To obtain reliable results a large number of evaluations is needed depending on problem dimensionality.

Another sampling technique is the *Latin Hypercube Sampling* (LHS). LHS basic idea is the stratification of the probability density functions to improve the coverage of the input space. Every PDF is divided into *n* non-overlapping intervals of equal probability [15]. The number of intervals in a specific parameter region will depend on the shape of the probability distribution. One value from each interval is selected at random and then combined with the others. The Latin Hypercube Sampling derives from the augmentation in *N* dimensions of the "Latin Square" technique. The latter operates in 2D space, dividing the distributions of the variables in a grid, forming a matrix, and picking one sample from each row and column. LHS shows faster rate convergence than Monte Carlo sampling without any significant reduction of statistical responses accuracy. In addition, LHS ensures that the entire range of each input variable is completely covered, even in the tail regions of the probability distribution. Despite that, LHS still requires a large number of evaluations, making the algorithm inappropriate for code with a time-demanding single run.

2.3.2 Reliability Methods

Reliability methods provide a different approach in uncertainty quantification that is less computational and time demanding than classical sampling techniques. These methods are mainly focused on computing threshold problems, i.e. the probability that an aleatory function lies under a certain limit, as shown in figure 2.1. More precisely, given random variables $\boldsymbol{x}$ and a generic scalar function $g(\boldsymbol{x})$, reliability methods estimate the probability that g is below a certain level $\bar{z}$;

$$P[g(\boldsymbol{x}) \leq \bar{z}] = F_g(\bar{z}) \tag{2.3}$$

This probability calculation involves a multi-dimensional integral over the domain of interest. A mapping transformation can turn, the, usually non-normal and correlated, input variable distributions into independent Gaussian distributions with zero mean and unit standard deviation (standard normal distribution). This procedure allows more tractable calculations, making the multidimensional integrals function of a single parameter β called the reliability index. In the transformed space, β is the minimum Euclidean distance from the origin to the response surface.

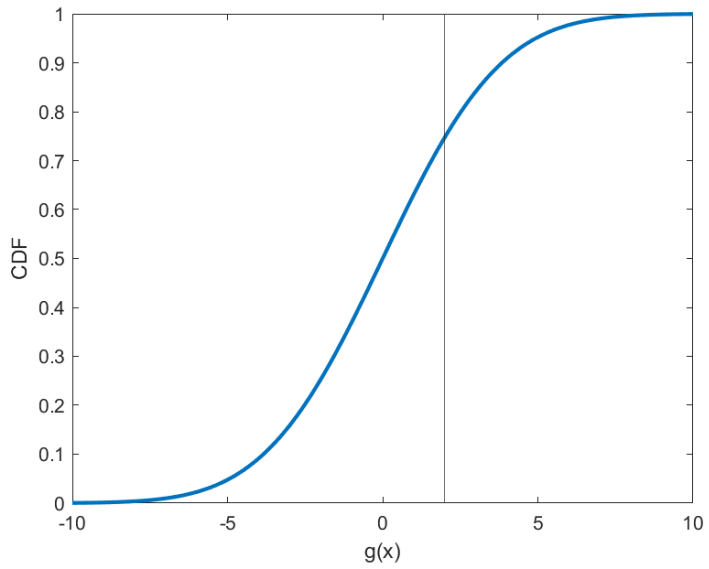

Figure 2.1: Threshold probability problem

Mean value method (MV, also called MVFOSM in [16]) is the simplest and less expensive of the reliability family. It evaluates all the statistical measures of interest (Mean, Standard Deviation, cumulative distribution function CDF, complementary cumulative distribution function CCDF) and response levels from a single evaluation of response functions and their derivatives computed at the uncertainty input variable means. The efficacy and accuracy depend on the particular numerical model, with less trusted results in the case

of non-linear models and response probabilities far from Gaussian ones. The expression for approximated means and variances are:

$$\mu_g = g(\mu_x) \tag{2.4}$$

$$\sigma_g^2 = \sum_i \sum_j Cov(i,j) \frac{dg}{dx_i}(\mu_x) \frac{dg}{dx_j}(\mu_x) \tag{2.5}$$

where x are the aleatory input variables, and $g(x)$ the response functions for which the statistical measures are needed. These two moments are then used for mapping from response thresholds to reliability indices and vice-versa:

$$\bar{z} \rightarrow \beta: \quad \beta_{CDF} = \frac{\mu_g - \bar{z}}{\sigma_g}, \quad \beta_{CCDF} = \frac{\bar{z} - \mu_g}{\sigma_g} \tag{2.6}$$

$$\beta \rightarrow \bar{z}: \quad z = \mu_g - \sigma_g \bar{\beta}_{CDF}, \; z = \mu_g + \sigma_g \bar{\beta}_{CCDF} \tag{2.7}$$

where β_{CDF} and β_{CCDF} are the reliability indices related to the cumulative distribution function and complementary distribution function (CDF/CCDF).
The CDF probability $p(g \leq z)$ and CCDF probability $p(g > z)$ are related to the reliability indices

$$p(g \leq z) = \Phi(-\beta_{CDF}) \tag{2.8}$$

$$p(g > z) = \Phi(\beta_{CCDF}) \tag{2.9}$$

with $\Phi()$ indicating the standard cumulative distribution function and recalling the assumption of response function Gaussian distributions.
Mean Value method's main advantage is the low computational cost since only a few model evaluations are needed to derive responses and derivatives with respect to input variables. It is the optimal solution in the case of long-running computer models. Since linear Taylor series approximation is adopted to recover statistical measures, other algorithms can be best fitting when high nonlinear behavior is expected from the simulation model.

MPP Search Methods are local reliability methods that apply an equality-constrained non linear optimization algorithm to compute the Most Probable Point (MPP), integrating around the latter to extrapolate probabilities. Given a limit state surface (the space domain where the response function is below the threshold), the MPP is defined as the shortest distance of the limit state function from the origin. Recalling that the transformed domain has the origin at the mean response value, this distance has the meaning of number of input standard deviations separating the mean response from the limit threshold. A mapping transformation $u = T(x)$ is defined to transform the general correlated non-normal distributions of the variables (x space) to uncorrelated standard

normal distributions (u space). Rosenblatt [17] and Nataf [18] transformations are generally used for this purpose. In the transformed domain the usually irregular-shaped probability contours are circular in nature, allowing more simple calculations. The Most Probable Point in the standard normal space has the highest probability of producing the limit state function, and it is the point contributing the most to the integral for statistical evaluations. The inverse transformation $x = T^{-1}(u)$ is then applied to retrieve the original correlated input variables. The Most Probable Point search involves an optimization problem with equality constraints, while different integral approximations can be used to extrapolate statistics measures. Two main reliability analysis types are defined according to what is specified and what is searched for.

The *Reliability index approach* (RIA) is a forward reliability analysis algorithm for computing probability/reliability levels and CDF/CCDF from prescribed response levels. The *Performance measure approach* (PMA) is an inverse reliability analysis algorithm for obtaining response levels starting from probability/reliability levels and CDF/CCDF. The difference lies in the definition of the objective function and equality constraint.

For RIA, the MPP search is formulated as computing the minimum distance in the u space from the origin to the $\bar{z}$ contour of the limit state response to achieve the response level $\bar{z}$:

$$\begin{aligned} &minimize\ u^T u \\ &with\ G(u) = \bar{z} \end{aligned} \tag{2.10}$$

where u is the vector centered at the origin in u-space and $g(x) = G(u)$ by definition. The optimal MPP solution u^* defines the reliability index β used to find the CDF/CCDF probabilities.

For PMA, the MPP search is about finding the minimum/maximum response function value after prescribing the distance from the origin in u space:

$$\begin{aligned} &minimize\ \pm G(u) \\ &with\ u^T u = \bar{\beta}^2 \end{aligned} \tag{2.11}$$

After obtaining the reliability index, integration approaches are needed to infer probabilities.
First Order Reliability Methods (FORM) [19] estimates the failure probability as:

$$P_f == \Phi(-\beta) \tag{2.12}$$

where Φ is the Gaussian distribution of the standard normal law. The precision of such approximation depends on the non-linearity of the model and surface of interest. More

precise evaluations can be obtained by *Second Order Reliability Methods* (SORM) involving curvature correction obtained by asymptotic analysis near the limit state surface:

$$p = \Phi(-\beta)\prod_{i=1}^{N-1}\frac{1}{\sqrt{1+\beta k_i}} \tag{2.13}$$

with k_i indicating the principal curvatures of the limit state function.

2.3.3 Stochastic Expansion Methods

Stochastic expansion methods are another class of non-sampling techniques for uncertainty quantification. They rely on the ability to approximate system input-output responses to assess statistical contents. Different methodologies are present in the literature according to the way such estimates are derived such as Stochastic Collocation (SC) and Polynomial Chaos Expansion (PCE). For the sake of brevity, only an introduction of the latter will be presented, referring to other literature for more details [20,21].

Polynomial Chaos Expansion (PCE) methods use orthogonal polynomial functions from Wiener-Askey [22,23] schemes to capture the functional relationship between a set of responses and a set of input variables. The value of these polynomials is to be orthogonal with respect to the specific set of probability distribution functions (PDFs), forming a basis for the latter, and allowing the solution of second-order random processes with finite variance. It is straightforward to understand the relevance of such methods since most of the engineering problems exhibit finite variance behavior.

A second order random process $Y(\boldsymbol{x})$, function of random parameters $\boldsymbol{x}$ can be expressed according to the PCE model as:

$$Y(\boldsymbol{x}) = \sum_{i=0}^{\infty} c_i \psi_i(\mathbf{x}) \tag{2.14}$$

where

- c_i is the i-th real deterministic coefficient
- ψ_i is the i-th Wiener-Askey polynomial basis of order *P*

The choice of the i-th adopted multivariate polynomial ψ_i depends on the statistical content of the model aleatory variables $\boldsymbol{x}$ in form of their probability distribution functions. As shown in table 2.1, for each PDF a particular Wiener-Askey polynomial class

is associated with orthogonal properties. Each of the multivariate polynomials involves products of related one-dimensional polynomials. How these are built is beyond the scope of this work and can be found in [24].
Ideally, the equality equation (2.14) is an exact representation of the process in the case of infinite summation. In reality, such operation is impossible and the summation must be truncated after *N* values, leading to the approximation

$$\tilde{Y}(\boldsymbol{x}) = \sum_{i=0}^{N} c_i \psi_i(\mathbf{x}) \tag{2.15}$$

The polynomial chaos expansion contains the complete set of bases up to the fixed specified order. The total number of terms N_T depends on the number of variables n and on the particular order of the expansion P:

$$N_T = 1 + N = \frac{(n+P)!}{n!\,P!} \tag{2.16}$$

The calculation of the deterministic coefficients of the expansion c_i depends on the specific application. Several methods are used and usually rely on the possibility of reducing some defined error between the real model and the approximated one. The coefficient tuning can be performed by projection methods, that follow the definition (eq. 2.14) and make use of the orthogonal properties, by computing the coefficients as expected values of the inner product between system responses and polynomials, that

DISTRIBUTION	POLYNOMIAL	SUPPORT RANGE
Normal	Hermite	$[-\infty\,,+\infty]$
Uniform	Legendre	$[-1\,,+1]$
Beta	Jacobi	$[-1\,,+1]$
Exponential	Laguerre	$[0\,,+\infty]$
Gamma	Generalized Laguerre	$[0\,,+\infty]$

Table 2.1: Wiener-Askey polynomials for different distributions

can be solved by integral approximation scheme techniques.

$$c_i = E[\psi_i(\mathbf{x}) \cdot \mathrm{Y}] \tag{2.17}$$

Whatever methodology is selected, the coefficient tuning needs function evaluations by direct simulation of the computer code to collect data points. Usually, PCEs are non-intrusive type algorithms and the coefficients are the result of post-processing of the set

of model evaluations. The required number of samples is well below with respect to sampling techniques since they are only required for tuning purposes and not to recover the entire probabilistic specter.
Statistical moments can be directly computed from expansion coefficients as

$$E[Y] = c_0 \tag{2.18}$$

$$\sigma_Y^2 = \sum_{i=1}^{N} c_i^2 \tag{2.19}$$

respectively the output mean and variance. Also, higher moment statistics can be extrapolated from direct coefficient operations.

2.3.4 Interval Analysis Method

All the methods discussed until now require input variables statistical description in order to extrapolate the uncertainty-related quantities of the output. This input-related knowledge not always is accessible since a great number of information is required to assess the statistical contents of the variables. Furthermore, uncertainties can be related to a mere lack of knowledge about some characteristics, instead of inherent randomness. When this is the case, epistemic variables must be considered within interval analysis. Since no probability distributions (PDF/CDF) can be assigned, an interval is linked to each variable. In this situation, it is not assumed that the value has a uniform probability within the interval, but that every value within the interval is a possible realization of the variable. Interval analysis purpose is to determine the resulting bounds of the output given the input ones.

Dempster-Shafter Theory of Evidence (DSTE) is the main method used to infer response function intervals. Mathematical details can be found in [25]. DSTE allows the user to specify more than one interval per variable, each of them linked to a basic probability assignment (BPA). BPA is a weight indicating how likely it is that the variable falls within the interval. The weights are not associated with some real probability but derive from the critical sense acquired from experience. Two primary and complementary uncertainties quantities are pulled out from evidence analysis, i.e, Beliefs, and Plausibility.

- *Belief* $B(\zeta)$ is interpreted to be the minimum likelihood that is associated with event ζ
- *Plausibility* $P(\zeta)$ is interpreted to be the maximum amount of likelihood that can be associated with event ζ

Belief and Plausibility define the lower and upper limits or intervals on probability values.

2.4 Inverse Propagating UQ

Inverse Propagating UQ is the process of quantifying the uncertainties of input parameters starting from experimental data or statistical quantities of the response functions. In this view, the problem can be seen as similar to calibration. The main difference relies on the information acquired after the process. Calibration (usually defined as deterministic calibration) determines the values of unknown variables resulting in a "best-fit" with respect to experimental data, i.e., trying to reduce as low as possible the difference between data and code responses. On the other hand, Inverse Propagating UQ tools produce statistical descriptions of the variables (distributions and moments), which allow more insight into the problem considering the uncertainties related to the model. In the sections below two methods will be explored:

1. *Bayes Calibration*
2. *Calibration Under Uncertainty (CUU)*

An implementation of the Forward Propagating UQ techniques listed above is also possible in their inverse form, though they are not very practical and efficient as their forward counterpart.

2.4.1 Bayes Calibration (BC)

Bayes calibration is a calibration method different from conventional ones. Instead of minimizing the difference between observed data and model outputs, it evaluates uncertainties of input parameters, in form of probability distributions, that produce output distributions in which test data are more likely. This is done by updating prior probability distributions of the input parameter, through the experimental data, in order to derive posterior ones and their statistical moments.
Given input parameters θ, a model M used to make predictions of the process and a set of experimental data D, the problem can be formalized using Bayes Theorem [26]:

$$P(\theta) = \frac{\pi(\theta)L(\theta)}{Z} \tag{2.20}$$

Each term presented in equation 2.20 has its own meaning and importance in inferring the calibration [27].

- $\pi(\theta) = P(\theta|M)$ is the prior distribution. This describes the probability of having a particular set of values θ given the model M before considering the experimental data D. Since it is indipendent from the data, the prior distribution

is an estimate of parameter values based on previous experiments, physical considerations and experiences about the problem under consideration.

- $L(\theta) = P(D|\theta, M)$ is the Likelihood function, i.e., the probability of seeing the data D collected, assuming (conditioned on) a particular choice of parameter values and the model. Different variable values will give different likelihoods, giving an insight into which parameter choice best describes the data.
- $Z = \int P(D|\theta, M)P(\theta|M)d\theta$ is the evidence or marginal Likelihood of the model. It quantifies, integrating over all possibile parameter values, if the model M is a good model to describe the observed data D.
- $P(\theta) = P(\theta|D, M)$ is the posterior variable distribution, that describes parameter probabilities after combining prior belief, likelihood function and normalizing with respect evidence.

Generating the posterior distribution is useful to derive model predictions, quantify uncertainties, and make correct guesses running the code in conditions not allowed or yet tried with experiments.
More than computing the posterior distribution itself, the main focus of Bayes Calibration, and of the sampling techniques used to generate results, is to derive probability moments and estimate the expected value of generic functions $f(\theta)$ when θ follows the posterior $P(\theta)$.

$$E_P[f(\theta)] = \int f(\theta)P(\theta)d\theta \tag{2.21}$$

This requires the computation of integrals with some numerical approximation methods. Most of them make use of the *Markov Chain Montecarlo* (MCMC) technique, and its related algorithms, to infer unknown posterior distributions by sampling and obtain quantities of interest. A detailed explanation of MCMC derivation is given in Appendix.

2.4.2 Bayes Calibration in Dakota

The Likelihood function used in Dakota is of Gaussian type, meaning that the differences between the model responses and the observed data are gaussian

$$d_i = q_i(\theta) + \varepsilon_i \tag{2.22}$$

where θ are the uncertain parameters, q_i are the model responses, d_i the experimental data, and ε_i is a random variable describing both measurements and modeling errors.

Considering n observation, the likelihood function is expressed as

$$L(\theta) = \frac{1}{\sqrt{(2\pi)^n |\Sigma_d|}} \exp\left(-\frac{1}{2} r^T \Sigma_d^{-1} r\right) \tag{2.23}$$

with r the residual vector ($r_i = d_i - q_i(\theta),\ i = 1, .., n$) and Σ_d the covariance matrix of the gaussian data uncertainties.

The negative log-likelihood is the misfit function

$$M(\theta) = \frac{1}{2} r^T \Sigma_d^{-1} r \tag{2.24}$$

Dakota uses as proposal distribution, to move into the parameter space, a multivariate normal (MVN) distribution with mean value centered at the current chain point and a covariance matrix designed to describe accurately the local curvature of the posterior distribution. The proposal covariance is the inverse of the Hessian of the negative log posterior

$$\nabla_\theta^2(-\log(P(\theta)) = \nabla_\theta^2 M(\theta) - \nabla_\theta^2(\log(\pi(\theta)) \tag{2.25}$$

The Hessian of the misfit function is computed from code simulations through Gauss-Newton approximation

$$\nabla_\theta^2 M(\theta) \approx \nabla_\theta q(\theta)^T \Sigma_d^{-1} \nabla_\theta q(\theta) \tag{2.26}$$

while the Hessian of the negative log prior distribution is assumed to be the inverse of the initial covariance

$$-\nabla_\theta^2(\log(\pi(\theta)) = \Sigma_0^{-1} \tag{2.27}$$

Through this equality is exact for normal prior distributions, it will be used also for other prior distribution types.

Dakota allows the implementation of different MCMC algorithms including Delayed Rejection DR, Adaptive Metropolis AM, and the combination of the two DRAM.

2.4.3 Calibration Under Uncertainty (CUU)

Calibration is the process of estimating unknown parameters according to experimental data. This allows the computer code to reproduce as close as possible the observed measurements when run with the calibrated variables and with the same conditions as the real test. Moreover, the acquired knowledge of the missing input values enables the

possibility of obtaining further information about the system behavior also in situations where no experiments have been performed.

The system output y, function of the input variables x, can be written as

$$y = f(x, \theta) + \delta \tag{2.28}$$

where $f(x, \theta)$ is the, usually nonlinear, model output, θ the parameters to be calibrated, and δ is an error term.

Calibration can be seen as an optimization algorithm where the objective function trying to be minimized is the residual between observed data and system outputs. Usually, the problem is formulated as a non-linear least square algorithm with the objective function $S(x)$ as the sum of the square of the residual terms.

$$S(x) = \frac{1}{2}\sum_{i=1}^{n} R_i^2 = \frac{1}{2}\sum_{i=1}^{n}(y_i - f_i)^2 \tag{2.29}$$

Calibration under Uncertainty (CUU) algorithm is concerned with identifying model parameters that result in a "best fit" between experimental data and simulation results in a non-deterministic sense. Statistical assessments of data can be derived by a test campaign in which multiple simulations are observed to extrapolate the required measurements. These could be statistical moments (mean, variance…) or percentiles of a CDF/CCDF. In the same way, the calibrated variables can be uncertain quantities themself characterized by specific distributions, whose parameters may be unknown.

In Dakota, CUU methods belong to "nested" models, in which, an inner loop algorithm exchanges information with an external algorithm. The inner one is usually an uncertainty quantification method that derives statistical measures given input probability distributions, the outer one is a calibration algorithm that confronts the uncertainty results of the model with their experimental respectives, changing accordingly the calibrated parameters. The process is repeated in a loop until stopped by a convergence criterion [28]. Figure 2.2 shows a visual scheme of the nested model where

- $d =$ design variables
- $u =$ uncertainty variables characterized by probability distributions
- $r_u(d, u) =$ response functions from the simulation
- $s_u(d) =$ statistics generated from UQ on the responses r_u

Any of the uncertainty quantification methods described so far can be used as an inner loop in a CUU study. Since every calibration iteration is performed after the statistical assessment, the overall number of iterations required for the entire process can become

impractical when sample techniques are used for uncertainty quantification. The choice, therefore, falls into one of the other algorithms in order to reduce the computational cost.

As for the inner case, also the outer loop can be equipped with one of the many least-squares algorithms present in the literature. Since it is beyond the scope of this work, an overview of these methods is not presented here and only the main concepts related to the one that will be used later on are reported.
Consider a generic non-linear least square problem

$$min\emptyset_x = \frac{1}{2}|g(x)|^2 \tag{2.30}$$

where x is an n-dimensional real vector and $g(x)$ is a m-dimensional real vector function. The Jacobian and Hessian of $\emptyset(x)$ are respectively

$$\nabla\emptyset(x) = J(x)^T g(x) \tag{2.31}$$

$$\nabla^2\emptyset(x) = J^T(x)J(x) + Q(x) \tag{2.32}$$

with $J(x) = g'(x)$ and $Q(x)$ denoting the second term

$$Q(x) = \sum_{i=1}^{m} g_i(x)\nabla^2 g(x) \tag{2.33}$$

In many cases the second term is difficult and impractical to compute and, as in the case of residuals, it tends to zero faster than the other terms. Ignoring this term will lead to the classical Gauss-Newton algorithm [29] which can be described in a few steps:

1. Choose an initial x_0
2. Repeat until convergence:

   Solve $J(x_k)^T J(x_k)s_k = -J(x_k)^T g(x_k)$
   Set $x_{k+1} = x_k + s_k$

In practice, the method is based on solving the linearized version of the problem

$$min\frac{1}{2}|J(x_k)s_k + g(x_k)|^2 \tag{2.34}$$

i.e., a linear Taylor expansion of $g(x)$ around x_k.

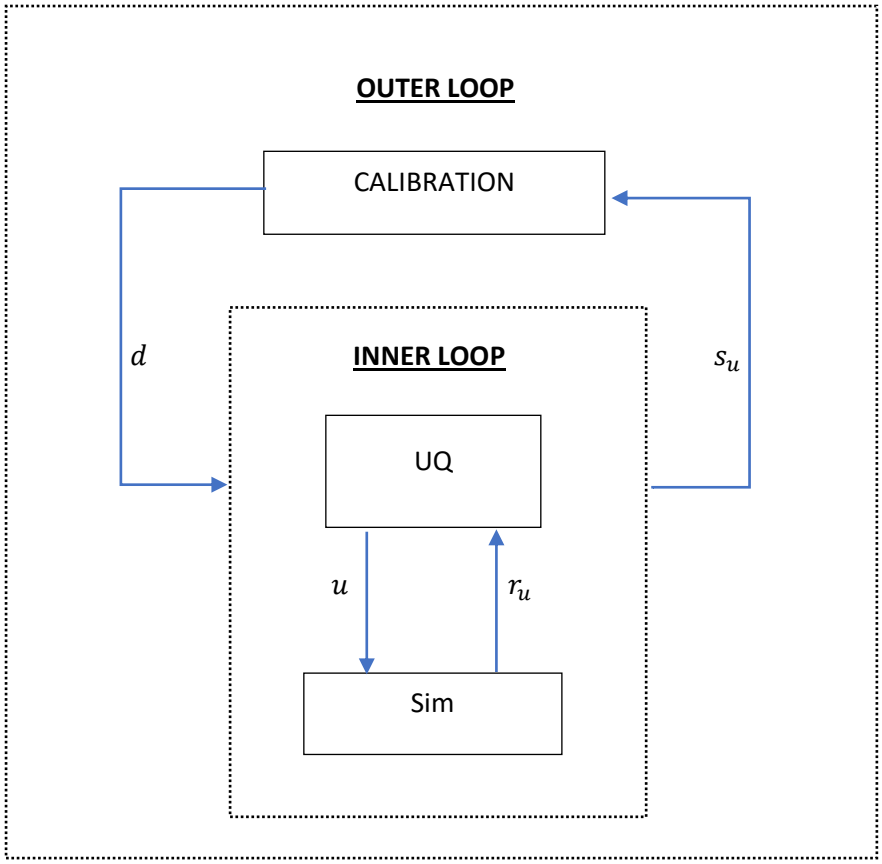

Figure 2.2: Dakota CUU nested model

The solution is sought through a quadratic approximation in which $Q(x)$ is not involved:

$$q_k = \frac{1}{2} g(x_k)^T g(x_k) + s_k^T J(x_k) g(x_k) + \frac{1}{2} s_k^T J(x_k)^T J(x_k) s_k \qquad (2.35)$$

The calibration method that will be used is NL2SOL [30], which is a secant-based least-squares algorithm that iteratively chooses between the Gauss-Newton Hessian approximation and this approximation augmented by a correction secant term S_k.

$$q_k = \frac{1}{2} g(x_k)^T g(x_k) + s_k^T J(x_k) g(x_k) + \frac{1}{2} s_k^T [\, J(x_k)^T J(x_k) + S_k \,] s_k \qquad (2.36)$$

This method provides more reliable results and convergence rates even when the initial residuals are large.

2.5 Surrogate Models

When running a method of any kind, time is one of the main features to keep in mind. Despite the utility and the number of information that can be gotten from a specific analysis, the latter could become meaningless if the required time is excessive. In this sense, time is one of the elements that tip the scales between doing or not a job. For example, Bayes Calibration algorithms could necessitate hundreds of thousands of iterations to converge or to, at least, obtain reasonable and accurate results, making them impractical when put besides physical models with time-demanding single runs.

Surrogate Models are inexpensive approximate models with the purpose of acquiring the main features of the real high-fidelity model. A surrogate model can be used as a tool to explore system responses in different configurations without relying on the real model itself. Obviously a specified set of responses and analyses are necessary to allow them to capture what is needed to approximate results. A wide choice and classes of surrogate models are present in literature, according to the approximation methodology. Without going into the merits, here a review of the Kringing emulator that will be used later on is given.

Gaussian Process or Kringing Interpolation is a data fits type surrogate model [31]. It uses a specified number of response functions from the high-fidelity model to construct a response surface from which to extrapolate the approximated results. The approximation of a real function $f(x)$ can be written as

$$\hat{f}(x) = g(x)^T\beta + \varepsilon(x) \tag{2.37}$$

where

$\hat{f}(x)$ is the Kringing emulator
$g(x)^T\beta$ is a trend function (usually a least squares fit to data)
$\varepsilon(x)$ is a Gaussian error model used to correct the trend function

The Gaussian error model $\varepsilon(x)$ is added in order to have zero uncertainty and error at the data points. The trend function value determines the various types of Kringing models:

1. Simple Kringing assumes that the trend function is a known constant, usually zero ($g(x)^T\beta = 0$)
2. Universal Kringing [32] uses a general polynomial model with the coefficients determined by least squares regression.
3. Ordinary Kringing is essentially Universal Kringing when the polynomial order of the trend function is zero ($g(x) = 1$)

The trend function is in some sense a "global" model for the entire design space based on the N given observations. The $\varepsilon(x)$ Gaussian error, instead, is used to "locally" deviate from the global model. It is assumed as having zero mean and a covariance error matrix depending on the distance between the two analyzed points:

$$Cov\big(\varepsilon(x), \varepsilon(x')\big) = \sigma^2 r(x, x') \tag{2.38}$$

where σ^2 is a known variance and $r(x, x')$ is the distance between points x and x'
If measurement errors are available, these can be explicitly added to the model through the covariance matrix

$$Cov\big(\varepsilon(x), \varepsilon(x')\big) = \sigma^2 r(x, x') + \Delta^2 \delta(x - x') \tag{2.39}$$

where δ is the Dirichlet function that is equal to 1 if $x = x'$ and 0 otherwise, and Δ^2 is the variance of the measurement error.

Chapter 3

Validation Analysis

Virtual engineering implementation with computer models is a foundation for establishing approaches that can drive and support decisions for all the rotorcraft lifecycle [33]. In whatever field of competence, from the initial structural design to the last certification requirements, building robust and reliable models is a useful step to confirm and acquire knowledge, without using a large number of physical tests. To achieve such benefits, a high level of confidence must be addressed to the used toolkits, in order to rely on them without incurring excessive errors. According to the guideline published by the American Society of Mechanical Engineers (ASME), "*validation is the process of determining the degree to which a model is an accurate representation of the real world from the perspective of the intended uses of the model"* [34]. In practice, the validation process aims to assess the predictive capability of the model. This is inferred by comparing the model responses with some real experiment data, usually obtained considering simple and not time demanding configurations. If the model faithfully reflects, according to some criteria, the real experimental setup and its results can be used to make the desired predictions with confidence. Validation is a necessary step when working with not well-known instruments or toolkits, to derive if a particular model, method, or algorithm can correctly assess the main aspects of the problem under investigation.

Before trying to reproduce experimental data, a validation analysis is performed to ensure the level of fidelity of the mathematical models. This is not intended as a way to verify the correct operation of Dakota toolkits, which have already been fully tested in different fields and with different dynamic solvers [35,36] but to give a perspective on how the methodologies work and how to assess and choose the different parameters governing the behavior of the numerical tools. As already stated, validation usually relies on specifically built experiments, used to extrapolate data to be compared with model responses. Since the interest of the presented work concerns uncertainty quantification and calibration, data are created directly on computer models assuming parameter distributions. This allows the comparison of the original and correct user-defined distributions with the derived and calibrated ones, giving an insight into the confidence level of the simulations.

3.1 Multibody Model

Validation analysis has been performed using a multi-body model of the Bo105 on MBDyn. The Bo105 is a small helicopter used for different purposes thanks to its extreme

maneuverability, such as transport, offshore, police, and military missions [37]. It has a four-bladed hingeless main rotor and a two-bladed teetering tail rotor working as a pusher on the left side of the helicopter. For simplicity reasons, only the main rotor has been implemented. The multibody model consists of a rotor with four rigid blades connected to the hub through spherical and deformable hinges, which allow rotations about feathering, lead, and flapping axes. The blades are at a constant chord (untapered) and twisted through a constant linear slope, with nodes positioned at the tip. Table 3.1 summarizes the main technical features of the model with blades inertia matrix written with respect to the hub. A flexible pitch link, connected to a complete swashplate mechanism, allows collective and cyclic controls to be applied to the rotor. The latter rotates counterclockwise with constant rotational speed. The aerodynamic model description relies on the definition of an aerodynamic body element, which assumes a rigid aerodynamic surface that takes its configuration from a single node.

Description	Symbol	Value	Unit
Angular Velocity	Ω	44.4	$[rad/s]$
Rotor Radius	R	4.912	$[m]$
Hinges offset	d	0.746	$[m]$
Flap Stiffness	K_f	15035.467	$[N/rad]$
Flap Damping	C_f	0.0	$[N \cdot s/rad]$
Lag Stiffness	K_l	70698.042	$[N/rad]$
Lag Damping	C_l	4093.662	$[N \cdot s/rad]$
Pitch Link Length	L_p	0.2	$[m]$
Pitch Link Stiffness	K_p	$1.0e9$	$[N/m]$
Pitch Link Damping	C_p	100.0	$[N \cdot s/m]$
Pitch Inertia	I_p	0.257	$[Kg \cdot m^2]$
Flap Inertia	I_f	140.001	$[Kg \cdot m^2]$
Lag Inertia	I_l	140.257	$[Kg \cdot m^2]$
Blade Span	b	3.812	$[m]$
Blade Chord	c	0.270	$[m]$
Twist slope	θ_r	$-6.2 * \pi/180$	$[rad/m]$

Table 3.1: Model technical features

The aerodynamic center and the velocity measurement point (the point where aerodynamic boundary conditions are evaluated) are aligned to the centerline of the element, which is assumed to be at 25% of the airfoil chord. Each blade is equipped with four Gauss integration points over the span, which will be used by the solver to evaluate forces at specific spanwise locations. Aerodynamic loads are computed based on Blade Element/Momentum Theory [38] using the c81 data file of NACA23012, used as airfoil for the entire blades. The inflow of the rotor is represented by a uniform model which works well in hover conditions. The overall rotor multibody model is shown in figure 3.1.

The validation carried out here mainly concerns the replication of a Bayes Calibration analysis to test the accuracy level of the method itself. The basic idea is to manually

generate, from known variables distributions, a number of test data and to give them back to the calibration model in order to retrieve the original distributions from the defined prior ones. Input variables are chosen from both aerodynamic and structural fields, in particular:

- Aerodynamic Lift Coefficient Slopes C_{L_α} at different Mach numbers
- Structural Stiffnesses from Pitch Link K_P , Flap Hinge K_F and Lag Hinge K_L

This particular choice tries to reflect a real engineering situation, in which aerodynamic conditions and manufacturing tolerances are characterized by uncertainties on their nominal values. As output responses/experimental data, mean Thrust $\bar{T}$ and Torque moment $\bar{Q}$ generated by the rotor is selected. These are extrapolated by averaging the last two rotor laps over 10 seconds of simulation, with constant collective and longitudinal commands of 6° and -2° respectively.

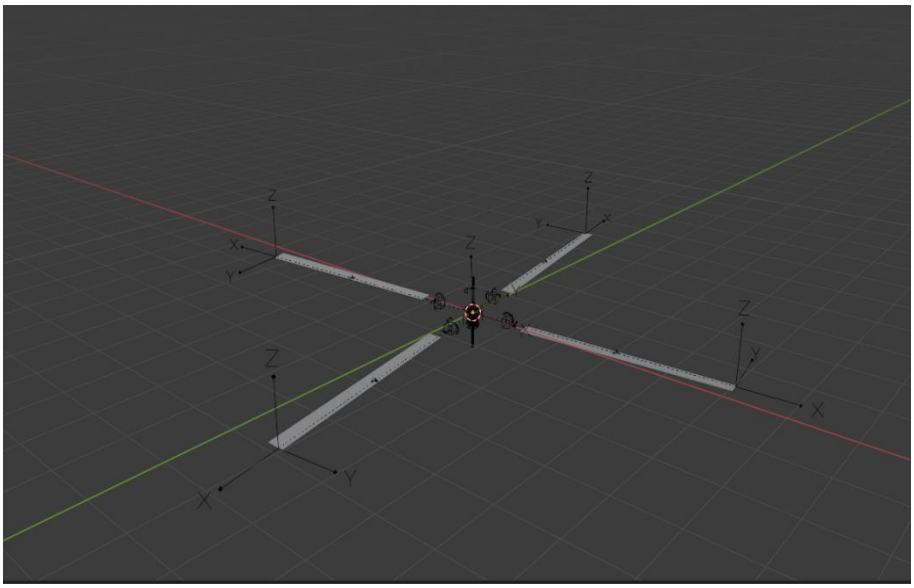

Figure 3.1: Bo105 model representation

3.1.1 Aerodynamic Interpolation

To retrieve C_{L_α} coefficients a linear interpolation scheme has been applied to the aerodynamic Lift coefficients contained in the c81 data file

$$C_L = C_{L_\alpha}\alpha + C_{L_0} \tag{3.1}$$

at Mach numbers $M \in [0.1\,;0.6]$ and with angle of attacks $\alpha \in [-10°\,;10°]$, considering the commands amplitudes and the tip velocity of the rotor $v_{TIP} = \Omega R$. Results of such linear assumption are displayed in table 3.2 and figures from 3.2 to 3.7, which compare the nominal lift coefficient values with the interpolated ones. The interpolation scheme

has been implemented using the "*polyfit*" function on Matlab. A python script, "*airfoil.py*" rewrites the overall c81 data file, changing the Lift Coefficient values in the previous range of application.

Mach values	C_{L_α}	C_{L_0}
0.1	0.1075	0.1270
0.2	0.1075	0.1270
0.3	0.1075	0.1270
0.4	0.1126	0.1270
0.5	0.1172	0.1405
0.6	0.1135	0.1327

Table 3.2: Interpolated aerodynamic lift coefficients

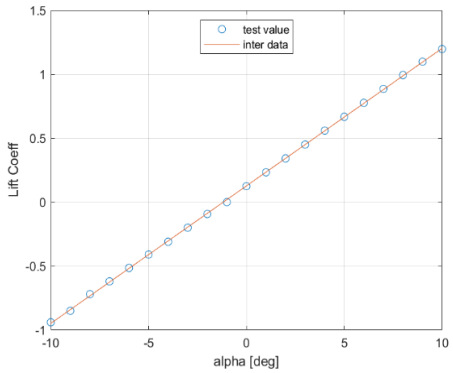

Figure 3.2: $C_L(0.1)$ comparison
(° test data, -- interpolated data)

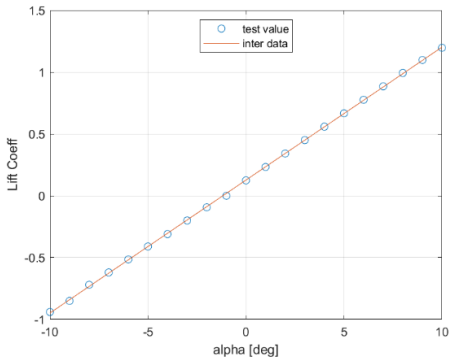

Figure 3.3: $C_L(0.2)$ comparison
(° test data, -- interpolated data)

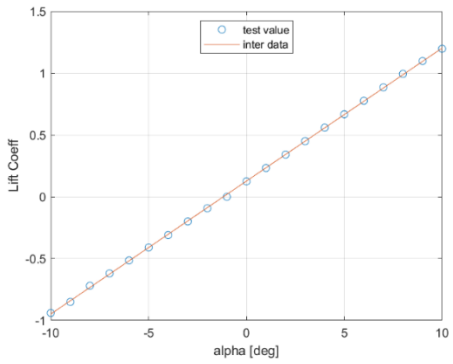

Figure 3.4: $C_L(0.3)$ comparison
(° test data, -- interpolated data)

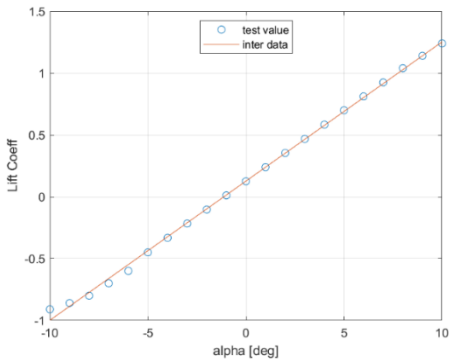

Figure 3.5: $C_L(0.4)$ comparison
(° test data, -- interpolated data)

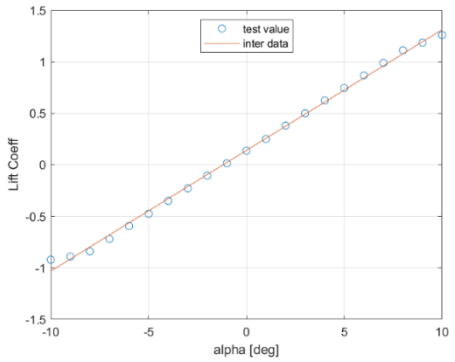

Figure 3.6: $C_L(0.5)$ comparison
(° test data , -- interpolated data)

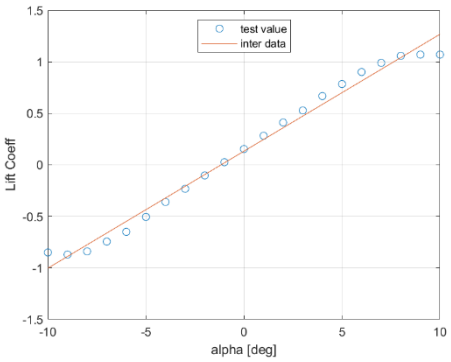

Figure 3.7: $C_L(0.6)$ comparison
(° test data , -- interpolated data)

3.2 Dakota-MBDyn

As already mentioned, the main advantage of the Dakota toolkit is of being external to dynamic simulation codes, considered as "*black boxes*". Regardless of the type of study conducted, Dakota provides a set of, user-defined, input variables and collects the response functions to derive final results, without concerning about the simulation tool's internal structure. This is possible thanks to a user-defined interface that allows Dakota to call and communicate with external scripts and solvers. Usually, Dakota writes and iteratively changes the input parameters of the dynamic solver and retrieves each response after these have been transferred to specific text files. All the operations involved are written in a shell script, namely *"driver.sh"*, which is called by Dakota input file and contains all the terminal commands that have to be executed to run the external scripts or software. Figure 3.8 shows a graphical representation of the steps involved in running a Dakota-MBDyn simulation with both aerodynamic and structural parameters, contained respectively in the *c81* data file and *parameters.set* text file. These are independent of the type of analysis required and will be the same for all the studies. The steps can be described in sequence as:

1. Specification of input variables, expected outputs, and analysis type, with a related algorithm, in Dakota input file *Dakota.in*
2. Insertion of input values in *airfoil.py* and *parameters.set*, creating two new files, respectively *airfoil_set.py* and *parameters_final.set*
3. Execution of *airfoil_set.py* to create the new c81 aerodynamic data file
4. Execution of Mbdyn with multibody model conditioned by *parameters_final.set* file
5. Execution of *data.sh* shell script to select the desired response quantities from the simulation and write them in *Loads.out* text file

6. Execution of *mean.py* python script to average *Loads.out* values and write the final mean responses in *results.out* text file

Steps from 2. to 6. are iteratively repeated until the number of iterations, specified in the Dakota input file, is achieved. Final analysis outputs are extrapolated at the end by Dakota, considering all the evaluated mean responses.

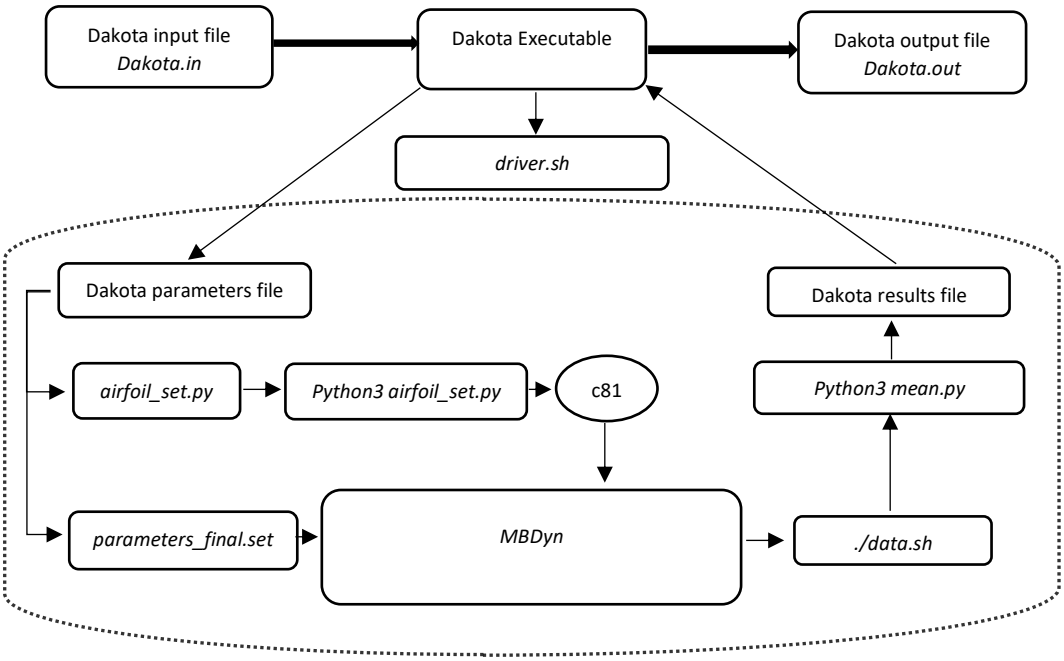

Figure 3.8: Dakota-MBDyn interaction scheme

3.3 Sensitivity Analysis

Uncertainty quantification analysis is usually run in tandem with sensitivity one. The latter relates how the uncertainty in the output of a mathematical model or system can be divided into different sources of uncertainties in its inputs [39]. Determining the impact of a variable under sensitivity analysis can be useful for different purposes:

- Testing the robustness of a model in presence of uncertainties
- Increasing the understanding of the relationships between inputs and outputs
- Reducing the uncertainty by identifying the variables that most affect outputs and should therefore be given more attention

Dakota allows sensitivity studies under its design and analysis of computer experiments (DACE) capabilities, which seek to extract as much trend data from a parameter space as possible using a limited number of sample points. Many DACE techniques have been

developed in order to characterize the behavior of the response functions of interest through the parameter ranges of interest. Here sensitivity analysis is carried out by computing sensitivity indices through Variance-based Decomposition (VBD). VBD is a global sensitivity method that summarizes how the uncertainties in model outputs Y can be distributed to uncertainties of single input variables x. Two primary measures are used: the main effect sensitivity index S_i and the total effect sensitivity index T_i:

$$S_i = \frac{Var_{x_i}[E(Y|x_i)]}{Var(Y)} \tag{3.2}$$

$$T_i = \frac{E[Var(Y|x_{-i})]}{Var(Y)} = \frac{Var(Y) - Var(E[Y|x_{-i}])}{Var(Y)} \tag{3.3}$$

where $Y = f(x)$, and $x_{-i} = (x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_m)$

The main effect sensitivity index S_i corresponds to the fraction of the uncertainty in the output Y, that can be related to the input variable x_i alone, while, the total effect sensitivity index T_i, corresponds to the fraction of output uncertainty that can be attributed to input x_i and its interaction with other variables. These are obtained by comparing the variance of the conditional expectation against the total output variance.

Three different sensitivity analyses have been conducted to separate aerodynamic and structural contributions. Latin Hypercube Sampling (LHS) technique generates samples from the defined probability distributions shown in table 3.3 to derive sensitivity indices. For a specified number of samples N in Dakota input file, and a number of non-deterministic variables M, variance based decomposition requires the evaluation of $N(M + 2)$ samples. $N = 200$ has been set for all the simulations. Input variables are iteratively updated following Gaussian distributions with mean value μ equal to the nominal values, and standard deviation σ such that the overall parameter space, defined as $\mu \pm 3\sigma$ produces a 15% deviation from nominal values. Results of sensitivity analyses with respect to mean thrust $\bar{T}$ are shown in tables 3.4, 3.5, 3.6. Firstly aerodynamic and structural variables are considered separately in order to derive the most useful ones on each field. Then a mixed sensitivity analysis extrapolates the indices on the parameters having the higher sensitivity indices from the aerodynamic and structural fields respectively. This final set of variables will be the one used in the uncertainty quantification later on. No big differences can be noted on main sensitivity index values S_i with respect to total ones T_i. This means that the output uncertainty can be mainly linked to single variables, with no particular interaction contributions. As expected Lift coefficients dominate the behavior of forces and moments generated by the rotor, with half of the output variance dictated by C_{L_α} coefficient at $M = 0.3$. This will be helpful in understanding how Bayes Calibration works in different configurations and with variables of different nature and importance.

Variable	Distribution Type	Mean	Std. Deviation
$C_{L_\alpha}(0.1)$	Gaussian	0.1075	$5.375e-3$
$C_{L_\alpha}(0.2)$	Gaussian	0.1075	$5.375e-3$
$C_{L_\alpha}(0.3)$	Gaussian	0.1075	$5.375e-3$
$C_{L_\alpha}(0.4)$	Gaussian	0.1126	$5.630e-3$
$C_{L_\alpha}(0.5)$	Gaussian	0.1172	$5.860e-3$
$C_{L_\alpha}(0.6)$	Gaussian	0.1135	$5.675e-3$
$K_f\ [N/rad]$	Gaussian	15035	751
$K_l\ [N/rad]$	Gaussian	70698	3534
$K_p\ [N/m]$	Gaussian	$1.0e9$	$5.0e7$

Table 3.3: Input variable distributions

Variable	Main Sensitivity Index S_i	Total Sensitivity Index T_i
$C_{L_\alpha}(0.1)$	$7.013e-4$	$2.615e-4$
$C_{L_\alpha}(0.2)$	$2.270e-3$	$3.709e-3$
$C_{L_\alpha}(0.3)$	$5.424e-1$	$5.153e-1$
$C_{L_\alpha}(0.4)$	$2.502e-3$	$2.107e-3$
$C_{L_\alpha}(0.5)$	$1.924e-1$	$1.905e-1$
$C_{L_\alpha}(0.6)$	$2.395e-1$	$2.457e-1$

Table 3.4: Aerodynamic sensitivity results with respect to $\bar{T}$

Variable	Main Sensitivity Index S_i	Total Sensitivity Index T_i
$K_f\ [N/rad]$	$8.938e-1$	$8.893e-1$
$K_l\ [N/rad]$	$7.458e-2$	$8.763e-2$
$K_p\ [N/m]$	$2.417e-4$	$1.185e-5$

Table 3.5: Structural sensitivity results with respect to $\bar{T}$

Variable	Main Sensitivity Index S_i	Total Sensitivity Index T_i
$C_{L_\alpha}(0.3)$	$5.424e-1$	$5.153e-1$
$C_{L_\alpha}(0.6)$	$2.396e-1$	$2.457e-1$
$C_{L_\alpha}(0.5)$	$1.924e-1$	$1.905e-1$
$K_f\ [N/m]$	$7.435e-2$	$8.093e-2$

Table 3.6: Mixed sensitivity results with respect to $\bar{T}$

3.4 Bayes Calibration

Sensitivity analysis has been helpful in identifying the suitable parameters for Bayes Calibration, namely, aerodynamic Lift Slope Coeffients $C_{L_\alpha}s$ at $Mach = 0.3|0.5|0.6$ and the Flap Hinge Stiffness K_f. As discussed in the previous chapter, to derive posterior and most probable density distributions of input parameters, Bayes Calibration techniques make use of experimental data. Since the scope of validation analysis is to compare results with known quantities, these latter are manually created by taking out response function samples of mean thrust $\bar{T}$ and mean torque moment $\bar{Q}$ from a forward uncertainty quantification analysis. By doing so, a direct comparison of calibrated and original distributions is available. To clarify, a visual scheme of the steps involved in the validation process is shown in figure 3.9. Experimental data consist of 100 thrust and moment samples, extrapolated by Latine Hypercube Samples (LHS) technique, with Gaussian input distributions. Parameter means and standard deviations are chosen to be the same as sensitivity analysis. Statistical normal distributed error with zero mean and variance of 0.25 is added to the data in order to simulate measurement errors. A Bayes Calibration study is then executed through a DRAM algorithm, starting with an initial uniform distribution for all the parameters and a total chain length of 100.000 samples. An initial "burn-in" of 5000 samples is set up in order to eliminate points in low probability regions. The large number of chain samples, required to derive reliable results, makes it mandatory to use an emulator. For this reason, a Gaussian process surrogate model makes use of 500 real model evaluations to collect load outputs and extrapolate response surfaces that will be used for calculating likelihood functions for the Bayes calibration. The trend function of the emulator is a reduced quadratic polynomial that includes main effects but not mixed interaction terms. In addition to the lower computational cost, another advantage of using a surrogate model is the possibility of having easy access to derivatives. These are computed by default in a global optimization process to obtain the hyperparameters of the Gaussian process and will be used to derive the covariance of the proposal distribution used to move into the chain space. As discussed, Dakota uses a multivariate normal (MVN) jumping distribution to create new points within a Markov chain. That is, a new point in the chain is determined by sampling within an MVN probability density with prescribed covariance that is centered at the current chain point. Derivatives allow to approximate, through the Gauss-Newton technique, the Hessian of the misfit function (negative log-likelihood), and to use it as proposal covariance, decreasing the rejection rate of the chain. Table 3.7 shows the results of the Bayes Calibration process with the first four statistical moments of the computed posterior distributions. Markov Chain MonteCarlo methods, as explained above, are not intended to derive distributions directly, but to approximate integrals of the latter. Table 3.8, instead, reports a comparison between prior and calibrated variable distributions considering only means and standard deviations.

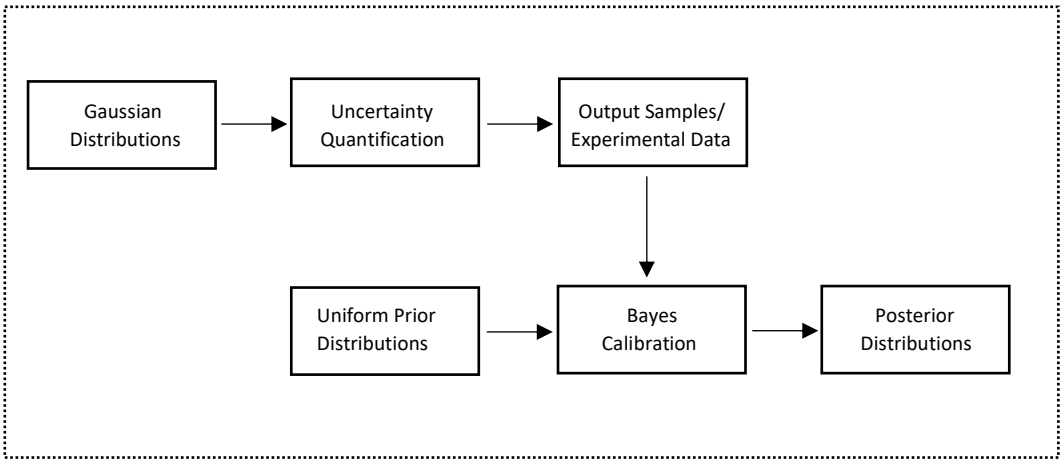

Figure 3.9: Bayes Calibration validation scheme

Variable	Mean	Std. Deviation	Skewness	Kurtosis
$C_{L_\alpha}(0.3)$	$1.082e-1$	$7.844e-3$	$-1.468e-1$	$-6.556e-1$
$C_{L_\alpha}(0.5)$	$1.170e-1$	$1.298e-2$	$-1.471e-1$	$-9.951e-1$
$C_{L_\alpha}(0.6)$	$1.154e-1$	$1.034e-2$	$3.146-2$	$-5.321e-1$
$K_f\ [N/m]$	14995.33	269.75	$1.936e-2$	-1.1016

Table 3.7: Bayes calibration results

	Uniform distribution		Calibrated distribution	
Variable	**Lower Bound**	**Upper Bound**	**Mean**	**Std Deviation**
$C_{L_\alpha}(0.3)$	0.09	0.14	0.1082	$7.844e-3$
$C_{L_\alpha}(0.5)$	0.09	0.14	0.1170	$1.298e-2$
$C_{L_\alpha}(0.6)$	0.09	0.14	0.1154	$1.034e-2$
$K_f\ [N/m]$	11000.00	18000.00	14995.33	269.75

Table 3.8: Comparison between prior and calibrated distributions

In the end, table 3.9 shows the difference in terms of means and standard deviations between the real Gaussian distribution, used to derive experimental data, and the calibrated one. Mean values are accurately evaluated by the calibration process with a relative error of the order of 1% with respect to nominal values. Standard deviations are overestimated for the aerodynamic coefficients and underestimated for the flap stiffness. However, these incorrect evaluations are not of concern since calibration's main goal is to quantify unknown parameters that have to be set for running computer models. A

visual representation of the various distribution types for each input parameter is shown in figures going from 3.10 to 3.13, where a Gaussian distribution approximation is performed for the calibrated density probabilities.

	Real distribution		Calibrated distribution	
Variable	**Mean**	**Std Deviation**	**Mean**	**Std Deviation**
$C_{L_\alpha}(0.3)$	0.1075	$5.375e-3$	0.1082	$7.844e-3$
$C_{L_\alpha}(0.5)$	0.1172	$5.860e-3$	0.1170	$1.298e-2$
$C_{L_\alpha}(0.6)$	0.1135	$5.675e-3$	0.1154	$1.034e-2$
$K_f\ [N/m]$	15035.467	751.774	14995.33	269.75

Table 3.9: Comparison between real and calibrated distributions

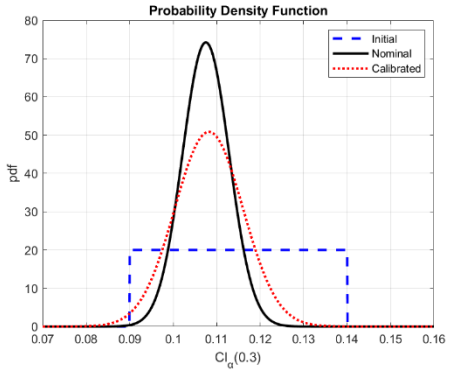

Figure 3.10: $C_{L_\alpha}(0.3)$ distribution comparison
(-- initial ; -- nominal ; -- calibrated)

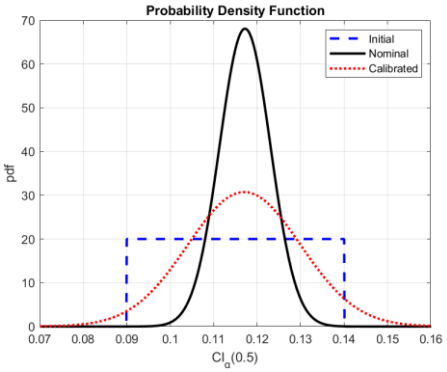

Figure 3.11: $C_{L_\alpha}(0.5)$ distribution comparison
(-- initial ; -- nominal ; -- calibrated)

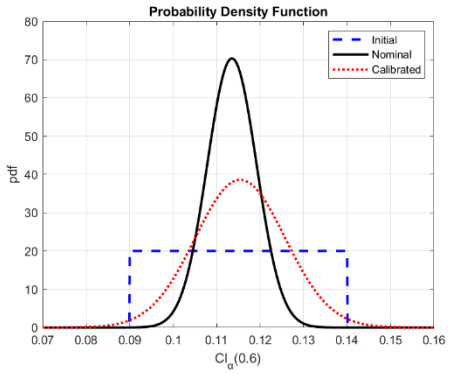

Figure 3.12: $C_{L_\alpha}(0.6)$ distribution comparison
(-- initial ; -- nominal ; -- calibrated)

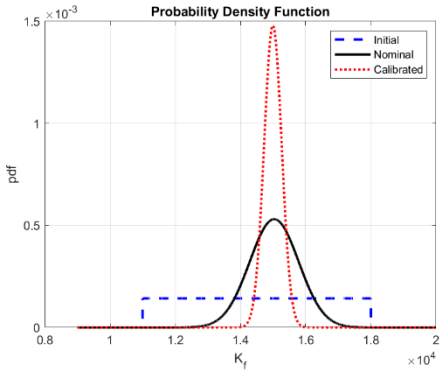

Figure 3.13: K_f distribution comparison
(-- initial ; -- nominal ; -- calibrated)

3.5 Calibration Under Uncertainty

The last validation step concerns another often met situation in engineering application, that is, the evaluation of parameter statistics given experimental ones. In other words, the study tries to answer the following question: "*what are the acceptable input variable uncertainties that ensure the measured response function uncertainties* ? ". This is a well-established problem since many rotorcraft certification requirements set boundaries or specific ranges for loads in different fly conditions, and establishing acceptable value intervals for given parameters allows to ensure these criteria are imposed upstream. Calibration under uncertainty (CUU) toolkits permit such evaluations since, as discussed, these try to identify model variables resulting in a "best fit" between responses and data in a nondeterministic sense. Assuming Gaussian distributions, input standard deviations can be calibrated according to output ones, giving an estimate of the acceptable uncertainties. Boundaries and ranges are satisfied if they fall respectively beyond or inside of, a certain multiple of the standard deviation. As shown in the scheme presented in figure 3.14, experimental data standard deviations are obtained through LHS analysis with 300 samples, considering, as before, mean thrust and mean torque values as output responses and the three aerodynamic slope lift coefficients and the flap hinge stiffness as input variables. These follow Gaussian distributions with means derived from Bayes calibration and standard deviations such that the overall parameter space of $\mu \pm 3\sigma$ produces a 10% deviation from mean values. A Dakota nested model is constructed in which an inner uncertainty quantification method exchanges information and statistical results with the least squares algorithm to infer the calibration. The loop keeps going until the convergence criteria, usually set in terms of difference threshold between model responses and calibration data, is obtained. The Mean Value method is chosen as the inner uncertainty quantification algorithm due to its capability of inferring statistical contents with few iterations, while, an NL2SOL algorithm is used as a gradient-based optimization process to move into the input variable space. Also in this case input variables are forced to follow Gaussian distributions with means and initial standard deviations taken from Bayes Calibration results to give continuity between the two studies. Table 3.10 compares the nested model results with the real ones, while table 3.11 highlights the difference between experimental data means and standard deviations, with the same obtained using a forward uncertainty LHS technique with 300 samples from calibrated variables. As can be seen, even if the statistical measures are accurately reproduced, input standard deviations are far from being the real ones, except for the C_{L_α} at Mach equal to 0.6. This miscalculation is due to the fact that more than one statistical combination leads to the same result when working with variables of different sensitivities. To derive more reliable results, the same steps have been repeated considering only aerodynamic coefficients thanks to their comparable order of magnitudes in sensitivity indices. Tables 3.12 and 3.13 report again the comparison between real and calibrated variables and uncertainty output measures. In this case, more accurate results are obtained since no input parameter overhangs others in the

optimization process. Finally figures 3.15 and 3.16 show a comparison between cumulative distribution functions (CDFs) obtained for mean thrust and mean torque in the case of nominal input distributions and calibrated ones. Figures 3.17 and 3.18 plot probability distribution functions (PDFs) for both output responses using CUU aerodynamic results. The forward uncertainty quantification is essentially run to confirm the obtained results and check if the response function statistical descriptions are in line with the certification conditions.

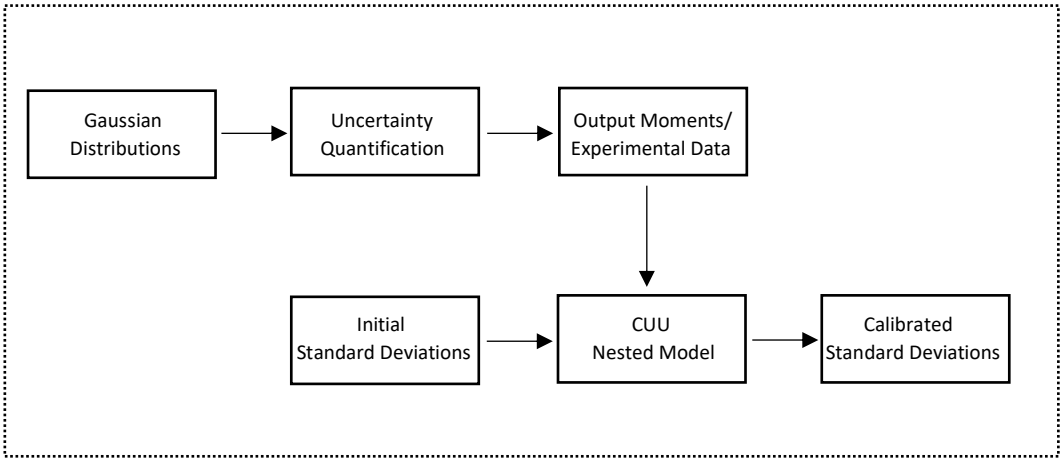

Figure 3.14: CUU validation scheme

	Experimental Input Data		Calibrated Input Data	
Variable	**Mean**	**Std Deviation**	**Mean**	**Std Deviation**
$C_{L_\alpha}(0.3)$	0.10825	$3.608e-3$	0.10825	$1.896e-3$
$C_{L_\alpha}(0.5)$	0.11700	$3.900e-3$	0.11700	$6.870e-3$
$C_{L_\alpha}(0.6)$	0.11538	$3.867e-3$	0.11538	$3.506e-3$
$K_f\ [N/m]$	14995.33	499.844	14995.33	1523.05

Table 3.10: Mixed CUU results

	Experimental Output Data		Calibrated Output Data	
	Mean	**Std Deviation**	**Mean**	**Std Deviation**
Thrust $[N]$	2009.065	11.453	2009.182	11.533
Torque $[Nm]$	-2360.925	2.100	-2360.945	2.120

Table 3.11: Load statistical moments

	Experimental Input Data		Calibrated Input Data	
Variable	**Mean**	**Std Deviation**	**Mean**	**Std Deviation**
$C_{L_\alpha}(0.3)$	0.10825	$3.608e-3$	0.10825	$3.223e-3$
$C_{L_\alpha}(0.5)$	0.11700	$3.900e-3$	0.11700	$4.356e-3$
$C_{L_\alpha}(0.6)$	0.11538	$3.867e-3$	0.11538	$3.803e-3$

Table 3.12: Aerodynamic CUU results

	Experimental Output Data		Calibrated Output Data	
	Mean	**Std Deviation**	**Mean**	**Std Deviation**
Thrust $[N]$	2009.067	11.423	2009.102	11.470
Torque $[Nm]$	-2360.925	2.093	-2360.931	2.106

Table 3.13: Final load statistical moments

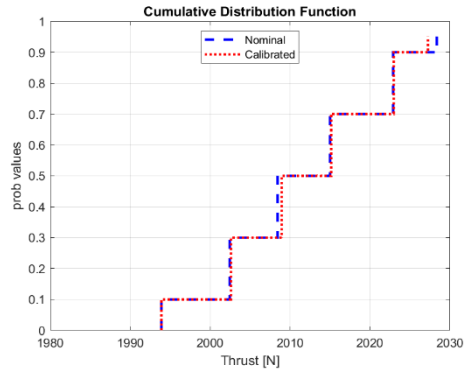

Figure 3.15: Thrust CDF
(-- Nominal ; -- Calibrated)

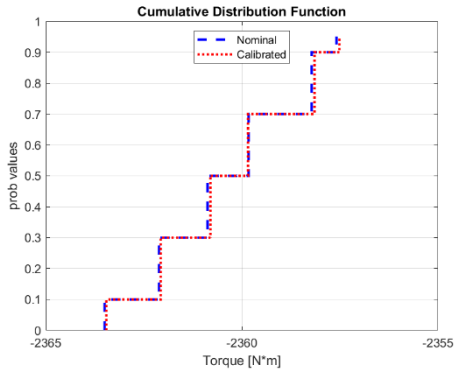

Figure 3.16: Torque CDF
(-- Nominal ; -- Calibrated)

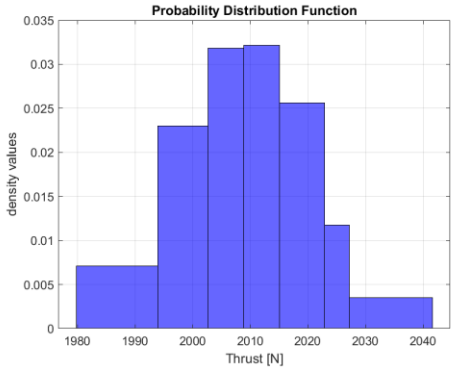

Figure 3.17: Calibrated Thrust PDF

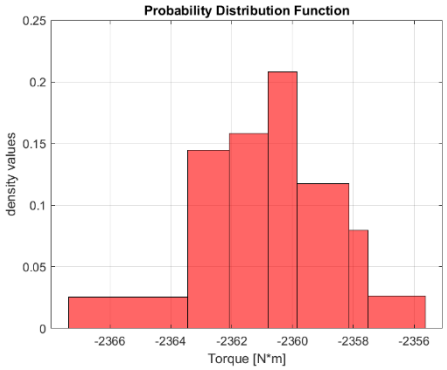

Figure 3.18: Calibrated Torque PDF

Chapter 4

Experimental Analysis

Validation analysis has been helpful in determining the confidence level of uncertainty quantification and calibration methods, showing optimal and accurate results under specific conditions and observations. Bayes calibration estimates with high precision parameter mean values, while lower confidence can be assigned to derived standard deviations. The level of accuracy mainly depends on parameter influence about related response functions and on the number of experiments available. In most cases few experiments can be conducted, based on the number of accessible resources, and critical judgment is required to evaluate results. Nevertheless, Bayes calibration is a well-established technique to infer unknown parameter values when various kinds of uncertainties can be related to them. On the other hand, calibration under uncertainty (CUU) nested models are characterized by correct and optimal operation only when variables with comparable sensitivity indices are taken into account in the problem development. All the techniques explained in the previous chapters can now be used in real engineering applications, where no complete prior knowledge is assumed and where the computational model is covered by uncertainty. The chapter starts with a brief introduction and explanation of the helicopter experiment and its related multibody constructed model. Numerical methods for uncertainty quantification and calibration are then exploited in a Dakota-MBDyn interface environment to derive unknown input parameters and statistical descriptions of the desired response functions.

4.1 Experiment Setup

The experiment was conducted in the large test chamber of GVPM (Galleria del Vento Politecnico di Milano) with 13.84 m wide, 3.84 m high, and length of 35 m. Figure 4.1 shows the setup mounted for performing the test cases and the measurements. The experiment considered is part of a larger one concerning the simulation and testing of helicopter-ship interactions [40]. The helicopter model is composed of a fuselage and a rotor with four untapered and untwisted rectangular blades made of carbon-fiber composite materials. The sectional profile of the blade is obtained through a 3D scan with NACA 0012 airfoils. The rotor has a diameter of 970 mm and includes a complete swashplate mechanism. In this way, collective and cyclic commands can be applied to the blades to obtain trim conditions at specific forces and moments. The fuselage is made of polycarbonate and tries to represent a 1:10 scaled model of a Bo105, considered as generic medium-size helicopter. A visual representation of the model is shown in figure 4.2. The fuselage is mounted on an internal metallic structure with a six-component strain

gauge balance (Koris F6D-80e-60) measuring forces and moments in all the six degrees of freedom ($F_x, F_y, F_z, M_x, M_y, M_z$), and it is specialized for laboratory and research applications. A driving motor system, inside the fuselage structure, is installed as well. It consists of a brushless motor with 3.3kW continuous power directly connected to the rotor shaft by means of a joint coupling. The rotor rotational speed is kept thanks to an Electronic Speed Controller (ESC) connected to the rotor. To obtain load coefficients based on instantaneous RPM measures, the ESC is recorded with the same sampling frequency of the loads. The helicopter model was attached to a horizontal strut connected to a system of two orthogonal sliding guides, allowing the translation of the helicopter in both vertical and longitudinal directions.

Figure 4.1: Experiment setup

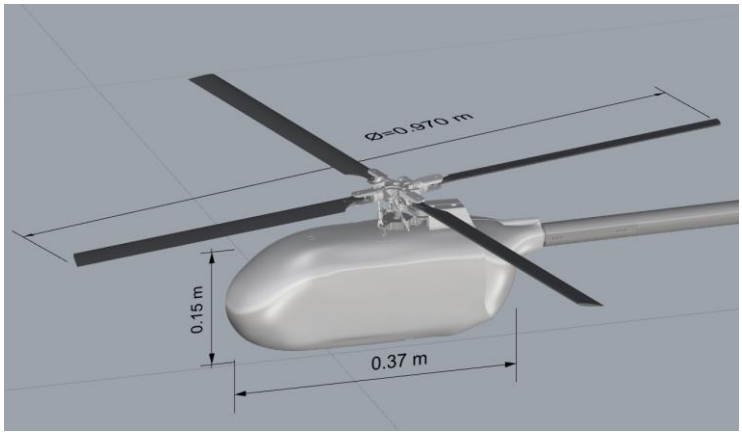

Figure 4.2: Helicopter model

The experiment setup allows to reproduce helicopter aerodynamic loads in different environmental conditions, including in and out of ground effect, hover, and forward flight. For this particular application, the helicopter is considered in hover condition at four different altitudes, in order to simulate both OGE (Out of Ground Effect) and IGE (In Ground Effect) cases. Measurements of Thrust T, Roll M_x, Pitch M_y and Torque M_z moments are obtained from 10 second simulations for each height H, respectively of four, three, two, and one time the rotor radius R. Test data for calibration and uncertainty quantification are then extrapolated after averaging process. Figures 4.3 and 4.4 show the time histories of aerodynamic loads and rotor RPM for the OGE case at $H = 4R$. Force and Moment signals are obtained after being sampled at 100 Hz frequency and filtered with a lowpass filter of 15 Hz. Table 4.1 reports the averaged load values for all the different altitudes, normalized with respect to rotor radius.

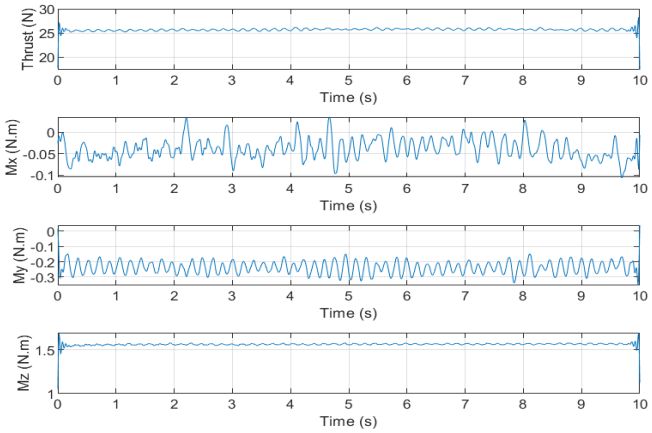

Figure 4.3: Load time histories

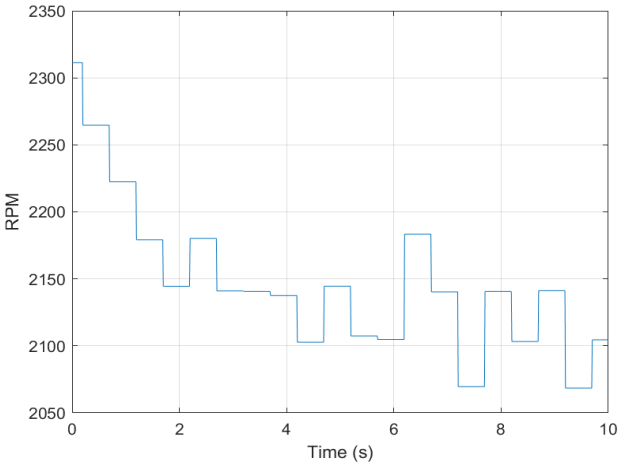

Figure 4.4: RPM time history

Height $[H/R]$	Thrust $[N]$	Roll $[Nm]$	Pitch $[Nm]$	Torque $[Nm]$
4	25.6849	−0.0388	−0.2371	1.5621
3	25.8241	−0.1483	−0.2193	1.5835
2	29.0067	−0.2525	−0.3082	1.7789
1	35.8256	−0.2401	−0.2490	1.8520

Table 4.1: Experimental load mean values

4.2 Multibody Model

A multibody model of the experimental helicopter has been developed using MBDyn. The model, lacking fuselage description, consists of a hingeless, stiff-in plane rotor with four elastic blades connected to the hub through a revolute hinge, allowing the rotation about the feathering axis of the blade. This degree of freedom, along with a flexible pitch link, allows pitch control. The pitch link is connected to a complete swashplate mechanism which consists of two static structural nodes, connected together through a revolute hinge allowing relative rotation about the vertical axis. Technical features of the rotor model are reported in table 4.2. Each blade is modeled by three finite volume beam elements composed of three nodes [41]. Constitutive properties can be defined separately. MBDyn provides maximum topological flexibility since each node of the beam can be related to a structural node by an offset and an optional relative orientation. A linear viscoelastic constitutive law matrix is associated with each beam section so that internal forces can be computed as functions of the straining of the reference line and orientation at selected points, called evaluation points, which lie between two pairs of beam nodes. To introduce the aerodynamic model, Aerodynamic Beam Element is used, which relies on the structural beam element to compute the aerodynamic section configuration at each integration point. A c81 data file of NACA 0012, provides aerodynamic lift, drag and moment coefficients at different Mach numbers and angles of attack. These are used to compute aerodynamic loads based on Blade element/Momentum theory. Blade element theory assumes that blades can be divided into smaller elements that work independently and as two-dimensional airfoils so that aerodynamic forces can be computed based on local flow conditions. The elements then can be summed together over the span to provide the total forces and moments exerted on the rotor. On the other hand, momentum theory estimates the induced velocity by momentum lost in the rotor plane, affecting the inflow in the rotor plane and therefore the forces calculated by element theory. The coupling between the two sets up an iterative process to determine aerodynamic loads and induced velocities near the rotor [42]. The inflow is represented by a Glauert-Trees model assuming linear distribution over the rotor disk.

This work aims for establishing a correct statistical calibration analysis in conditions as general as possible. For this reason, a large number of parameters from different fields have been chosen to be representative of the uncertainty related to the model, in order

to derive a stable methodology when poor information is given. The variables are picked up from:

- Aerodynamic field
- Structural field
- Command control inputs

In particular, lift slope coefficients C_{L_α}s and zero lift drag coefficients C_{D_0}s at different Mach numbers and angles of attack, pitch link stiffness K_p, flapping bending stiffness EI_y, collective and cyclic commands. These are representative of the uncertainty related to almost the entire helicopter applications, validating the study for different kinds of operating environments.

Description	Symbol	Value	Unit
Blade span	b	0.425	$[m]$
Hub radius	r	0.06	$[m]$
Rotor radius	R	0.485	$[m]$
Blade chord	c	0.042	$[m]$
Axial stiffness	EA	$1.0e6$	$[N]$
Shear stiffness	GA_y	$1.0e6$	$[N]$
Shear stiffness	GA_z	$1.0e6$	$[N]$
Bending stiffness	EI_y	7.0	$[Nm^2]$
Bending stiffness	EI_z	$2.26e3$	$[Nm^2]$
Torsional stiffness	GJ	$1.0e2$	$[Nm^2]$
Pitch Link stiffness	K_p	10.0	$[N/m]$
Pitch Link damping	C_p	0.05	$[Ns/m]$

Table 4.2: Model technical features

4.2.1 Aerodynamic Interpolation

Before simulations run, aerodynamic variables have been extrapolated from the c81 data file using interpolation function *"polifit"* on Matlab. Considering the maximum tip velocity of the rotor $v_{tip}^{max} = 108.9\ m/s$ for the $4R$ altitude case, a specific range of interest have been delimited for the aerodynamic coefficients, with $M \in [0\ ;\ 0.4]$ and $\alpha \in [-10°\ ;\ 10°]$. For C_{L_α} values a linear interpolation is assumed, while a quadratic one is used to derive C_{D_0}s due to function profile shape in the domain.

$$C_L = C_{L_\alpha}\alpha + C_{L_0} \tag{4.1}$$

$$C_D = kC_L^2 = C_{D_{\alpha^2}}\alpha^2 + C_{D_\alpha}\alpha + C_{D_0} \tag{4.2}$$

with α in degrees. Tables 4.3 and 4.4 show the obtained results for Lift and Drag coefficients after interpolation, while figures from 4.5 to 4.12 compare the interpolated

functions with c81 data. An interpolation process is also used by MBDyn to derive aerodynamic coefficients at Mach values and angle of attacks not listed in the c81 data file. As for the validation case, a python script, namely *"airfoil.py"* iteratively substitutes the original data with the new ones, creating a new c81 data file, when interfacing with Dakota simulation.

Mach Number	C_{L_α}	C_{L_0}
0.0	0.1055	0.0
0.2	0.1055	0.0
0.3	0.1080	0.008
0.4	0.1112	0.0

Table 4.3: Interpolated aerodynamic lift coefficients

Mach Number	$C_{D_{\alpha^2}}$	C_{D_α}	C_{D_0}
0.0	$9.2e-5$	0.0	0.0079
0.2	$9.2e-5$	0.0	0.0079
0.3	$1.025e-4$	0.0	0.0077
0.4	$1.554e-4$	0.0	0.0068

Table 4.4: Interpolated aerodynamic drag coefficients

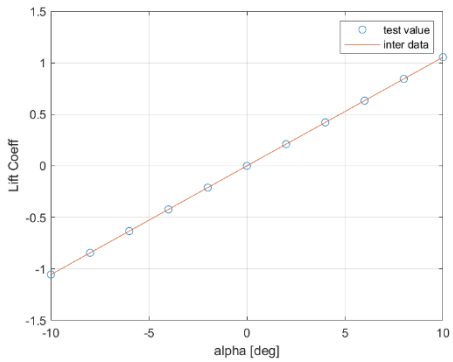

Figure 4.5: $C_L(0.0)$ comparison
(° test data , -- interpolated data)

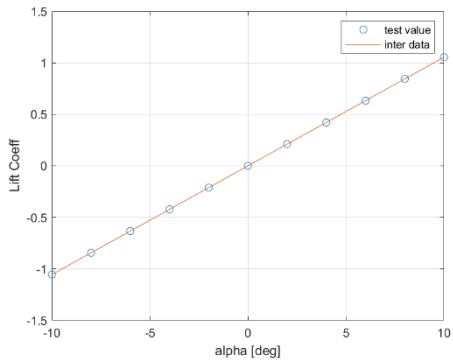

Figure 4.6: $C_L(0.2)$ comparison
(° test data , -- interpolated data)

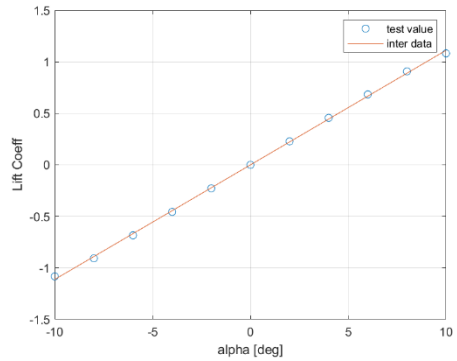

Figure 4.7: $C_L(0.3)$ comparison
(° test data , -- interpolated data)

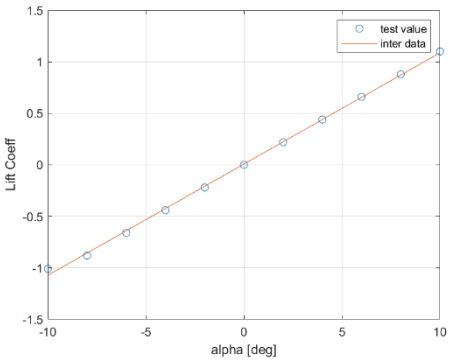

Figure 4.8: $C_L(0.4)$ comparison
(° test data , -- interpolated data)

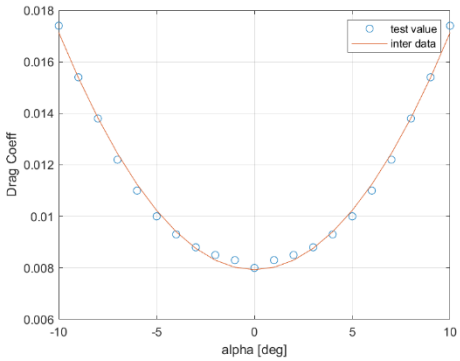

Figure 4.9: $C_D(0.0)$ comparison
(° test data , -- interpolated data)

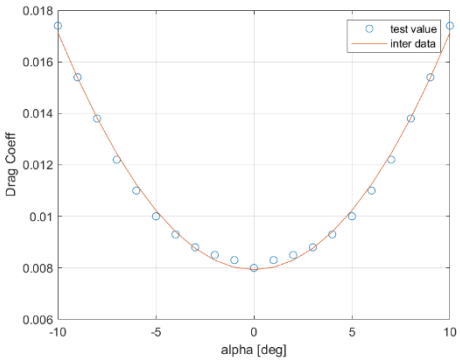

Figure 4.10: $C_D(0.2)$ comparison
(° test data , -- interpolated data)

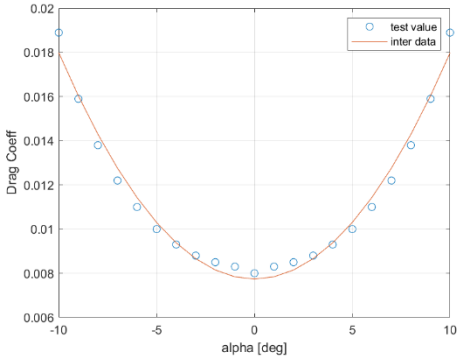

Figure 4.11: $C_D(0.3)$ comparison
(° test data , -- interpolated data)

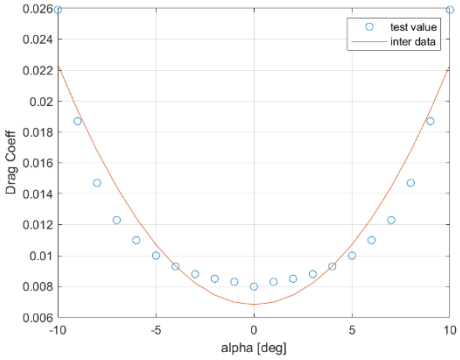

Figure 4.12: $C_D(0.4)$ comparison
(° test data , -- interpolated data)

4.3 Sensitivity Analysis

As already stated in the validation chapter, uncertainty analysis is usually preceded by sensitivity analysis in order to reduce the problem size, extrapolate information and derive the most appropriate and influential variables under investigation. Objective functions and calibration data are chosen to be the mean values of Thrust T, roll moment M_x, pitch moment M_y, and rotor power P obtained by multiplying the torque moment M_z with the rotational speed Ω expressed in rad/s $(P = M_z\Omega)$. This last has been chosen due to its primary demand in certification requirements. Mean RPM value at 4R height is used as rotational rotor speed. As first step, a sensitivity analysis considering command controls is run. This is used to assess their influence on objective functions, and to establish with more precision the pitch link stiffness value. No direct measurements on the component were available, so its order of magnitude has been selected by comparing the multibody model behavior at assigned control values with the experimental helicopter one. Sensitivity indices are obtained through Variance Based Decomposition (VBD), sampling with LHS technique from command density distributions listed in table 4.5. VBD will be used for all the sensitivity analysis later on. Table 4.6 shows the result obtained assuming pitch link stiffness of 10 N/m. These are in line with the observed helicopter behavior since roll and pitch moments are mainly produced respectively by lateral and longitudinal swashplate rotations. A dynamic coupling is also present as expected. After obtaining pitch link stiffness order of magnitude, a complete sensitivity analysis can be inferred considering both aerodynamic and structural parameters. The simulations are performed considering 200 samples, LHS uncertainty quantification technique and control inputs set with collective, lateral and longitudinal commands respectively of $5°, -1°, -1°$. Gaussian probability distributions are assumed for all the parameters, as shown in table 4.7, with means μ equal to the nominal values and standard deviations σ such that the parameter space defined as $\mu \pm 3\sigma$ produces a 15% deviation from mean values. A first aerodynamic sensitivity analysis is performed to eliminate Lift and Drag coefficients with minor impact on objective functions (table 4.8), then a mixed analysis with both structural and aerodynamic related quantities pulls out the final set of sensitivity indices (table 4.9). The high pitch link flexibility heavily affects the rotor performances and its dynamic behavior, with a net contribution in all the loads especially in roll and pitch moments. As expected Zero Lift Drag coefficients C_{D_0} exclusively show their influence in power, since, to keep the rotor in rotation, it is necessary to overcome the profile drag acting on the rotating blades and to supply energy to develop the necessary induced speed to generate thrust. To get a complete overview and comparison of the obtained results, figure 4.13 shows a tornado diagram with data categories listed vertically with the same order of table 4.9 starting from the bottom. These last parameters, along with control commands will be used to infer Bayes calibration.

Command	Distribution Type	Lower Bound	Upper Bound
$Coll$	Uniform	4.0	8.0
Lat	Uniform	-2.0	2.0
$Long$	Uniform	-2.0	2.0

Table 4.5: Command probability distributions

Command	Thrust T	Roll M_x	Pitch M_y	Power P
$Coll$	$9.06e-1$	$6.38e-4$	$2.41e-3$	$9.10e-1$
Lat	$2.16e-3$	$7.72e-1$	$1.86e-1$	$2.75e-3$
$Long$	$3.95e-4$	$1.16e-1$	$7.93e-1$	$1.12e-3$

Table 4.6: Command sensitivity results

Variable	Distribution type	Mean	Std Deviation
$C_{L_\alpha}(M=0)$	Gaussian	0.1055	$5.275e-3$
$C_{L_\alpha}(M=0.2)$	Gaussian	0.1055	$5.275e-3$
$C_{L_\alpha}(M=0.3)$	Gaussian	0.1080	$5.4e-3$
$C_{L_\alpha}(M=0.4)$	Gaussian	0.1112	$5.56e-3$
$C_{D_0}(M=0)$	Gaussian	0.0079	$3.95e-4$
$C_{D_0}(M=0.2)$	Gaussian	0.0079	$3.95e-4$
$C_{D_0}(M=0.3)$	Gaussian	0.0077	$3.85e-4$
$C_{D_0}(M=0.4)$	Gaussian	0.0068	$3.4e-3$
K_p	Gaussian	15.0	0.75
EI_y	Gaussian	10.0	0.5

Table 4.7: Overall probability distributions

Variable	Thrust T	Roll M_x	Pitch M_y	Power P
$C_{L_\alpha}(M=0)$	$1.55e-2$	$3.26e-3$	$5.04e-4$	$1.23e-2$
$C_{L_\alpha}(M=0.2)$	$2.33e-1$	$6.22e-1$	$1.84e-1$	$1.05e-1$
$C_{L_\alpha}(M=0.3)$	$8.11e-1$	$3.95e-1$	$7.55e-1$	$3.67e-1$
$C_{L_\alpha}(M=0.4)$	$8.44e-5$	$3.77e-5$	$1.75e-4$	$1.14e-5$
$C_{D_0}(M=0)$	$1.50e-6$	$4.56e-6$	$3.05e-7$	$7.46e-4$
$C_{D_0}(M=0.2)$	$4.29e-5$	$8.01e-5$	$3.17e-4$	$3.4e-1$
$C_{D_0}(M=0.3)$	$3.31e-5$	$2.10e-5$	$1.88e-4$	$2.83e-1$
$C_{D_0}(M=0.4)$	$2.37e-7$	$3.04e-7$	$5.68e-7$	$5.24e-5$

Table 4.8: Aerodynamic sensitivity results

Variable	Thrust T	Roll M_x	Pitch M_y	Power P
$C_{L_\alpha}(M=0.2)$	$3.46e-2$	$1.67e-1$	$6.25e-2$	$2.48e-2$
$C_{L_\alpha}(M=0.3)$	$7.82e-1$	$4.21e-1$	$1.03e-1$	$4.78e-1$
$C_{D_0}(M=0.2)$	$6.22e-3$	$2.69e-3$	$3.81e-3$	$1.06e-1$
$C_{D_0}(M=0.3)$	$1.62e-2$	$7.62e-3$	$8.00e-4$	$2.58e-1$
K_p	$1.40e-1$	$4.61e-1$	$7.86e-1$	$1.09e-1$
EI_y	$5.43e-2$	$6.84e-3$	$9.03e-2$	$4.28e-2$

Table 4.9: Overall sensitivity results

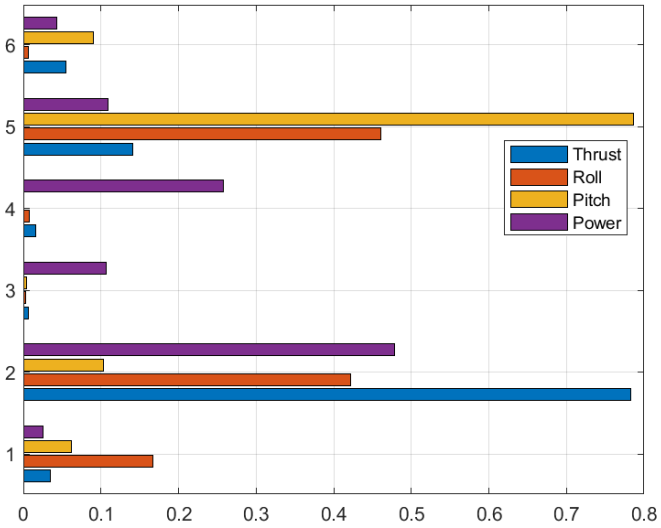

Figure 4.13: Sensitivity tornado diagram
(1: $C_{L_\alpha}(0.2)$; 2: $C_{L_\alpha}(0.3)$; 3: $C_{D_0}(0.2)$; 4: $C_{D_0}(0.3)$; 5: K_P; 6: EI_y)

4.4 Bayes Calibration

After delimiting and defining the variables of interest, a Bayes calibration analysis is performed to derive their statistical descriptions in terms of distribution function moments. Nine input parameters are selected to describe, as best as possible, the uncertainties characterizing helicopter performances and to include all the different fields in which these uncertainties can be collocated into. As a reminder, the variables are:

- Aerodynamic Lift Slope and Zero Lift Drag coefficients at Mach number equal to 0.2 and 0.3:

$$\left[C_{L_\alpha}(0.2), C_{L_\alpha}(0.3),\ C_{D_0}(0.2), C_{D_0}(0.3)\right]$$

- Pitch Link Stiffness K_p and Blade Bending Stiffness EI_y
- Collective and cyclic control commands, namely $Coll\ Lat\ Long$

Experimental helicopter measurements of mean thrust, roll, pitch, and power in hover and at altitude $H = 4R$ (OGE condition) are used as calibration data by the method. As for the validation part of the study, Bayes Calibration is inferred making use of the DRAM algorithm with a total chain of 100.000 samples. Gaussian process emulator interpolates 500 real model function evaluations to obtain response surfaces used to derive Likelihood Functions and proposal distribution covariance matrices. 5000 "burn in" samples are also selected to remove lower probability regions from the chain. Pre-existing parameter knowledge and assumptions are contained in the prior distributions listed in table 4.10 and used by the calibration process. Uniform distributions are attributed to all the variables, with bounds defined by considering usually assumed uncertainties in measurements, and confidence obtained from previous simulations of the multibody model under specific input values. Lift Coefficients are delimited by a near 10% uncertainty from nominal values, which is selected as a trusted region in aerodynamic-related estimations. A larger bound is assigned to the zero lift drag coefficients in order to add the hub contribution in drag which has not been modeled. This is necessary to not end up in torque, and consequently power, underestimation. To match all the experimental measures and obtain feasible results, a first Bayes calibration analysis is run considering only Roll and Pitch as calibration data and Pitch Link Stiffness K_p, Blade Bending Stiffness EI_y and the cyclic commands $Lat, Long$ as input variables. The choice has been determined by the lower order of magnitudes of moment values with respect to Thrust and Power, which could lead to erroneous and too small estimations of cyclic mean values. The means and standard deviations obtained from this first analysis are then used to characterize gaussian distributions, selected as new prior distributions for the four previously cited variables, in the overall Bayes Calibration. Results of the final analysis are listed in table 4.11, reporting the first four statistical moments describing the posterior variable distributions. Figures from 4.14 to 4.22 compare the initial uniform distributions with the posterior normal type ones, constructed by taking the means and standard deviations from the calibration results.

Variable	Distribution Type	Lower Bound	Upper Bound
$C_{L_\alpha}(M = 0.2)$	Uniform	0.09	0.125
$C_{L_\alpha}(M = 0.3)$	Uniform	0.09	0.125
$C_{D_0}(M = 0.2)$	Uniform	0.007	0.015
$C_{D_0}(M = 0.3)$	Uniform	0.007	0.015
K_p	Uniform	10.0	20.0
EI_y	Uniform	6.0	10.0
$Coll$	Uniform	4.0	8.0
Lat	Uniform	-2.0	2.0
$Long$	Uniform	-2.0	2.0

Table 4.10: Variable probability distributions

Variable	Mean	Std. Deviation	Skewness	Kurtosis
$C_{L_\alpha}(M = 0.2)$	0.1072	$9.378e-3$	$-2.778e-2$	-1.085
$C_{L_\alpha}(M = 0.3)$	0.1078	$9.238e-3$	$4.008e-2$	-1.028
$C_{D_0}(M = 0.2)$	0.0192	$2.151e-3$	$4.601e-2$	-1.083
$C_{D_0}(M = 0.3)$	0.0195	$2.157e-3$	$8.497e-3$	-1.123
K_p	15.024	2.658	$6.830e-3$	-1.051
EI_y	8.499	0.824	$2.523e-2$	-1.209
$Coll$	5.411	0.331	$-9.214e-1$	1.556
Lat	0.1698	0.727	$-1.187e-1$	$-3.830e-1$
$Long$	-0.2955	0.756	$3.345e-1$	$-2.209e-1$

Table 4.11: Bayes Calibration results

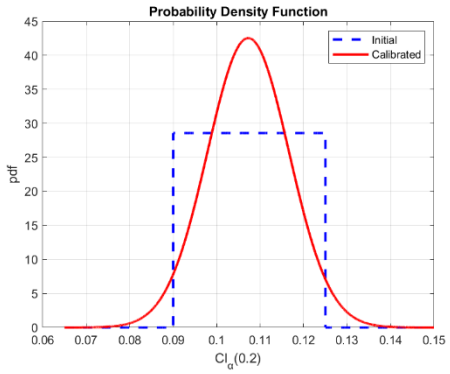

Figure 4.14: $C_{L_\alpha}(0.2)$ distribution comparison
(-- initial ; -- calibrated)

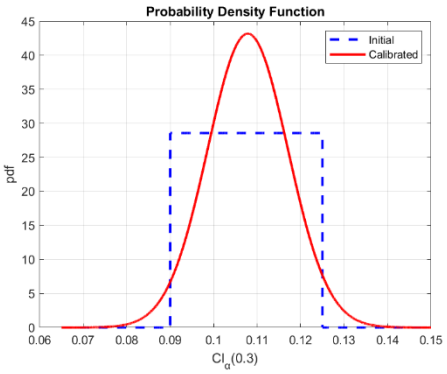

Figure 4.15: $C_{L_\alpha}(0.3)$ distribution comparison
(-- initial ; -- calibrated)

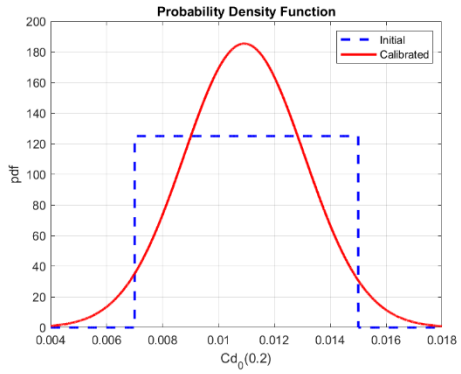

Figure 4.16: $C_{D_0}(0.2)$ distribution comparison
(-- initial ; -- calibrated)

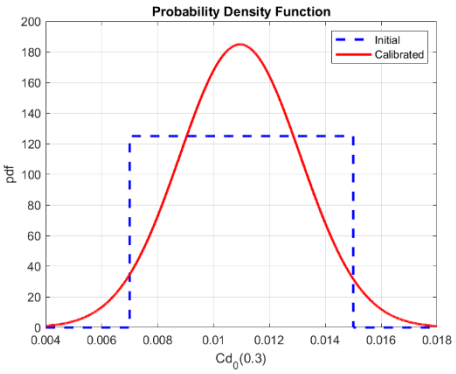

Figure 4.17: $C_{D_0}(0.3)$ distribution comparison
(-- initial ; -- calibrated)

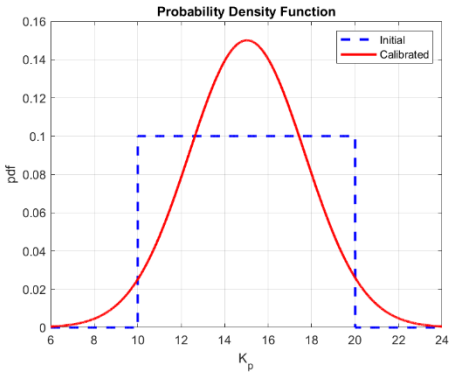

Figure 4.18: K_P distribution comparison
(-- initial ; -- calibrated)

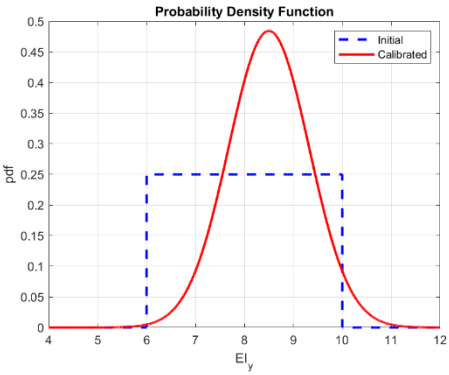

Figure 4.19: EI_y distribution comparison
(-- initial ; -- calibrated)

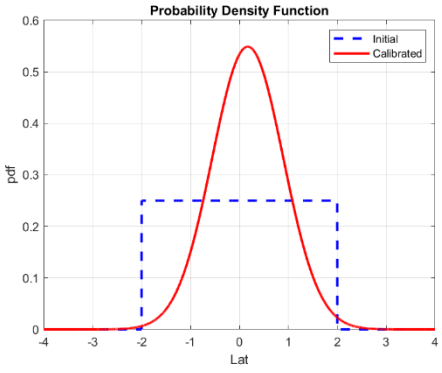

Figure 4.20: Lat distribution comparison
(-- initial ; -- calibrated)

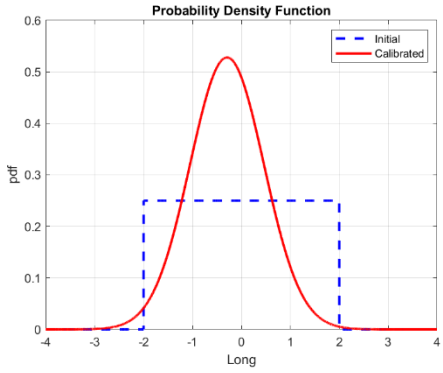

Figure 4.21: $Long$ distribution comparison
(-- initial ; -- calibrated)

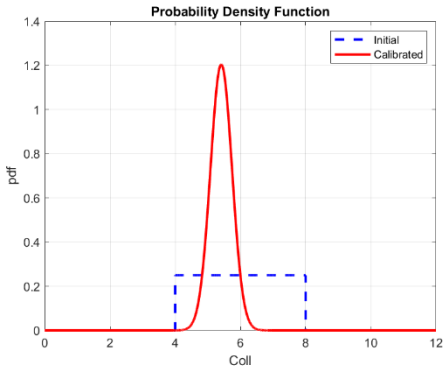

Figure 4.22: $Coll$ distribution comparison
(-- initial ; -- calibrated)

As can be seen, calibrated drag coefficient mean values are far from the respective nominal values obtained through c81 data file interpolation due to hub contribution. Instead, aerodynamic lift coefficients have not been changed by the calibration process. High uncertainty, in form of standard deviation, characterizes cyclic commands due to the low-value moments acting on the rotor in the Hover condition. Of different nature is instead the collective command which is accurately evaluated thanks to its primary importance in thrust and power development. As well as deducing unknown model parameters, Bayes calibration analysis allows obtaining a complete statistical description of response functions. To do so, and also to confirm the obtained results, a forward uncertainty quantification analysis is run where calibrated variables, described by probability distributions, are used as input parameters to obtain the statistical assessment of the loads. Normal distributions with means and standard deviations obtained from Bayes calibration are assumed for all the variables, while the LHS technique with 500 total samples derives the uncertainties related to quantities of the outputs. Table 4.12 compares the load distribution means and standard deviations along with the average calibration data obtained from the experiment. The last column indicates the number of standard deviations separating the two means for each load. Optimal agreement between experimental data and simulation results can be appreciated since the former is away less than one standard deviation from the mean value of the probability distributions for all the loads, making them probable realizations. Figures from 4.23 to 4.30 show a complete uncertainty loads description through Cumulative Distribution Functions (CDF) and Probability Distribution Functions (PDF) extrapolated from LHS results.

Load	Mean	Std Deviation	Exp Data	N° Std Deviation
Thrust T $[N]$	26.1148	3.5505	25.6849	0.12
Roll R $[Nm]$	−0.0069	0.5604	−0.0388	0.057
Pitch L $[Nm]$	−0.2501	0.5661	−0.2371	0.023
Power P $[W]$	353.4290	47.0523	350.9062	0.05

Table 4.12: Model and experimental loads comparison

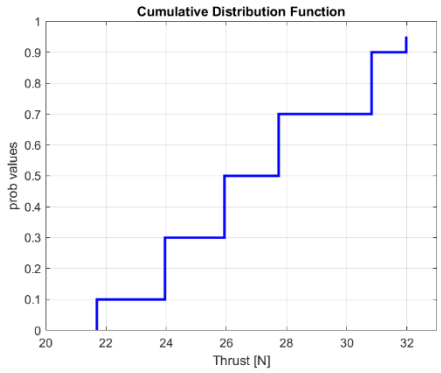

Figure 4.23: Thrust CDF

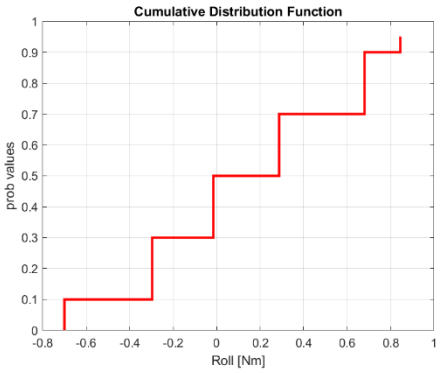

Figure 4.24: Roll CDF

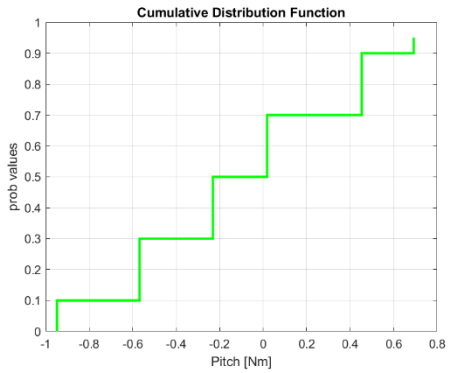

Figure 4.25: Pitch CDF

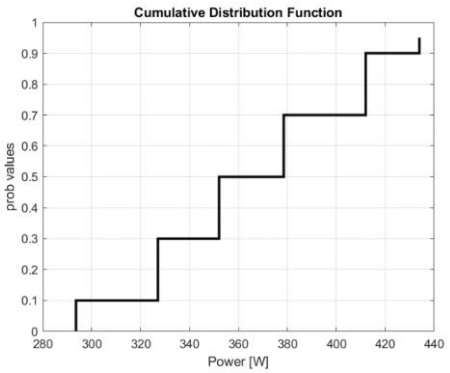

Figure 4.26: Power CDF

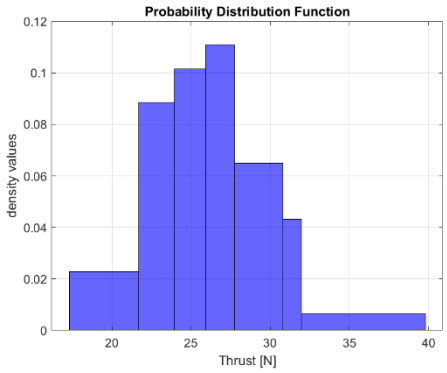

Figure 4.27: Thrust PDF

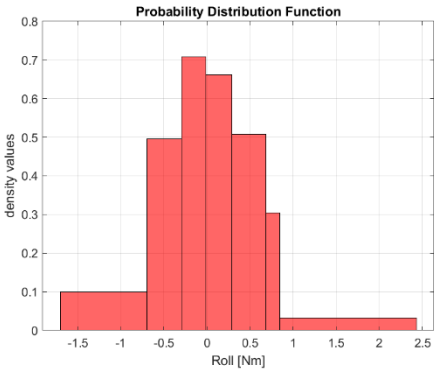

Figure 4.28: Roll PDF

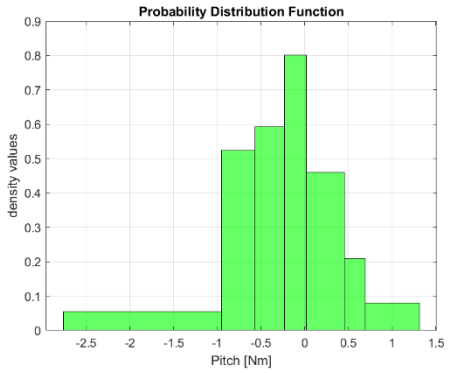

Figure 4.29: Pitch PDF

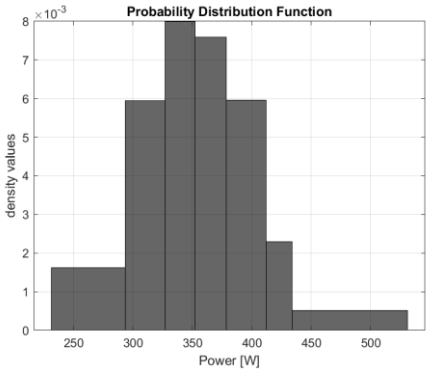

Figure 4.30: Power PDF

4.5 Calibration Under Uncertainty

The performed calibration and uncertainty analysis does not take into account the actual load trends measured in the laboratory. Experimental data are extrapolated after an averaging process which cuts out all model dynamic fluctuations and numerical errors. These can be recovered by assigning standard deviations in load distributions, such that, sampling from the derived distributions has the same effect of taking an actual measurement at a generic time. The problem solution falls back into the Calibration Under Uncertainty (CUU) model already studied in the validation analysis, in which input variables statistical moments are derived from response function ones. Then, running a forward uncertainty analysis with the derived parameter statistical measures, ensures the observation of the desired response function uncertainties. As already stated the method suffers from a great limitation given by the optimization/least squares algorithm used by the outer loop. Especially working with derivative-based optimization algorithms, using variables with comparable sensitivity indices, is a necessary condition to ensure the correct method operation, without incurring misleading solutions, since only one parameter, or related quantities, combination leads to the desired output. The problem has been set in determining input control commands standard deviations from defined load ones, leaving all the other parameters constant at their previously calibrated mean values, for the 4R altitude dynamic case. Apart from the sensitivity condition, knowing manual controls range of application is a fundamental step to ensure specific aerodynamic loads acting on the rotor in operating conditions. Roll and Pitch standard deviations σ have been chosen so that the maximum deviation, defined as 3σ, is about $0.5\ Nm$, while Thrust and Power standard deviations are computed by considering a maximum difference of 10% around mean values, as observed in experiments. Dakota input file consists of a nested model, as for the validation case, with Mean Value Method (MPP) as uncertainty quantification internal algorithm and NL2SOL least-squares method as the outer one. Normal command distributions are assumed with means and initial standard deviations equal to the ones obtained in Bayes Calibration. Results of such analysis are

shown in table 4.13 which compares the initial first two command distribution moments with the ones obtained through the least-squares process. Table 4.14, instead, reports a complete overview of the different load's standard deviation results. The first ones, denoted as "Initial Std Dev", are obtained through LHS 500 samples uncertainty quantification, considering commands following normal distributions characterized by Bayes calibration results. The "Exp Std Dev" are the ones used as calibration data to infer the CUU analysis, and the "Calibrated Std Dev" reported the ones obtained after running the same LHS uncertainty quantification but with commands normal distributions constructed with the CUU results.

Command	Mean	Initial Std Dev	Calibrated Std Dev
$Coll\ [deg]$	5.4114	0.3311	0.1292
$Lat\ [deg]$	0.1698	0.7268	0.2273
$Long\ [deg]$	-0.2955	0.7557	0.2282

Table 4.13: Command standard deviations comparison

Load	Initial Std Dev	Exp Std Dev	Calibrated Std Dev
$Thrust\ T\ [N]$	3.000	0.8600	1.1638
$Roll\ R\ [Nm]$	0.5368	0.1670	0.1665
$Pitch\ L\ [Nm]$	0.5469	0.1670	0.1669
$Power\ P\ [W]$	30.3619	11.6600	11.6875

Table 4.14: Loads standard deviations comparison

4.6 Helicopter Ground Interaction

Crucial point is to assess if the calibrated variables, and their probability distributions, are optimal estimations even under different operating and environmental conditions. More precisely, assuming the same parameter distribution evaluated in Out of Ground Effect (OGE), is it possible to obtain a correct statistical description of the loads when the model is run in ground effect (IGE)? Operating helicopters close to the ground introduces special features in flight dynamic behavior [43]. The more significant is the effect in inflow velocity and hence in thrust and power required, especially in hover conditions. Close to the ground, the rotor downwash is highly influenced by the surface, and since it has to be zero at the ground, its value at the rotor disk will be lower than in the OGE case. Considering the same thrust required, the decrease of induced velocity will lead to a minor induced power needed to generate it when operating in Hover near the ground [44]. A complete description of the phenomenon would require a dynamic and nonlinear

aerodynamic description, which is far beyond the scope of this work which seeks a simple estimation and comparison between model responses and averaged experimental data. Forward uncertainty quantification is run, characterized by the same features explained above, but with the model operating at reduced altitudes. Particularly heights of two and one times the rotor radius are chosen as done in the laboratory. The 3R case is not reproduced since it is still in OGE condition and no particular variations from the previous results are expected. Rotational rotor speeds are updated for all the simulations, considering the corresponding averaged RPM value obtained from experiments. 500 samples for each analysis are iteratively extrapolated through the LHS technique from the input distribution functions listed in table 4.15. These are just Gaussian distribution obtained considering, as always, input parameters means and standard deviations from Bayes calibration. Tables 4.16 and 4.17 compare loads distributions with experimental data, validating the analysis in terms of number of standard deviations separating means and averaged measures.

Variable	Distribution	Mean	Std Deviation
$C_{L_\alpha}(0.2)$	Normal	0.1072	$9.378e-3$
$C_{L_\alpha}(0.3)$	Normal	0.1075	$9.238e-3$
$C_{D_0}(0.2)$	Normal	0.0109	$2.151e-3$
$C_{D_0}(0.3)$	Normal	0.0109	$2.157e-3$
K_p	Normal	15.0239	2.658
EI_y	Normal	8.4990	0.824
$Coll$	Normal	5.4114	0.331
Lat	Normal	0.1698	0.727
$Long$	Normal	-0.2955	0.756

Table 4.15: Input variable normal distributions

As expected no particular variations of Roll and Pitch moments are obtained with respect to the OGE case independently of the altitude considered. On the other hand, an increase in thrust and power is observed especially in the 1R case, since the ground effect acquires greater prominence at altitudes equal to or less than the rotor radius. While a growth with constant slope can be noted for power from 2R to 1R case (number of standard deviations separating mean and experiment is constant at 0.7), a more marked and net increase in thrust is observed (number of standard deviations from 0.3 to 1.01), with model prediction underestimating experimental data. This is in line with what has been explained before, with more thrust generated at constant rotor power. Nevertheless, the great number of variables and uncertainty considered allows keeping the differences below the one standard deviation threshold. The increase of thrust and power standard deviations, maintaining constant input variable distributions, suggests that the latter assume greater influence on loads as the altitude decreases. As a final result, figures from 4.31 to 4.34 show a complete statistical description of the thrust and power in the 1R

altitude case considering as before Cumulative Distribution Functions (CDF) and Probability Distribution Functions (PDF).

Load	Mean	Std Deviation	Exp Data	N° Std Deviation
Thrust	27.8528	3.8174	29.0067	0.3022
Roll	−0.0071	0.5583	−0.2525	0.4395
Pitch	−0.2491	0.5649	−0.3082	0.1046
Power	362.8009	49.1283	399.4418	0.7458

Table 4.16: 2R altitude loads comparison

Load	Mean	Std Deviation	Exp Data	N° Std Deviation
Thrust	31.4166	4.3563	35.8256	1.0121
Roll	−0.0081	0.5528	−0.2401	0.4197
Pitch	−0.2472	0.5631	−0.2490	0.0032
Power	375.9794	52.5775	413.0807	0.7056

Table 4.17: 1R altitude loads comparison

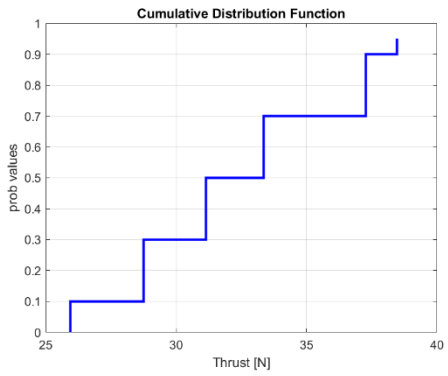

Figure 4.31: 1R altitude Thrust CDF

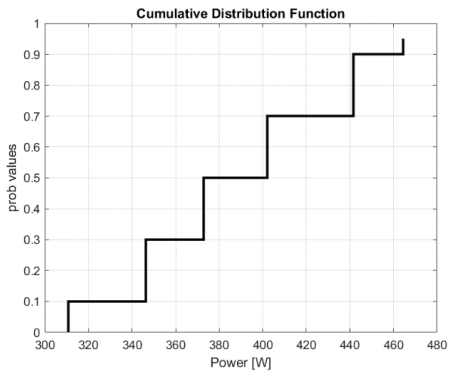

Figure 4.32: 1R altitude Power CDF

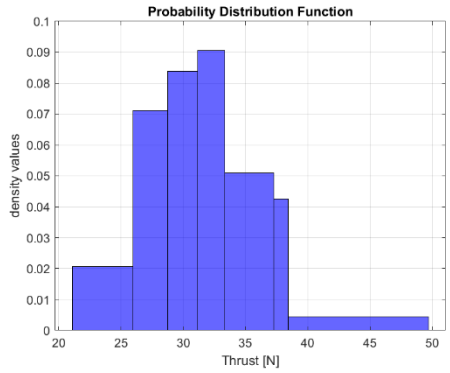

Figure 4.33: 1R altitude Thrust PDF

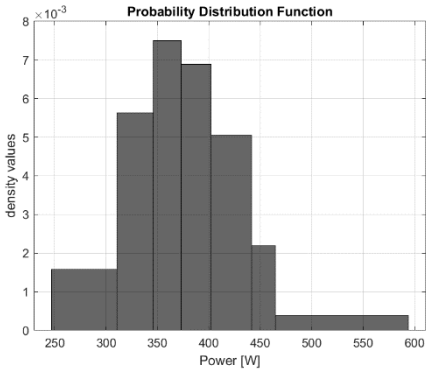

Figure 4.34: 1R altitude Power PDF

Assessing and modeling correctly aerodynamic interaction between helicopters and obstacles is of primary interest for safety reasons. Rotorcraft are, in fact, very versatile flying machines that are usually required to operate close to ground or structures [45]. According to the accident database collected by the Joint Helicopter Safety Analysis Team (JHSAT) [46], take-off and landing are the flight conditions causing the greatest number of accidents, making In Ground Effect (IGE) analysis, a mandatory step in experimental campaigns and numerical simulations. As can be noted, model predictions of thrust and power (and equivalently torque) in IGE conditions, underestimate the experimental data. This trend is especially visible at 1R altitude, in which, the proximity of the ground heavily affects the aerodynamic flow and consequently the rotor behavior. Despite this, the great number of parameters and related uncertainties considered in the present work, succeed in assessing the loads also in the IGE case in a statistical sense. Experimental data are, in fact, in the range of one standard deviation from the mean distribution values, and so possible realizations. This solution is, however, useful in early development stages or in all the conditions in which little model information are accessible, and the knowledge is covered in uncertainties. In all other cases, a specific cause identification for misleading solutions is necessary to develop a higher phenomenon understanding. The idea is to address the difference between model computed loads and data in the uncertainty related to the pitch link value, which can change in ground proximity due to swashplate flexibility and its non-linear behavior. To set up the problem, only pitch link stiffness is assumed as uncertainty variable, following Normal probability distribution constructed from Bayes Calibration results, with mean $\mu(K_p) = 15.02\ N/m$, and standard deviation $\sigma(K_p) = 2.66\ N/m$. All other input parameters are stuck at their mean values and considered deterministic. In order to consider also RPM change from OGE to IGE condition, thrust and torque coefficients are chosen as quantities of interest. These are respectively defined as:

$$C_T = \frac{T}{\rho A v_{tip}^2} \tag{4.3}$$

$$C_Q = \frac{Q}{\rho A R v_{tip}^2} \tag{4.4}$$

in which T is the thrust, Q the torque, ρ the air density, R the rotor radius, A the rotor disk area used as reference section and v_{tip}^2 is the reference speed chosen as the tip one, equal to ΩR with Ω indicating the rotor rotational speed, expressed in rad/s. The air density is kept constant at its standard condition value: $\rho = 1.225\ Kg/m^3$, while Ω is changed according to the altitude using the mean experiment RPM values. Statistical load coefficients descriptions are obtained through forward uncertainty quantification analysis using LHS technique with 200 samples for each environment condition. Tables 4.18 and 4.19 show the obtained results in terms of means and standard deviations for each altitude. Thrust and torque coefficients are normalized with respect to the same obtained using 4R test data (C_T/C_{T_n}), in order to compare results with respect to a reference condition. As before, the last column is used to identify the number of standard deviations dividing the experimental data from the mean values.

Height	Mean	Std Deviation	Exp Data	N° Std Dev
$4R$	1.0069	$5.588e-2$	1.000	0.1235
$3R$	1.0068	$5.579e-2$	1.0060	0.0143
$2R$	1.0760	$6.267e-2$	1.1303	0.8664
$1R$	1.2305	$8.018e-2$	1.4148	2.2985

Table 4.18: Thrust Coefficient

Height	Mean	Std Deviation	Exp Data	N° Std Dev
$4R$	0.9970	$4.367e-2$	1.000	0.0687
$3R$	0.9970	$4.359e-2$	1.0143	0.3968
$2R$	1.0251	$4.844e-2$	1.1397	2.3658
$1R$	1.0837	$6.076e-2$	1.2026	1.9568

Table 4.19: Torque Coefficient

Figures 4.35 and 4.36 show a comparison between model simulations and test data considering all the parameter space, defined as three times the standard deviations, inside the vertical bars. For the thrust coefficient case also the Fradenburgh [47] equation is shown, which is usually used as a reference result for rigid rotor test cases. Starting from the rigid case, adding flexibility to the model leads to an increase in both thrust and torque coefficients when crossing the OGE/IGE boundary. In this view, the pitch link stiffness decrease can be a possible cause of the difference between model predictions and test measurements. However, especially for the 1R case, experimental data fall into the considered range only when this is constructed assuming 2/3 times the standard

deviation. This large uncertainty bound, leaves the possibility that other non considered sources can be responsible as well.

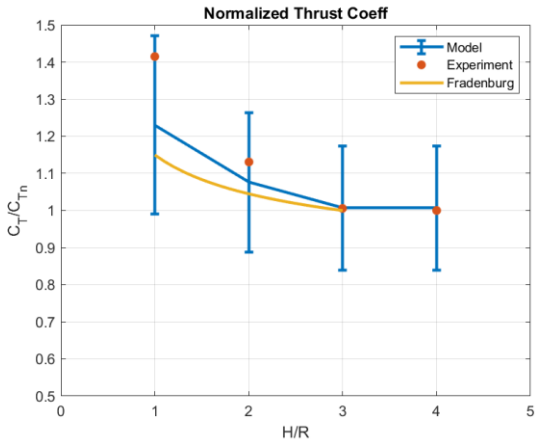

Figure 4.35: Normalized thrust coefficient
(° Experiment; -- Model; -- Fradenburg)

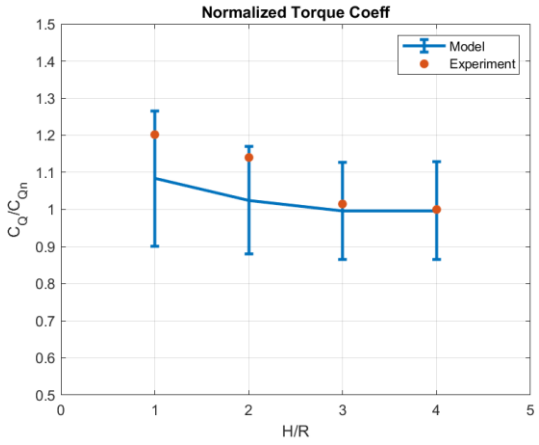

Figure 4.36: Normalized torque coefficient
(° Experiment; -- Model)

Chapter 5

Conclusions

Dakota-MBDyn interaction has provided a comprehensive investigation tool to validate and assess different types of analyses on helicopter multi-body models. Uncertainty quantification (UQ) capability allows deriving statistics on the quantity of interest when different uncertainties affect the model in the form of both inherent randomness and lack of knowledge. When no precise variable information is available, Bayesian Calibration is capable of deriving input parameter main statistical moments from experimental data, allowing the possibility of reproducing the test conditions within an uncertainty perspective. This initial information can be augmented by a Calibration under Uncertainty nested model, which in return provides the limit uncertainty conditions that can be accepted in order to obtain a pre-set margin on response functions. The combined use of the two tools can, in principle, supply a complete statistical description starting from conditions as general as possible. The correct operational behavior is, however, only ensured when applied to variables with the same influence on responses. Critical judgment is necessary to establish if the obtained results make sense depending on the application scope. The cases taken into consideration in this work are of simple nature, where the rotorcraft, in hover, changes its altitude. More complete and difficult conditions, as for certification requirements, are left for future researches.

Appendix

Bayes Calibration methods

This appendix provides a detailed description of the methods and numerical algorithms involved in the Bayes Calibration analysis run in the thesis. Other techniques that do not rely on MCMC estimations are possible, even if they are not explained here.

Markov Chain Montecarlo (MCMC)

MCMC estimates integral quantities by sampling over the posterior distribution. Obviously, the posterior distribution is not known at priori and must be quantified in some manner before operating. The explanation of how this quantification is performed and all the operations involved require a basic background about sampling techniques to understand how they work and for which purpose they have been implemented [48].
Starting from equation 2.21 in one dimension, a possible and standard numerical technique to approximate the integral is the Riemann sum over a discrete grid of points:

$$E_P[f(\theta)] = \int f(\theta)P(\theta)d\theta \approx \sum_{i=1}^{n} f(\theta_i)P(\theta_i)\Delta\theta_i \qquad (A.1)$$

where $\Delta\theta_i = \theta_{j+1} - \theta_j$ is the spacing between the set of $j = 1 \dots n+1$ grid points and θ_i is the mid point.
This concept can be generalized to higher dimensions in which, instead of dividing the space into n segments, it can be decomposed into n N-dimensional cuboids each one contributing to the integral by the product of the "height" $f(\theta_i)P(\theta_i)$ and the volume

$$\Delta\theta_i = \prod_{j=1}^{d} \Delta\theta_{i,j} \qquad (A.2)$$

This simple approach however is not efficient since the number of grid points needed for evaluations increases exponentially with the problem dimensionality. In addition, since the posterior density is an unknown quantity, the grid can be partitioned in an uneven way leading to wrong estimations. The method so not only depends on the number of grid points but also on where they are allocated. Equation A.1, in fact, can be seen as a weighted mean, with each weight w_i equal to $\hat{P}(\theta_i)\Delta\theta_i$. Here $\hat{P}(\theta_i)$ is the unnormalized

posterior $\hat{P}(\theta_i) = \pi(\theta)L(\theta)$ and, recalling that $Z = \int \hat{P}\,(\theta)d\theta$, the same equation can be written as

$$E_P[f(\theta)] \approx \frac{\sum_{i=1}^{n} f(\theta_i)\hat{P}(\theta_i)\Delta\theta_i}{\sum_{i=1}^{n} \hat{P}(\theta_i)\,\Delta\theta_i} = \frac{\sum_{i=1}^{n} f_i w_i}{\sum_{i=1}^{n} w_i} \qquad (A.3)$$

From equation A.3, it is evident the connection between the estimation of $E_P[f(\theta)]$ using a grid of n points and using a series of n samples f_i with associated weights w_i. Moreover, there is a strict connection between the weights and the posterior distribution. Although not proven here, it is convenient to have uniform weights, i.e., smaller grid resolution where the posterior distribution is higher in order to increase the accuracy of the estimation. In the limit case of $n \to \infty$ this ends up in defining a continuous grid function $Q(\theta)$ inversely proportional to $\Delta\theta_i$. With some substitutions the expected value can be computed as

$$E_P[f(\theta)] = \frac{\int f(\theta)\hat{P}(\theta)d\theta}{\int \hat{P}(\theta)d\theta} = \frac{E_Q[\frac{f(\theta)\hat{P}(\theta)}{Q(\theta)}]}{E_Q[\frac{\hat{P}(\theta)}{Q(\theta)}]} \qquad (A.4)$$

The expected value of $f(\theta)$, according to the posterior distribution $P(\theta)$, is now divided into two new expected values computed with respect to a new distribution $Q(\theta)$, called the *proposal distribution*. This simple mathematical trick allows its estimation since, as reminded, nothing is known at the beginning about the posterior distribution, while the proposal distribution is already defined and in practice it is possible to derive the expression explicitly by sampling over $Q(\theta)$. Obviously the ideal would be having the proposal distribution equal to the posterior one $Q(\theta) = P(\theta)$ since in this way the weights would be constant and equal to the evidence Z.

This solution is what Markov Chain Montecarlo (MCMC) method seeks to do, creating a chain of parameter values $\{\theta_1 \to \cdots \to \theta_n\}$ over n iterations such that the number of iterations $r(\theta_i)$ spent in a particular region $\Delta\theta_i$ centered on θ_i is proportional to the posterior distribution. The density of generated samples is equal to

$$\rho(\theta_i) = \frac{r(\theta_i)}{n} \qquad (A.5)$$

and the posterior integral in the parameter region can be approximated by

$$\int P(\theta)d\theta \approx \int \rho(\theta)d\theta \approx n^{-1}\sum_{j=1}^{n} \mathrm{I}\,[\,\theta_j \in \Delta\theta_j] \qquad (A.6)$$

where I is the indicator function giving 1 if the parameter is inside the region and 0 otherwise.
Increasing the number of iterations $\rho(\theta_i)$ will be closer to the posterior distribution, allowing the expected value evaluations by simply averaging over the samples

$$E_P[f(\theta)] \approx E_\rho[f(\theta)] = n^{-1}\sum_{i=1}^{n} f_i \qquad (A.7)$$

The only question left is how to generate the chain sample to reconstruct the posterior distribution and its related quantities. Various algorithms of different complexity succeed in this goal.

Metropolis Hastings Algorithm

The basic idea is to generate new chain samples $\theta_i \rightarrow \theta_{i+1}$ such that the inferred distribution is stationary (converges) and is equal to the posterior distribution $P(\theta)$.
The stationary condition can be satisfied using *detailed balance*, enabling the conservation of the probability when moving from one sample to another, i.e., the probability must be reversible. This concept can be written down as

$$P(\theta_{i+1}|\theta_i)P(\theta_i) = P(\theta_{i+1},\theta_i) = P(\theta_i|\theta_{i+1})P(\theta_{i+1}) \qquad (A.8)$$

$P(\theta_{i+1}|\theta_i)$ is the probability of moving from position $i+1$ to position i, while $P(\theta_i|\theta_{i+1})$ is the reverse probability of moving from position i to position $i+1$.

The procedure to generate new chain samples is divided into two steps. A first position proposal is made $\theta_i \rightarrow \theta'_{i+1}$ according to the proposal distribution $Q(\theta'_{i+1}|\theta_i)$, which is accepted or not by some *transition probability* $T(\theta'_{i+1}|\theta_i)$. If the iteration is successful then $\theta_{i+1} = \theta'_{i+1}$ and the algorithm proceeds, otherwise, if it is rejected, $\theta_{i+1} = \theta_i$ with a consequent new proposal.
The Metropolis criterion satisfies the balance constraint, staring the Transition probability as

$$T(\theta_{i+1}|\theta_i) = \min\left[1, \frac{P(\theta_{i+1})}{P(\theta_i)}\frac{Q(\theta_i|\theta_{i+1})}{Q(\theta_{i+1}|\theta_i)}\right] \qquad (A.9)$$

The Metropolis Hastings algorithm [49,50] can be then described as following

1. Propose a new chain sample θ'_{i+1} from the proposal distribution $Q(\theta'_{i+1}|\theta_i)$
2. Compute the transition probability $T(\theta'_{i+1}|\theta_i)$
3. Generate a random number $u \in [0,1]$
4. If $u \leq T(\theta'_{i+1}|\theta_i)$ accept the sample and $\theta_{i+1} = \theta'_{i+1}$, otherwise $\theta_{i+1} = \theta_i$

5. Increase i and repeat

Since no information is available at the beginning about the posterior, the initial chain values can be in a low probability region. For this reason, generally, a *burn in samples* process is executed, in which a specified number of initial samples are eliminated from the chain in order to not compromise the final result.

DRAM algorithm

DRAM (Delayed Rejaction Adaptive Metropolis) combines two different algorithms ideated to improve the efficiency of a classic Metropolis Hastings [51].

Delayed Rejection is a local adaptive strategy in which, after a proposal chain value rejection, instead of retaining the same position, a second proposal stage is performed. The transition probability for this second stage will be dependent not only on the current position, but also on what has been accepted and rejected previously. Since all acceptance probabilities are computed so that the reversibility property of the posterior distribution is maintained, the delayed rejection strategy can be interrupted at any stage or can be iterated for a fixed or random number of stages.

Adaptive Metropolis is a global adaptive strategy in which a Gaussian proposal distribution is created with a covariance matrix, calibrated using the previous samples of the chain. After an initial non-adaptation period, the Gaussian proposal is centered on the current chain position θ_i , with its covariance matrix defined as

$$C_i = s_d Cov(\theta_0, \dots, \theta_{i-1}) + s_d \varepsilon I_d \qquad (A.10)$$

where s_d is a parameter depending on the problem dimensionality, $\varepsilon > 0$ is a constant, I_d is the identity matrix and

$$Cov(\theta_0, \dots, \theta_k) = \frac{1}{k}\left(\sum_{j=0}^{k} \theta_j \, \theta_j^T - (k-1)\bar{\theta}_k \bar{\theta}_k^T\right) \qquad (A.11)$$

with $\bar{\theta}_k = \frac{1}{k+1}\sum_{i=0}^{k} \theta_i$

The choice of the length of the initial non-adaptation period is free to be chosen by the user with more increasing time adaptation cost as the period gets bigger. It has been found that the adaptation process should not be done at each time step but at given time intervals.

MCMC methods mainly depend on the proposal distribution, since it is the element that governs the acceptance rate during the chain movement. In its basic form DR employs a given number of proposals used at different stages and therefore its operation relies on

the probability that at least one of these is successfully calibrated. On the other hand, AM tunes the covariance matrix, and so the proposal distribution, considering the previous chain steps. DRAM algorithm works combining AM adaptation process with a m-stages of DR:

- The proposal at the first stage is adapted. The covariance C_n^1 is constructed from n points of the sample chain
- The proposal covariance C_n^i at stage $i = 2, \dots, m$ is a scaled version of the first one $C_n^i = \gamma_i C_n^1$

The scaled parameter γ_i can be freely chosen, allowing smaller or higher covariance matrices at the following stages.
The combination of DR and AM has drastically increased the acceptance rate and MCMC functionality, making DRAM one of the most used algorithms to assess Bayes Calibration.

Bibliography

[1] Lier, Max & Krenik, Alex & Kunze, Philipp & Schwinn, Dominik & Kohlgrüber, Dieter & Lützenburger, Marius. (2015). A Toolbox for Rotorcraft Preliminary Design.

[2] Anderson, J. D.: Aircraft Performance and Design. McGraw-Hill, Boston, 1999.

[3] A. Ragazzi, R. B. Mengotti, P. Sabato, G. Afruni, and C. Hyder, "AW169 loss of tail rotor effectiveness simulation," 43rd Eur. Rotorcr. Forum, ERF 2017, vol. 1, pp. 401–413, 2017.

[4] CS-29 – Certification Specification for Large Rotorcraft, Amendment 2 dated 17/11/2008

[5] Limbourg, Philipp. (2008). Dependability Modelling under Uncertainty - An Imprecise Probabilistic Approach. 10.1007/978-3-540-69287-4.

[6] EASA, "Notification of a Proposal to issue a Certification Memorandum: Modelling and Simulation – CS-25," no. 01, pp. 1–55, 2020.

[7] Quaranta, Giuseppe & Hoff, Stefan & Jones, Michael & Lu, Linghai & White, Mark. (2021). Challenges and Opportunities Offered by Flight Certification of Rotorcraft by Simulation.

[8] Masarati, P., Piatak, D. J., Quaranta, G., Singleton, J. D., and Shen, J., "Soft Inplane Tiltrotor Aeromechanics Investigation Using Two Comprehensive Multibody Solvers," Journal of the American Helicopter Society, Vol. 53, (2), 2008, pp. 179–192.

[9] Caccia, Claudio & Masarati, Pierangelo. (2021). Coupling multi-body and fluid dynamics analisys with preCICE and MBDyn. 10.23967/coupled.2021.014.

[10] Adams, B.M., Bohnhoff, W.J., Dalbey, K.R., Ebeida, M.S., Eddy, J.P., Eldred, M.S., Hooper, R.W., Hough, P.D., Hu, K.T., Jakeman, J.D., Khalil, M., Maupin, K.A., Monschke, J.A., Ridgway, E.M., Rushdi, A.A., Seidl, D.T., Stephens, J.A., Swiler, L.P., and Winokur, J.G., "Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.14 User's Manual," Sandia Technical Report SAND2021-5822, May 2021.

[11] Kennedy, Marc & O'Hagan, Anthony. (2001). Bayesian Calibration of Computer Models. Journal of the Royal Statistical Society Series B. 63. 425-464. 10.1111/1467-9868.00294.

[12] C.J. Roy, W.L. Oberkampf. "A comprehensive framework for verification, validation, and uncertainty quantification in scientific computing" Comput Methods Appl Mech Eng, 200 (25) (2011), pp. 2131-2144

[13] Guerroni, F., (2020). Development of a Stochastic parametric tiltorotor whirl flutter numerical predictor using general-purpose multibody and optimization software (Master's Thesis, Poitecnico di Milano, Milano, Italia). Retrieved from http://hdl.handle.net/10589/165243

[14] Dalbey, K.R., Eldred, M.S., Geraci, G., Jakeman, J.D., Maupin, K.A., Monschke, J.A., Seidl, D.T., Swiler, L.P., Tran, A., Menhorn, F., and Zeng, X., "Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.14 Theory Manual," Sandia Technical Report SAND2021-5821, May 2021.

[15] Stein, M., "Large Sample Properties of Simulations Using Latin Hypercube Sampling,"Technometrics,Vol. 29, (2), 1987, pp. 143–151.

[16] Haldar, A., and Mahadevan., S.,Probability, Reliability, and Statistical Methods in Engineering Design, Wiley,New York, 2000.

[17] M. Rosenblatt. Remarks on a multivariate transformation. Annals of Mathematical Statistics, 23(3):470–472, 1952.

[18] A. Der Kiureghian and P. L. Liu. Structural reliability under incomplete information. J. Eng. Mech., ASCE, 112(EM-1):85–104, 1986.

[19] Kadry, Seifedine & Chateauneuf, Alaa. (2007). Probabilistic Transformation Method in Reliability Analysis. Env. Sci. 31. 135-148.

[20] R. Ghanem and J. R. Red-Horse. Propagation of probabilistic uncertainty in complex physical systems using a stochastic finite element technique. Physica D, 133:137–144, 1999.

[21] R. G. Ghanem and P. D. Spanos. Stochastic Finite Elements: A Spectral Approach. Springer-Verlag, New York, 1991.

[22] N. Wiener. The homogoeneous chaos. Amer. J. Math., 60:897–936, 1938.

[23] R. Askey and J. Wilson. Some basic hypergeometric polynomials that generalize jacobi polynomials. In Mem. Amer. Math. Soc. 319, Providence, RI, 1985. AMS.

[24] Dongbin Xiu and George Em Karniadakis. "The Wiener-Askey polynomial chaos for stochastic differential equations". In: SIAM journal on scientific computing 24.2 (2002), pp. 619-644

[25] W.L. Oberkampf and J. C. Helton. Evidence theory for engineering applications. Technical Report SAND2003-3559P, Sandia National Laboratories, Albuquerque, NM, 2003.

[26] E. T. Jaynes and G. Larry. Bretthorst. Probability theory: the logic of science. Cambridge University Press, Cambridge, UK; New York, NY, 2003.

[27] HAMADA M S, WILSON A, REESE C S, MARTZ H. Bayesian reliability [M]. Berlin: Springer Science & Business Media, 2008: 43–50.

[28] Swiler, Laura & Adams, Brian & Eldred, Michael. (2008). Model Calibration Under Uncertainty: Matching Distribution Information. 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, MAO. 10.2514/6.2008-5944.

[29] Gratton, Serge & Lawless, Amos & Nichols, Nancy. (2007). Approximate Gauss–Newton Methods for Nonlinear Least Squares Problems. SIAM Journal on Optimization. 18. 106-132. 10.1137/050624935.

[30] Dennis, J. E., Gay, D. M., and Welsch, R. E.,ALGORITHM 573: NL2SOL–an adaptive nonlinear least squaresalgorithm, ACM Trans. Math. Software, 1981.

[31] N. Cressie. Statistics of Spatial Data. John Wiley and Sons, New York, 1991.

[32] G. Matheron. The theory of regionalized variables and its applications. Les Cahiers du Centre de morphologie math´ematique de Fontainebleau. ´ Ecole national sup´erieure des mines, 1971.

[33] White, M. D., Cameron, N., Lumsden, B., and Padfield, G. D., “Virtual Engineering in Skills Acquisition and Development in the Career of the Rotorcraft Engineer,” Conference on Rotorcraft Virtual Engineering, Liverpool, UK, November 2016.

[34] American Society of Mechanical Engineers, Guide for verification and validation in computational solid mechanics, 2006.

[35] Eldred, M.S., Bichon, B.J., and Adams, B.M., "Overview of Reliability Analysis and Design Capabilities in DAKOTA,Proceedings of the NSF Workshop on Reliable Engineering Computing (REC 2006), Savannah, GA, February 22-24, 2006.

[36] Weickum, G., Eldred, M.S., and Maute, K., "Multi-point Extended Reduced Order Modeling For Design Optimization and Uncertainty Analysis, paper AIAA-2006-2145 in Proceedings of the 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference (2nd AIAA Multidisciplinary Design Optimization Specialist Conference), Newport, Rhode Island, May 1 - 4, 2006.

[37] Dieterich, Oliver & Götz, Joachim & Dangvu, Binh & Haverdings, Henk & Masarati, Pierangelo & Pavel, Marilena & Jump, Michael & Gennaretti, Massimo. (2008). Adverse Rotorcraft-Pilot Coupling: Recent Research Activities in Europe.

[38] Jeremy Ledoux, Sebastián Riffo, Julien Salomon. Analysis of the Blade Element Momentum Theory. 2020. hal-02550763

[39] Wikipedia contributors. (2021, November 11). Sensitivity analysis. In Wikipedia, The Free Encyclopedia. Retrieved 11:51, November 14, 2021, from https://en.wikipedia.org/w/index.php?title=Sensitivity_analysis&oldid=1054712826

[40] N. Taymourtash, A. Zanotti, G. Gibertini, G. Quaranta "Simulation and Testing of Helicopter-Ship Aerodynamic Interaction", 47th European Rotorcraft Forum, Glasgow, Scotland (2021)

[41] Ghiringhelli, G. L., Masarati, P., and Mantegazza, P., "Multibody Implementation of Finite Volume C Beams," AIAA Journal, Vol. 38, (1), January 2000, pp. 131–138.

[42] Moriarty, Patrick & Hansen, A.. (2005). Aerodyn Theory Manual. National Renewable Energy Laboratory. 10.2172/15014831.

[43] Padfield, G. D., Helicopter Flight Dynamics: The Theory and Application of Flying Qualities and Simulation Modelling, 2nd ed., Blackwell Publishing Ltd, Oxford, UK, 2007. https://doi.org/10.1002/9780470691847.

[44] Arra M. L'elicottero. Hoepli Editore, Milano, 2001.

[45] Zanotti, Alex & Gibertini, Giuseppe & Grassi, Donato & Zagaglia, Daniele. (2015). Experimental investigation on the aerodynamic interaction between a helicopter and ground obstacles. Proceedings of the Institution of Mechanical Engineers Part G Journal of Aerospace Engineering. 229. 1395-1406. 10.1177/0954410014550501.

[46] International Helicopter Safety Team. The U.S. JHSAT baseline of helicopter accident analysis, www.ihst.org/portals/54/US_JSHAT_Compendium_Report1.pdf (2011, accessed August 2011).

[47] Fradenburgh EA. The helicopter and the ground effect machine. J Am Helicopter Soc 1960; 5(4): 24–33

[48] Speagle, J. S., "A Conceptual Introduction to Markov Chain Montecarlo Methods," Technical report, 2020.

[49] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., and Teller, E. (1953). "Equations of state calculations by fast computing machines." J. Chem. Phys., 21(6): 1087–1092.

[50] Hastings, W. (1970). "Monte Carlo sampling methods using Markov chains and their application." Biometrika, 57: 97–109.

[51] Haario, Heikki & Laine, Marko & Mira, Antonietta & Saksman, Eero. (2006). Dram: efficient adaptive MCMC. Statistics and Computing. 16. 339-354. 10.1007/s11222-006-9438-0.