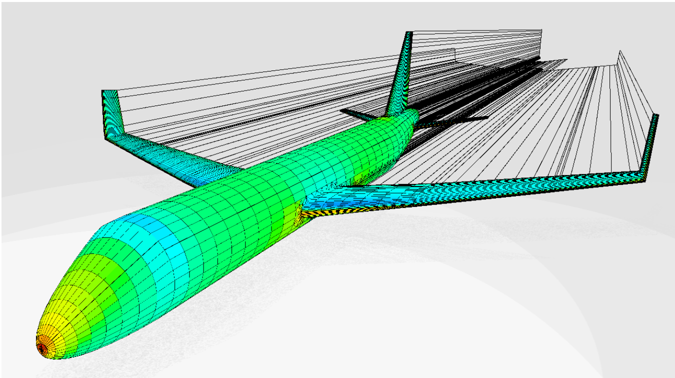


Stochastic Optimization in Aircraft Design

Luis Filipe Barreiras Amândio

Thesis to obtain the Master of Science Degree in

Aerospace Engineering

Examination Committee

Chairperson:	Professor Fernando José Parracho Lau
Supervisor:	Doutor André Calado Marta
Supervisor:	Professor Afzal Suleman
Members of the Committee:	Professor Aurélio Lima Araújo
	Doutor José Lobo do Vale

November 2013

Dedicated to my family.

Acknowledgments

Even though I cannot name everyone who has contributed in someway to this thesis and to me finishing my master's degree, I thank them all.

I would like to show my deepest appreciation to Doctor André Marta, not only for the knowledge of optimization he lent me, his constant support, but most of all, for the availability to help that he showed during the course of this work.

I would like to thank Doctor Afzal Suleman for his support as well as his key guidance which led to this thesis being what it is today.

I cannot thank Frederico Afonso enough for his patience, his insight and mostly, for the invaluable help he provided me, without which I would not have been able to finish this work.

Finally, I would like to thank my family and friends, especially to my parents, for all the support they have given me over the years.

Resumo

Esta tese foca-se na análise das vantagens e desvantagens associadas à utilização de optimização estocástica, especialmente em problemas de projecto de aeronaves. Primeiro, foi feita uma revisão bibliográfica de forma a escolher os métodos mais comuns e promissores de projecto robusto, projecto baseado em fiabilidade e projecto robusto e baseado em fiabilidade. Os métodos escolhidos foram o Monte Carlo, o método dos momentos, o ponto Sigma, aproximação de índice de fiabilidade, aproximação de medição de performance, optimização sequencial e avaliação de fiabilidade e espaço de projecto fiável. Em seguida, estes métodos foram testados em duas funções analíticas e as suas performances foram comparadas. Quatro destes métodos foram posteriormente escolhidos, tendo em conta a sua performance, para serem implementados numa ferramenta de optimização multidisciplinar desenvolvida para problemas de projecto de aeronaves. Por forma a avaliar os métodos escolhidos num ambiente mais realístico, desenvolveram-se dois casos de estudo baseados em fiabilidade e relacionados com optimização multidisciplinar de aeronaves. Nestes casos, foram utilizados modelos de aproximação em detrimento das análises disciplinares computacionalmente dispendiosas, tendo o objectivo principal sido o estudo da variação da eficiência de cada método com o número dos parâmetros de incerteza. Os resultados obtidos revelaram que a eficiência de cada método está muito relacionada com o tipo de problema. Enquanto nos casos analíticos, para níveis elevados de incerteza, o método de projecto robusto mostrou dificuldades em obter a fiabilidade desejada, nos casos de projecto de aeronaves, este mostrou ser o melhor método em termos da relação qualidade dos resultados/custo computacional.

Palavras-chave: Propagação de incerteza, Projecto robusto óptimo, Projecto baseado em fiabilidade, Optimização multidisciplinar, Comparação de métodos

Abstract

This thesis focuses on analysing the advantages and disadvantages of using stochastic optimization, especially in aircraft design problems. First, a literature review served as a starting point to choosing some of the most common and promising methods of robust design optimization, reliability based design optimization and robust and reliability based design optimization. The chosen methods were Monte Carlo, method of moments, Sigma point, reliability index approach, performance measure approach, sequential optimization and reliability assessment, and reliable design space. After implementing these methods, they were tested for two analytic functions and their performances compared. Four of these methods were then chosen based on their performances to be implemented in a multidisciplinary optimization tool specially tailored to solve aircraft optimization problems. To evaluate the chosen methods in a more realistic environment, two new reliability based test cases related to aircraft design were developed. In these test cases, surrogate models were employed instead of the more computationally expensive disciplinary analysis, with the main objective being the study of how the efficiency of each method changed with the number of uncertainty parameters. The obtained results revealed that the efficiency of each method is closely related to the type of problem solved. While in the analytic cases, for high levels of uncertainty, the robust optimization method showed some difficulties in achieving the target reliability, in the aircraft design cases, it proved to be the best method in terms of the relation between accuracy and computational cost.

Keywords: Uncertainty Propagation, Robust Design Optimization, Reliability-Based Design Optimization, Multidisciplinary optimization, Benchmarking

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α Angle of attack.

β Reliability index, Angle of side-slip.

χ Sigma point.

γ Angle of climb/descent.

μ Mean value.

Φ Variance.

ϕ Set of hyper-spherical angular coordinates.

σ Standard deviation, stress.

σ^2 Variance.

σ_{yield} Yield stress.

ε Error.

Roman symbols

c Chord.

$c.o.v.$ Coefficient of variance.

C_D Coefficient of drag.

C_L Coefficient of lift.

c_t Specific fuel consumption.

$E(...)$ Expectation operator.

f, F Objective function.

f_X Probability density function.

g, G Vector of inequality constraints.

h Vector of random variables and/or parameters.

HT_{inc} Horizontal tail incidence.

$K(...)$ Kurtosis.

$N, \#$ Number of.

P Probability.

p Vector of random parameters.

R Range.

r Vector of parameters (random and/or deterministic).

s Shifting vector.

t_s Skin thickness.

t_w Web thickness.

t_{max} Maximum thickness.

u Vector of random variables transformed into the standard normal space.

V Airspeed.

X Random variable.

x Vector of design variables.

D Drag.

L Lift.

T Thrust.

t Value of the random variable.

W Weight.

Subscripts

0 Initial point.

$allow$ Allowable.

DV Design variable.

f Failure.

i, j, k Computational indexes.

MPP Solution of the MPP.

$reqd$ Required.

RV Random/stochastic variable.

Superscripts

0 Initial point.

d Deterministic constraint.

LB Lower bound.

rc Reliability constraint.

UB Upper bound.

* Evaluated at the MPP.

-1 Inverse.

T Transpose.

Glossary

ADA	Adaptive Decoupling Approach.
AMV	Advanced Mean Value.
CCSA	Conservative Convex Separable Approximations.
CDF	Cumulative Density Function.
CFD	Computational Fluid Dynamics .
CG	Center of Gravity.
DR	Dimension Reduction.
FE	Finite Element.
FORM	First Order Reliability Method.
FSQP	Feasible Sequential Quadratic Programming.
GA	Genetic Algorithm.
GCMMA	Globally Convergent Method of Moving Asymptotes.
LHS	Latin Hypercube Sampling.
MC	Monte Carlo.
MDO	Multidisciplinary Optimization.
MM	Method of Moments.
MPP	Most Probable Point.
PCE	Polynomial Chaos Expansion.
PDF	Probability Density Function.
PMA	Performance Measure Approach.
QMC	Quasi Monte Carlo.
QP	Quadratic Programming.
R^2BDO	Robust and Reliability Based Design Optimization.
RBDO	Reliability Based Design Optimization.
RDO	Robust Design Optimization.
RDS	Reliable Design Space.
RIA	Reliability Index Approach.
ROC	Rate of Climb.
ROD	Rate of Descent.
SBDO	Surrogate Based Design Optimization.

SORA Sequential Optimization and Reliability Assessment.

SORM Second Order Reliability Method.

SP Sigma Point.

SQP Sequential Quadratic Programming.

Chapter 1

Introduction

1.1 Motivation

Optimization is the act of obtaining the best result under given circumstances [Rao, 2009]. In fact, everybody does it to some extent. Be it thinking about the fastest route between two points or arranging furniture in a way that maximizes space in a room. Even nature does it when physical systems tend to states of minimum energy and rays of light follow paths that minimize their travel time. It is only natural that optimization has also become an important tool for decision making, in a vast number of areas such as engineering and economics.

To optimize something, first it is necessary to identify some objective, i.e., a quantitative measure of the performance of the system under study. This objective will depend on certain characteristics of the system, called variables, that may or may not be constrained. In the optimization, the goal is to find values for the variables that optimize the objective. The process of identifying the objective, variables and constraints is known as modeling and is only the first step in the optimization. Creating an appropriate model is very important, as it will determine the quality of the optimization itself. If the model is too simple, the optimization may not provide useful insights into the practical problem, but if it is too complex, it may become too difficult to solve.

Once the model has been formulated, the next step concerns the choice of the optimization algorithm to find the solution of the problem. There is no single method to solve every optimization problem efficiently. Instead, a number of optimization methods have been developed, and some are still in development, to cater to different needs that are specific to certain problems. The study of these optimum seeking methods is part of operations research, which is a branch of mathematics concerned with the application of scientific techniques to decision making problems and with establishing the best or optimal solutions [Rao, 2009].

The existence of optimization methods can be traced back to the days of Newton and Lagrange, but the real progress in this area did not start until the middle of the twentieth century, when high-speed digital computers made implementation of the optimization procedures possible and stimulated further research on the subject.

From then on, major developments in the area of numerical methods were made. The foundations for further development were laid, which led the ever evolving methods to become even more capable and robust. Optimization, especially deterministic, consolidated its position as an important tool for numerous areas, specially engineering. Optimization in the design of aircraft and aerospace structures for minimum weight or design of civil engineering structures such as frames, bridges or dams for minimum cost became a standard procedure.

In the course of time, these deterministic methods have become more and more reliable until a point where the industry felt comfortable in using them. Even though these methods have proven useful during nearly six decades of design, they have several shortcomings. Specially when it comes to account for uncertainty, by means of a combination of safety factors and knockdown factors.

When specifically talking about the aerospace industry, the traditional procedures may be difficult to apply to vehicles with unconventional configurations and that use new material systems. Also, since the measures of both robustness and reliability are not provided in the design process, it is not possible to determine the relative importance that the design options have in these measures. In addition to this, it is impossible to maintain consistency in terms of reliability throughout the whole vehicle, which probably leads to excess weight with no corresponding improvement in overall reliability [Zang et al., 2002].

To overcome these problem, it is necessary to start implementing efficient uncertainty based methods for optimization. Instead of just using both safety and knockdown factors, if the uncertainty is properly characterized and then used to optimize, designers will be able to produce a consistent level of reliability and performance throughout the vehicle, avoiding unnecessary over designs in some areas. In addition, with an uncertainty-based design procedure it will be possible to determine the sensitivity of the reliability t some design changes, which in turn are linked to changes in cost. As a result, it will be possible to make trade-offs between reliability and cost. For example, for the same cost, airframes can be made safer than with traditional design approaches, or, for the same reliability, the airframe can be made at a lower cost [Zang et al., 2002].

The industry is currently comfortable with the traditional methods for solving multidisciplinary optimization problems, and because these newly developed uncertainty based procedures still have a lot of problems that need to be overcome, there is still a long way to go before they can become more popular than conventional deterministic methods. It is necessary to provide compelling demonstrations of the benefits of using uncertainty in optimization, to reduce the complexity of these methods, thus making them less computationally expensive, and also find ways to better characterize uncertainties.

This work will specially focus on how the uncertainty based optimization methods perform and what kind of benefits they have. Comparisons between them and deterministic methods are made to understand how their increased complexity translates into the obtained results. It is of utmost importance to carry out such studies if uncertainty based methods are to become more important in design and optimization processes.

1.2 Uncertainty Models

Uncertainty, which is generally defined as ”that which is not precisely known”, is present everywhere and in everything. Even though its characterization, quantification and management have not always been taken into account in science, awareness to its importance in decision and policy making is starting to increase.

Since the distant days, when the scientific community saw uncertainty as an undesirable state, until today, a lot of things have changed. When physicists realized that Newtonian mechanics did not address problems at the molecular level, they developed a statistics based method where statistical averages could replace specific manifestations of microscopic entities, thus accounting for uncertainty. This was one of the first ever created models to quantify uncertainty, and it was based on probability theory. Since then, research in the area increased, leading to the creation of several different methods, besides probabilities, that were able to quantify uncertainty to aid in decision making processes (such as uncertainty based optimization). A partial list of these mathematical uncertainty theories includes Probability Theory, Zadeh Fuzzy Sets and Logic Theory, Possibility Theory, Dempster-Shafer Evidence Theory, Imprecise Probability Theory and Random Intervals. These theories can be divided in three main categories, as illustrated in Fig.1.1, depending on how they approach uncertainty:

- Methods that specify uncertainties by means of interval bounds. These should only be used when little information is known about a certain system. In these methods, variables are represented by two scalars: a lower bound value and an upper bound value. These numbers reflect a measure of uncertainty in the knowledge of the actual value of the variable. Even though interval analysis is ideally suited to deal with uncertainties in parameters, it is only applicable to simple problems of small order. For medium to large scale problems, interval arithmetic produces potentially conservative results, thus rendering it less suitable [Zang et al., 2002];

- Membership functions, which represents the degree of membership of the fuzzy variable within the fuzzy set. These methods provide an intermediate level of detail and are mainly used when data necessary to quantify parameter uncertainties is limited. Fuzzy logic allows one to create models based on inexact, incomplete, or unreliable knowledge or data, and, moreover, to infer approximate behavior of the system from such models. Fuzzy logic is appealing for its simplicity of application and implementation, and while it may be useful in certain cases where inferences are acceptable, such as conceptual design or early preliminary design stages, it does not have the necessary level of accuracy for most aerospace applications in late preliminary and detailed design [Zang et al., 2002];

- Methods based on the probability density function (PDF), which are the most detailed ones. These should only be used when there is enough sample data in order to make generalizations about the populations from which the samples were obtained. The probabilistic methods are the most well developed approaches of uncertainty analysis. They can either be based on samples, by using some type of Monte Carlo (MC) simulation, or based on the solution of stochastic differential equations. Throughout this work, this is the kind of methods that is going to be used, for they are the ones that provide the most accurate results [Zang et al., 2002].

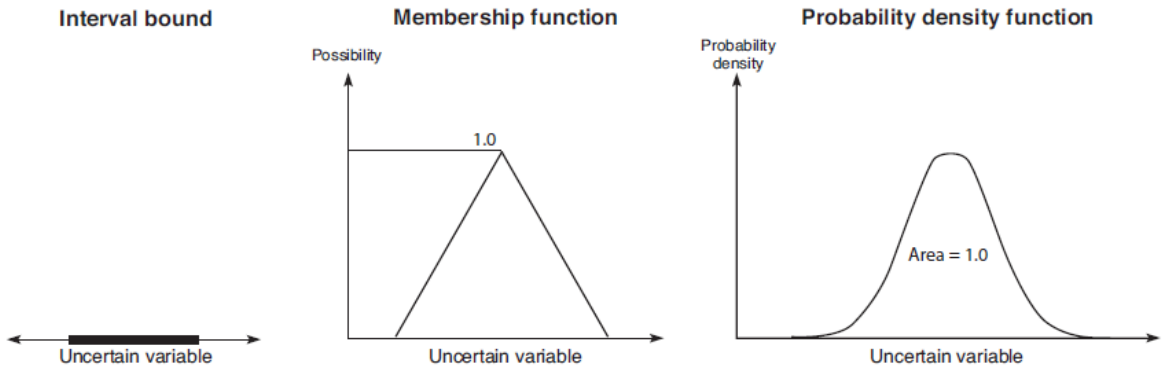


Figure 1.1: Uncertainty descriptions [Zang et al., 2002]

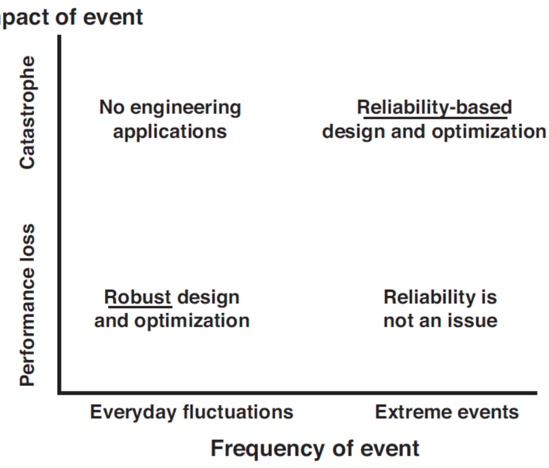
It is important to note that these theories concern a type of uncertainty called parameter uncertainty, which is tied to either the input data (boundary conditions or initial conditions) to a computational process or with basic parameters that define a given computational process, such as the coefficients of phenomenological models. There is also another type of uncertainty called model form uncertainty, which is associated with model validity, i.e., whether the nominal mathematical model adequately captures the physics of the problem, but this will not be approached in this work as the strategies for characterizing this type of uncertainty are far less developed than those for parameter uncertainty [Zang et al., 2002].

1.3 Uncertainty-based Optimization

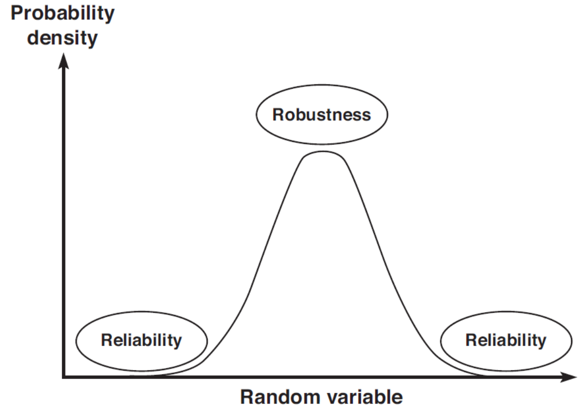
There are two major classes of uncertainty-based optimization methods, robust design optimization (RDO) and reliability based design optimization (RBDO). While robust optimization seeks a design insensitive to small changes in the uncertain quantities, the design sought by reliability optimization is one that has a probability of failure that is less than some acceptable value.

In order to achieve these different designs, not only their mathematical formulation is different, but also their domains of applicability. As can be seen in Fig.1.2(a), while robust optimization focus on obtaining designs whose performances are insensitive to everyday fluctuations, reliability optimization ensures that the events that may lead to failure are extremely unlikely to happen. As for the area of the PDF that concerns each class of uncertainty-based optimization, robust design focus primarily on the event distribution near the mean value, while reliability design is more concerned with the event distribution in the tails of the PDF.

As stated before, probabilistic optimization can rely on sampling methods such as MC. The only problem associated with MC is the fact that, depending on the problem, in order to obtain accurate results, the number of samples may need to be higher than one million, thus rendering this procedure too computationally expensive. Although methods such as Latin hypercube sampling [Mckay et al., 2000]



(a) Uncertainty-based design domains [Huyse, 2001]



(b) Reliability versus robustness in terms of the probability density function [Zang et al., 2002]

Figure 1.2: Robust vs reliability design optimization

or quasi-Monte Carlo (QMC) method [Niederreiter, 1978] can significantly improve the MC convergence rate, their applicability is limited.

As a means to overcome this, several new and more efficient methods of uncertainty propagation, that can be utilized in conjunction with uncertainty based optimization methods, have been developed. Since both robust and reliability design focus on different zones of the PDF, as can be seen in Fig.1.2(b), the kinds of methods that these two classes of stochastic optimization utilize to propagate uncertainty are also different.

There is a reasonable number of different uncertainty propagation methods to be used in conjunction with RDO. Some of them more simplistic like the Gaussian Quadrature [Padulo et al., 2007], others more complex to implement like the ones based on the Fast Probability Integration technique, such as the Advanced Mean Value (AMV) method [Mavris and Bandte, 1997]. It all depends on the complexity of the system under study, the level of uncertainty that is present and the desired precision of the results. In addition to these two methods, there are also others that are often used in the literature, such as the Taylor-based method of moments [Padulo et al., 2007] [Menshikova, 2010], the Sigma Point method [Padulo et al., 2007] [Bertuccelli and How, 2008], Polynomial Chaos Expansion [Xiong et al., 2011] or Dimension Reduction (DR) methods [Xu and Rahman, 2004]. Even though the goal of these methods is the same, they are very different in terms of the way they approach the problem and in terms of performance.

Much like what happened with RDO, RBDO also has some methods that have been specially developed to better approximate its focus region of the PDF (its tails). The most common methods to perform reliability analysis, are based on the Most Probable Point (MPP) problem. Methods such as the first order reliability method (FORM) and second order reliability method (SORM) are two popular methods in this category [Agarwal, 2004] [Du, 2005]. These methods usually involve a reformulation of the RBDO statement by using either the Reliability Index Approach (RIA) or the Performance Measure Approach (PMA) [Agarwal, 2004] [Tu et al., 1999]. Other methods such as Sequential Optimization Reliability Assessment (SORA) [Du and Chen, 2002], are also based on the same principles as the previous ones, but by decoupling the reliability assessment from the main optimization cycle, it makes the whole process faster. On the other hand, there are methods like the Reliability Design Space (RDS), which approach the MPP problem in a different way, and instead of using the reliability assessment cycle, turns the reliability constraints into deterministic ones and than does a single deterministic optimization [Shan and Wang, 2008].

As research in the area of uncertainty-based optimization increases, the trend seems to either be the improvement of the uncertainty propagation methods that already exist (in terms of numerical robustness and computational efficiency) or the adaptation of these same methods to very specific problems.

Even though there is more development in the RBDO field compared to RDO, this can also be seen for some of the RDO methods. Studies like [Wang, 2011] where changes are made to existing methods (DR in this case) in order to make them more efficient, are an example of the first trend. As for the second trend, an example is the development of an approach to deal with potential stamping problems during the early phase of product and tooling design [Hou et al., 2010].

In terms of RBDO, a lot of improvements have been made as well. Be it decoupled methods such as SORA, or single loop methods like RDS, all have been improved over the last years. Examples of this are, SORA being modified in order to be more computational efficient [Huang et al., 2012a] or to be able to perform optimizations with varying design variance [Yin and Chen, 2006]. Similar to SORA, a new decoupling method called Adaptive Decoupling Approach (ADA), that improves computational efficiency by adopting a new update angle strategy and a novel feasibility-checking method has been developed [Chen et al., 2013]. As for the the single loop methods class, a new method similar to RDS has been developed [Li et al., 2013], as well as one which uses a recently developed Matrix-based System Reliability (MSR) method in order to efficiently and accurately compute system failure probabilities and parameter sensitivities [Nguyen et al., 2010]. Even on the FORM front, some developments have been made that take advantage of specif properties of the problems, like monotony [de Rocquigny, 2009]. Much like in RDO, some of these methods are not only being applied, but also specially adapted to specific problems in certain areas such as civil engineering [Huang et al., 2012b] [Spence and Gioffre, 2012].

It is also important to note that some efforts have been made, in order to develop efficient methods that perform both robust and reliability optimization at the same time [Du and Chen, 2002] [Yu et al., 2013] [Yadav et al., 2010].

1.4 Relevance to Aeronautics

In the ever evolving and very demanding area of aerospace engineering, optimization, or in this case multidisciplinary design optimization (MDO), is already a well established tool that designers use for the design, manufacture, operation and disposal phases, to obtain aerospace vehicles that have better performance, higher reliability and robustness, and lower cost and risk. There are several examples in the literature that showcase the improved results that can be obtained through deterministic optimization, such as, the aerostructural optimizations of wings [Chiba et al., 2005] [Gumbert et al., 2005a] [Gumbert et al., 2005b], novel configurations of wingtips [Jansen et al., 2010] and even optimizations on the whole aircraft, with different methods and objective functions [Kaufmann et al., 2010] [Liem et al., 2012] [Martins et al., 2004] [Mastroddi and Gemma, 2013].

As competitiveness increases in the aerospace industry and manufacturers search for even better designs, the need for taking into account uncertainties in the optimization process, as a way to overcome the limitations of deterministic optimization, arises. For example, in structural design, uncertainties include prediction errors induced by design model assumption and simplification, performance uncertainty arising from material properties and manufacturing tolerance, and uncertainties of load conditions applied on the structure during operation. These uncertainties may cause system performance to fluctuate, or even cause failure, therefore making it important for them to be taken into account during the optimization process [Yao et al., 2011].

Even though uncertainty-based optimization still has some issues and barriers that need to be overcome [Zang et al., 2002], there is already a lot of work done in the aerospace field with results that showcase its potential. In the RDO side of things, methods for robust optimization specifically for aircraft applications are being developed [Meng and Ma, 2010], while other studies apply already existing methods in order to improve the efficiency of the optimizations [Padulo et al., 2008]. As for RBDO, literature shows that the problems are little bit more practical. Examples of research in this area include the optimization of wings using the SORA algorithm [Yu and Du, 2006], optimization of a wingtip with great results in comparison with the original model [Patnaik et al., 2010] and optimization of aeroelastic structures with the purpose of showing the advantages that including uncertainties in the design process brings, when comparing to deterministic optimization [Allen and Maute, 2004].

Since one of the problems that uncertainty-based optimization faces is the fact that it is much more computationally expensive than conventional (deterministic) optimization, and because aircraft are also complex and multidisciplinary systems that are computationally expensive to analyze, both the RDO and RBDO problems presented in the previous paragraph are considerably less complex than their deterministic counterparts. A solution that has been encountered is to use surrogate models. Surrogate models are computationally cheap functions to analyze, that are constructed based on data gathered through a limited number of runs of the aircraft high fidelity models. This way the optimization process becomes faster and the problems can grow in complexity and size.

1.5 Thesis Outline

This thesis is comprised of 7 chapters. Chapter 1 introduces the reader to the evolution of optimization over the years and to how uncertainty began to be introduced in it. It also shows a literature review on the subject of stochastic optimization and discusses about its importance in the aeronautics field.

Chapter 2 briefly explains the different classes of uncertainty and how it can be quantified. It also introduces the readers to some statistical concepts, that are used in this quantification.

In Chapter 3, the concepts of robustness and reliability in optimization are explained (RDO and

RBDO formulations), as well as an hybrid formulation that utilizes both this concepts (R^2BDO). Some methods of stochastic optimization are introduced and then tested for a couple of analytic functions. The obtained results are used to compare the different methods and formulations.

Chapter 4 presents the MDO tool and its capabilities. It explains why this was the chosen program to implement some of the stochastic optimization method, and also which of them were actually implemented. Finally, it briefly describes some more realistic optimization test cases, that were developed by using the MDO tool.

In Chapter 5, the first of two test cases is presented. This one concerns the optimization of the operation conditions that maximize the range of a certain aircraft during cruise. Because surrogate models were used, several optimizations were performed in order to analyze each method's performance in different situations.

In Chapter 6, a similar test case to that of Chapter 5 is presented. Once again surrogates are used and the objective is to maximize a quantity similar to the range. In this test case, not only the different methods are compared, but also different configurations of winglets. While the main objective of this test case is still the assessment of the efficiency of each method, comparisons between stochastic and deterministic optimization are also made.

Chapter 7 summarizes all the important conclusions that were obtained throughout the thesis and presents some future research that should be done in order to improve this work.

Chapter 2

Uncertainty Quantification

As modern engineering systems increase in complexity, so does the necessity of using increasingly sophisticated multidisciplinary analysis and optimization tools that take into account uncertainty in their design. This is often achieved by formulating uncertainty in terms of probability theory and then using that information to compute the system response to it. In this chapter, a brief explanation on this subject is given and then some probability and statistical concepts are introduced.

2.1 Uncertainty Types and Quantification

Even though every real engineering system has associated to it a certain level of uncertainty, said uncertainty can arise from many different sources. It is known that these variations can come from anywhere, they are present during design, manufacturing and operations [Du and Chen, 2005]. More specifically, in an aeronautics structures case, there can be uncertainties in operations (aerodynamic loads, pay-loads, flight speed and altitude), uncertainties in material properties (the strength and modulus), uncertainties in manufacturing processes (component dimensions and shapes) and uncertainties in model structures and parameters [Yu and Du, 2006].

Although analysts may disagree on terminology, uncertainty can be divided in two main classes [Wojtkiewicz et al., 2001], namely epistemic and aleatory. Epistemic uncertainty is the one that originates from a lack of information about some aspect of the problem being considered. This type of uncertainty can also be called reducible, as it can be reduced if enough data to characterize said problem can be gathered. On the other hand, aleatory uncertainty, as the name clearly indicates, is the one that is associated with fluctuations that are part of the physical system being studied. It is also known as irreducible uncertainty, since it cannot be reduced no matter how much information about it is gathered.

For uncertainty to be used in design and optimization, it first needs to be quantified. Uncertainty quantification, as a process, is divided in three different key areas [Wojtkiewicz et al., 2001]. The first step consists of characterizing uncertainty in system parameters and the external environment. This involves the development of methods that can model both epistemic and aleatory uncertainty. Regardless of the type of uncertainty, characterization relies heavily on data gathered experimentally. Second comes the propagation phase which consists of using a certain method to propagate uncertainty through the system and then evaluating its response. Although there are some methods to do this, the most common and well established approach is to use probability and statistics to propagate uncertainty. Finally, it is necessary to both verify and analyze the computational models in order to incorporate their own uncertainty into the global uncertainty assessment.

2.2 Statistical Concepts

In order for the reader to better understand some of the methods that are introduced throughout this work, here follows a brief introduction to some important definitions of probability and statistical background.

2.2.1 Random Variables and Probability Density Functions

A random variable X is a variable that, instead of having a single fixed value, can take on a set of possible x values, each associated with a given probability. Random variables can either be discrete, i.e., they can assume any number from a determined set of numbers, or they can be continuous, i.e., they may assume any number inside a given interval.

The mathematical function describing the distribution of the possible values x of X and their respective probabilities, is called the Probability Density Function (PDF), $f_X(X)$. This function assigns a certain probability density to each value of the random variable, which means that the total probability of a variable X lying inside the interval $[x_1, x_2]$ is:

$$P(x_1 \leq X \leq x_2) = \int_{x_1}^{x_2} f_X(t)\, dt \tag{2.1}$$

where P stands for probability. If the interval of Eq.(2.1) is modified to $[-\infty, x]$, one obtains the cumulative density function (CDF), denoted by $\Phi(x)$, and which, as its name indicates, yields the sum of all possible value's probabilities up until x.

Any PDF satisfies the two following conditions:

$$f_X(X) \geq 0 \text{ for all values of } X \text{ and} \tag{2.2}$$

$$\int_{-\infty}^{+\infty} f_X(t)\, dt = 1 \tag{2.3}$$

and two important properties of the CDF are:

$$P(-\infty < t < +\infty) = 1 \tag{2.4}$$

$$P(t = a) = 0 \ \forall a \in \mathbb{R} \tag{2.5}$$

It is important to note that both Eq.(2.4) and 2.3 are the same property.

There are some commonly used distributions in engineering, like the Laplace distribution or the Log Normal distribution, though the one that is going to be used throughout this work is the Normal distribution, also known as Gaussian distribution and characterized by its density function:

$$f_X(X) = \frac{1}{\sigma_X\sqrt{2\pi}} \exp\left(\frac{1}{2}\left(\frac{X-\mu_X}{\sigma_X}\right)^2\right) \tag{2.6}$$

where μ_x and σ_X are respectively the mean and the standard deviation of the random variable X.

2.2.2 Expected Value, Variance and Higher Order Moments

The mathematical expectation, $E(X)$ or mean value μ_X of a random variable, defines the center of its distribution and is given by:

$$E(X) = \mu_X = \int_{-\infty}^{+\infty} t\, f_X(t)\, dt \tag{2.7}$$

for a continuous random variable.

The variance or second central moment is a measure of dispersion of a distribution and is denoted by:

$$\begin{aligned} V(X) = \sigma_X^2 &= E[(X-\mu_X)]^2 \\ &= E(X^2) - E(X)^2 \\ &= \int_{-\infty}^{+\infty} (t-\mu_X)^2\, f_X(t)\, dt. \end{aligned} \tag{2.8}$$

The positive square root of the variance is called the standard deviation of X (σ_X).

As for higher order moments, for any positive integer n, X's n^{th} order central moment is defined by:

$$E[(X-\mu_X)]^n = \int_{-\infty}^{+\infty} (t-\mu_X)^n\, f_X(t)\, dt. \tag{2.9}$$

Even though the most used statistical parameters are the mean and standard deviation, there are some other central moments that are worth mentioning, such as the Skewness (third central moment $S(X)$) and the Kurtosis (fourth central moment $K(X)$). The Skewness is a measure of the symmetry of the distribution (a normal distribution has a Skewness of zero since it is symmetric about its mean). On the other hand, Kurtosis can be thought of as the degree of ”peakedness” of the probability distribution.

It is also important to note that for sampling methods, the discrete definitions for both expected value and variance are used and they assume the form of:

$$E(X) = \mu_X = \frac{1}{n}\sum_{i=1}^{n} t_i \tag{2.10}$$

$$V(X) = \sigma_X^2 = E[(X-\mu_X)^2] = \frac{1}{n}\sum_{i=1}^{n}(t_i-\mu_X)^2 \tag{2.11}$$

where n is the number of samples taken.

2.2.3 Multivariate Distributions

In order to expand on the concepts first introduced in Eqs.(2.7) to (2.9), for the case of multiple variables, one has to take in consideration their joint probability function $f_{X_1,...,X_N}(X_1,...,X_N)$, where N is the number of random variables. For independent random variables, this function is defined as the product of each individual PDF as in:

$$f_{X_1,X_2,...,X_N}(X_1,X_2,...,X_N) = f_{X_1}(X_1)f_{X_2}(X_2)...f_{X_N}(X_N). \tag{2.12}$$

Thus as in Eq.(2.1), the probability that the random variables X_1 to X_N are respectively within the intervals $[x_{1_a}, x_{1_b}]$ and $[x_{N_a}, x_{N_b}]$ is:

$$P(x_{1_a} \leq X_1 \leq x_{1_b}, ..., x_{N_a} \leq X_N \leq x_{N_b}) = \int_{x_{1_a}}^{x_{1_b}} ... \int_{x_{N_a}}^{x_{N_b}} f_{X_1,...,X_N}(t_1,...,t_N)\, dt_1,...,dt_N \tag{2.13}$$

If $Z = g(X_1,...,X_N)$ is another random variable, its mean is:

$$E(Z) = E[g(X_1,...,X_N)] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} g(t_1,...,t_n)\, f_{X_1,...,X_N}(t_1,...,t_N)\, dt_1,...,dt_N \tag{2.14}$$

While a generic formulation for the multivariate central moments is given by:

$$M^n_{m_1,...,m_N} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f_{X_1,...,X_N}(t_1,...,t_N) \prod_{i=1}^{N} (x_i - \mu_{x_i})_i^m dt_1,...,dt_N \tag{2.15}$$

2.2.4 Normal Distribution and Sigma Levels

Finally, the last concept that is important to retain has to do with reliability targets. While optimizing with uncertainty, sometimes one has to make sure that the probability of violating the constraints lies within certain prescribed values. Assuming that both objective and constraint functions have normal distributions, these probabilities are associated with different Sigma levels (or standard deviations σ). This behavior is better understood by looking at Fig.2.1 and Tab.2.1, where for example, 99.73% of the values drawn from the normal distribution are within $\pm 3\sigma$ from the mean value. By taking advantage of this, it is possible to achieve certain probabilities of failure by making sure that a design lies in a region that is characterized by a certain sigma level.

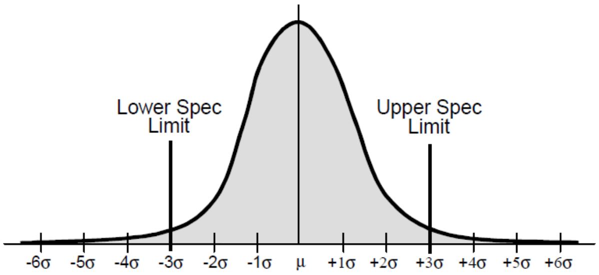


Figure 2.1: Normal distribution, 3σ design [Koch et al., 2004]

Sigma Level	Percent variation
$\pm 1\sigma$	68.26
$\pm 2\sigma$	95.46
$\pm 3\sigma$	99.73
$\pm 4\sigma$	99.9937
$\pm 5\sigma$	99.999943
$\pm 6\sigma$	99.9999998

Table 2.1: Sigma level as percent variation

Chapter 3

Robust and Reliable Design

Accounting for uncertainty in design optimization implies solving a slightly modified version of the deterministic optimization problem. These modifications are made according to the method that is being employed. Even though nowadays there are many different methodologies available to choose from, they can be categorized as belonging to one of two main groups, which are Robust Design Optimization (RDO) and Reliability Based Design Optimization (RBDO). The main difference between these two groups are their area of interest of the response distribution function [Allen and Maute, 2004].

RDO focus on optimizing the mean response of a system while minimizing its standard deviation, i.e. its sensitivity to fluctuation of input variables, thus requiring stochastic analysis tools able to appropriately approximate the influence of stochastic variations about the mean of the system. Tools such as Monte Carlo method (MC), Taylor Based Method of Moments (MM) or Sigma Point Methos (SP) are well suited to be used in conjunction with RDO.

On the other hand, RBDO is much more focused on the constraints of the problem, as instead of reducing the system's response variation, it tries to shift the response distribution away from constraint boundaries [Koch et al., 2004], thus ensuring that the target reliability is met. This method requires tools to predict the likelihood of extreme events at the tails of the response distribution. Besides MC, methods such as First Order Reliability Methods (FORM) or Second Order Reliability Methods (SORM) are usually used in RBDO.

Finally, it is important to note that, in this work, two assumptions are made regarding random variables. The first one is that all random variables have a normal distribution. According to [Paiva, 2010], even though there may be variables that do not have this kind of distribution, it is fairly easy to transform them into normal ones using for example the Rosenblatt transformation. The second assumption is that every random variable is independent from each other, meaning that they are not correlated and that their covariance matrix is diagonal. It has also been stated by [Paiva, 2010] that it is possible to deal with this problem in case the variables are not independent, though the impact on the sequence of operations required is considerable.

3.1 Deterministic Optimization

In a deterministic design optimization, the designer seeks the optimum set of design variable values for which the objective function is the minimum and the deterministic constraints are satisfied [Agarwal, 2004]. A common way to formulate such a problem is [Marta, 2013]:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x) \\ \text{subject to :} \quad & g_i(x) \leq 0,\ i = 1, 2, ..., N_g \end{aligned}$$

where f is the objective function, x is the vector of design variables, which can or cannot be restricted to a certain interval by means of $x_k^{LB} \leq x_k \leq x_k^{UB}$, $k = 1, 2, ..., N_{DV}$ where LB and UB are the lower and upper bounds of the design space respectively.

3.2 Robust Design Optimization (RDO)

To perform RDO, at least some of the components of the design vector need to be considered as being stochastic, and to account for this, some modifications to the deterministic problem statement have to be made. The robust attribute of the design is achieved by simultaneously minimizing the variance (σ_f^2) and expected value (μ_f) of the objective function, while ensuring probabilistic satisfaction of the constraints. Probabilistic bounds can also be set for the independent variables [Padulo et al., 2008]. The final result is the following statement:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & F(\mu_f(x, r), \sigma_f(x, r)) \\ \text{subject to :} \quad & G_i(\mu_{g_i}(x, r), \sigma_{g_i}(x, r)) \leq 0,\ i = 1, 2, ..., N_g \\ & P(x_k^{LB} \leq x_k \leq x_k^{UB}) \geq P_{bounds},\ k = 1, 2, ..., N_{DV} \end{aligned} \tag{3.1}$$

where μ and σ represent the mean and standard deviation of either the objective function or the constraint functions (depending on their subscript), and r is a vector of parameters that may or may not be deterministic. The robust objective and constraints are now designated by capital letters (F and G respectively), since in RDO they depend on their mean and standard deviation, which in turn depend on the probabilistic distribution of the variables. In Eq.(3.1), P stands for the probability of the input variables residing within their bounds.

In RDO, since both objective and constraint functions are evaluated in terms of their respective probabilistic parameters μ and σ, it is now necessary to compute their values. These parameters, in turn, depend on the random variable's own μ and σ according to Eqs.(2.14) and (2.15). Since it is impractical to analiticaly solve the integrals in most cases, a numerical procedure is in order. In the next few sections, some of these numerical methods are explained.

3.2.1 Monte Carlo Method (MC)

Monte Carlo algorithms are known to generate pseudorandom numbers, thus taking their name from the city of Monte Carlo which has been famous for its casinos, where games are based on random results as well. As it has been stated before, MC simulations belong to the sampling category of uncertainty propagation methods. This means that to propagate uncertainty through the system, they use random samples. The idea of using it to compute mean values and standard deviations of functions can be better understood by taking a closer look at Fig.3.1.

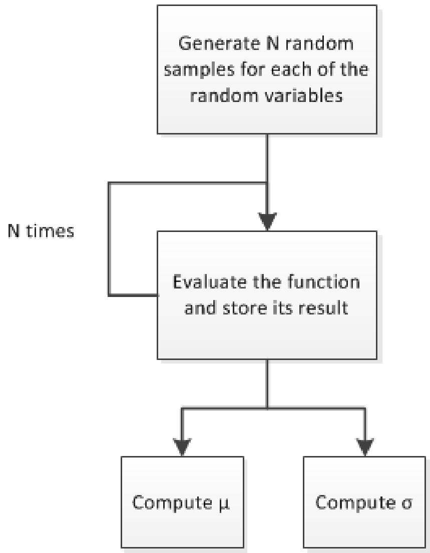


Figure 3.1: Representation of MC procedure in RDO

First, N random samples are generated and stored for each individual design variable and/or parameter, based on their own mean and standard deviation. Then, the function response is evaluated at each of the N stored points and its results are saved. Finally, resorting to Eqs.(2.10) and (2.11) both the expected value and the variance of the function can be computed. The method accuracy is tied to the number of samples N that are generated. The higher this number is, the better the results. Even though it is possible to generate a great amount a samples to get the best accuracy, the computational effort greatly escalates with the increase in function evaluations, meaning that the bigger the vector of stochastic design variables, the more costly it is to use a method such as MC in an optimization loop.

3.2.2 Taylor Based Method of Moments (MM)

The idea behind MM is to approximate the distribution of a given function $y = g(x)$ in terms of its derivatives by using Taylor approximations of the statistical moments [Menshikova, 2010]. By taking the

Taylor expansion of $g(x)$ about its mean μ_x:

$$\begin{aligned} g(x) &= g(\mu_x) + \sum_{i=1}^{n} \frac{\partial g}{\partial x_i}(x_i - \mu_i) + \\ &+ \frac{1}{2!}\sum_{i=1}^{n}\sum_{j=1}^{n} \frac{\partial^2 g}{\partial x_i \partial x_j}(x_i - \mu_i)(x_j - \mu_j) + \\ &+ \frac{1}{3!}\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} \frac{\partial^3 g}{\partial x_i \partial x_j \partial x_k}(x_i - \mu_i)(x_j - \mu_j)(x_k - \mu_k) + \\ &+ \ldots \end{aligned} \tag{3.2}$$

where all the function evaluations are made at μ_x. By applying the expectation operator to Eq.(3.2) and assuming that all vector x's entries are independent and have symmetric distributions, i.e.:

$$E(x_i - \mu_i) = 0 \tag{3.3}$$

$$E((x_i - \mu_i)(x_j - \mu_j)) = 0,\ i \neq j \tag{3.4}$$

$$E((x_i - \mu_i)(x_j - \mu_j)) = \sigma_{x_i}^2,\ i = j \tag{3.5}$$

$$E((x_i - \mu_i)(x_j - \mu_j)(x_k - \mu_k)) = 0 \tag{3.6}$$

the mean value of the function $g(x)$ becomes:

$$\mu_g = g(\mu_x) + \frac{1}{2!}\sum_{i=1}^{n} \frac{\partial^2 g}{\partial x_i^2}\sigma {x_i}^2 + \ldots \tag{3.7}$$

By squaring Eq.(3.7) and subtracting it from the squared Taylor approximation of the same function $g(x)$, thus satisfying the definition of variance first introduced in Eq.(2.8), the following is obtained:

$$\begin{aligned} \sigma_g^2 &= \sum_{i=1}^{N_{RV}} \left(\frac{\partial g}{\partial x_i}\right)^2 \sigma_{x_i}^2 + \\ &+ \sum_{i=1}^{N_{RV}} \left[\frac{\partial^3 g}{\partial x_i^3}\frac{\partial g}{\partial x_i}\frac{K(x_i)}{3} + \left(\frac{\partial^2 g}{\partial x_i^2}\right)^2 \frac{K(x_i) - 1}{4}\right]\sigma_{x_i}^4 + \\ &+ \sum_{i=1}^{N_{RV}}\sum_{j=1,\, i\neq j}^{N_{RV}} \left[\left(\frac{\partial^3 g}{\partial x_i^2 \partial x_j}\right)\left(\frac{\partial g}{\partial x_i}\right) + \frac{1}{2}\left(\frac{\partial^2 g}{\partial x_i \partial x_j}\right)^2\right]\sigma_{x_i}^2\sigma_{x_j}^2 + \\ &+ \ldots \end{aligned} \tag{3.8}$$

This method is classified according to the order of the truncated series, which is directly related to the number of derivatives used to calculate the mean and variance of the target function [Padulo et al., 2007]. By using higher order derivatives, the accuracy can be increased, especially for variables with high variance. Because complex algorithms are required to numerically compute high order derivatives of highly nonlinear functions, it is usual to only use the first order MM.

3.2.3 Sigma Point Method (SP)

This method is based on the idea that it is easier to approximate the probabilistic distribution of the input variables, rather than that of the target function [Padulo et al., 2007]. To do this, a Gaussian quadrature like method is employed, that computes both the mean and standard deviation by using appropriate weights and by evaluating the function at a sufficient number of key points (Sigma Points). Assuming both symmetric and independent input variables, the Sigma points are located symmetrically about the mean of each of the inputs depending on the input covariance matrix, as follows:

$$\chi_0 = \mu_x \tag{3.9}$$

$$\chi_{i+} = \mu_x + h\sigma e_i \tag{3.10}$$

$$\chi_{i-} = \mu_x - h\sigma e_i \tag{3.11}$$

where h is a suitable scalar, σ is the covariance matrix and e_i is the i^{th} column of the identity matrix of size $N_{RV} \times N_{RV}$. This method's weights are:

$$W_0 = \frac{h^2 - N_{RV}}{h^2} \tag{3.12}$$

$$W_i = \frac{1}{2h^2} \tag{3.13}$$

It is important to mention that W_i is the same for χ_{i+} and χ_{i-} and that $h = \sqrt{K(x)}$, which for normally distributed inputs equals $\sqrt{3}$. After having computed all the weights and points, the function evaluations are used to estimate the mean and variance of the function itself:

$$\widehat{\mu_f} = W_0 f(\chi_0) + \sum_{i=1}^{N_{RV}} W_i \left[f(\chi_{i+}) + f(\chi_{i-})\right] \tag{3.14}$$

$$\widehat{\sigma_f^2} = \frac{1}{2}\sum_{i=1}^{N_{RV}} \{W_i \left[f(\chi_{i+}) - f(\chi_{i-})\right]^2 + \left(W_i - 2W_i^2\right)\left[f(\chi_{i+}) + f(\chi_{i-}) - 2f(\chi_0)\right]^2\} \tag{3.15}$$

The SP method requires $2N_{RV} + 1$ function evaluations for each analysis, it does not require to compute derivatives in contrast to MM, but it can only approximate up to the second central moment, which is the variance.

3.3 Reliability Based Design Optimization (RBDO)

A typical RBDO formulation involves the minimization of an objective function subject to reliability constraints and deterministic constraints. Its equivalent to Eq.(3.1) can be mathematically represented by [Padmanabhan et al., 2006] [Frangopol and Maute, 2003]:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & f(x,r) \\ \text{subject to :} \quad & g_i^{rc}(x,r) \leq 0,\ i=1,2,...,N_{rc} \\ & g_i^{d}(x,r) \leq 0,\ j=1,2,...,N_{d} \\ & x_k^{LB} \leq x_k \leq x_k^{UB}\ k=1,2,...,N_{DV} \end{aligned} \tag{3.16}$$

where g_i^{rc} and g_i^d are respectively the reliability and deterministic constraints. Another way to formulate a RBDO problem is one in which the objective is to minimize the probability of the objective function to exceed/not exceed a predetermined value [Allen and Maute, 2004]:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & P(f(x,r) - target \geq 0)\ or\ P(target - f(x,r) \geq 0) \\ \text{subject to :} \quad & g_i^{rc}(x,r) \leq 0,\ i=1,2,...,N_{rc} \\ & g_i^{d}(x,r) \leq 0,\ j=1,2,...,N_{d} \\ & x_k^{LB} \leq x_k \leq x_k^{UB}\ k=1,2,...,N_{DV} \end{aligned} \tag{3.17}$$

The reliability constraints used in the two previous equations are defined as:

$$g_i^{rc} = P_{f_i} - P_{allow_i} = P(g_i(x,r) \geq 0) - P_{allow_i} \tag{3.18}$$

$$P(g_i(x,r) \geq 0) = \int_{g_i(x,r)\geq 0} p_{X,r}(t)dt \tag{3.19}$$

where P_{f_i} is the probability of failure, i.e., the probability of the originally deterministic constraint being greater than or equal to zero, P_{allow_i}, as the name indicates, is the allowable value for the probability of failure and $p_{X,r}$ is the joint probability density function. By ensuring that $g_i^{rc}(x,r) \leq 0$, these two formulations aim at a probability of failure less than or equal to a target value. One way to compute this probability is to solve the integral of Eq.(3.19) analytically, but since in most cases this is not feasible, numerical methods are employed. One type of methods that can be employed, as was the case in RDO, is MC simulations. It is fairly simple to implement and effective on problems that are highly non-linear with respect to the uncertainty parameters but, once again, it requires a high computational effort. To overcome the computational cost and still maintain good accuracy, first and second order reliability methods (FORM/SORM) can be used. These methods are able to compute the reliability indexes of the systems being studied, by solving a most probable point (MPP) problem. The main difference between them is the higher accuracy that SORM has when it comes to highly non linear systems, at the expense of requiring higher order information, which makes it more computationally expensive and thus less used compared to FORM. It is also possible to combine these two types of methods (MC and FORM/SORM) to create new and improved methodologies [Padmanabhan et al., 2006].

3.3.1 Monte Carlo Method (MC)

Much like in RDO, Monte Carlo algorithms can be used to generate random numbers with a certain distribution, in order to evaluate stochastic properties. In this case, instead of estimating the mean and standard deviation of the objective function and its constraints, the probability of them exceeding or not a predetermined value is evaluated. This process is exemplified in the diagram of Fig.3.2.

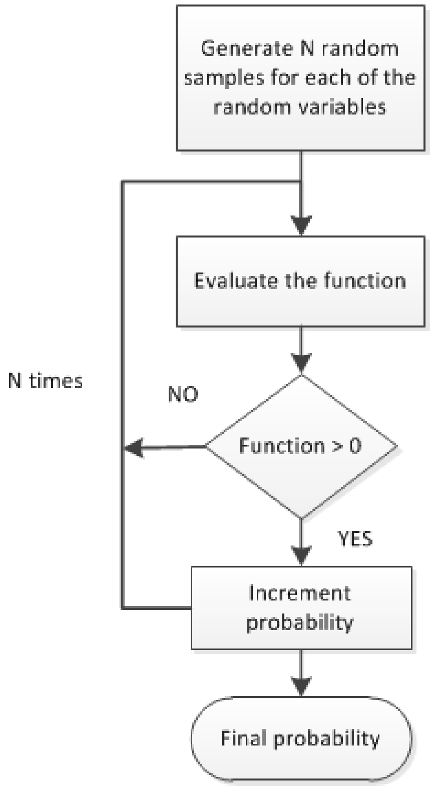


Figure 3.2: Representation of MC procedure in RBDO

As it can be seen, after generating the samples of the random variables, they are substituted into the function of interest to evaluate its response. The difference is that, instead of storing these values, now what is stored is the number of times the function is bigger than zero. After that, the probability of it being higher than zero is computed according to:

$$P(OZ) = \frac{N_{samples_{OZ}}}{N_{samples}} \tag{3.20}$$

where $N_{samples_{OZ}}$ is the number of samples for which the function was higher than zero and $N_{samples}$ is the total number of samples generated through the Monte Carlo Method. It is important to note that in the diagram the computed probability refers to the function being greater than zero but that this is not necessarily the only probability that can be computed.

3.3.2 First Order Reliability Method (FORM)

FORM basically consists of linearly approximating the limit state surface $g(h) = 0$, where h is now a vector of random variables (which encompasses uncertainty in both design variables and parameters), by means of a first order Taylor expansion at the Most Probable Point (MPP) of failure (this is the point where $g(h)$ has the highest probability of being zero). After that, the correspondent probability of failure can be approximated by:

$$P_{f_i} = P(g(h) \geq 0) = \Phi(-\beta) = 1 - \Phi(\beta) \tag{3.21}$$

where Φ is the cumulative distribution function of the standard normal distribution and β is the so called reliability index. The reliability index is the distance between the MPP of failure and the origin of the standard normal space u, and is given by $\beta = (u^T u)^{\frac{1}{2}}$. It can be found by solving the following optimization sub-problem:

$$\begin{aligned} &\underset{u}{\text{minimize}} && (u^T u)^{\frac{1}{2}} \\ &\text{subject to :} && g(h(u)) = 0 \end{aligned} \tag{3.22}$$

where u is the vector of random variables h, transformed into the standard normal space, as illustrated in Fig.3.3, through:

$$u = T(h) \tag{3.23}$$

Since in the optimization sub-problem of Eq.(3.22), what is being used is h and not u, the transformation that is necessary is the inverse of the one showed in Eq.(3.23). The transformation depends on the type of distribution considered for the random variables, though since only normal distributions are used throughout this work, only that one is presented:

$$h_k = T^{-1}(u_k) = \mu + \sigma u_k \tag{3.24}$$

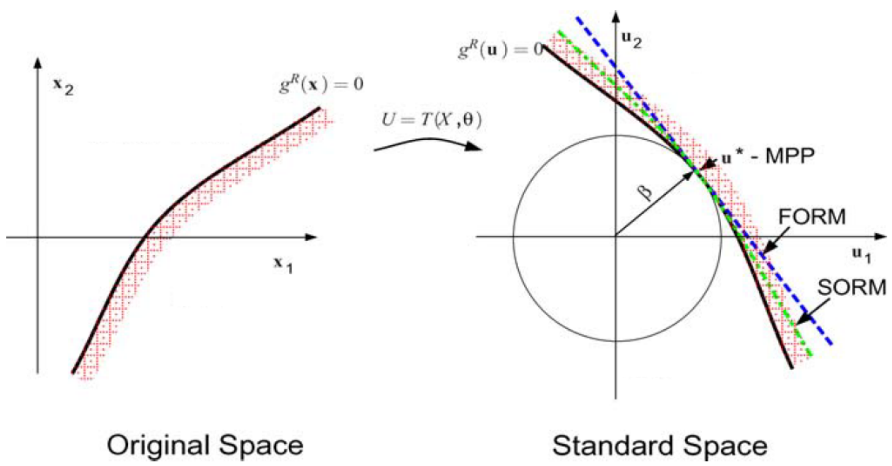


Figure 3.3: Original to standard normal space transformation and MPP determination [Padmanabhan et al., 2006]

At this point, it is preferable to formulate the constraints (and also the objective function if formulation 3.17 is being used) in terms of their reliability indexes instead of their probabilities. The mathematical expression for the reliability constraints should now be transformed into the equivalent:

$$g_i^{rc} = \beta_{reqd} - \beta_i \tag{3.25}$$

where β_{reqd} is the required reliability index (that corresponds to a given P_{allow}) and β_i is the reliability index of the current iterate. This approach to the RBDO problem is called the Reliability Index Approach (RIA). The problem with RIA is that it cannot always guarantee a solution for the MPP problem. This happens because the RIA numerical robustness is affected when either failure does not occur at all for a particular set of values of the design variables, or when the limit state surface is too far from the origin. To solve this problem, RIA's formulation can be slightly changed, which originates a new method called Performance Measure Approach (PMA), also known as the inverse MPP problem. By changing the optimization sub-problem of Eq.(3.22) to its inverse [Tu et al., 1999]:

$$\begin{aligned} &\underset{u}{\text{minimize}} && -g(h(u)) \\ &\text{subject to :} && (u^T u)^{\frac{1}{2}} - \beta_{reqd} = 0 \end{aligned} \tag{3.26}$$

one obtains the PMA. Solving RBDO by the PMA formulation is usually more efficient and robust than the RIA formulation where the reliability is evaluated directly. The efficiency lies in the fact that the search for the MPP of an inverse reliability problem is easir to solve than the search for the MPP corresponding to an actual reliability [Agarwal, 2004]. It is important to note that this method immediately returns the reliability constraint's value $g_i^{rc} = g^*(h(u))$ (the $*$ superscript means that it is evaluated at the MPP). Since PMA is more efficient at evaluating the inactive probabilistic constraints and RIA is more efficient at evaluating the violated probabilistic constraints, numerical robustness when solving RBDO problems can be achieved if the right method is used for each situation.

Another way to formulate the PMA problem is to confine the values of the vector u to a hyper-spherical surface of radius β_{reqd}, thus eliminating the necessity for the equality constraint $(u^T u)^{\frac{1}{2}} - \beta_{reqd} = 0$ in Eq.(3.26) and reducing the dimension of the sub problem to $N_{RV} - 1$. This results in the following statement:

$$\underset{u}{\text{minimize}} \quad -g(h(u(\phi))) \tag{3.27}$$

where ϕ is the set of hyper-spherical angular coordinates $\phi = \{\phi_1, \phi_2, ..., \phi_{N_{RV}-1}\}$. This formulation requires another coordinate transformation defined as:

$$\begin{aligned} u_1 &= \beta_{reqd} cos(\phi_1) \\ &\vdots \\ u_{N_{RV}-1} &= \beta_{reqd} \prod_{i=1}^{N_{RV}-2} sin(\phi_i) cos(\phi_{N_{RV}-1}) \\ u_{N_{RV}} &= \beta_{reqd} \prod_{i=1}^{N_{RV}-1} sin(\phi_i) \end{aligned} \tag{3.28}$$

Considering the lack of a constraint, plus the lower problem dimension, the alternative formulation of the PMA ought to allow for faster convergence. Considering the high number of function evaluations associated with the reliability assessment of each constraint that characterize these double-loop approaches, this property can turn out to be really important. In section 3.5.3, tests are performed on a numerical test case and the results are compared to those of the classical PMA.

3.3.3 Improved RBDO methods

Even though the methods described in section 3.3.2 are much more efficient than MC when it comes to assess the probability of failure at any given design point, they still require a high number of function evaluation, which makes practically impossible for them to be used to solve complex MDO problems. They belong to a class of methods called double-loop methods, since they have an outer loop that minimizes the objective function while several inner loops conduct the reliability analysis (one for each reliability constraint). This behavior can be seen in Fig.3.4.

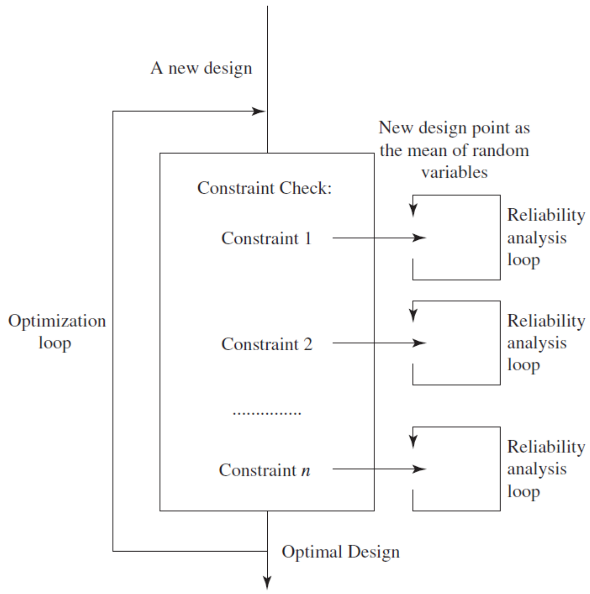


Figure 3.4: Double-loop method for RBDO [Shan and Wang, 2008]

The fact that, at each design point iteration, all the inner loops must be called, makes the number of function evaluations increase too much. To deal with this, two new types of approach have been proposed. In the first approach, the two optimization loops are decoupled into sequential deterministic optimization and reliability analysis [Du and Chen, 2002]. While in the second, all the reliability constraints are converted into deterministic ones and then only a single optimization problem is solved using these same constraints. Both these methods drastically reduce the number of required function evaluations, while still maintaining good accuracy.

Sequential Optimization and Reliability Assessment (SORA)

As the name indicates, this method is part of the first type of methods that were proposed to increase the efficiency of RBDO. The goal in SORA is to use serial single loops to efficiently optimize the objective function and assess its reliability. The difference between this and other similar methods is the way it uses inverse MPP to establish deterministic constraints that are equivalent to the probabilistic ones.

In order to better understand the principle behind SORA, one first needs to understand the relation between both the deterministic and probabilistic constraints. SORA is based on a formulation similar to that of the PMA method just described:

$$\begin{aligned} &\underset{\mu_x}{\text{minimize}} && f(\mu_x) \\ &\text{subject to :} && g_i^{rc}(x_{MPP}, p_{MPP}) \geq 0 \qquad i = 1, 2, ..., N_{rc} \end{aligned} \tag{3.29}$$

where $g_i^{rc}(x_{MPP}, p_{MPP})$ is the result of the inverse MPP problem corresponding to the desired reliability, as was described in subsection 3.3.2, x are random variables and p are random parameters. It must be noted that in this formulation, failure occurs for $g \leq 0$, unlike what has been done in this work up until this point. The relation between the two types of constraints can be further explained by considering Fig.3.5.

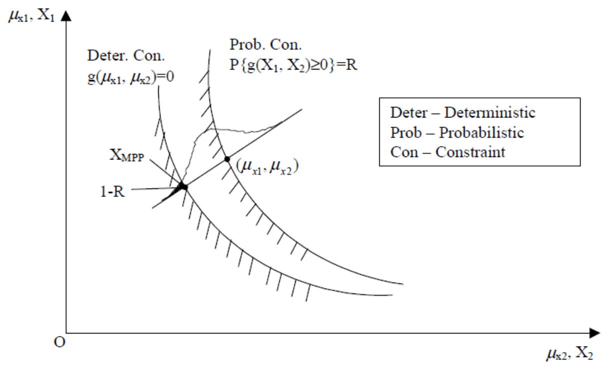


Figure 3.5: Deterministic and Probabilistic Constraints [Du and Chen, 2002]

As can be seen in this example comprising two random variables (X_1 and X_2), the feasible region of a probabilistic design is a reduced region of the deterministic feasible region and evaluation of the probabilistic constraint at the mean values (μ_{x_1}, μ_{x_2}) is equivalent to evaluating the deterministic constraint at the MPP, which is what $g^{rc}(x_{MPP}, p_{MPP})$ really does. Also, the MPP that corresponds to a certain design point on the probabilistic constraint, lies exactly on the boundary of the deterministic constraint, meaning that when $g^{rc}(x_{MPP}, p_{MPP}) = 0$ the shaded area near the tail of the distribution, corresponds exactly to the maximum probability of failure. Because of this, in order to maintain feasibility, the MPP must always lie within the deterministic feasible region. With this in mind, what SORA aims for is to rapidly approximate the original deterministic constraint of the problem to the probabilistic one by using a shifting vector $s = \mu_x - x_{MPP}$. This vector is created during each main cycle and uses the information of the MPP gathered in the previous loop. In the end, since the constraints are evaluated at the mean values of the random variables, the objective is to shift the deterministic constraint all the way to the place where the probabilistic constraint lies, so that when it becomes active, the target reliability is met.

This is accomplished by an algorithm with a flowchart like the one in Fig.3.6. Since during the first cycle there is yet no information about the MPPs, they are initiated as μ_p^0 and μ_x^0. After this step, SORA computes the shifting vector which is zero during the first cycle for the MPPs are initiated as the initial values. This means that the deterministic constraint remains exactly the same as the original problem constraint during the first cycle:

$$\begin{aligned} &\underset{\mu_x}{\text{minimize}} && f(\mu_x) \\ &\text{subject to}: && g_i^{rc}(\mu_x, \mu_p) \geq 0 \qquad i = 1, 2, ..., N_{rc} \end{aligned} \tag{3.30}$$

After the deterministic optimization concludes, some of the constraints might have become active like it is shown in Fig.3.7.

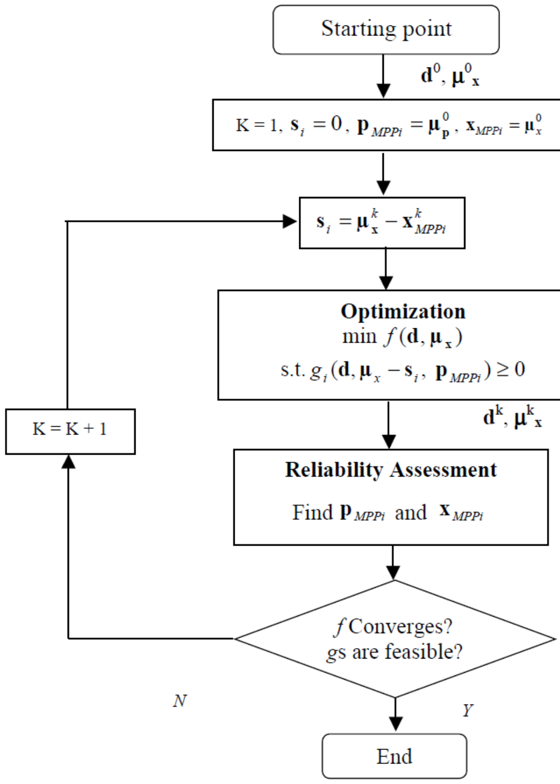


Figure 3.6: Flowchart of the SORA method [Du and Chen, 2002]

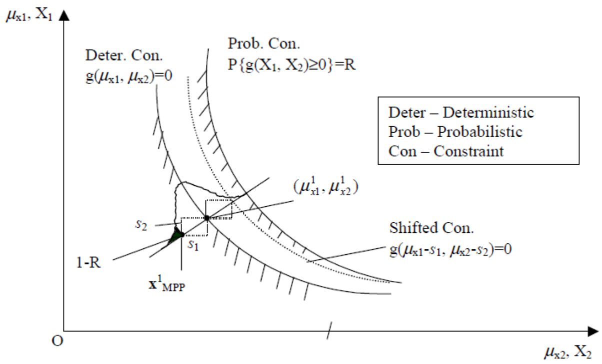


Figure 3.7: Shifting constraint boundary [Du and Chen, 2002]

Because of this, and considering that the random variables have a normal distribution, it means that the actual reliability is probably around 0.5 (which in most cases is way lower than the target reliability). The next step is to assess the reliability of the optimum design point just found. As expected, the MPP now lies out of the deterministic feasible region, which means that the design is unfeasible. Therefore, when establishing the constraint for the next outer loop, the algorithm tries to make it so that the the MPP may lie in the deterministic boundary. This is achieved by shifting the previous constraint using the shifting vector s, that in this specific case of two random variables, is formulated as $s = (s_1, s_2) = \mu_x^1 - x_{MPP} = (\mu_{x_1}^1 - x_{1MPP}^1, \mu_{x_2}^1 - x_{2MPP}^1)$. In the second cycle, because the shifting vector will not be zero anymore, the deterministic optimization stated in Eq.(3.29) becomes:

$$\begin{aligned} &\underset{\mu_x}{\text{minimize}} && f(\mu_x) \\ &\text{subject to}: && g_i^{rc}(\mu_x - s, p_{MPP}) \geq 0 \qquad i = 1, 2, ..., N_{rc} \end{aligned} \tag{3.31}$$

This way, the reliability of those violated probabilistic constraints will greatly improve with each passing cycle. The outer loop is repeated and the deterministic constraints are updated with each new MPP, until all probabilistic constraints are satisfied. Deterministic variables are treated as special random variables with no variance and no shifting distance. As for parameters, since they cannot be controlled during the optimization process, but still need to have their MPP on the deterministic feasible region, the algorithm uses p_{MPP} found in the previous cycle to update the deterministic constraint of the current cycle as in Eq.(3.31). It is important to note that since each probabilistic constraint has its own MPP, it also has its own shifting vector. To further increase the efficiency of the algorithm, both the starting point for the deterministic optimization and MPP search are the optimum found in the previous cycle.

In terms of stopping criteria, SORA stops when first, the difference in the objective function between two consecutive cycles is small enough and second, all the reliability requirements are satisfied.

To conclude, SORA is a simple to implement method, that totally decouples the reliability assessment from the optimization loop and that, by using a shifting mechanism based on the MPP, drastically reduces the number of reliability analysis required to find a feasible design point.

Reliable Design Space (RDS)

A different approach from the two previous methods introduced (double-loop and decoupled loops) is the RDS approach [Shan and Wang, 2008]. To put it simple, what this method aims for is to convert the original deterministic constraint into a probabilistic one. This way, only one single optimization loop needs to be done to reach the solution, as illustrated in Fig.3.8. Therefore, the number of required function evaluations is expected to drastically reduce, thus making this the most efficient method in theory.

By using the Lagrange multiplier method, the reliability index β can be approximated by [Shan and Wang, 2008]

$$\beta \approx \frac{-\sum_k u_k^*(\partial g_i/\partial u_k)_*}{\sqrt{\sum_k (\partial g_i/\partial u_k)_*^2}} \tag{3.32}$$

where the derivatives $(\partial g_i/\partial u_k)_*$ are evaluated at the MPP as the $*$ indicates. Since it was also found

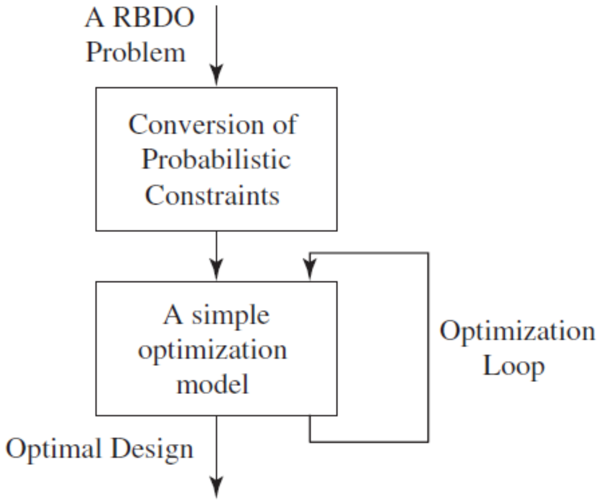


Figure 3.8: Single-loop method [Shan and Wang, 2008]

that each component k of the MPP on the failure surface in the standard normal space becomes:

$$u_k^* = -\frac{(\partial g_i/\partial u_k)_*}{\sqrt{\sum_k (\partial g_i/\partial u_k)_*^2}}\beta \tag{3.33}$$

and taking into account both the transformation first introduced in Eq.(3.24) and that:

$$\frac{\partial g_i}{\partial u_k} = \frac{\partial g_i}{\partial x_k}\frac{\partial x_k}{\partial u_k} = \sigma_{u_k}\frac{\partial g_i}{\partial x_k} \tag{3.34}$$

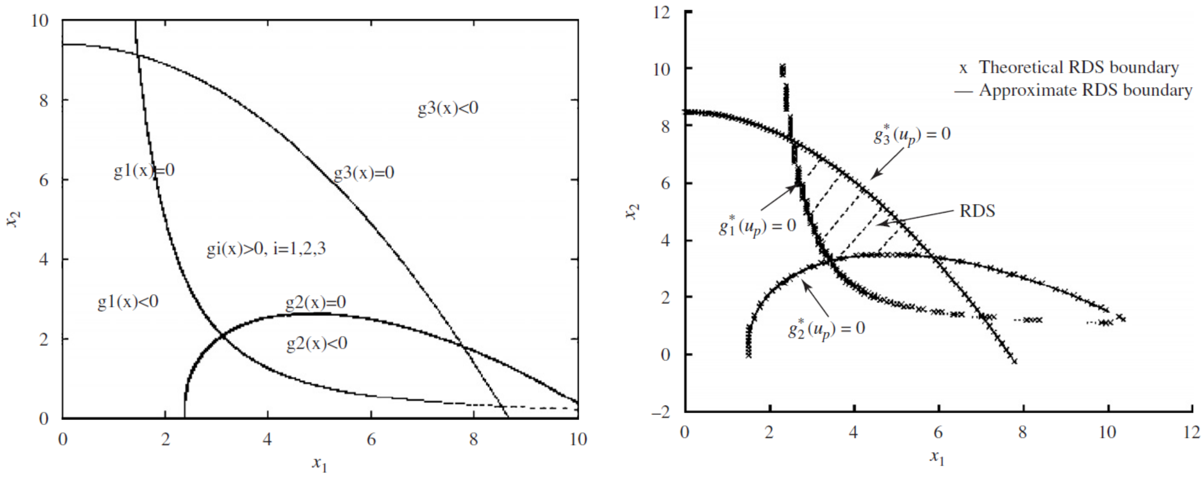
it is possible to transform Eq.(3.33) back into the original design space:

$$x_k^* = \mu_{x_k} - \beta\sigma_{u_k}^2 \frac{(\partial g_i/\partial u_k)_*}{\sqrt{\sum_k (\partial g_i/\partial u_k)_*^2}} \tag{3.35}$$

where x_k^* denotes the point in the original design space that corresponds to the MPP in the standard normal space. What sets the RDS method apart from others is the way it calculates the MPP without needing to solve an optimization subproblem like in double-loop or in decoupled methods. Instead of solving the inverse MPP problem that was first introduced in section 3.3.2, RDS approximates the direction cosine part $\sigma_{u_k}\frac{(\partial g_i/\partial u_k)_*}{\sqrt{\sum_k (\partial g_i/\partial u_k)_*^2}}$ of Eq.(3.35) by evaluating it at the corresponding mean point μ_{u_k}, which originates the following equation:

$$x_k^* \approx \mu_{x_k} - \beta\sigma_{u_k}^2 \frac{\partial g_i/\partial u_k}{\sqrt{\sum_k (\partial g_i/\partial u_k)^2}} \tag{3.36}$$

This way, it is now possible to directly calculate the inverse MPP x_k^* at any design point μ_{x_k}, without the need for an inner loop. We already know from SORA that in order for the design to be feasible, one has to make sure the MPP lies within the deterministic region. And since it is now possible to directly compute the MPP through Eq.(3.36), based only on the values of μ_x and μ_p, it is just a matter of rewriting the deterministic constraint as a function of the calculated MPP values, thus converting them into the theoretical probabilistic constraints g_i^* as can be seen in Fig.3.9. By forcing the MPP to lie within the



(a) Deterministic design space (b) Probabilistic design space or reliability design space

Figure 3.9: Design Spaces [Shan and Wang, 2008]

deterministic feasible region like in SORA, the reliability design space within which the corresponding design points should lie is being created.

The fact that this method completely eliminates the need for inner loops to assess the reliability for each constraint greatly improves its overall efficiency. Nevertheless, it is important to note that to implement this method, partial derivatives are required, which may not always be possible to obtain. In cases where analytic methods cannot be used to find these derivatives, the whole efficiency of the method may depend on the chosen numerical approach.

3.4 Robust and Reliability Based Design Optimization (R^{2}BDO)

A new stochastic optimization formulation called R^{2}BDO was proposed by [Paiva, 2010] to overcome some of the RDO and RBDO shortcomings. It is known that sometimes the RBDO formulation has problems with its objective function, if the initial guess is too far from a feasible design. As for RDO, some problems may arise when trying to use appropriate weights to mimic the behavior of probabilistic constraints.

In an attempt to bring together the best of both formulations, R^{2}BDO comprises RDO objective function treatment and RBDO constraint treatment in a single problem statement, resulting in:

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & F(\mu_f(x,r), \sigma_f(x,r)) \\ \text{subject to :} \quad & g_i^{rc}(x,r) \leq 0, \ i = 1,2,...,N_{rc} \\ & g_i^{d}(x,r) \leq 0, \ j = 1,2,...,N_d \\ & x_k^{LB} \leq x_k \leq x_k^{UB} \ k = 1,2,...,N_{DV} \end{aligned} \tag{3.37}$$

This formulation is tested and its results are compared and discussed in section 3.5.4.

3.5 Numerical Test Case - Rosenbrock Function

In order to both validate and understand the differences between the previously introduced methods, they have been tested with the Rosenbrock function and their results compared to each other.

The Rosenbrock function is known to be used to test the performance of optimization algorithms. Its minimum lies in a long, narrow, parabolic shaped valley, which is trivial to find, but its global minimum, not so much.

3.5.1 Deterministic Solution

The first test case consists of finding the minimum of the Rosenbrock function within the unit circle [Paiva, 2010]. The problem can be posed as:

$$\begin{aligned} &\underset{x}{\text{minimize}} && f(x) = 100(x_2 - x_1^2) + q(1 - x_1)^2 \\ &\text{subject to :} && g(x) = x_1^2 + x_2^2 - 1 \leq 0 \end{aligned} \tag{3.38}$$

where the parameter $q = 1$ was chosen. The results for the deterministic optimization problem are $x_1 = 0.7864$ and $x_2 = 0.6177$, and the function takes on the value of $f = 0.0457$. The solution to this problem has been achieved by using MATLAB®'s function fmincon with a SQP algorithm and starting point $x_0 = (0.1, 0.1)$.

3.5.2 RDO Solution

By doing the necessary modifications to the problem of Eq.(3.38), the new robust optimization problem looks like this:

$$\begin{aligned} &\underset{x}{\text{minimize}} && F(x, q) = \mu_f + \sigma_f \\ &\text{subject to :} && G(x) = \mu_g + 2\sigma_g \leq 0 \end{aligned} \tag{3.39}$$

where the f and g subscripts stand for the objective and constraint functions respectively. In this case, x_2 is a deterministic variable, x_1 is a random variable and q a random parameter. The objective of this formulation is to minimize both the mean value of the function and its standard deviation, while ensuring that approximately 97.73% of the times, the constraint is satisfied. This last part is accomplished by summing two standard deviations of the constraint to its mean (see Tab.2.1).

Throughout the tests, different combinations of uncertainty for x_1 and q are tried out. The parameter q is characterized by a mean $\mu_q = 1$ and different standard deviations ranging from 0.005 to 0.1. As for x_1, its statistical parameters are defined by means of the coefficient of variance ($c.o.v.$) , i.e., its standard deviation obeys $c.o.v. = \frac{\sigma}{|\mu|}$ while its mean is equal to the value of the iterate. For x_1 the value of $c.o.v.$ also varies from 0.005 to 0.1.

During these tests, three different methods are used to perform the optimization problem that has been established in Eq.(3.39). In a first stage, both MM and SP are used to optimize the objective function and to compute its stochastic parameters (both μ_f and σ_f). Afterwards, a post optimal analysis is conducted using MC simulations (with 6×10^6 samples) in order to assess the error of the stochastic parameters that these two methods calculated. In the second stage of this RDO test case, the MC method is used twice. Once to perform the optimization and to compute the stochastic parameters and once to assess the reliability of the final design point. This reliability assessment is tied to the probabilistic satisfaction of the constraint and is also done by using a post optimal analysis using MC simulations and 6×10^6 samples. It is important to note that these post optimal analysis are done at the optimal points that were found during the different optimization.

The results from the tests with MM and SP can be seen in Tab.3.1, while the ones with MC are in Tab.3.2.

		Method of Moments						Sigma Point					
$c.o.v.\ x_1$	σ_q	x_1	x_2	$\widehat{\mu_f}$	$\varepsilon_{\mu_f}[\%]$	$\widehat{\sigma_f}$	$\varepsilon_{\sigma_f}[\%]$	x_1	x_2	$\widehat{\mu_f}$	$\varepsilon_{\mu_f}[\%]$	$\widehat{\sigma_f}$	$\varepsilon_{\sigma_f}[\%]$
0.005	0.005	0.7831	0.6119	0.0472	-7.4	2.36E-04	-95.6	0.7813	0.6093	0.0517	$\lvert\varepsilon\rvert < 0.1$	5.31E-03	$\lvert\varepsilon\rvert < 0.5$
	0.01	0.7831	0.6119	0.0472	-7.4	4.73E-04	-91.2	0.7813	0.6093	0.0517	$\lvert\varepsilon\rvert < 0.1$	5.33E-03	$\lvert\varepsilon\rvert < 0.5$
	0.05	0.7831	0.6120	0.0472	-7.4	2.36E-03	-59.6	0.7813	0.6093	0.0517	$\lvert\varepsilon\rvert < 0.1$	5.82E-03	$\lvert\varepsilon\rvert < 0.5$
	0.1	0.7831	0.6120	0.0472	-7.4	4.71E-03	-34.1	0.7813	0.6093	0.0517	$\lvert\varepsilon\rvert < 0.1$	7.15E-03	$\lvert\varepsilon\rvert < 0.5$
0.01	0.005	0.7797	0.6065	0.0487	-23.4	2.43E-04	-98.8	0.7760	0.6013	0.0648	$\lvert\varepsilon\rvert < 0.1$	2.06E-02	$\lvert\varepsilon\rvert < 0.5$
	0.01	0.7797	0.6065	0.0487	-23.4	4.86E-04	-97.7	0.7760	0.6013	0.0648	$\lvert\varepsilon\rvert < 0.1$	2.06E-02	$\lvert\varepsilon\rvert < 0.5$
	0.05	0.7797	0.6065	0.0487	-23.4	2.43E-03	-88.5	0.7760	0.6013	0.0648	$\lvert\varepsilon\rvert < 0.1$	2.08E-02	$\lvert\varepsilon\rvert < 0.5$
	0.1	0.7796	0.6065	0.0487	-23.4	4.86E-03	-77.5	0.7760	0.6013	0.0648	$\lvert\varepsilon\rvert < 0.1$	2.12E-02	$\lvert\varepsilon\rvert < 0.5$
0.05	0.005	0.7530	0.5653	0.0613	-84.1	3.05E-04	-99.9	0.4777	0.2266	0.3260	$\lvert\varepsilon\rvert < 0.1$	7.55E-02	$\lvert\varepsilon\rvert < 1$
	0.01	0.7530	0.5653	0.0613	-84.1	6.10E-04	-99.9	0.4777	0.2266	0.3260	$\lvert\varepsilon\rvert < 0.1$	7.56E-02	$\lvert\varepsilon\rvert < 1$
	0.05	0.7530	0.5653	0.0613	-84.1	3.05E-03	-99.3	0.4798	0.2286	0.3247	$\lvert\varepsilon\rvert < 0.1$	7.79E-02	$\lvert\varepsilon\rvert < 1$
	0.1	0.7530	0.5653	0.0613	-84.1	6.10E-03	-98.7	0.4850	0.2338	0.3216	$\lvert\varepsilon\rvert < 0.1$	8.44E-02	$\lvert\varepsilon\rvert < 1$
0.1	0.005	0.7221	0.5196	0.0776	-93.4	3.86E-04	-100.0	0.3291	0.1055	0.4998	$\lvert\varepsilon\rvert < 0.1$	7.16E-02	≈ -2
	0.01	0.7221	0.5196	0.0776	-93.4	7.72E-04	-100.0	0.3293	0.1056	0.4997	$\lvert\varepsilon\rvert < 0.1$	7.18E-02	≈ -2
	0.05	0.7221	0.5196	0.0776	-93.4	3.86E-03	-99.8	0.3326	0.1080	0.4971	$\lvert\varepsilon\rvert < 0.1$	7.76E-02	≈ -2
	0.1	0.7221	0.5196	0.0776	-93.4	7.72E-03	-99.5	0.3399	0.1134	0.4917	$\lvert\varepsilon\rvert < 0.1$	9.17E-02	≈ -2

Table 3.1: RDO Rosenbrock function - Method of Moments and Sigma Point Method

From Tab.3.1 it is perceptible that MM is very inaccurate even for low values of uncertainty. This is shown by the big errors obtained with the post optimal analysis, especially the ones concerning the standard deviation of the objective function. As for SP, which takes roughly the same function evaluations to converge as MM, by looking at the same table, it is possible to see that it is a really accurate method at predicting the desired statistical measures. For all input uncertainties, the errors obtained are really low, increasing only slightly as uncertainty increases. As expected, especially since such discrepancies in terms of errors exist, the values for which the two design variables converged during the optimization are different for the two methods. It can also be observed that the higher the uncertainty, the bigger the difference.

The results from the second stage of the numerical test case can be seen in Tab.3.2. Since the SP method is so accurate when compared to the post optimal MC analysis, the values found in the four first columns of the MC table are similar to the ones found in the SP method's table. It is important to note that a small difference starts to show as the uncertainty increases though. As for the last column, it shows the reliability index obtained at the optimal point. While at low uncertainties RDO wields βs of about 1.99, which roughly corresponds to a probability of failure of 2.33% (a bit higher than the the target 2.27%), as the uncertainty increases, so does the reliability index, ultimately reaching a point, so far away from the robust constraint that the probability of failure is nearly zero. This behavior is best shown in Fig.3.10, where it can be seen that the higher the uncertainty (either in x_1 or q) the further from the constraint the design point is.

$c.o.v.\ x_1$	σ_q	x_1	x_2	$\widehat{\mu_f}$	$\widehat{\sigma_f}$	β
0.005	0.005	0.7830	0.6121	0.0510	0.0054	≈ 1.99
0.005	0.01	0.7830	0.6121	0.0510	0.0054	≈ 1.99
0.005	0.05	0.7830	0.6121	0.0510	0.0059	≈ 1.99
0.005	0.1	0.7830	0.6121	0.0510	0.0071	≈ 1.99
0.01	0.005	0.7794	0.6067	0.06354	0.0210	≈ 1.99
0.01	0.01	0.7795	0.6067	0.0636	0.0211	≈ 1.99
0.01	0.05	0.7792	0.6069	0.0636	0.0213	≈ 1.99
0.01	0.1	0.7795	0.6067	0.0636	0.0216	≈ 1.99
0.05	0.005	0.4780	0.2267	0.3261	0.0763	∞
0.05	0.01	0.4749	0.2235	0.3278	0.0739	∞
0.05	0.05	0.4447	0.1943	0.3496	0.0592	∞
0.05	0.1	0.5403	0.2913	0.2974	0.1248	∞
0.1	0.005	0.3269	0.1031	0.5024	0.0705	∞
0.1	0.01	0.3258	0.1033	0.5024	0.0704	∞
0.1	0.05	0.3354	0.1115	0.4940	0.0827	∞
0.1	0.1	0.3337	0.1088	0.4962	0.0888	∞

Table 3.2: RDO Rosenbrock function - Monte Carlo Method

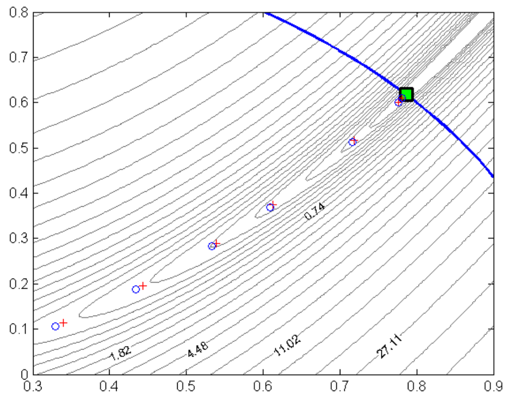


Figure 3.10: Problem solution of the RDO example for values of $c.o.v.$ of x_1 ranging from 0.005 to 0.1, with fixed $\sigma_q = 0.005$ (red '+' symbol) and fixed $\sigma_q = 0.1$ (blue 'o' symbol). Contour plot belongs to the deterministic objective function, blue line is the constraint and green square is the deterministic optimum

3.5.3 RBDO Solution

Similarly to RDO, a few changes were made to the deterministic problem to get the following RBDO problem statement:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & F(x) = f(\mu_x) \\ \text{subject to :} \quad & g^{rc} \leq 0 \end{aligned} \tag{3.40}$$

In this problem, the objective function is evaluated at the mean values of the random variables and parameters, which means that since $\mu_q = 1$, the parameter q does not have any real influence this time.

As for the reliability constraint, two approaches were used. In case of RIA, the reliability constraint takes on the form of $g^{rc} = \beta_{reqd} - \beta$, where β is computed by solving the minimization subproblem of Eq.(3.22) with $(u^T u)^{\frac{1}{2}}$ as the objective function and $x_1^2(u) + x_2^2 - 1 = 0$ as the constraint. When PMA is used instead, the constraint is directly obtained by solving the subproblem of Eq.(3.26), that has $-(x_1^2(u) + x_2^2 - 1)$ as the objective function and $\beta - \beta_{reqd} = 0$ as the constraint.

Because the parameter q does not have any influence in the final results, no different uncertainties were considered for it. Instead, different starting points and different required reliability indexes (β_{reqd}) were used to understand their effect on both the numerical robustness and accuracy of the methods, respectively.

The optimizations of this test case were conducted by using three different methods (RIA, PMA and MC). The results that were gathered comprise the optimal point (x_1 and x_2), the value of the objective function at that same point (F) and also the error between the required reliability index and the real one. Much like in the RDO test case, this error was computed using a post optimal analysis using MC simulations with 6×10^6 samples.

Results from Tab.3.3 can be used to draw several conclusions. First, it is possible to see that RIA was not able to find a feasible solution for the case where the initial design was too far from the constraint, which proves that RIA is not as numerically robust as PMA. Second, it can be seen that for the cases where both methods were able to find a solution, the results are really similar. Finally, both tables also show that despite the different levels of uncertainty and the required reliabilities, the error is always low (maximum of 0.2%).

		RIA				PMA			
x_0	$c.o.v.\ x_1$	x_1	x_2	F	$\varepsilon_\beta[\%]$	x_1	x_2	F	$\varepsilon_\beta[\%]$
$\beta_{reqd} = 2$									
	0.005	0.7829	0.6122	0.0472	$\|\varepsilon\| < 0.2$	0.7829	0.6122	0.0472	$\|\varepsilon\| < 0.2$
(0.1,0.1)	0.01	0.7794	0.6067	0.0487	$\|\varepsilon\| < 0.2$	0.7794	0.6067	0.0487	$\|\varepsilon\| < 0.2$
	0.05	0.7511	0.5634	0.0620	$\|\varepsilon\| < 0.2$	0.7511	0.5634	0.0620	$\|\varepsilon\| < 0.2$
	0.1	0.7159	0.5118	0.0808	$\|\varepsilon\| < 0.2$	0.7159	0.5118	0.0808	$\|\varepsilon\| < 0.2$
	0.005					0.7829	0.6122	0.0472	$\|\varepsilon\| < 0.2$
(-0.1,-0.1)	0.01		No solution found			0.7794	0.6067	0.0487	$\|\varepsilon\| < 0.2$
	0.05					0.7511	0.5634	0.0620	$\|\varepsilon\| < 0.2$
	0.1					0.7159	0.5118	0.0808	$\|\varepsilon\| < 0.2$
$\beta_{reqd} = 3$									
	0.005	0.7811	0.6094	0.0480	$\|\varepsilon\| < 0.2$	0.7811	0.6094	0.0480	$\|\varepsilon\| < 0.2$
(0.1,0.1)	0.01	0.7758	0.6012	0.0503	$\|\varepsilon\| < 0.2$	0.7758	0.6012	0.0503	$\|\varepsilon\| < 0.2$
	0.05	0.7335	0.5372	0.0711	$\|\varepsilon\| < 0.2$	0.7335	0.5372	0.0711	$\|\varepsilon\| < 0.2$
	0.1	0.6815	0.4637	0.1015	$\|\varepsilon\| < 0.2$	0.6815	0.4637	0.1015	$\|\varepsilon\| < 0.2$
	0.005					0.7811	0.6094	0.0480	$\|\varepsilon\| < 0.2$
(-0.1,-0.1)	0.01		No solution found			0.7758	0.6012	0.0503	$\|\varepsilon\| < 0.2$
	0.05					0.7335	0.5372	0.0711	$\|\varepsilon\| < 0.2$
	0.1					0.6815	0.4637	0.1015	$\|\varepsilon\| < 0.2$

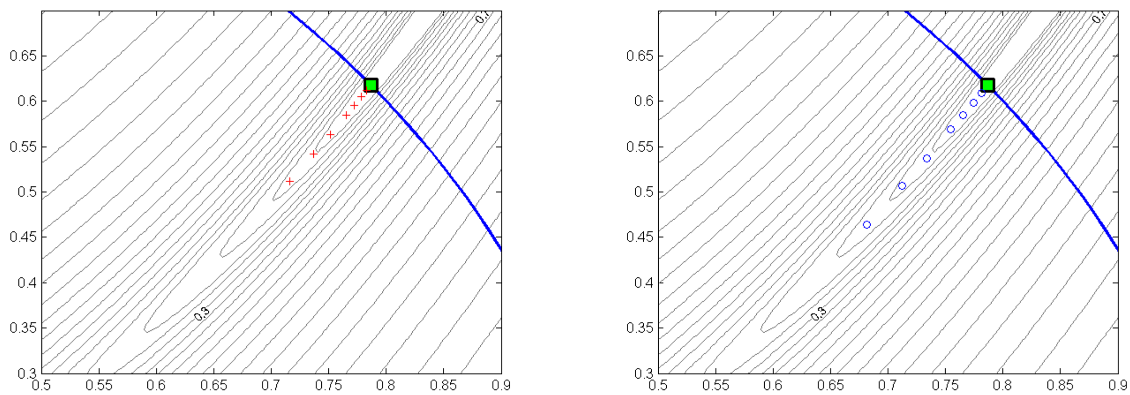
Table 3.3: RBDO Rosenbrock function - Reliability Index Approach and Performance Measure Approach

As in the RDO test case, the MC method, with once again 6×10^6 samples, is also used to perform the optimization. In this specific case, it was used to compute the probability of failure for every iteration, thus eliminating the need to have a reliability assessment subproblem. The results are presented in Tab.3.4 and can be used to validate the ones obtained with RIA and PMA. It can be seen that although the methodology with MC suffers from the same problems as RIA, since it cannot provide a solution when the starting point is $(-0.1, -0.1)$, the results seem to be in accordance with the ones that were previously obtained.

The behavior of the RBDO method with $\beta_{reqd} = 2$ and $\beta_{reqd} = 3$, as well as with different uncertainties, is presented in Figs.3.11(a) and 3.11(b) respectively. It is possible to see that as uncertainty grows, so does the distance between the optimal point and the deterministic constraint. This happens because the design is trying to keep the same reliability. The bigger the target reliability is, the more pronounced this behavior becomes. Unlike RDO, for this specific case, RBDO is able to properly approximate the probabilities of failure for every uncertainty level tested, which means that despite the uncertainty being higher in some cases, it can always achieve the target reliability.

x_0	$c.o.v.\ x_1$	x_1	x_2	F	$\varepsilon_\beta[\%]$
$\beta_{reqd} = 2$					
	0.005	0.7829	0.6122	0.0472	$\lvert\varepsilon\rvert < 0.2$
(0.1,0.1)	0.01	0.7794	0.6067	0.0487	$\lvert\varepsilon\rvert < 0.2$
	0.05	0.7511	0.5635	0.0620	$\lvert\varepsilon\rvert < 0.2$
	0.1	0.7160	0.5118	0.0807	$\lvert\varepsilon\rvert < 0.2$
	0.005				
(-0.1,-0.1)	0.01		No solution found		
	0.05				
	0.1				
$\beta_{reqd} = 3$					
	0.005	0.7811	0.6094	0.0480	$\lvert\varepsilon\rvert < 0.2$
(0.1,0.1)	0.01	0.7759	0.6012	0.0503	$\lvert\varepsilon\rvert < 0.2$
	0.05	0.7335	0.5372	0.0711	$\lvert\varepsilon\rvert < 0.2$
	0.1	0.6817	0.4639	0.1014	$\lvert\varepsilon\rvert < 0.2$
	0.005				
(-0.1,-0.1)	0.01		No solution found		
	0.05				
	0.1				

Table 3.4: RBDO Rosenbrock function - Monte Carlo Method



(a) $\beta_{reqd} = 2$ (b) $\beta_{reqd} = 3$

Figure 3.11: Problem solution of the RBDO example for values of $c.o.v.$ of x_1 ranging from 0.005 to 0.1. Contour plot belongs to the deterministic objective function, blue line is the constraint and green square is the deterministic optimum

Alternative PMA

In this particular section, both PMA and its alternative formulation are tested and their results are compared. To better understand the impact of the proposed altered formulation in terms of function evaluations, it was decided to increase the uncertainty in the constraint function. To do this, the RBDO constraint was changed to:

$$g(x) = x_1^2 + (x_2 - h)^2 - 1 \leq 0 \tag{3.41}$$

where once again x_1 and x_2 are the design variables and h is a newly introduced parameter. During these tests, two cases with different uncertainties were considered. First, one where x_2 is a deterministic variable, x_1 is a random variable and h is a random parameter, and second, one where both x_1 and x_2 are random variables and h is still a random parameter. It is important to note that in both cases, the mean value of h is constant $\mu_h = 0.5$, the target reliability is 3, and that in the second case, both x_1 and x_2 have the same $c.o.v.$.

To have a better idea where the optimum should lie, a deterministic optimization using the new constraint of Eq.(3.41) was solved. This optimization started from the point $x_0 = (0.1, 0.1)$ and only took 66 function evaluations to converge to the optimum $x = (0.9306, 0.8659)$.

After the stochastic optimizations had finished, the results were gathered and presented in Tab.3.5. It is important to note that, because all the optimum points found by the two methods were so similar, in each table, only one value for x_1, x_2, F and $\varepsilon_\beta[\%]$ is presented.

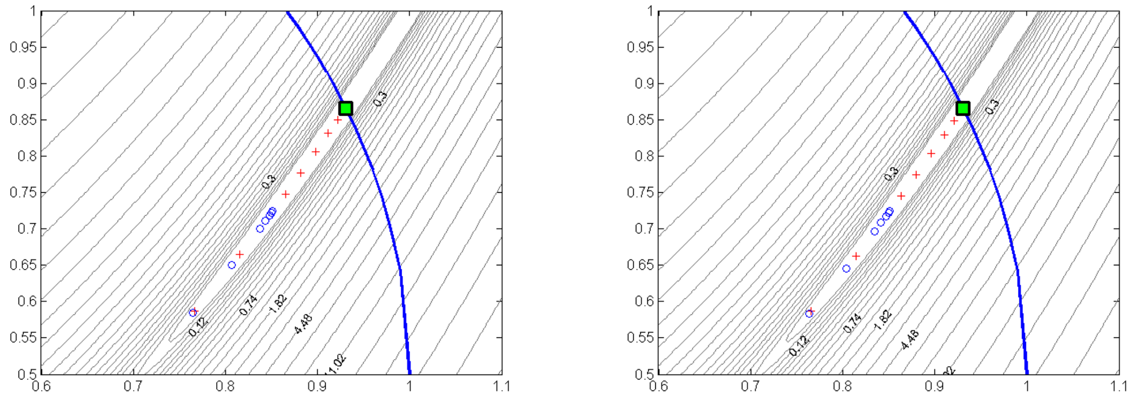
		$N_{RV}=2$						$N_{RV}=3$					
$c.o.v.\ x_1$	σ_h	x_1	x_2	F	$\varepsilon_\beta[\%]$	$\#eval$ PMA	$\#eval$ PMA_alt	x_1	x_2	F	$\varepsilon_\beta[\%]$	$\#eval$ PMA	$\#eval$ PMA_alt
0.005	0.005	0.9219	0.8497	0.0061	$\lvert\varepsilon\rvert < 0.2$	3212	590	0.9215	0.8489	0.0062	$\lvert\varepsilon\rvert < 0.1$	4365	975
	0.01	0.9201	0.8465	0.0064	$\lvert\varepsilon\rvert < 0.1$	3508	591	0.9197	0.8457	0.0064	$\lvert\varepsilon\rvert < 0.4$	4959	1406
	0.05	0.8931	0.7974	0.0114	$\lvert\varepsilon\rvert < 0.3$	2524	674	0.8930	0.7972	0.0115	$\lvert\varepsilon\rvert < 0.1$	4071	1254
	0.1	0.8512	0.7241	0.0222	$\lvert\varepsilon\rvert < 0.2$	4736	739	0.8512	0.7240	0.0222	$\lvert\varepsilon\rvert < 0.2$	3285	828
0.01	0.005	0.9141	0.8355	0.0074	$\lvert\varepsilon\rvert < 0.1$	3028	512	0.9133	0.8339	0.0075	$\lvert\varepsilon\rvert < 0.3$	3923	681
	0.01	0.9132	0.8338	0.0075	≈ -0.3	2960	592	0.9124	0.8322	0.0077	$\lvert\varepsilon\rvert < 0.3$	4292	696
	0.05	0.8913	0.7941	0.0118	≈ -0.3	2856	690	0.8908	0.7933	0.0119	≈ -0.4	4598	1465
	0.1	0.8506	0.7230	0.0224	$\lvert\varepsilon\rvert < 0.1$	10407	721	0.8503	0.7226	0.0224	$\lvert\varepsilon\rvert < 0.2$	7690	943
0.05	0.005	0.8483	0.7195	0.0230	$\lvert\varepsilon\rvert < 0.1$	2021	392	0.8467	0.7167	0.0235	≈ -0.5	4206	489
	0.01	0.8482	0.7193	0.0230	$\lvert\varepsilon\rvert < 0.1$	1986	428	0.8466	0.7165	0.0235	≈ -0.6	3201	534
	0.05	0.8450	0.7138	0.0240	≈ -1.1	1749	544	0.8430	0.7103	0.0247	≈ -1.6	3610	746
	0.1	0.8290	0.6868	0.0293	≈ -2.1	1807	704	0.8259	0.6816	0.0303	≈ -2.1	3252	809
0.1	0.005	0.7663	0.5871	0.0546	$\lvert\varepsilon\rvert < 0.1$	1934	454	0.7659	0.5864	0.0548	≈ -0.7	3253	210
	0.01	0.7663	0.5871	0.0546	$\lvert\varepsilon\rvert < 0.1$	1478	396	0.7659	0.5864	0.0548	≈ -0.9	3637	210
	0.05	0.7660	0.5866	0.0547	$\lvert\varepsilon\rvert < 0.1$	1395	395	0.7655	0.5859	0.0550	≈ -1.5	2737	613
	0.1	0.7647	0.5845	0.0554	≈ -2.6	1452	492	0.7637	0.5829	0.0559	≈ -3.8	3314	484

Table 3.5: RBDO Rosenbrock function ($N_{RV} = 2$ and $N_{RV} = 3$) - PMA vs alternative PMA

By looking at these tables, the first thing that can be noticed is the drastic reduction in function evaluations that is associated with the alternative PMA. Although it is not clear in the tables, as expected, these reductions of about 80% or higher, concern only the constraint function evaluations. The number of objective functions evaluations is roughly the same for both methods. Since the results that were obtained with these two methods are practically the same, this reduction in overall function evaluations, makes the alternative PMA formulation much more efficient when compared to the classic PMA. In terms of function evaluations, there is still another thing that is interesting, which is the fact that as uncertainty increases, the number of necessary function evaluations seem to decrease.

It is also possible to see from the tables, that adding another random variable (x_2), in this specific problem did not have a big influence on the optimized results. This behavior can be better seen in Figs.3.12(a) and 3.12(b), which are indeed really similar to each other, despite having been obtained from different cases.

Finally, it can be seen that as uncertainty increases, especially in σ_h, so does the reliability error. It is important to note that once again this error was computed with a post optimal analysis using MC method with 6×10^6 samples.



(a) $N_{RV} = 2$

(b) $N_{RV} = 3$

Figure 3.12: Problem solution of the RBDO example for values of $c.o.v.$ of x_1 ranging from 0.005 to 0.1, with fixed $\sigma_h = 0.005$ (red '+' symbol) and fixed $\sigma_h = 0.1$ (blue 'o' symbol). Contour plot belongs to the deterministic objective function, blue line is the constraint and green square is the deterministic optimum

3.5.4 R^{2}BDO Solution

The last test case that uses the Rosenbrock function consists of using both a R^{2}BDO formulation and an equivalent RDO formulation to solve the same problem and then compare their results. This way, it can be verified whether the hybrid formulation has any real benefits over the other two. The problem statement is as follows:

$$\begin{aligned} \underset{x}{\text{minimize}} \quad & F(x) = \mu_f + \sigma_f \\ \text{subject to}: \quad & g^{rc} \leq 0 \end{aligned} \tag{3.42}$$

where the objective is to minimize both the objective function and its standard deviation, while ensuring that the target reliability is met by the optimum design. It is important to note that the constraint function of this problem is the same as Eq.(3.41), with x_1 as a random variable, x_2 as a deterministic variable and h as a random parameter with $\mu_h = 1$.

To handle the computation of the mean and standard deviation of the objective function, both formulations used the SP method. This method was chosen because it proved to be more accurate than MM and less computationally expensive than the MC method. As for the way the constraints were handled, there were some significant differences between the two formulations.

In R^{2}BDO, the reliability constraint was handled by the alternative PMA, as among the methods that had been implemented so far, it was the one that proved to be the most robust and efficient. The chosen target reliability index was $\beta_{reqd} = 3$, which corresponds to a $P_f = 0.13\%$.

As for RDO, the target reliability is sought for by using an appropriate weight (K_σ) in the classic RDO constraint:

$$G(x) = \mu_g + K_\sigma \sigma_g \leq 0 \tag{3.43}$$

Just like in section 3.5.2, by replacing the weight by an appropriate scalar (in the RDO case this scalar was 2), the constraint is able to ensure that the optimum design has a certain probability of failure. In this case, to obtain a reliability index of 3, a weight of $K_\sigma = 3$ was also used, which according to Tab.2.1, corresponds to roughly the same probability of failure. For the same reasons that have been stated for the objective function, both the mean and standard deviation of the constraint were computed using the SP method.

Like in the previous tests, the optimization is performed for different combinations of uncertainty in variables and parameters alike. Both the $c.o.v.$ of the random variable x_1 and the standard deviation of the parameters q and h varied between 0.005 and 0.1. It is important to note that, in order not to create a big number of uncertainty combinations, it was decided that the standard deviation of both parameters should be equal between the two at all times. The results to this test case can be seen in Tabs.3.6 and 3.7. Once again the errors shown in the tables were computed with a post optimal analysis performed with MC at the optimal point, using 6×10^6 samples.

$c.o.v.\ x_1$	$\sigma_h\ and\ \sigma_q$	x_1	x_2	$\widehat{\mu_f}$	$\varepsilon_{\mu_f}[\%]$	$\widehat{\sigma_f}$	$\varepsilon_{\sigma_f}[\%]$	$\varepsilon_\beta[\%]$	$\#eval$
$\beta_{reqd} = 3$									
0.005	0.005	0.9219	0.8497	0.0133	$\|\varepsilon\| < 0.1$	0.0103	$\|\varepsilon\| < 0.1$	$\|\varepsilon\| < 0.2$	1205
0.005	0.01	0.9201	0.8464	0.0136	$\|\varepsilon\| < 0.1$	0.0102	$\|\varepsilon\| < 0.1$	$\|\varepsilon\| < 0.1$	1190
0.005	0.05	0.8932	0.7974	0.0178	$\|\varepsilon\| < 0.1$	0.0091	$\|\varepsilon\| < 0.1$	$\|\varepsilon\| < 0.3$	1142
0.005	0.1	0.8513	0.7240	0.0274	$\|\varepsilon\| < 0.1$	0.0078	$\|\varepsilon\| < 0.1$	$\|\varepsilon\| < 0.2$	1321
0.01	0.005	0.8720	0.7602	0.0396	$\|\varepsilon\| < 0.1$	0.0328	$\|\varepsilon\| < 0.1$	$\beta >> \beta_{reqd}$	1437
0.01	0.01	0.8720	0.7602	0.0396	$\|\varepsilon\| < 0.1$	0.0328	$\|\varepsilon\| < 0.1$	$\beta >> \beta_{reqd}$	12850
0.01	0.05	0.8721	0.7604	0.0396	$\|\varepsilon\| < 0.1$	0.0329	$\|\varepsilon\| < 0.1$	$\beta \approx 4.4 > \beta_{reqd}$	1354
0.01	0.1	0.8505	0.7231	0.0434	$\|\varepsilon\| < 0.1$	0.0298	$\|\varepsilon\| < 0.1$	$\|\varepsilon\| < 0.3$	1408
0.05	0.005	0.4777	0.2266	0.3260	$\|\varepsilon\| < 0.1$	0.0755	≈ -0.6	$\beta >> \beta_{reqd}$	746
0.05	0.01	0.4777	0.2266	0.3260	$\|\varepsilon\| < 0.1$	0.0756	≈ -0.6	$\beta >> \beta_{reqd}$	848
0.05	0.05	0.4798	0.2286	0.3247	$\|\varepsilon\| < 0.1$	0.0779	≈ -0.6	$\beta >> \beta_{reqd}$	861
0.05	0.1	0.4850	0.2338	0.3216	$\|\varepsilon\| < 0.1$	0.0844	≈ -0.5	$\beta >> \beta_{reqd}$	873
0.1	0.005	0.3291	0.1055	0.4998	$\|\varepsilon\| < 0.1$	0.0716	≈ -1.8	$\beta >> \beta_{reqd}$	725
0.1	0.01	0.3293	0.1056	0.4997	$\|\varepsilon\| < 0.1$	0.0718	≈ -1.8	$\beta >> \beta_{reqd}$	637
0.1	0.05	0.3326	0.1080	0.4971	$\|\varepsilon\| < 0.1$	0.0776	≈ -1.8	$\beta >> \beta_{reqd}$	716
0.1	0.1	0.3399	0.1134	0.4917	$\|\varepsilon\| < 0.1$	0.0917	≈ -1.7	$\beta >> \beta_{reqd}$	765

Table 3.6: R^2BDO Rosenbrock function - SP + alternative PMA

Tables (3.6) and (3.7) show that even though the results obtained with R^2BDO and RDO appear to be similar, some conclusions can still be drawn. The first thing that can be seen is the difference in the necessary function evaluations to converge. As expected, R^2BDO requires more function evaluations than RDO because it uses an optimization subproblem to assess reliability at each iteration. In terms of the mean and standard deviation of the objective function, SP once again proves to be an accurate method as the error obtained for these parameters are small.

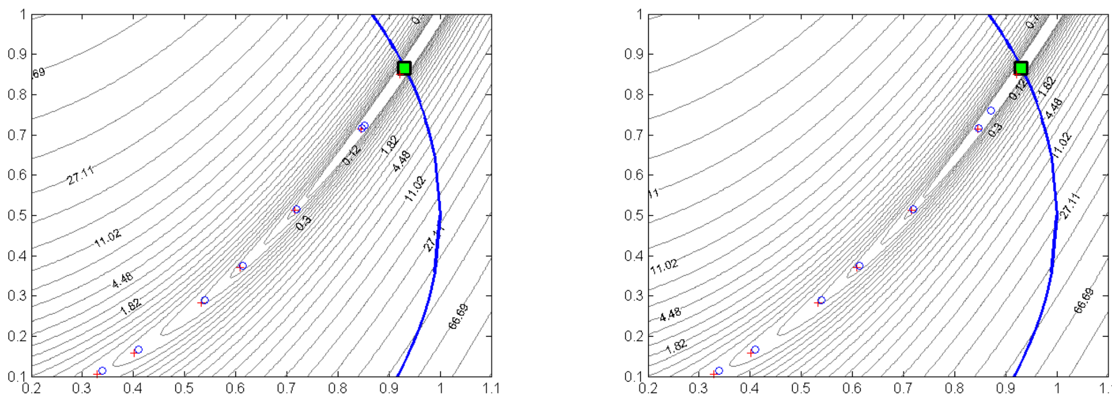
As for the reliability error, it can be seen that overall, R^2BDO has lower error when compared to RDO. For lower uncertainties R^2BDO obtains designs with a reliability that is close to the target one. On the other hand, RDO proves once again that, even for lower uncertainties and for this specific problem, it

$c.o.v.\ x_1$	$\sigma_h\ and\ \sigma_q$	x_1	x_2	$\widehat{\mu_f}$	$\varepsilon_{\mu_f}[\%]$	$\widehat{\sigma_f}$	$\varepsilon_{\sigma_f}[\%]$	$\varepsilon_\beta[\%]$	$\#eval$
$\beta_{reqd} = 3$									
0.005	0.005	0.9220	0.8498	0.0133	$\lvert\varepsilon\rvert < 0.1$	0.0103	$\lvert\varepsilon\rvert < 0.1$	≈ 0.6	840
0.005	0.01	0.9203	0.8466	0.0136	$\lvert\varepsilon\rvert < 0.1$	0.0102	$\lvert\varepsilon\rvert < 0.1$	≈ -1.3	870
0.005	0.05	0.8985	0.8070	0.0168	$\lvert\varepsilon\rvert < 0.1$	0.0093	$\lvert\varepsilon\rvert < 0.1$	≈ -13.6	830
0.005	0.1	0.8717	0.7593	0.0223	$\lvert\varepsilon\rvert < 0.1$	0.0084	$\lvert\varepsilon\rvert < 0.1$	≈ -23.4	770
0.01	0.005	0.8720	0.7602	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0328	$\lvert\varepsilon\rvert < 0.1$	$\beta >> \beta_{reqd}$	950
0.01	0.01	0.8720	0.7602	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0328	$\lvert\varepsilon\rvert < 0.1$	$\beta >> \beta_{reqd}$	840
0.01	0.05	0.8721	0.7604	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0329	$\lvert\varepsilon\rvert < 0.1$	$\beta \approx 4.6 > \beta_{reqd}$	830
0.01	0.1	0.8703	0.7572	0.0399	$\lvert\varepsilon\rvert < 0.1$	0.0326	$\lvert\varepsilon\rvert < 0.1$	≈ -22.6	890
0.05	0.005	0.4777	0.2266	0.3260	$\lvert\varepsilon\rvert < 0.1$	0.0755	≈ -0.6	$\beta >> \beta_{reqd}$	580
0.05	0.01	0.4777	0.2266	0.3260	$\lvert\varepsilon\rvert < 0.1$	0.0756	≈ -0.5	$\beta >> \beta_{reqd}$	620
0.05	0.05	0.4798	0.2286	0.3247	$\lvert\varepsilon\rvert < 0.1$	0.0779	≈ -0.5	$\beta >> \beta_{reqd}$	550
0.05	0.1	0.4850	0.2338	0.3216	$\lvert\varepsilon\rvert < 0.1$	0.0844	≈ -0.5	$\beta >> \beta_{reqd}$	550
0.1	0.005	0.3291	0.1055	0.4998	$\lvert\varepsilon\rvert < 0.1$	0.0716	≈ -1.8	$\beta >> \beta_{reqd}$	550
0.1	0.01	0.3293	0.1056	0.4997	$\lvert\varepsilon\rvert < 0.1$	0.0718	≈ -1.9	$\beta >> \beta_{reqd}$	510
0.1	0.05	0.3326	0.1080	0.4971	$\lvert\varepsilon\rvert < 0.1$	0.0776	≈ -1.8	$\beta >> \beta_{reqd}$	480
0.1	0.1	0.3399	0.1134	0.4917	$\lvert\varepsilon\rvert < 0.1$	0.0917	≈ -1.7	$\beta >> \beta_{reqd}$	480

Table 3.7: RDO Rosenbrock function - SP method

is not the best method when it comes to obtaining designs with a target reliability in mind. The higher the input uncertainty gets, the less the solution depends on the constraint behavior, since the optimum points move away from it. This means that for both formulations, the dominant factor becomes the RDO objective function, which in turn makes the designs more and more conservative. When this happens, both methods yield the same optimum results.

Once again, Figs.3.13(a) and 3.13(b) show the behavior of both the formulations graphically.



(a) R^2BDO (b) RDO

Figure 3.13: Problem solution for values of $c.o.v.$ of x_1 ranging from 0.005 to 0.1. Contour plot belongs to the deterministic objective function, blue line is the constraint and green square is the deterministic optimum

3.6 Numerical Test Case - Problem with 3 Constraints

The previous test cases used a rather difficult objective function (Rosenbrock function) and focused mainly on assessing how the accuracy of each method was affected by different uncertainty levels and target reliabilities. In this test case, the main goal is to introduce both SORA and RDS, which were not used in the previous test case, and to provide information about the efficiency of each method, in terms of required function evaluations. In this test case, instead of using the Rosenbrock function and one constraint to test these methods, a rather simple objective function is used in conjunction with three nonlinear constraints (the design space of this problem is that of Fig.3.9). This way, not only it is possible to see how the methods perform with a simpler function but also if they still remain accurate when more than one constraint exists. This problem is posed as [Shan and Wang, 2008]:

$$
\begin{aligned}
\underset{\mu_1,\mu_2}{\text{minimize}} \quad & f(\mu_1,\mu_2) = \mu_1 + \mu_2 \\
\text{subject to :} \quad & P(g_i(X) \geq 0) \geq R_i,\ i = 1,2,3 \\
& g_1(X) = X_1^2 X_2/20 - 1, \\
& g_2(X) = (X_1 + X_2 - 5)^2/30 + (X_1 - X_2 - 12)^2/120 - 1, \\
& g_3(X) = 80/(X_1^2 + 8X_2 + 5) - 1, \\
& 0 \leq \mu_j \leq 10,\ j = 1,2 \\
& \sigma_1 = \sigma_2 = 0.3\ \beta_i = 3\ i = 1,2,3
\end{aligned}
\tag{3.44}
$$

where μ_1, μ_2, σ_1 and σ_2 are the mean values and standard deviations of the two random design variables X_1 and X_2 respectively, and R_i is a the required reliability, which is the same for every constraint. Since the RBDO methods presented use reliability indexes instead of reliabilities, the constraints were adapted according to each method's own formulation. A target reliability index ($\beta_{reqd} = 3$) was chosen for this test case, as for the standard deviation of the random variables, a fixed value of 0.3 was chosen.

It is also important to note that, since the formulation of Eq.(3.44) concerns a RBDO problem, it had to be converted to be used with RDO and R^{2}BDO. To do this, the approaches were the same as the ones used in the test case of section 3.5.4. Also, the error is once more calculated by a post optimal analysis using MC with 6×10^6 samples.

	RBDO						RDO		R^{2}BDO
Method	RIA	PMA	PMA_alt	SORA	SORA_alt	RDS	MM	SP	SP + PMA_alt
Design Variables									
μ_1	3.4391	3.4391	3.4391	3.4391	3.4391	3.4406	3.6333	3.6291	3.4391
μ_2	3.2866	3.2866	3.2866	3.2866	3.2866	3.2800	3.4442	3.4164	3.2866
Objective function	6.7257	6.7257	6.7257	6.7257	6.7257	6.7205	7.5017	7.46987	7.1499
Constraint Reliability									
$\varepsilon_{\beta_1}[\%]$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	25.41	24.50	$\lvert\varepsilon\rvert < 2$
$\varepsilon_{\beta_2}[\%]$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	$\lvert\varepsilon\rvert < 2$	11.01	8.39	$\lvert\varepsilon\rvert < 2$
$\varepsilon_{\beta_3}[\%]$	∞	∞	∞	∞	∞	∞	∞	∞	∞
$\#Obj.\,func.\,eval$	18	18	21	51	51	18	15*	75**	105**
$\#Const.\,func.\,eval$	1137	1956	688	367	234	54*	45*	225**	688
$\#Total.\,func.\,eval$	1155	1974	709	418	285	72*	60*	300**	709**

$*$ the number of function evaluations required to determine partial derivatives, were not accounted for
$**$ the number of necessary function calls within the main function evaluations, were taken into account

Table 3.8: Comparison between different stochastic optimization methods

From Tab.3.8 a lot of conclusions can be drawn. First of all, in terms of the reliability error, it can be seen that, while the RDO methods once again struggled to achieve the target reliability, both the RBDO and R^{2}BDO were able to achieve it, apparently without any major problems.

In terms of the optimum points that were found, all RBDO and R^{2}BDO methods reached the same solutions, except for the RDS method. This solution was slightly different, but not enough to increase the reliability error of this method. Despite the fact that the R^2BDO method found the same solution as most of the other RBDO methods, the value of its objective function was different. This happens because in this method, the objective function is comprised of its mean value plus its standard deviation, instead of just it mean value like in the rest of the RBDO methods. As for the RDO methods, they reached different solutions when compared to each other, just like in the Rosenbrock test case. Both solutions were too conservative.

In terms of the number of required function evaluations, some trends can also be seen. Both RDO methods are among the ones that have the lowest number of function evaluations, since they do not have reliability assessment cycles. As for RBDO, the classic approaches PMA and RIA are the methods that have the highest number of evaluations. After them, comes the alternative PMA, that is indeed able to reduce the number of constraint evaluations. Both SORA and SORA_alt have even less function evaluations, and finally comes RDS. It can be seen that compared to the classic approached, both SORA, SORA_alt and RDS greatly reduce the number of required function evaluation, apparently at no cost, since the reliability errors remain low. These methods can even be compared to RDO in terms of function evaluation. As for the R^2BDO, it has the same constraint evaluations than PMA_alt (because it uses this strategy to handle the constraints) but has more objective evaluations because it uses the SP method to handle the objective function.

Chapter 4

MDO Framework and Test case definition

Even though commercial programs are typically robust and user friendly, their use in a context of MDO is unpractical for they take up too many computational resources, thus making the whole optimization process unacceptably slow. Of course it would be possible to use programs such as Matlab® in conjunction with a FE and CFD commercial codes to both analyze and optimize any aero-structural problem. The problem is that, the more complex the optimization became, the less effective this approach would be.

Because of this, it was decided to use the MDO Framework that is being developed at Instituto Superior Técnico [Suleman et al., 2014]. Not only has this tool been especially created for the preliminary analysis and design of novel configurations of aircraft, but it was also engineered to be as comprehensive as possible, while being versatile and user friendly. Its modular architecture allows the users to tailor the program to their needs, by creating new modules whenever necessary, as well as allowing the use of those that have already been created. This program, developed in C++, integrates different main air-

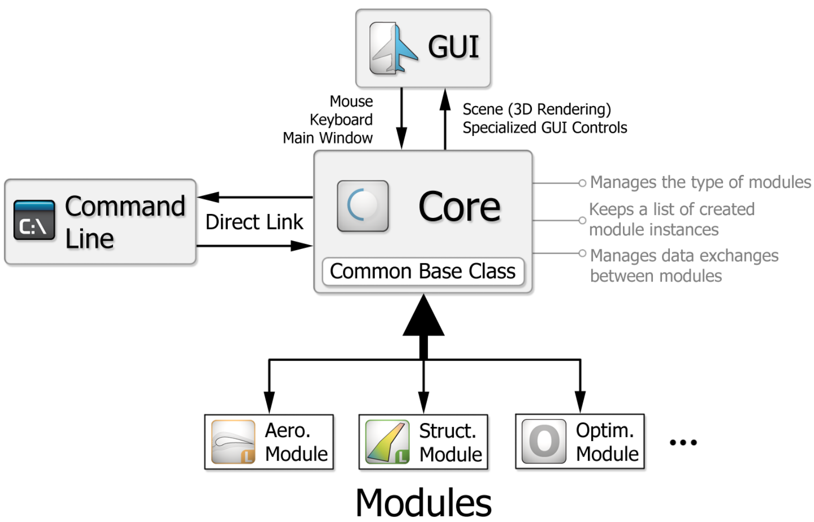


Figure 4.1: MDO Framework structure [Suleman et al., 2014]

craft disciplines (that are divided by modules) with optimization software, to efficiently solve performance optimization problems.

4.1 MDO Framework

As it has already been stated, the MDO framework has been implemented to work as a unified aircraft multidisciplinary optimization environment. With concepts like ease of use and versatility in mind, a modular architecture controlled by a program Core and a graphical user interface, was developed (as illustrated in Fig.4.1). The program Core manages all the data flow between the modules and the interface. It can either be accessed by command line, by the main graphical interface or by any other software capable of interfacing with Windows dynamic-link libraries (dll). Its main features are [Suleman et al., 2014]

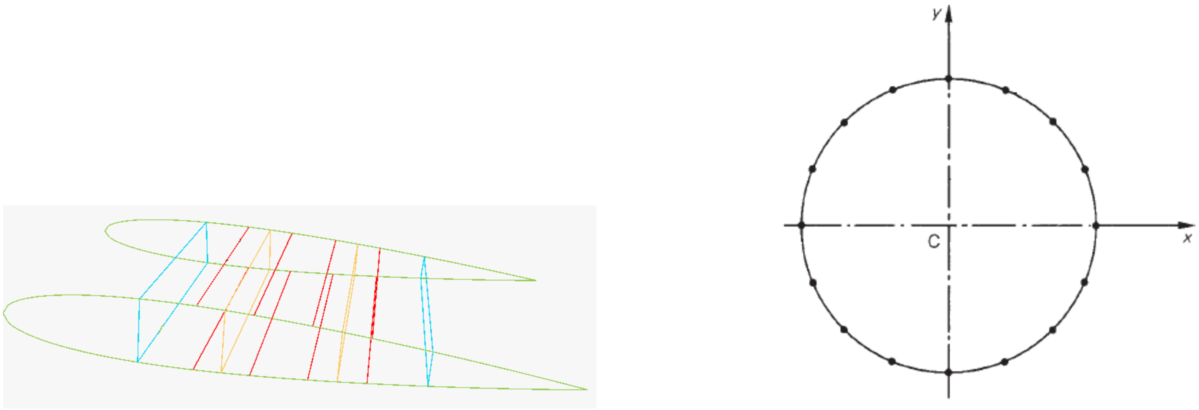
1. Manage the types of modules. Each module is contained in a separate dll file and is preloaded into the Core (e.e. the geometry module) or added by the user.
2. Manage a list of module instances. A Core can have several instances of a single module, that are kept in a list that can be saved into a project file.
3. Intermediates and encapsulates the data exchange between the module instances and the interface. The data exchange has to be managed due to the multi-threaded environment of the Program Core, where instances run in a separate CPU thread.

4.1.1 Modules

The MDO tool has already some pre-installed modules that are required to perform a complete analysis of the aircraft. Each module has a different function, and depending on the problem, they can be called in different order. The modules and their corresponding tasks are:

- Geometry Module - Definition of the geometric model of the aircraft;
- Payload Module - Evaluation of the distribution, mass, CG and moments of inertia of the payload;
- Propulsion Module - Choice of either a simplified turbofan or turbojet engine and computation of the propulsion forces, engine mass, CG, inertia and the fuel flow;
- Aerodynamics Module - Computation of both the aerodynamic forces and fuel distribution over the aircraft. Its code is based on an existing panel method code that accounts for friction and compressibility corrections. It also includes drag corrections and automatic mesh generation. This module was validated with commercial software [Paiva, 2010];
- Structures Module - Consists of an equivalent beam model, based on a 3D beam element (BEAM4 in ANSYS [ANSYS, Inc., 2010]) and it is used to compute the mass, CG, moments of inertia and stress distribution of the aircraft. The lift surfaces are modeled by wing-box cross sections with 10

stringers, while for the fuselage, frames with 10 booms are used. An example of these structures can be seen in Fig.4.2.



(a) Example of a wing with wingbox

(b) Example of an idealized section of a fuselage (frame)

Figure 4.2: Structural module

In the problems solved in this thesis, the order in which every task was performed is illustrated in Fig.4.3.

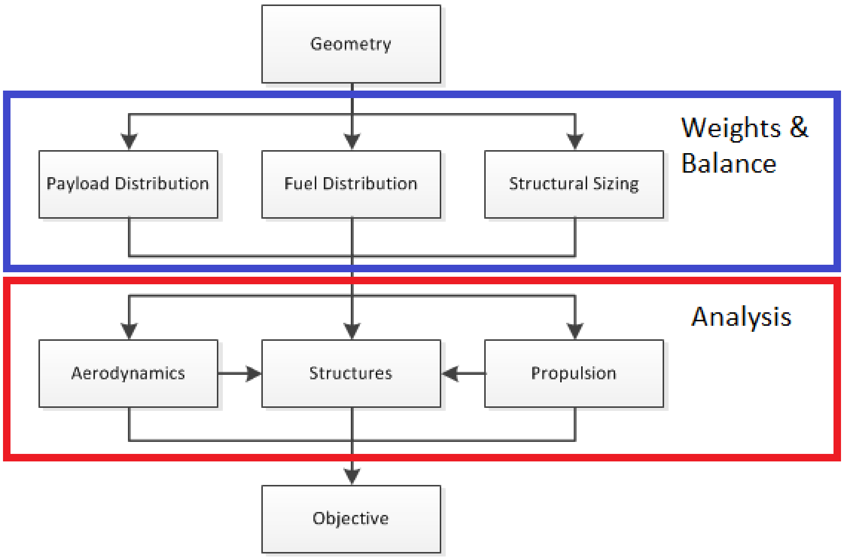


Figure 4.3: Task sequence of one analysis

It is important to note that the implemented MDO architecture has some simplifications. The way that these modules are linked to each other does not force convergence between the modules. For example, the structural displacement caused in the aerodynamic surfaces by the aerodynamic forces is not used to compute new aerodynamic forces, which in turn would generate a different structural displacement. Instead of iteratively computing these results until convergence is obtained, it is assumed that the differences between the results are so small, that only one analysis is required. This simplification is made to avoid making the use of the modules even more computationally expensive.

In addition to these preloaded modules, there is also the possibility to add new ones depending on the user's needs. In this case, since the existing optimization module was not able to cover stochastic optimization, a new module called R2BDO was developed to cater for this need.

Besides deterministic optimization, this module specially developed for this thesis includes the ability to perform robust optimization (SP Method), reliability based optimization (PMA as well as SORA) and robust and reliability based design optimization (SP+SORA). These methods were based on the conclusions of chapter 3. The SP method was chosen over MM because it proved to be much more accurate. As for the RBDO classic methods, since PMA is more numerically robust than RIA, it was the chosen method. In the improved RBDO methods class, the choice was down to SORA or RDS. Despite the potential shown by RDS, SORA was chosen because the problems that are being dealt with, from this section onward cannot be analytically differentiable. This means that to implement a method such as RDS, an additional numerical method such as finite differences would be required. As for R^2BDO, both SP+PMA and SP+SORA were implemented. The alternative approach to PMA and SORA introduced in chapter 3 was implemented in the R2BDO module, but ultimately, because of some numerical issues, it ended up not being used.

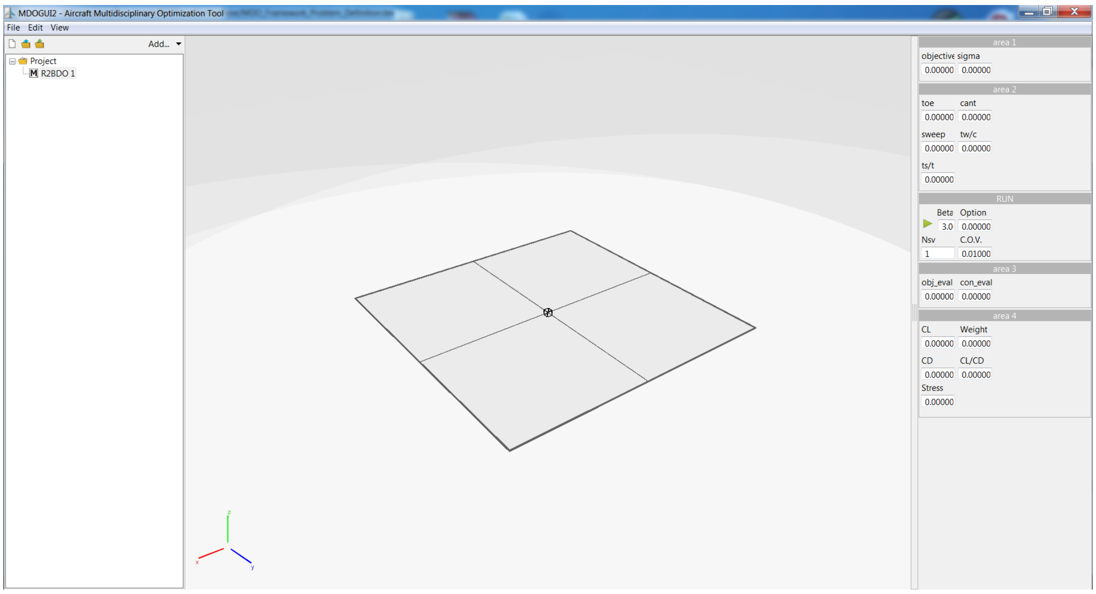


Figure 4.4: R2BDO module

Depending on the problem, the interface of the developed R2BDO module changes slightly. Figure 4.4 shows a screenshot of this module in the MDO Framework, for a certain problem. On top, it is possible to see the values of the objective function (its mean value and its standard deviation), just below are the values of the design variables, after that come the options of the stochastic optimization (consisting of the target reliability index, the option to choose the optimization method, the number of stochastic variables and their $c.o.v.$), down below is both the number of objective and constraint function evaluations and finally there can be seen some parameters of interest. The actual optimization problem is defined in the code, which means that in the MDO Framework the user only needs to change the options of the optimization. After running the program by clicking in the green arrow, the results start appearing in their respective boxes.

After programming the R2BDO module, in order for it to be validated, some of the test cases of chapter 3 were reproduced, and their results were compared to the ones previously obtained. The module not only proved to be a much faster alternative to Matlab®, but also that in terms of results, consistency was maintained despite a few minor discrepancies due to different algorithm options and tolerances.

4.1.2 Optimization Algorithms

In terms of available optimization algorithms, the MDO framework provides the users with three different choices.

- FSQP - Feasible Sequential Quadratic Programming;
- GCMMA - Globally Convergent Method of Moving Asymptotes;
- GA - Genetic Algorithm.

Sequential Quadratic Programming is a gradient-based iterative method for solving nonlinear optimization problems. The idea behind it consists of modeling the problem at the current point by a quadratic subproblem (QP) and use the solution of this subproblem to find the new point [Marta, 2013]. FSQP is a derivative of this method, which has the additional property of enforcing feasibility (i.e. constraint satisfaction) at each iteration. This behavior grants the algorithm a few advantages [Lawrence and Tits, 2001], namely: the QP problems are always consistent; the objective function may be used directly as a merit function in the line search; it has no problems when the objective function is undefined outside the feasible region; and, in case the optimization process is stopped after a few iterations, it always yields a feasible design point.

The GCMMA method is based on the conservative convex separable approximations (CCSA), for solving inequality constrained nonlinear programming problems. Each generated iteration point is a feasible solution (like in the FSQP) and the sequence of iteration points converges toward the set of Karush–Kuhn–Tucker points. The major advantage of this method is the fact that it can be applied to problems with a very large number of variables [Svanberg, 2002]. It is important to note that for both the FSQP and the GCMMA, the variables are scaled to vary between 0 and 1 for convergence purposes.

As for the included GA, it was developed exclusively for the framework. Unlike the two previous methods, GA is part of a class of optimization methods called heuristic, meaning that its algorithm mimics some kind of behavior found in nature. In the GA case, it mimics the principle of evolution of species through selection and mutation [Goldberg, 1989]. This algorithm also has the ability to solve non-linear inequality constrained problem by using a penalty method. In order to improve the convergence, Latin Hypercube Sampling (LHS) is used for the initial population. While GA has the advantages of being better at exploring the whole search space and also at optimizing certain cases where gradient-based methods fail, these come at the expense of a higher computational cost [Marta, 2013].

4.1.3 Surrogate Models

Running a simple analysis of the complete aircraft can already take some time by itself. For an optimization problem with a high number of variables, a large number of analysis might be required by the optimization algorithm. In addition to this,if stochastic optimization is implemented, it would only make the whole process even more costly. One way to reduce the computational costs of this kind of optimization is by using surrogate models, thus turning the problem into a Surrogate -Based Analysis and Optimization (SBAO) problem [Queipo et al., 2005]. These models mimic the behavior of simulation models as closely as possible while being computationally cheaper to evaluate. They provide a good solution for the preliminary design stage, even if at the expense of some loss of fidelity. The process of developing such models consists of three different steps:

- Sample selection;
- Surrogate model construction;
- Evaluation of the surrogate accuracy.

The first step in creating these models (Sample selection), consists of choosing the points (for the design variables) where the response of the simulations is to be evaluated. The locations of these points determines both the quality of the surrogate and how much computationally expensive it is to create them. Because of this, they must be chosen carefully. There are several methods to choose these points such as: MC sampling, Polynomial Chaos Expansion (PCE) sampling and LHC sampling [McKay et al., 1979] [Myers and Montgomery, 1995]. While being the simplest to implement, MC is the most computationally expensive of the three. PCE is the fastest in terms of convergence but it can sometimes generate particular forms of point replication [Suleman et al., 2014]. LHS is a space filling technique, that ensures that at least one sample lies in a particular region of the design space. This method developed in 1979 by McKay et al., while having some troubles with uniformity [Suleman et al., 2014], is still widely used for its simplicity and good overall performance. Because of these reasons, LHS was the chosen method in the MDO framework.

In terms of surrogate model construction, there are several available options as well, such as Quadratic Interpolation, Artificial Neural Networks or Kriging [Paiva, 2010]. For the MDO framework, Kriging was chosen because it demonstrates superior performance and stability when compared to the other two methods [Paiva, 2010]. Kriging is a statistical interpolation method, based on the assumption that the value of a given multivariate function, $y(x)$, can be decomposed into deterministic and stochastic components:

$$y(x) = f(x) + z(x) \tag{4.1}$$

where $f(x)$ is the deterministic model, $z(x)$ are the interpolated residuals and x is a location of interest, where no samples have been taken [Paiva, 2010]. The algorithm for the Kriging method used in the MDO framework is an adaptation to C++ of the code DACE [Lophaven et al., 2002].

To guarantee that both the sample selection and model construction were properly executed, the accuracy of the surrogate must be evaluated. This step is mandatory, as skipping it may lead to undesired results. To do this, some new sample points are chosen to run simulations. After having the results of these simulations, they are compared with the response of the surrogate models and the difference between exact simulation and approximation model defines the actual accuracy of the surrogate. If this is found to be unacceptable, additional points need to be used to build a new surrogate model.

4.2 Test Case Objectives

After the complete implementation of the R2BDO module, it was decided to apply it to some test cases. This was done with two objectives in mind: to assess the performance of stochastic optimization in an aircraft MDO environment; and to demonstrate the MDO Framework capabilities in optimizing novel aircraft configurations. A problem related to the NOVEMOR project was chosen not only influenced by the fact that it fits in the desired scope of problems, but also because both the EMB9MOR (aircraft developed by Embraer for the NOVEMOR project, that can be seen in Fig.4.5) models and some surrogates had already been created. This way, instead of focusing on modeling an aircraft and its surrogates, resources could be focused on achieving the two proposed objectives. NOVEMOR (Novel Air Vehicle Configurations From Fluttering Wings to Morphing Flight) [NOVEMOR, 2013] is a collaborative research project aimed at developing, implementing and assessing a range of numerical simulation technologies to accelerate future aircraft design. It is funded by the EU in the context of the 7th Framework Program. As for the aircraft in study, EMB9MOR is based on the EMBRAER 190 and is named after the project in which it is inserted. It has the purpose of providing comparable data to enable the investigation of morphing mechanisms and their implementation.

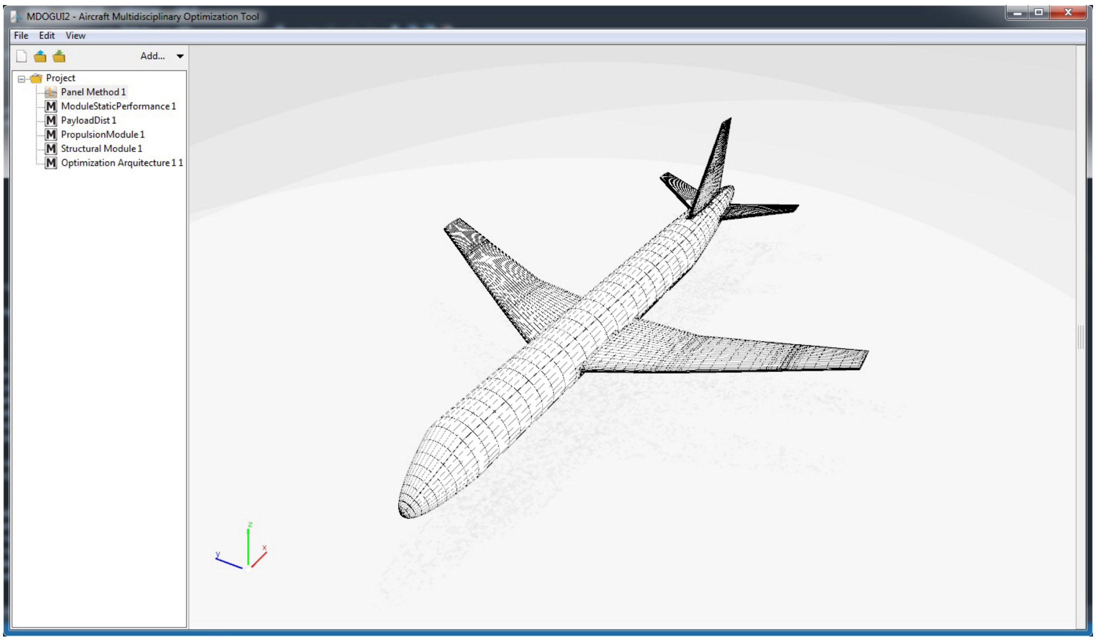


Figure 4.5: EMB9MOR - Baseline model

Two different test cases were devised to test different aspects about the implemented stochastic methods and the MDO Tool performance. These two cases are comprised of a single flight phase (cruise) and have a similar objective. Their main differences lie in the design variables that are used and in the constraints. They both use surrogate models, instead of the actual models, so that many optimizations with different uncertainty conditions and parameters can be performed at reasonable cost. These two test cases are fully described in Chapters 5 and 6.

Whenever the range is used in these test cases as the objective function, it is calculated with the Breguet range equation for thrust-rated aircraft. To obtain this equation, some assumptions must be made and the integral of Eq.(4.2) must be solved.

$$R = -\int_{W_1}^{W_2} \frac{V}{c_t D}\, dW \tag{4.2}$$

In the integral, R is the range, V is the airspeed, c_t is the specific fuel consumption, D is the drag force and W is the aircraft's total weight. By assuming constant specific fuel consumption, constant airspeed and constant angle of attack, we obtain from Eq.(4.2) the following:

$$R = \frac{V}{c_t}\frac{C_L}{C_D} ln \frac{W_1}{W_2} \tag{4.3}$$

where C_L and C_D are the coefficients of Lift and Drag, W_1 is the total initial weight of the aircraft when it starts the cruise flight phase and W_2 is the weight it has when it finishes the cruise phase. To simplify this problem, it was assumed that 100% of the fuel is available when the cruise phase starts, and 20% when it ends.

All the assumptions that were made lead to constant C_L and constant V. Because of this, as the aircraft gets lighter (due to fuel burn), the density is required to decrease, which leads to an increase in altitude. Because this happens, not only the drag is reduced but also the thrust. This way, everything is kept balanced, which leaves the throttle unchanged. This flight technique is called the "drift up" flight schedule.

Chapter 5

Test Case 1 - Baseline Surrogate Optimization

5.1 Problem Definition

This test case consists of maximizing the range of the baseline model of the aircraft EMB9MOR. Since it was decided to use the surrogates that had already been created for this model, the choice of both the design variables and constraints depended on them.

The surrogates were created with the computation of the aircraft performance in mind. A total of three surrogates were created: one that computes the aerodynamic forces and fuel mass distribution; one that computes the propulsion forces and engine mass distribution; and finally one that computes the payload and structural mass distribution.

By taking into account all the inputs and outputs of these surrogates, the following range maximization problem was devised:

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & Range \\ \text{subject to :} \quad & L = W - Tsin(\alpha) \\ & Tcos(\alpha) = D \\ & M \leq 0.8 \end{aligned} \tag{5.1}$$

where x are the design variables, Range is calculated with the Breguet Equation (4.3), L is the lift force, W is the total weight of the aircraft when it starts the cruise flight phase, T is the thrust generated by the aircraft engines, D is the drag force and M is the number of Mach at which the aircraft is flying.

It would have been interesting to have a constraint that evaluated the maximum stress on the aircraft as well, but since there was no available surrogate capable of calculating this output, it was not included.

Among all the inputs of the surrogates, some were chosen as design variables, while others remained fixed throughout the optimization. The design variables can be seen in Tab.5.1 and the fixed parameters can be seen in Tab.5.2. The bold values in Tab.5.1 represent the variables that during the stochastic optimizations have uncertainty.

		Lower Bound	Upper Bound
Design Variables	α (°)	0	1.9
	Airspeed (m/s)	**100**	**300**
	Altitude (m)	**5000**	**11000**
	Throttle	0	1

Table 5.1: Test case 1 - Design Variables

Parameter	Value	Parameter	Value
β (°)	0	Payload (%)	100
HT_{inc} (°)	1	t_w/c	5.25E-03
Fuel Tank (%)	100	t_s/t_{max}	5.25E-03

Table 5.2: Test case 1 - Fixed Parameters

In these tables, α is the angle of attack, V is the airspeed, β is the side-slip angle, HT_{inc} is the incidence of the horizontal stabilizer, t_w/c is the relative thickness of the web of the wing spars and t_s/t_{max} is the relative thickness of the skin of the aerodynamic surfaces, as illustrated in Fig.5.1.

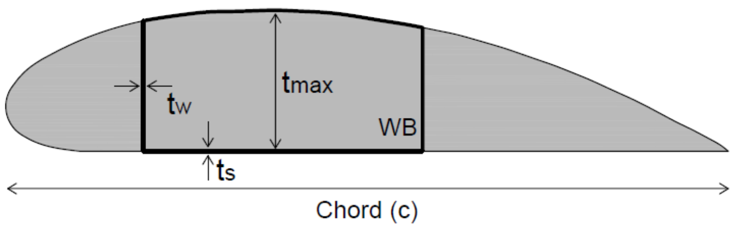


Figure 5.1: Test case 1 - Airfoil thickness parameters

It can be seen that only operational conditions were chosen to be design variables. This serves two purposes: first, it makes this test case a bit simpler; and second, it leaves all the geometric and configuration variables to be optimized in the next test cases, making every test case unique.

The upper bound of the angle of attack was chosen to be 1.9 ° because at 2 ° a shock wave, that the panel method used cannot detect, forms and drastically reduces lift. As for the other bounds, they were chosen for conveniently reduce the search space.

Since one of the objectives of this problem is to test the stochastic methods, some of the design variables need to carry uncertainty. For that purpose, and seeing that two of the constraints are equalities, i.e. they do not affect the outcome of stochastic optimization, the chosen variables (airspeed and altitude) were the ones that directly influence the inequality constraint. In a real context, the uncertainty tied to these variables, might exist due to errors in the aircraft sensors.

To better understand how the stochastic methods perform in different conditions, a lot of different optimizations were made. These optimizations are characterized by having different combinations of levels of uncertainty (applied to airspeed alone or airspeed+altitude) and target reliabilities.

It is also important to note that in this test case, only optimizations with target reliability were performed. Because the majority of the used methods perform reliability analysis, it was decided that the robust method (SP) would use the type of constraint of Eq.(3.43). This way, it is possible to directly compare all the tested methods.

5.2 Analysis of the Surrogate Models

To understand the influence of each variable on the range itself, a surrogate response study was conducted and its results can be seen in Figs.5.2 and 5.3. Since none of the inputs of the structures surrogate are design variables, only the aerodynamics and propulsion ones were tested.

By looking at these figures, one thing that immediately stands out is the fact that the surfaces of the surrogates apparently look smooth and without any noticeable weird behaviors. This leads us to conclude that, apparently, the surrogates were well generated.

After, by taking a closer look at each one of the figures individually, there are a few other things worth mentioning. An increase in thrust is verified as altitude gets lower, due to an increase of air density. With the exception of certain altitudes, increasing the thrust produces an increase in the airspeed. As expected the higher the throttle, the higher the thrust.

When it comes to the aerodynamic forces, it is all pretty straight forward. Both C_L and C_D are higher (modulus wise in the case of C_L) for lower altitudes, despite the fact that the air density increases. Likewise, as the angle of attack strays away from its $0°$ position, the higher the C_L (modulus wise) and C_D. Both C_L and C_D behave like a parabola when velocity varies, i.e. there is a maximum value for a given velocity in the middle of the scale. Because the panel method that was used, it is not capable of capturing the shock wave that forms for angles of attack higher than $2°$, both C_L and C_D keep increasing in a smooth fashion when indeed that was not supposed to happen.

After analyzing Figs.5.2 and 5.3, it becomes clear the sheer number of different effects each variable has on each parameter, which in turn affects the range. Because of this, it is not possible to determine the best operational conditions that maximize the range without resorting to MDO.

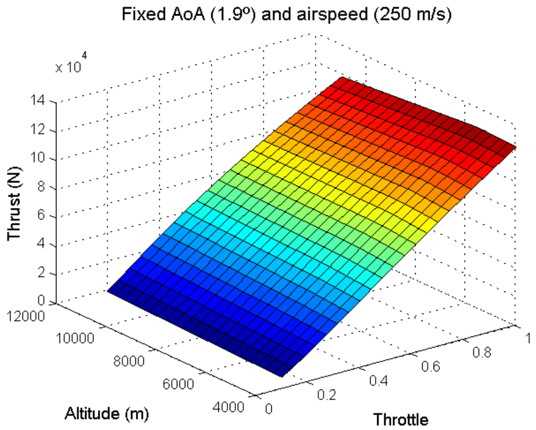
5.3 Optimization Results

All the conducted optimizations used the same FSQP algorithm and initial point ($\alpha = 0.9°$, airspeed $= 250m/s$, altitude $= 10900m$ and throttle $= 0.64$).

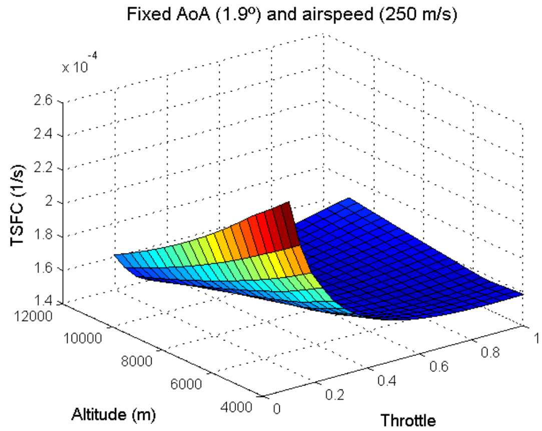
Among the obtained results, there are two different kinds: first, the ones that have to do with the performances of each method; and second, the ones that concern the optimized results.

5.3.1 Optimization Strategy

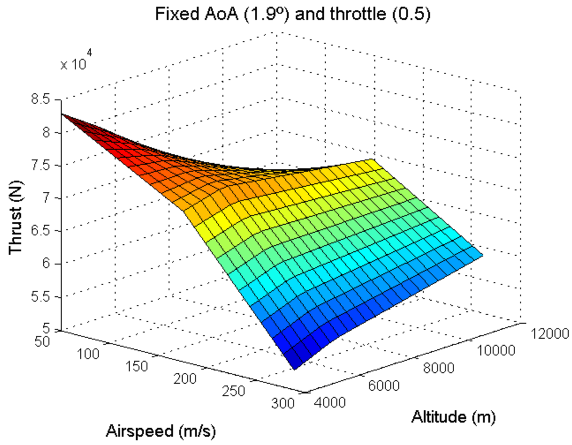
Several different optimization cases were tested as seen in Tab.5.3. Each optimization case is characterized by their unique combination of three parameters, namely, the level of uncertainty (defined by $c.o.v.$),



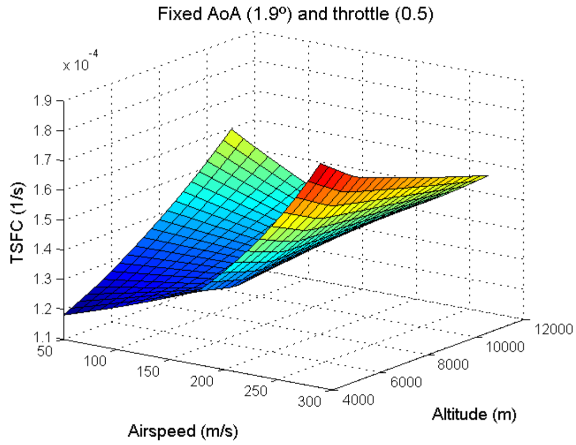
(a) Altitude vs throttle vs thrust



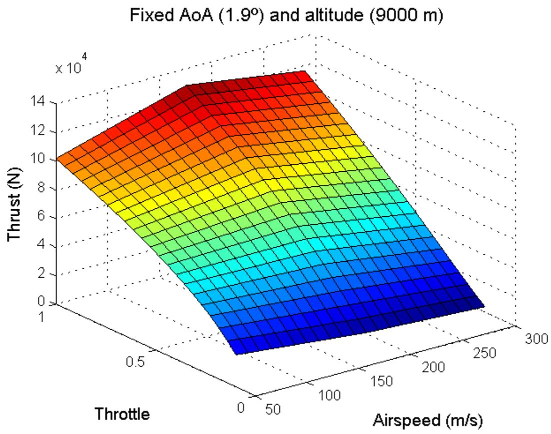
(b) Altitude vs throttle vs TSFC



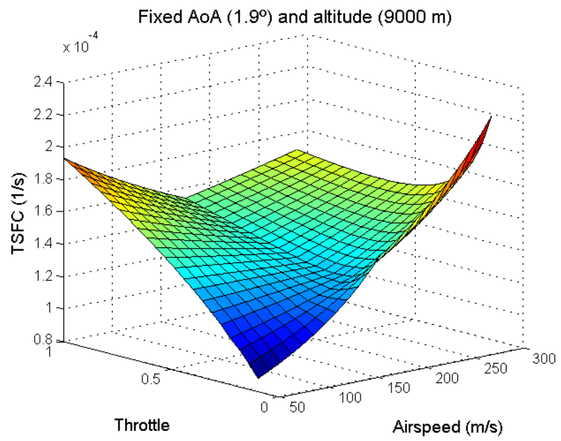
(c) Airspeed vs altitude vs thrust



(d) Airspeed vs altitude vs TSFC



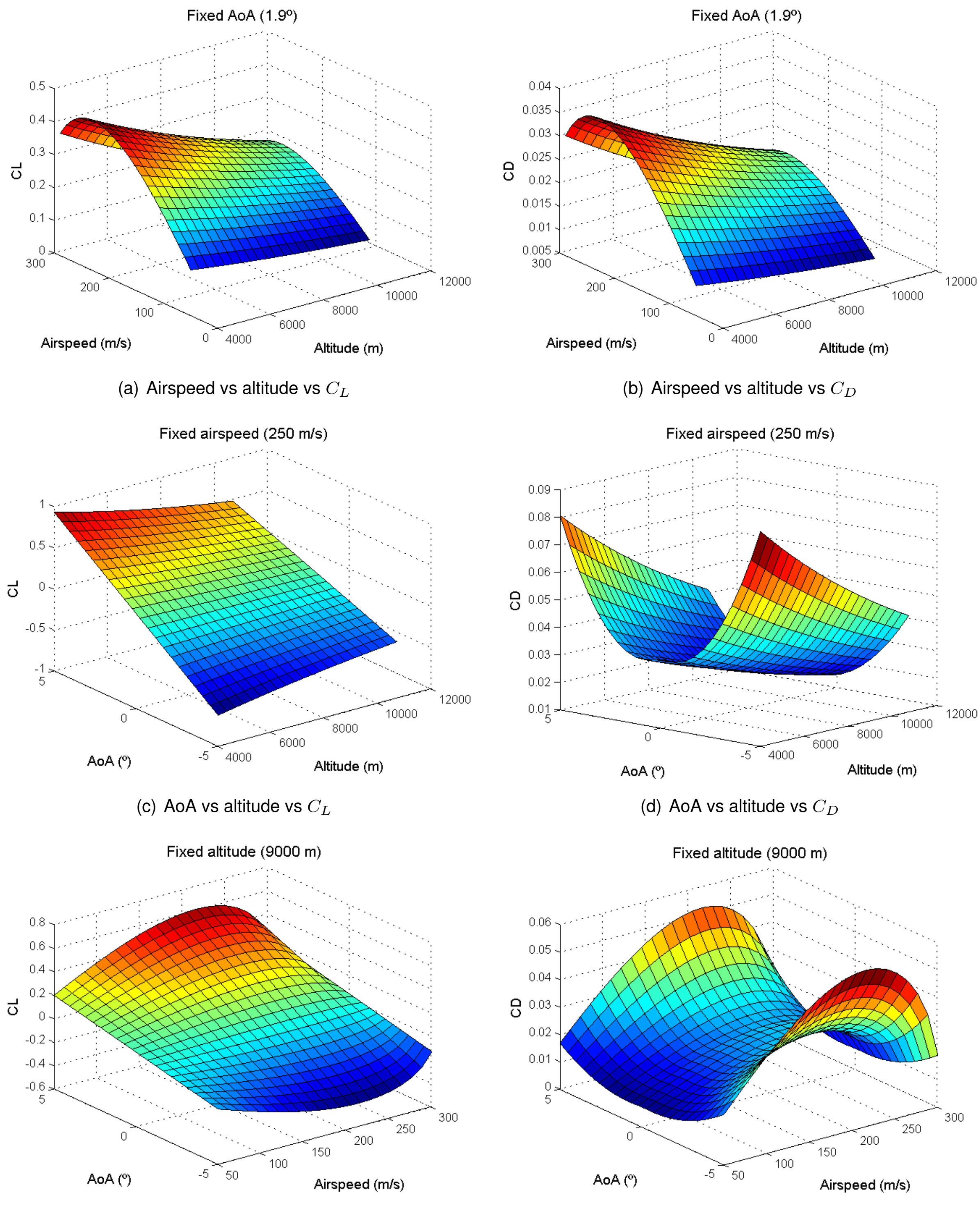
(e) Throttle vs airspeed vs thrust



(f) Throttle vs airspeed vs TSFC

Figure 5.2: Test case 1 - Thrust and TSFC surrogate response

the number of variables that have uncertainty (either only airspeed or airspeed plus altitude) and the target reliability of the optimization. For each case, the four different methods of stochastic optimization were utilized, thus making the complete test case be comprised of 72 optimizations. This was only possible because surrogates were used. Had the modules been used instead and it would be impossible due to time constraints. In order not to exceed 72 optimizations, it was decided that in the cases with



(a) Airspeed vs altitude vs C_L

(b) Airspeed vs altitude vs C_D

(c) AoA vs altitude vs C_L

(d) AoA vs altitude vs C_D

(e) AoA vs airspeed vs C_L

(f) AoA vs airspeed vs C_D

Figure 5.3: Test case 1 - Aerodynamic surrogate response

two random variables, both would have the same $c.o.v.$.

The numbers that were assigned to each different case in Tab.5.3, serve as a way to distinguish them.

Tables 5.4 and 5.5 show examples of the type of data gathered during each optimization. The first four rows refer to the values of the optimized design variables. The next two rows, refer to the expected

Case Number	$c.o.v.$ (%)	N_{RV}	β_{reqd}	Case Number	$c.o.v.$ (%)	N_{RV}	β_{reqd}
1	1	1	2	10	5	2	3
2	1	2	2	11	8	1	3
3	5	1	2	12	8	2	3
4	5	2	2	13	1	1	4
5	8	1	2	14	1	2	4
6	8	2	2	15	5	1	4
7	1	1	3	16	5	2	4
8	1	2	3	17	8	1	4
9	5	1	3	18	8	2	4

Table 5.3: Test case 1 - Different tested cases

value of the objective function and its standard deviation (in the case of SP and SP+SORA). After, comes the number of required function evaluations. If the modules were used instead of the surrogates, these function evaluations would correspond to the number of complete analysis that would be required. Finally, both the reliability of the design point and its error, compared to the target reliability, are shown. The reliability error was once again computed with the MC method using 6×10^6 samples.

Case - 8		$c.o.v.$ 1%			
	Deterministic	SP	PMA	SORA	SP+SORA
α (^{o})	1.9	1.9	**1.9**	1.9	1.9
Airspeed (m/s)	244.8386	238.3604	**238.3583**	238.1844	238.1844
Altitude (m)	8576.664	8362.631	**8362.554**	8356.132	8356.132
Throttle	0.466867	0.461349	**0.461347**	0.461199	0.461199
μ_f (km)	4602.761	4534.342	4534.544	4532.691	4532.467
σ_f (km)		23.02763			23.0203
Obj calls	60	325	55	155	750
Con calls	229	1235	1.81E+05	961	940
Total calls	289	1560	181172	1116	1690
β	0	2.9994	3.0001	3.0854	3.082
β error (%)		0.02	0.003333	2.846667	2.733333

Table 5.4: Test case 1 - Obtained values for $c.o.v.$ 1%, $\beta_{reqd} = 3$ and $N_{RV} = 2$

5.3.2 Analysis of the Optimization Methods

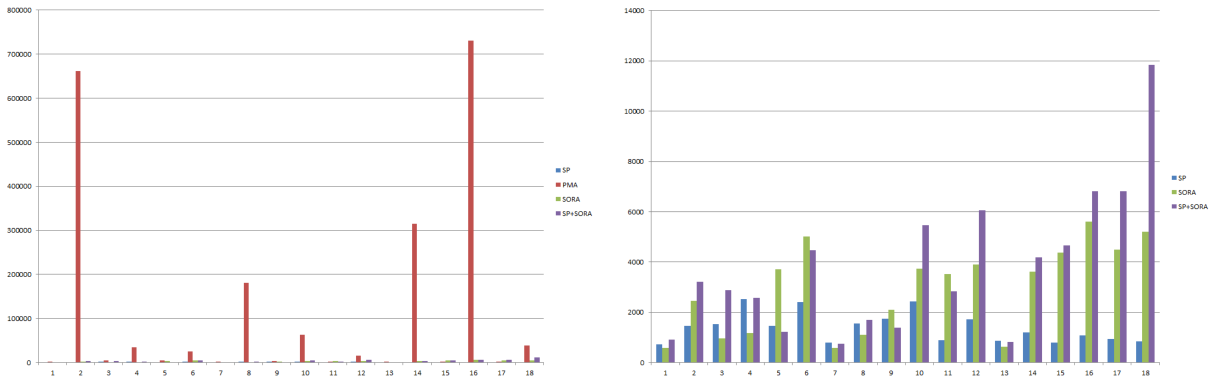
When it comes to assessing the qualities of each individual stochastic optimization method, two parameters were chosen to be evaluated: one that would determine the efficiency of the methods, which is the total number of function evaluations; and another that would determine whether or not the design point meet the target reliability, which is the reliability error.

While Fig.5.4(a) shows the number of function evaluations it took for each of the methods to complete a certain optimization, Fig.5.4(b) shows the same thing but excludes the PMA because of the different order of magnitude of its results. Figure 5.5 shows the average number of function evaluations for each of the methods, without accounting for both the higher and lower values produced by each of the methods. This was done to avoid being misled by numerical problems.

As can be seen either from both Figs.5.4 and 5.5, PMA is clearly the method that requires more function evaluations. This is due to the reliability assessment it has to do for every new iteration. Since

Case - 12		$c.o.v.$ 8%			
	Deterministic	SP	PMA	SORA	SP+SORA
α (°)	1.9	1.9	1.9	1.9	1.9
Airspeed (m/s)	244.8386	202.9679	202.9139	202.9139	202.9011
Altitude (m)	8576.6640	6461.0569	6457.3605	6457.3608	6456.4833
Throttle	0.4669	0.4303	0.4302	0.4302	0.4302
μ_f (km)	4602.7609	4127.1393	4139.5402	4139.5403	4126.3106
σ_f (km)		171.7406			171.8270
Obj calls	60	350	70	689	3175
Con calls	229	1375	1.58E+04	3213	2887
Total calls	289	1725	15874	3902	6062
β	0	2.995	2.9998	2.999	3.0005
β error (%)		0.166667	0.006667	0.033333	0.016667

Table 5.5: Test case 1 - Obtained values for $c.o.v.$ 8%, $\beta_{reqd} = 3$ and $N_{RV} = 2$



(a) Total number of module calls for all the methods

(b) Total number of module calls without PMA

Figure 5.4: Test case 1 - Total number of function calls

this reliability assessment is essentially an optimization problem within an optimization problem, the number of required function evaluations greatly increases. This is specially patent in the even numbered cases (two stochastic variables), where the complexity of the reliability assessment problem is bigger, due to the increased number of random variables.

As expected, SORA solves this problem by greatly reducing the number of necessary function calls. This behavior can be seen in both SORA and SP+SORA methods, which have similar numbers when compared to SP.

While not always true, SP is the method that has the lowest number of required function evaluations overall. It can be seen that, even though for the lower numbered cases (low target reliability), SP sometimes requires more functions than SORA or SP+SORA, as the target reliability increases, it seems to become the best method in terms of evaluations. Inversely, SORA and SP+SORA seem to need an increased number of function evaluations as the target reliability gets higher. This may have to do with the MPP problem becoming harder to solve.

In terms of reliability, the differences between all the methods can be seen in Figs.5.6 and 5.7. These figures show respectively, the reliability error produced by each method, for all of the cases, and the average of the reliability error. Once again, this average was calculated like that of Fig.5.5.

From Fig.5.6, it can be seen that while both SP and PMA have close to no error, things are a little bit

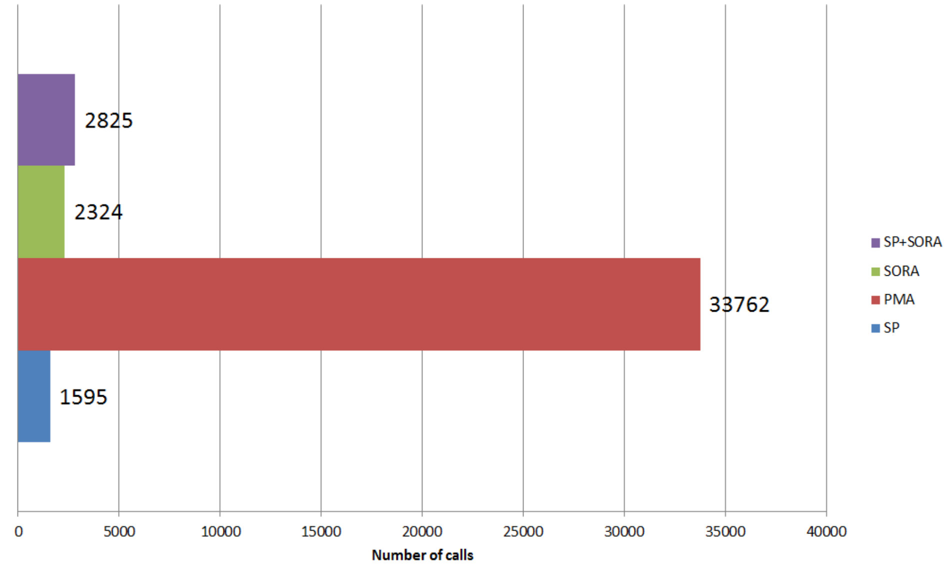


Figure 5.5: Test case 1 - Average number of function calls

different for SORA and SP+SORA. For $c.o.v.$ of 1% (cases 1,2,7,8,13 and 14), it seems that the error of both SORA and SP+SORA, grows (within acceptable values) with the target reliability. For other levels of uncertainty, the errors produced by these two methods, while being mostly small, sometimes without no apparent reason grow up to values of 10% or more. It seems that, for certain cases, the SORA algorithm, which is the common denominator between these two methods, is not numerically robust enough. Because of this, while for most of the cases these methods produce acceptable results (this can be seen in Fig.5.7), at times, they are just not able to achieve the target reliability.

As for SP and PMA, Fig.5.7 clearly shows that for this specific problem, both of them have very little error when it comes to target reliability. This was something expected from a method like PMA, but not necessarily from SP. Even at higher levels of uncertainty, the SP method maintains consistency, which is something really interesting considering the low number of function evaluations when compared to the other stochastic methods.

5.3.3 Optimal Solution

To analyze the evolution of each design variable during the optimizations, it is important to consider the parameters that promote such evolution. Parameters such as the target reliability, the different uncertainty levels and also the number of stochastic variables, all have different effects on how far the design variables position themselves relatively to the inequality constraint. In this section, these effects are discussed.

It is possible to see in Tabs.5.4 and 5.5 that even though the target reliability is the same, because the $c.o.v.$ is not, there are already some significant differences between the results obtained for some of the design variables. This difference can be especially seen for altitude and airspeed, which are the variables that are used to compute the inequality constraint (number of Mach). The angle of attack is at its allowed maximum of 1.9 $^\circ$ to compensate for the aircraft higher weight of the full payload/fuel

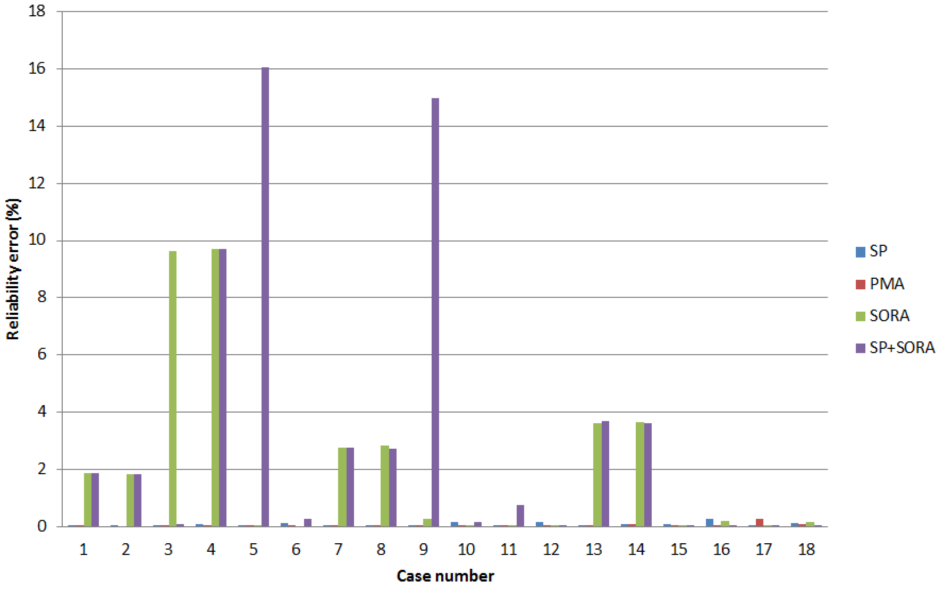


Figure 5.6: Test case 1 - Reliability error

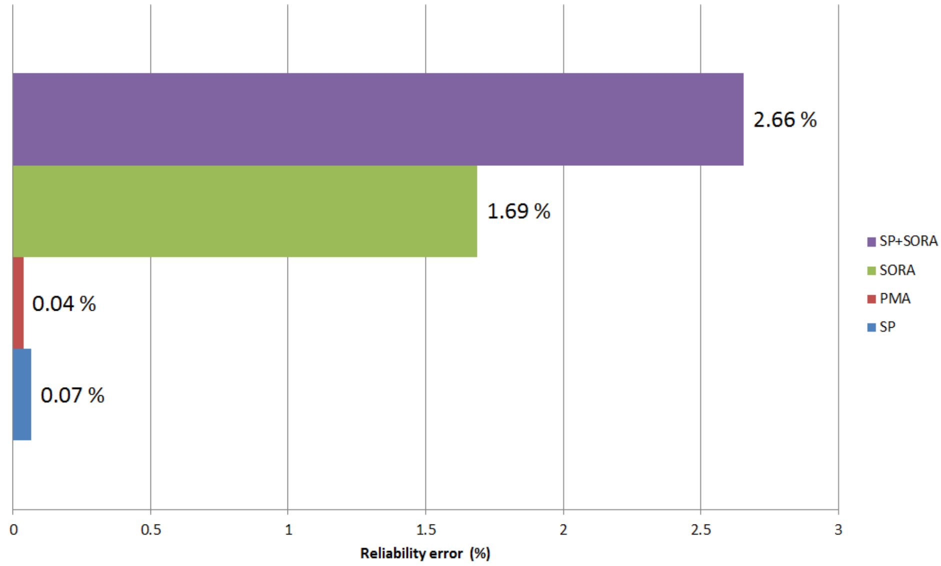


Figure 5.7: Test case 1 - Average reliability error

configuration. As for the throttle, it lowers slightly as uncertainty grows because the airspeed does too.

Still, in Tabs.5.4 and 5.5, it is possible to see that both SP and PMA have really similar results. Based on the previous section conclusions, this was expected since their reliability errors are practically the same. The only difference between them is the objective function expected value. This happens because, while PMA deals with the objective function deterministically, SP calculates its mean and standard deviation and presents them separately.

As for SORA and SP+SORA, they present similar results to one another. This suggests that the SORA algorithm is the most influencial during the optimizations. When comparing their results to the the ones obtained with SP and PMA, even though they are really close in these two tables, the same does not apply for every case. It can be seen in Fig.5.6 that for cases 8 and 12 (the ones that are shown in these tables), none of the methods have a large error. Because of this, it was expected that both SORA and SP+SORA would have similar results when compared to the other methods. For the cases where their reliability error increase, the results obtained become slightly different than the ones obtained with SP and PMA.

To better understand the behavior of the two random variables in this test case, a few graphics were plotted using values obtained with the PMA method, since it was the one that presented the lowest average error. Figure (5.8) shows the optimal airspeed and altitude for each case. It can be seen

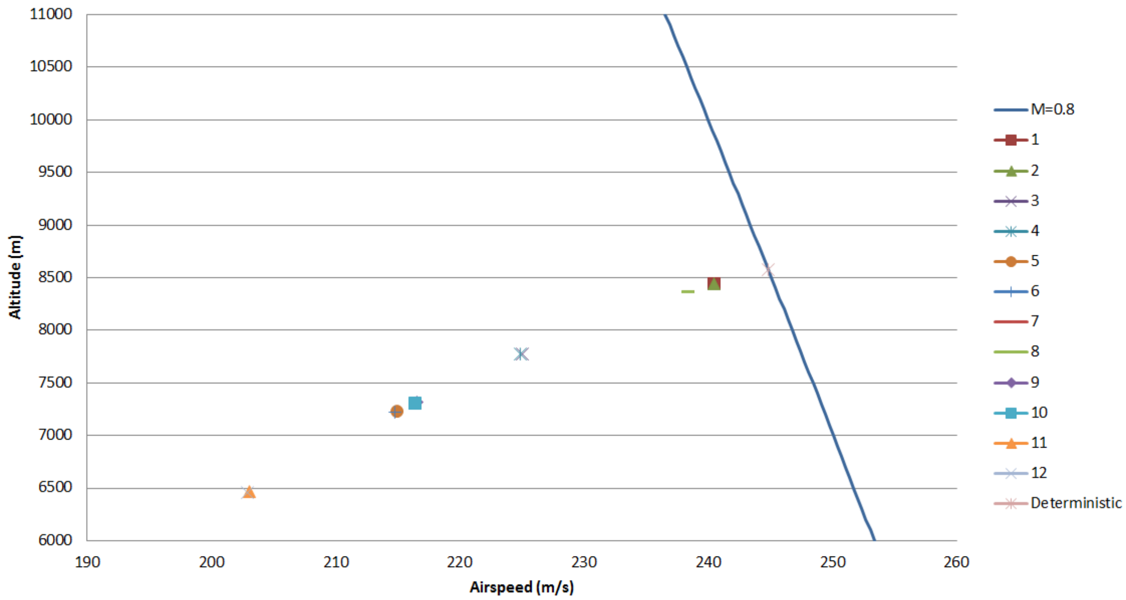


Figure 5.8: Test case 1 - Evolution of the DV's for each case

a gradual shift of the design variables away from the constraint $M \leq 0.8$, for increasing uncertainty, number of random variables and target reliability.

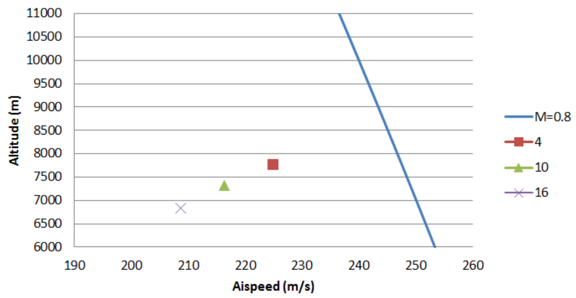
Because of how the cases are characterized and numbered, when all of them are plotted like in Fig.5.8, it is not possible to tell which parameters have more or less influence. To solve this, Fig.5.9 is presented. This figure is divided in three parts, each showing the influence of different parameters, such as the level of uncertainty, the number of random variables and the target reliability.

In Fig.5.9(a), both the uncertainty level and the number of random variables are fixed at $c.o.v.$ 5% and 2, respectively, which means that the influence of the target reliability can be studied. It can be seen that as the target reliability increases, the further away the design variables become from the constraint.

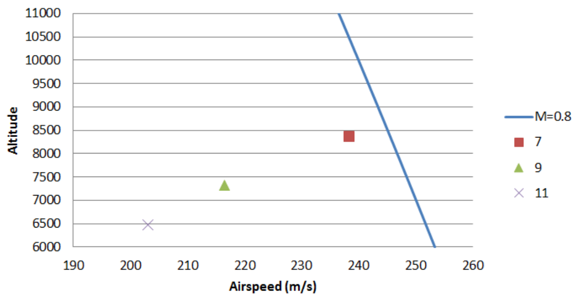
In Fig.5.9(b), the fixed parameters were both the target reliability and the number of random variables (3 and 2, respectively), meaning that the parameter in focus is the level of uncertainty. As $c.o.v.$ increases from 1% to 8%, the shift in the design point is even more prominent than in the previous case.

Finally, Fig.5.9(c) was obtained by fixing the target reliability at 3 and letting both the level of uncertainty and number of random variables vary. This way, it is possible to understand how the number of variables influence the position of the optimum design point, for different uncertainty levels. As it can be seen, this influence is so small that, for the same level of uncertainty, the points with 1 and 2 random variables are practically overlapping. Indeed, the points with 2 random variables are further away from the constraint than the ones with only 1 random variable, but only slightly. For different levels of uncertainty, this behavior remains the same.

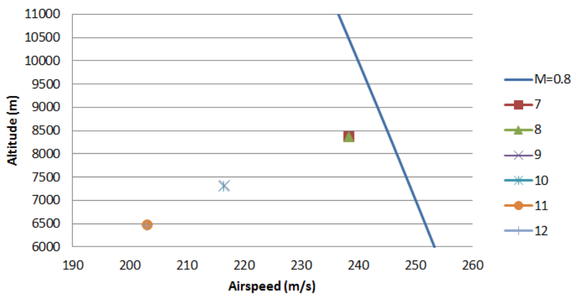
All the behaviors shown in Fig.5.9 also influence the value of the objective function. As can be seen in Fig.5.10, the range tends to decrease with increasing uncertainty levels, number of stochastic variables and target reliability. By paying close attention to the case numbers, it is possible to validate the conclusions drawn from Fig.5.9. The lowest influence comes from the number of random variables, then from the target reliability and finally from the $c.o.v.$. For the sake of comparison, the range obtained with the deterministic optimization has also been included in Fig.5.10 (red bar). Since in the deterministic optimization, the constraints do not need to be probabilistically satisfied, its range is always higher. For example, when compared to cases 12 and 18 (which is not shown in the figure), the deterministic range is respectively 11.2 % and 15.3 % higher.



(a) Evolution with target reliability index



(b) Evolution with uncertainty levels



(c) Evolution with number of stochastic variables

Figure 5.9: Test case 1 - Evolution of the stochastic design variables with different parameters

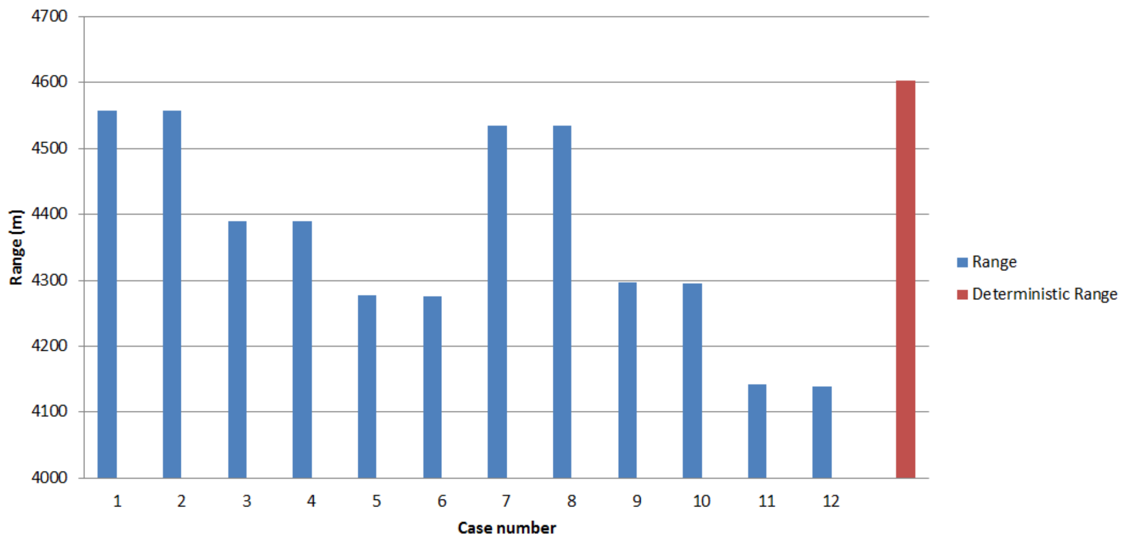


Figure 5.10: Test case 1 - Evolution of the range for each case

Chapter 6

Test Case 2 - Baseline + Winglet Surrogate Optimization

6.1 Problem Definition

In the second test case, it was decided to change the model under study. Instead of the baseline model of EMB9MOR of the previous test case, a baseline+winglet model was used. As stated before, the focus of this test case is still in assessing the capabilities of the stochastic methods and, because of that, surrogates were again employed. Since there were no available surrogates for this model, they had to be generated. As for the problem to be solved, it was devised while taking into account both the objectives of this thesis and the limitations of the surrogate models.

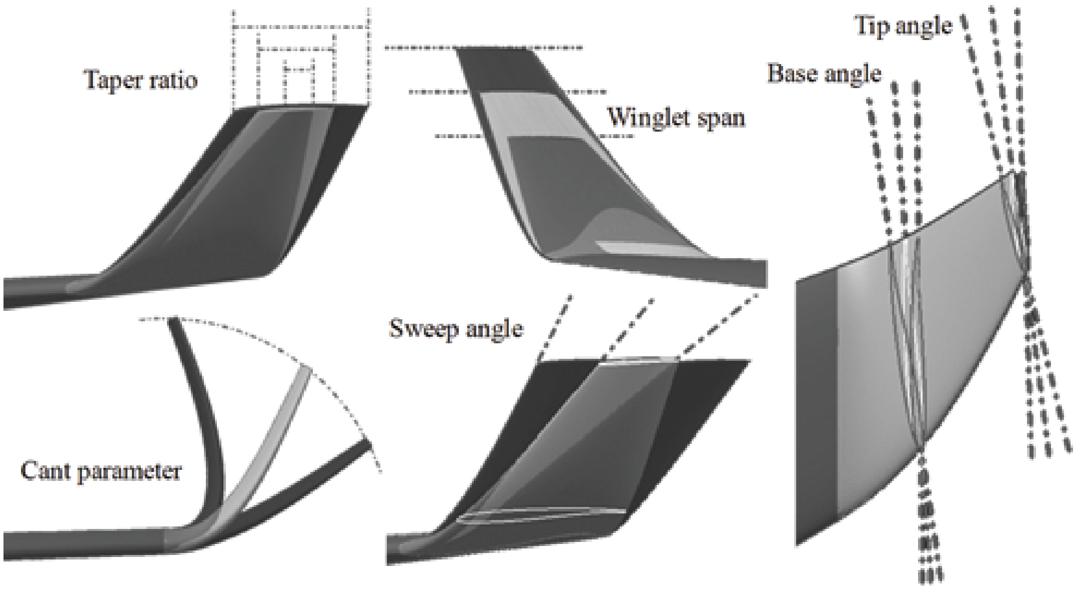


Figure 6.1: Test case 2 - Winglet parameterization [Cella and Romano, 2012]

The winglets used in this test case are similar to the one of Fig.6.1. They can be characterized by five different parameters: cant angle, toe angle (also known as twist), sweep angle, winglet span and tip chord. It is important to note that instead of the taper ratio shown in this figure, the tip chord length is used because in this test case the root chord of the winglet is fixed.

When developing this test case, the idea was to make it an extension of the previous one. By adding the winglet parameters and a maximum stress constraint to the design variables and constraints of the previous test case, it would be possible to understand how the stochastic optimization methods perform in a more complex problem. Unfortunately, since the computational time required to generate a surrogate increases exponentially with the number of inputs, it was not possible to have a problem with so many design variables. Because of this, a similar approach to that of test case 1 was used. Of all the existing variables, some were chosen as design variables, others as fixed parameters and finally, some were chosen as geometric variables. This division can be seen in Tabs.6.1 and 6.2.

		Lower Bound	Upper Bound
Design Variables	$\frac{t_w}{c}$	5E-04	1E-02
	$\frac{t_s}{t_{max}}$	5E-04	1E-02
	Toe (°)	-10	5
	Cant (°)	-80	80
	Sweep (°)	20	60
Geometric Variables	Span (m)	0.5	3
	Tip Chord (m)	0.7573	1.7276

Table 6.1: Test case 2 - Design and Geometric Variables

Parameter	Value	Parameter	Value
β (°)	0	α (°)	1.9
HT_{inc} (°)	1	Airspeed (m/s)	238.3583
Fuel Tank (%)	100	Altitude (m)	8362.554
Payload (%)	100	Throttle	0.461347

Table 6.2: Test case 2 - Fixed Parameters

Since in the first test case the operational conditions had already been design variables, they were fixed in this one. Their values were chosen to be the ones in bold in Tab.5.4, which correspond to a PMA optimization with $c.o.v.$ of 1%. The choice of these values was tied to the fact that they represent realistic operational conditions for this aircraft model. In terms of the rest of the fixed parameters, besides the relative thicknesses, which have a great influence on the maximum stress constraint, all the ones from the previous test case remained.

This left us with seven possible design variables, five that only had to do with the winglet and two relative thicknesses. Because seven was still a large number of inputs to have on a surrogate, they had to be further divided. The final choice was to have both thicknesses and winglet angles to be design variables, while the Span and Tip Chord were chosen to be the geometric variables. The way these geometric variables work is that, instead of studying their influence on the problem, by letting them vary within the surrogates, their influence is studied by resorting to several sets of surrogates. In this test case, instead of only one set of surrogates, nine different ones are used. Each set corresponds to a geometrically unique winglet characterized by a certain combination of the geometric variables.

By fixing the operational conditions and changing the model of the aircraft, it became impossible for the optimizer to find a feasible point where the trimming conditions were satisfied. Because of this, neither $L = W - Tsin(\alpha)$ nor $Tcos(\alpha) = D$ are constraints in this problem. On top of that, since both the airspeed and altitude have been fixed, the constraint $M \leq 0.8$ is not used either. These simplifications lead to the following optimization problem:

$$\begin{aligned} \underset{x}{\text{maximize}} \quad & \frac{C_L}{C_D} ln \frac{W_1}{W_2} \\ \text{subject to :} \quad & \sigma \leq \tfrac{\sigma_{max}}{SF} \end{aligned} \tag{6.1}$$

where the design variables are the ones mentioned before, the objective function consists of the range equation without the $\frac{V}{c_t}$ part, since none of the design variables actually influence it, and finally, the constraint for the maximum allowable stress. It is important to note that SF refers to the Safety Factor and that it is only used for the deterministic optimization.

Finally, it is important to mention that, all the simplifications that were made to this problem, in terms of fixing operational conditions and removing constraints, have a certain impact. Since the aircraft is not even trimmed, besides the information regarding the performance of the methods, the rest of the optimized results cannot really be compared to the ones of the previous test case. Nevertheless, it is still possible to compare all the winglets among them.

6.2 Analysis of the Surrogate Models

As previously stated, for this specific problem, nine sets of surrogates were used, each one corresponding to a certain winglet model. Each of these sets were comprised of five different surrogates. While some of these surrogates had to be specifically generated for this test case, others were just reused from the previous test case.

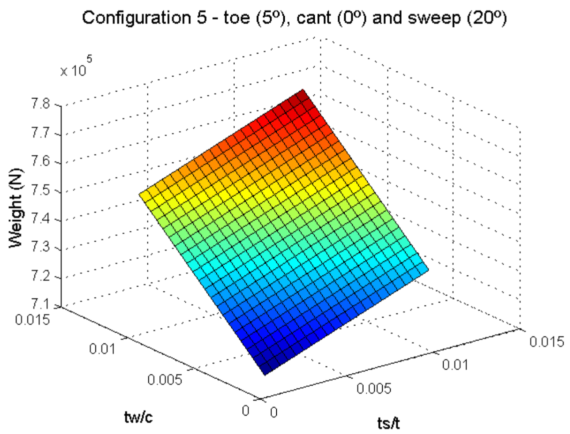
Since no modifications were made to the propulsion nor to the payload, both these surrogates were reused. In terms of the aerodynamics surrogate, a new one was created using three design variables (cant, toe and sweep), while all the other inputs remained fixed. Finally, both a structural mass distribution surrogate and a stress surrogate were generated, with all the design variables as inputs. It is important to remind that, these last three surrogates were generated once for each combination of winglet geometric variables.

Once all the surrogates for every winglet configuration had been generated, a similar study to the one performed in the last test case was conducted. The response of the structural mass distribution surrogate, the aerodynamics surrogate and the stress surrogate is computed and then shown in Figs.6.2 and 6.3. Like in the previous test case, the first thing that can be noticed is the apparent good quality of the surrogates. The relation between the inputs and outputs seem to be in accordance to what was expected and no weird behaviors can be observed. It is important to note that in order to obtain these graphics, only the surrogates relative to configuration 5 (Span=1.75m and Tip chord=1.242m) were used. Configuration 5 was chosen because its parameters lie in the middle of their range.

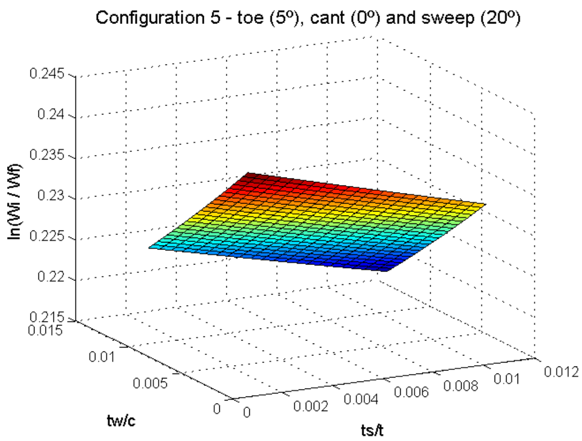
Figure 6.2 shows us how the relative thicknesses affect the initial weight of the aircraft, the logarithm of the initial weight of the aircraft divided by its final weight (after cruise) and also the maximum stress present in the aircraft. These figures were obtained with fixed winglet angles and for the fixed parameters of Tab.6.2.

As expected, it can be seen that the weight of the aircraft increases with the relative thicknesses of the wing box. Unlike the weight, the logarithm tends to decrease with the increase of the thicknesses, which of course means that, the lighter the structure the bigger the range.

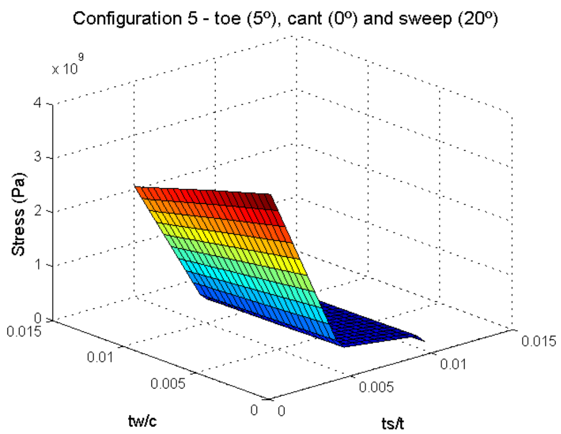
Finally, the last conclusion that can be drawn from the first set of figures has to do with how both relative thicknesses affect the maximum stress. As it can be seen, it appears that t_s/t_{max} has a bigger influence on the maximum stress than t_w/c. As the values of t_s/t_{max} decrease until a certain point, the stress is really low and does not increase that much, past this value (downwards) the stress starts to grow with a big slope. In terms of t_w/c, the lower it is, the higher the stress becomes, but not by much. Because of this, for this specific problem, it appears as if t_s/t_{max} is the thickness parameter that will suffer most changes in order to achieve a certain reliability requirement.



(a) $\frac{t_w}{c}$ vs $\frac{t_s}{t_{max}}$ vs weight (W_i)



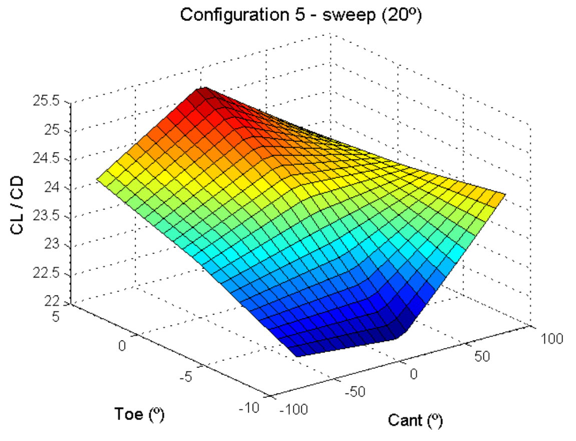
(b) $\frac{t_w}{c}$ vs $\frac{t_s}{t_{max}}$ vs log$\left(\frac{W_i}{W_f}\right)$



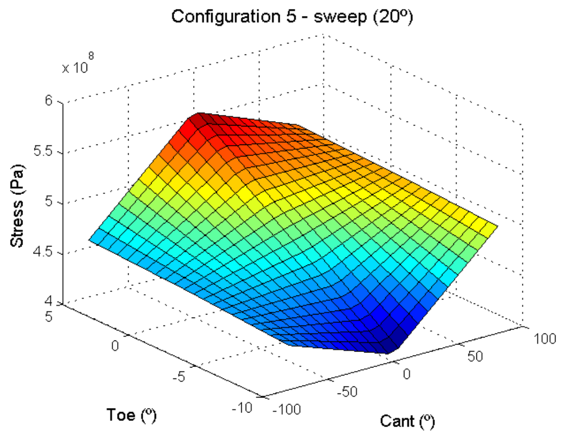
(c) $\frac{t_w}{c}$ vs $\frac{t_s}{t_{max}}$ vs stress

Figure 6.2: Test case 2 - Surrogate's Weight response

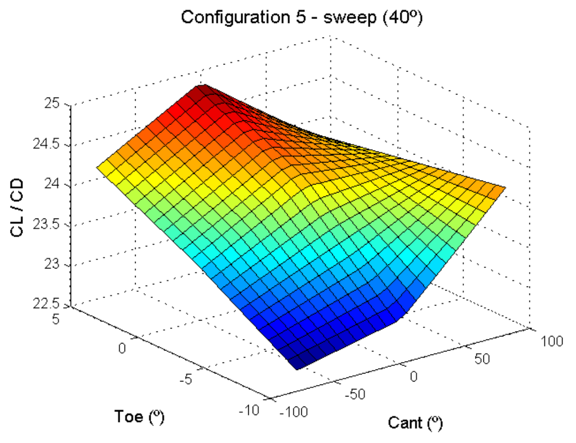
Figure (6.3) shows us a brief glimpse of how the winglet angles affect both C_L/C_D and the maximum stress. As can be seen, only the toe and cant angles are used as input variables. The influence of the sweep angle can be observed by looking at the different rows of figures, which were obtained for different sweep angles. It is important to note that these surrogates are relative to configuration 5 and that for the other configurations, these behaviors might not be the same.



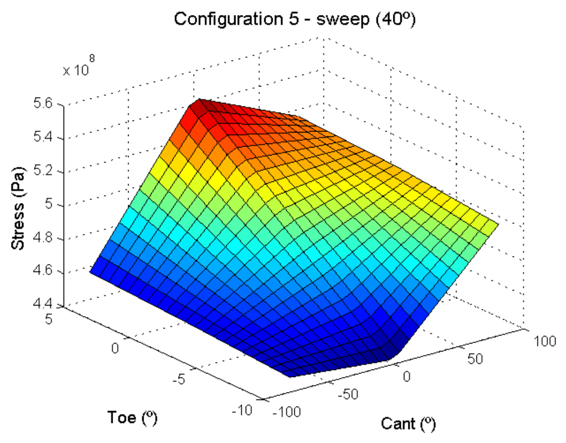
(a) Toe vs cant vs $\frac{C_L}{C_D}$



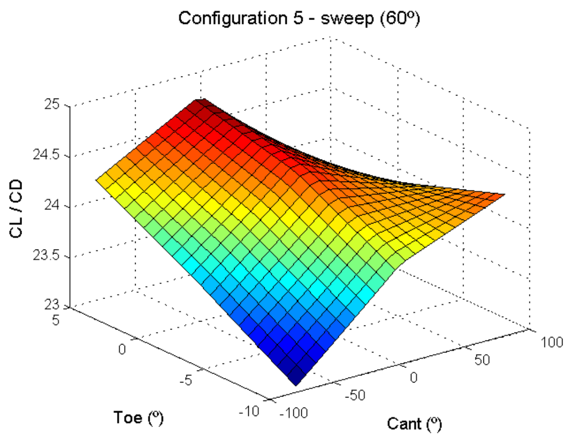
(b) Toe vs cant vs stress



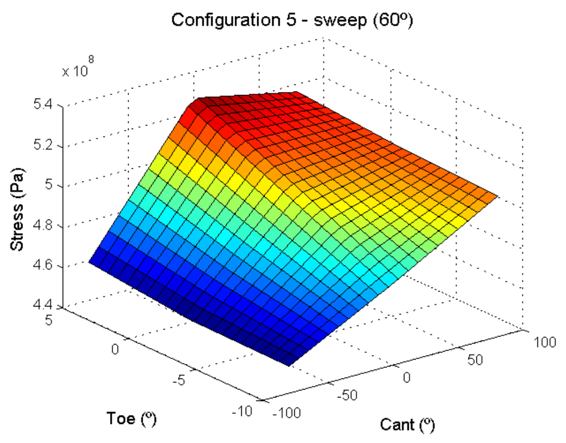
(c) Toe vs cant vs $\frac{C_L}{C_D}$



(d) Toe vs cant vs stress



(e) Toe vs cant vs $\frac{C_L}{C_D}$



(f) Toe vs cant vs stress

Figure 6.3: Test case 2 - Aerodynamic and Stress surrogate response

Even though there is not much to be said about the behavior evidenced in Fig.6.3, as expected, it can be seen that for every sweep angle (different rows), both C_L/C_D and stress have its maximum for the same combination of toe and cant angles ($5°$ and $0°$ respectively). Another thing that can be seen is that for this specific winglet configuration, both C_L/C_D and stress values seem to decrease with increasing sweep angle.

6.3 Optimization Results

Because there is more than just one configuration in this test case and, for each one of them, a different set of surrogate models, it was decided to divide it in two different parts (sections 6.3 and 6.4).

The first part is similar to the one of the previous test case. Optimizations using all the different methods are carried out and their results are shown in this section. This way, it is possible to draw conclusions about each method performance and results. Like in section 6.2, only configuration 5 is used, for the same reasons. Once again, the FSQP algorithm is used throughout the whole test case, with an initial point of $toe = 4.6°$, $cant = 0°$, $sweep = -18.4°$, $\frac{t_w}{c} = 0.001$ and $\frac{t_s}{t_{max}} = 0.08$.

As for the second part of the test case, it consists of the analysis of the different winglet configurations. This will be further explained in the proper section.

6.3.1 Optimization Strategy

Like in the previous test case, 72 different optimizations are solved with all the stochastic methods. Each optimization is characterized by a different combination of $c.o.v.$, number of stochastic variables and target reliability. These combinations can be seen in Tab.6.3.

Case Number	$c.o.v.$ (%)	N_{RV}	β_{reqd}	Case Number	$c.o.v.$ (%)	N_{RV}	β_{reqd}
1	1	2	2	10	5	5	3
2	1	5	2	11	8	2	3
3	5	2	2	12	8	5	3
4	5	5	2	13	1	2	4
5	8	2	2	14	1	5	4
6	8	5	2	15	5	2	4
7	1	2	3	16	5	5	4
8	1	5	3	17	8	2	4
9	5	2	3	18	8	5	4

Table 6.3: Test case 2 - Different tested cases

While the methods, the levels of uncertainty and reliability targets stay the same when compared to the previous test case, the number of stochastic variables is slightly different this time. Instead of 1 or 2, now there is the option to optimize with 2 variables with uncertainty (both the thicknesses) or 5 (thicknesses plus winglet angles). Much like in the previous test case, the $c.o.v.$ is the same for all the variables to simplify the problem.

Tables 6.4 and 6.5 are two examples of the studied cases (cases 10 and 11 respectively). The first 5 rows show the optimized values of the design variables. After that, come the expected value of the objective function and its standard deviation. Afterwards, both aerodynamic and structural values such as the C_L and the weight are presented. Then comes the information regarding the number of function calls, and finally, the information about the reliability index.

Case - 10	FS=1.5	$c.o.v.$ 5%			
	Deterministic	SP	PMA	SORA	SP+SORA
Toe (°)	4.60	4.60	4.60	-9.20	4.60
Cant (°)	0.04	0.02	0.00	73.60	0.00
Sweep (°)	21.60	21.60	21.60	55.20	21.60
$\frac{t_w}{c}$	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04
$\frac{t_s}{t_{max}}$	8.68E-03	6.30E-03	6.31E-03	6.13E-03	6.29E-03
μ_f	5.93	5.97	5.98	5.79	5.97
σ_f	0.00	0.011	0.000	0.000	0.011
C_L	0.30	0.299	0.299	0.294	0.299
C_D	0.01	1.19E-02	1.19E-02	1.21E-02	1.19E-02
$\frac{C_L}{C_D}$	25.11	25.11	25.11	24.31	25.11
Weight (N)	7.30E+05	7.24E+05	7.24E+05	7.24E+05	7.24E+05
Stress (Pa)	2.67E+08	3.62E+08	3.61E+08	3.38E+08	3.62E+08
Obj calls	76	550	66	289	10417
Con calls	91	495	18079	1687	4607
Total calls	167	1045	18145	1976	15024
β	0	2.9717	2.9937	2.9415	2.9364
β error (%)		0.0094	0.0021	0.0195	0.0212

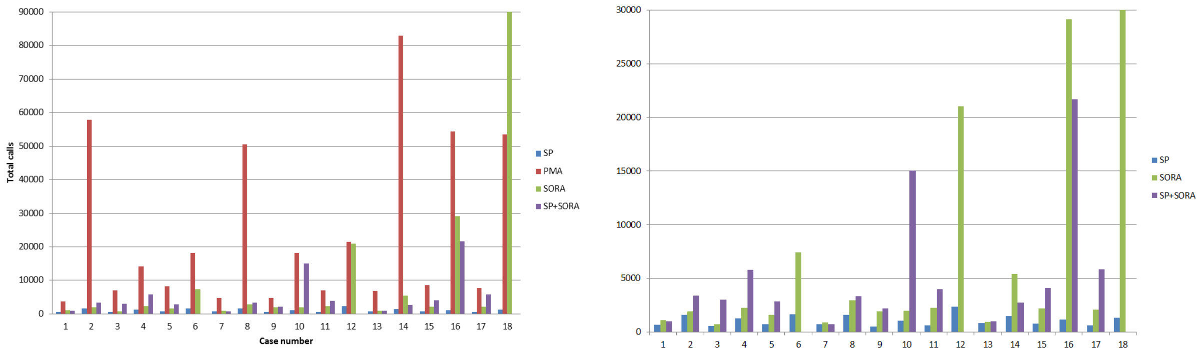
Table 6.4: Test case 2 - Obtained values for $c.o.v.$ 5%, $\beta_{reqd} = 3$ and $N_{RV} = 5$

Case - 11	FS=1.5	$c.o.v.$ 8%			
	Deterministic	SP	PMA	SORA	SP+SORA
Toe (°)	4.60	4.60	4.60	4.60	-9.20
Cant (°)	0.04	-0.10	0.02	0.00	73.60
Sweep (°)	21.60	21.60	21.60	21.60	55.20
$\frac{t_w}{c}$	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04
$\frac{t_s}{t_{max}}$	8.68E-03	7.03E-03	7.02E-03	7.02E-03	6.86E-03
μ_f	5.93	5.95	5.96	5.96	5.77
σ_f	0.00	0.012	0.000	0.000	0.011
C_L	0.30	0.299	0.299	0.299	0.294
C_D	0.01	1.19E-02	1.19E-02	1.19E-02	1.21E-02
$\frac{C_L}{C_D}$	25.11	25.11	25.11	25.11	24.31
Weight (N)	7.30E+05	7.26E+05	7.26E+05	7.26E+05	7.25E+05
Stress (Pa)	2.67E+08	3.32E+08	3.33E+08	3.33E+08	3.11E+08
Obj calls	76	330	110	853	2935
Con calls	91	260	6915	1404	1032
Total calls	167	590	7025	2257	3967
β	0	3.0064	2.9777	2.9733	3.0129
β error (%)		0.0021	0.0074	0.0089	0.0043

Table 6.5: Test case 2 - Obtained values for $c.o.v.$ 8%, $\beta_{reqd} = 3$ and $N_{RV} = 2$

6.3.2 Analysis of the Optimization Methods

To analyze the efficiency of each method in this specific optimization problem, three performance parameters were considered during the optimizations: the total numbered of function calls that each method requires is presented in Figs.6.4 and 6.5; the reliability error is presented in Figs.6.6 and 6.7; and finally, results concerning the value of the optimized design objective functions are shown in Fig.6.8.



(a) Total number of module calls for all the methods (b) Total number of module calls without PMA

Figure 6.4: Test case 2 - Total number of function calls

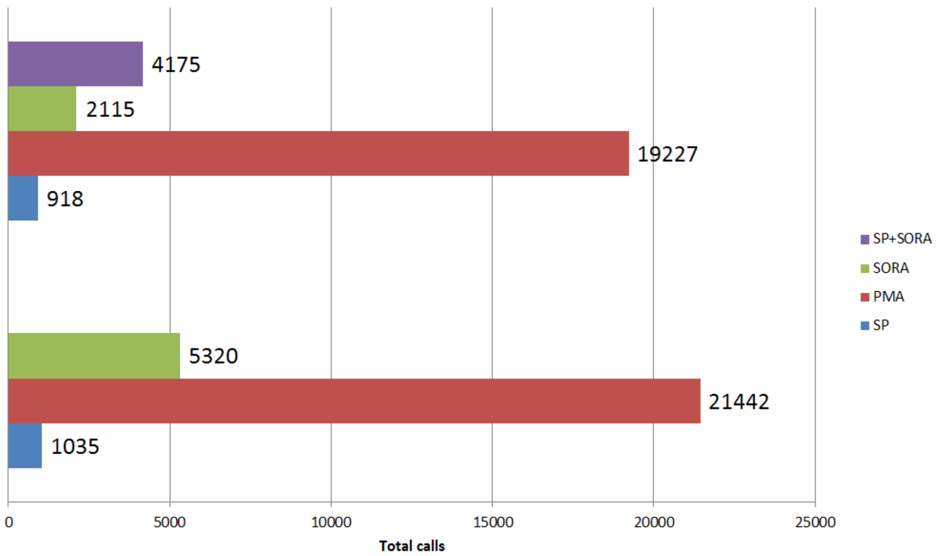


Figure 6.5: Test case 2 - Average number of function calls

First of all, it is important to note that in both Figs.6.4(a) and 6.4(a), for case 18, SORA required more function evaluations that those the vertical axis allow us to see. It was decided to reduce the maximum value of the vertical axis, so that the other results could be better seen.

By looking at Fig.6.4, some conclusions can be drawn. PMA is still the method that requires the highest number of function calls, while SP is the one that requires the least. PMA, SORA and SP+SORA seem to have trouble converging for cases with high target reliability, number of stochastic variables and uncertainty. This behavior is specially seen for the SP+SORA method, which was not even able to converge for all the cases that had 5 stochastic variables and $c.o.v.$ of 8%.

Figure (6.5) shows us two different average numbers of required function evaluations for each method. The average that is in the bottom does not include SP+SORA as it was not able to converge in some of the cases. In the top average, SP+SORA was included, but the average did not take into account those cases where this method was not able to converge. It is important to note that, once again, in order to avoid any misleading results (due to certain numerical problems each method might have), this average was computed like the ones from the previous test case, i.e., without the highest and lowest number of calls for each method. What can be seen in this figure, confirms the conclusions that were previously drawn. While PMA is the method that requires the highest number of calls, SP is the one that needs the least. Since comparing PMA to SP might not be fair, considering that one is a robust method and the other a reliability one, we can compare PMA to both SORA and SP+SORA and still find a huge difference in the total number of calls. This is once again a confirmation of the improvements (in terms of function evaluations) that SORA has over PMA. When comparing SORA to SP+SORA, it can be seen that the latter requires more function evaluations.

The results concerning the reliability error of each method, which was once again calculated with a post optimal analysis using the MC method with 10^6 samples, it can be seen in Figs.6.6 and 6.7. Figure (6.6) shows us the distribution of the reliability error across all test cases. Once again, the even numbered cases, which have 5 stochastic variables, seem to be the most problematic, as the reliability error tends to be a little higher. In two of these cases (14 and 18) some of the methods had difficulty converging and thus obtained a high reliability error. This behavior seems to indicate that a numerical error of unknown source might have occurred. These numerical errors may have to do with either the numerical robustness of the methods or with fact that surrogates are being employed. Surrogates are continuous functions that only have relevant information about the problem, between certain bounds. In case an iteration is close to these bounds, the stochastic optimization methods may try to evaluate the surrogate past these bounds, which could result in a numerical error.

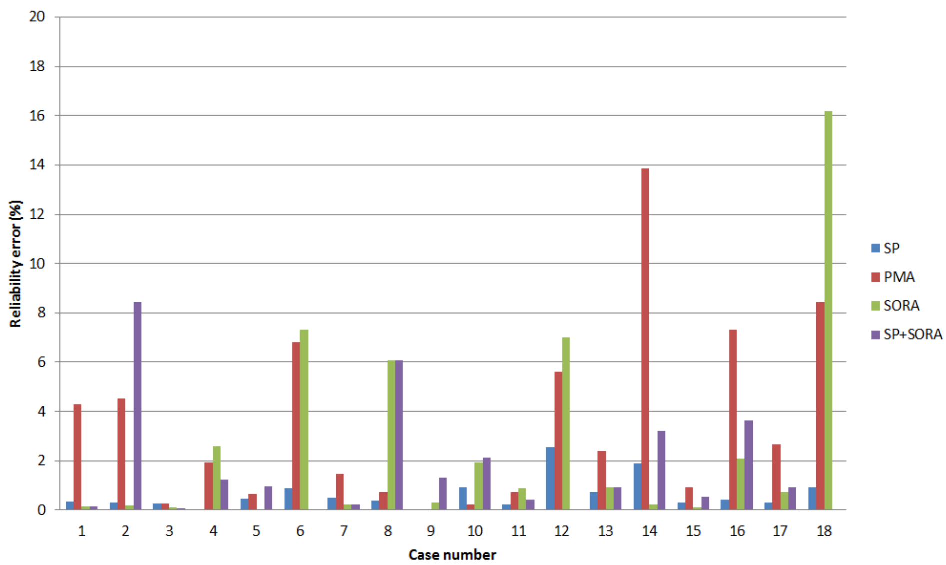


Figure 6.6: Test case 2 - Reliability error

Despite these problems, the overall results were acceptable, considering the higher number of design variables (both with and without uncertainty) when compared to the previous test case. This fact is proven in Fig.6.7, where the average of the error is presented for each method. Once again, both the top and bottom averages were obtained similarly to the ones of Fig.6.5.

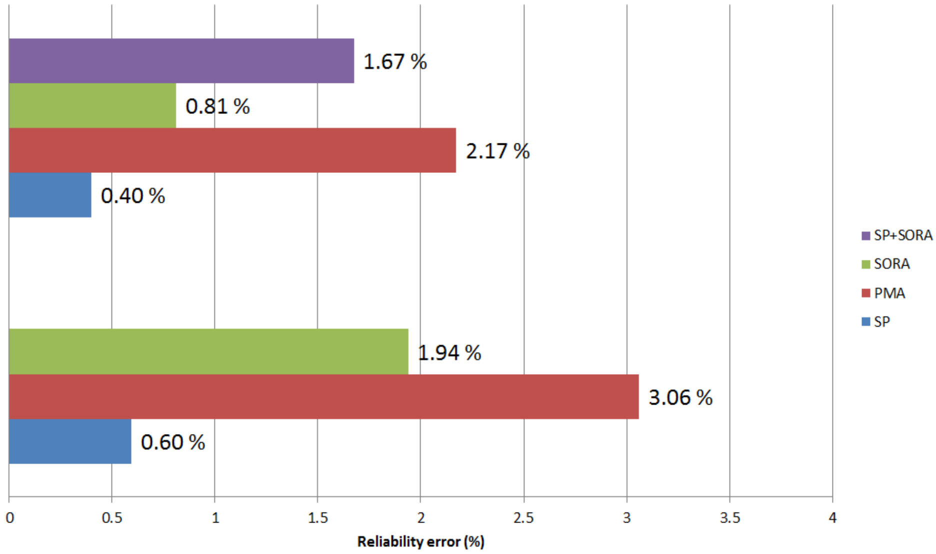


Figure 6.7: Test case 2 - Average reliability error

The first thing that should be noted is that, based on the information that has been gathered so far, by looking at the top average of Fig.6.7, it should be possible to assume that, had the method SP+SORA been able to converge for all the cases, its average reliability error should lie between SORA's and PMA's, in the bottom average. That being said, by comparing these values to the ones of the previous test case, it can be concluded that in this test case, all the methods present an higher reliability error. This can easily be explained by that fact that the number of variables both non stochastic and stochastic is higher in this test case, and because of the higher degree of non linearity that this constraint has.

Still, one thing is left unexplained. Figure (6.7) also shows a really interesting behavior of the PMA method, that is not explained by these facts. While the reliability error of the other three methods increase in a way that is consistent with the previous test case, PMA goes from being the method that had the lowest error, to the one that now has the highest. This can only be explained by the fact that PMA solely relies on the MPP problem to find the target reliability. Because some of the stochastic variables do not necessarily influence the reliability of the aircraft (they are not responsible for failures), but are still used for the reliability assessment of the PMA method, an error can be induced. PMA is usually able to deal with these problems, but since surrogate models were employed and some of the variables are usually close to their bounds, numerical errors occur, which sometimes lead to incorrect data regarding the influence of these variables. Even though it is not shown in any figure, the reliability results obtained with PMA are often below the target reliability, which confirms this theory. While both SORA and SP+SORA suffer from the same problem, they do not rely as much in this reliability assessment as PMA does. Because of that, their average error is lower. To solve a problem like this, a sensitivity analysis should be done prior to the optimization itself, and only the most critical variables should be included as stochastic design variables.

Despite the test case being more complex than the previous one, the SP method once again proved to be able to produce really good results, even for higher levels of uncertainty.

Finally, since unlike in the previous test case, the results of the objective function were considerably different for some of the methods, Fig.6.8 was included, where these differences can be seen.

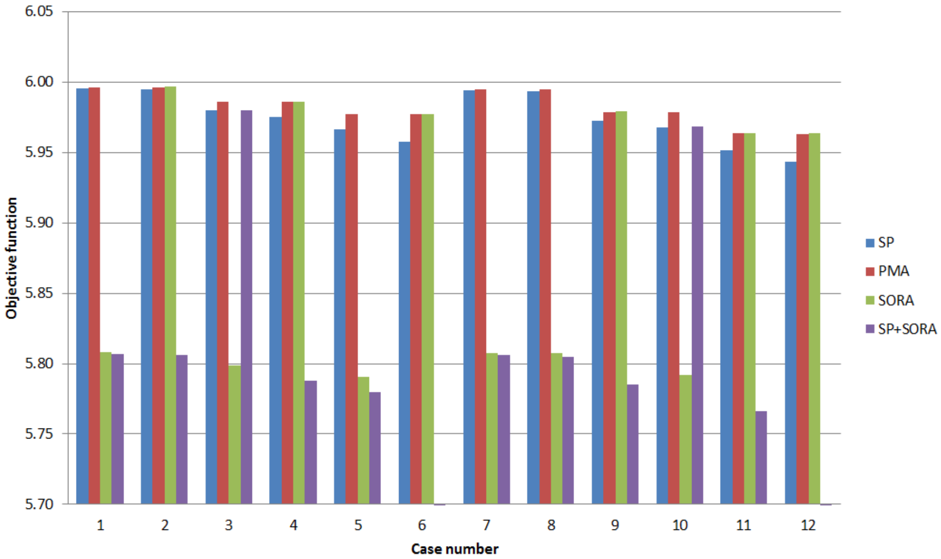


Figure 6.8: Test case 2 - Objective function results

Figure (6.8) shows that, for most cases, SP and PMA methods provide similar results to each other (with PMA's often being slightly higher because of the way the SP method uses to compute its objective function). As for SORA and SP+SORA, the same cannot be said. While both SP and PMA have always the highest results, SORA and SP+SORA sometimes converge to these same results and other times converge to values far below. It seems as if for this specific problem, using the FSQP algorithm in conjunction with the SORA method is not the best solution. It is hard to tell why this happens, but one of the reasons may be the fact that SORA's tolerances are not well calibrated for this problem. Even though both SORA and SORA+SP methods are working properly (this can be seen by their low reliability errors in 6.7), for some reason these methods have trouble finding the same optima as PMA and SP and often converge to "worse" global optima.

6.3.3 Optimal Solution

Since the constraint of this problem is not obtained through a simple equation, like the one of the previous test case, but is instead obtained using the stress surrogate, it is not possible to plot it as easily as it was in Figs.5.8 and 5.9. Instead, in this subsection, only a brief explanation as to how all the variables evolve, is given. This explanation is based on all the obtained results. Examples of these results are shown in Tabs.6.5 and 6.4.

Since the objective of this test case was to maximize $\frac{C_L}{C_D} ln \frac{W_1}{W_2}$, during a flight phase where normally not a lot of problems concerning the maximum stress tend to arise, one would assume that it simply meant maximizing C_L/C_D while decreasing the weight of the structure (which increases $ln \frac{W_1}{W_2}$). In fact, that is exactly what happens. The FSQP algorithm in conjunction with the different stochastic optimization methods, always try to find the combination of angles that provide the maximum C_L/C_D and then adapt the thicknesses so that the target reliability is met.

Everything happens as expected, but one thing. As stated before, Fig.6.2 shows us that t_w/c has considerably less influence in the maximum stress than t_s/t_{max}. This fact is evidenced throughout the optimizations since at all times t_w/c stays at its minimum (allowed by the surrogate). Only t_s/t_{max} changes, which means that for all the values (allowed by the surrogates) of t_w/c, no failure occurs, provided that t_s/t_{max} is large enough.

6.4 Comparison of Winglet Configurations

After the surrogate of configuration 5 had been optimized (in section 6.3) with different methods, levels of uncertainty numbers of random variables and target reliabilities, conclusions were drawn as to which method performed better, and how the different design variables tended to move. Because the whole process took so many optimizations be completed, and since there are in total 9 winglet configurations, it would be both difficult (in terms of time) and unnecessary to do the same for all of them.

In this section it was decided to follow a different approach, which consists of three different phases. First, since from subsection 6.3.3 it was concluded that the trend was for the optimizer to converge to the angles that provided the maximum C_L/C_D, those same angles were found, for each of the configurations and than converted into fixed parameters. After that, two methods were chosen based on their overall performance and used to optimize all the configurations. During these optimizations, which only had two design variables (the relative thicknesses), a target reliability of 3 is maintained and different levels of uncertainty are used. Finally, the results were compared and the most promising winglet configuration chosen.

6.4.1 Winglet angles for maximum C_L/C_D

Knowing beforehand that the optimizer always tries to find the winglet angles that produce the highest C_L/C_D, these values were found in a separate problem, to reduce the complexity of the optimization problem. To obtain them, 1331 angle combinations were generated for each configuration and then the best one was found. The results can be seen in Tabs.6.6 and 6.7.

Config #	1	2	3	4	5	6	7	8	9
Span (m)	0.5	1.75	3	0.5	1.75	3	0.5	1.75	3
Tip chord (m)	0.7573	0.7573	0.7573	1.242	1.242	1.242	1.7266	1.7266	1.7266
Toe (°)	5	-10	-10	5	5	5	5	5	5
Cant (°)	0	80	80	0	0	0	0	0	0
Sweep (°)	40	60	20	40	20	20	40	20	20
C_L	0.294	0.296	0.296	0.297	0.3	0.3010	0.2987	0.3032	0.3059
C_D	0.0119	0.0119	0.0118	0.0120	0.0119	0.01194	0.0119	0.0121	0.0121
$\frac{C_L}{C_D}$	24.8	24.9	25	24.8	25.2	25.21	25.07	25.13	25.20

Table 6.6: Test case 2 - Winglet configurations - angles and aerodynamic coefficients

In Tab.6.6 it is possible to see both the geometric parameters that differentiate each individual winglet configuration and the respective angles that maximize C_L/C_D. It can also be seen that, for different configurations, there are also different optimum angle combinations (for certain cases, this combination is the same though).

In terms of the aerodynamic coefficients, they are also shown in Tab.6.6 but, in order to better understand how C_L/C_D varies with both span and tip chord, Tab.6.7 is presented.

C_L/C_D			
Span\Tip chord	0.7573	1.242	1.7266
0.5	24.8	24.9	25
1.75	24.8	25.2	25.21
3	25.07	25.13	25.20

Table 6.7: Test case 2 - Winglet geometric combinations vs C_L/C_D

As can be seen, the behavior of C_L/C_D is not linear with neither of the geometric parameters. While it is true that (within the surrogate's boundaries) C_L/C_D grows (in a non linear fashion) with the increase of tip chord, the same cannot be said about the span. With an increase in the span, C_L/C_D grows up until a certain point and then decreases.

It is important to note that since the aircraft is not trimmed, in this test case, the obtained values of C_L, C_D and C_L/C_D may not be correct. Nevertheless, since all the same simplifications were used for each of the configurations, these values may be used (qualitatively) for comparison purposes between the different winglets.

With the knowledge obtained in this subsection alone, it is not yet possible to draw conclusions about which configuration is the best for this specific problem. It remains to be seen, whether the increase of C_L/C_D of some of the configurations, is enough to compensate the probable weight increase, in order to maintain a certain reliability.

6.4.2 Configuration Optimization

This brings us to the next phase of test case 2. In this part, there are two main objectives. One is to compare the performance of the two chosen methods, throughout the optimization of all the configurations. The other one is to see whether after the optimization of the thicknesses, configuration 6 is still the one that has the maximum objective function (this configuration was the one that showed in Tab.6.6 to have the highest C_L/C_D of all).

Since there are too many configurations, some simplifications to the problem had to be made. Besides the fact that only two design variables were considered, the target reliability was kept equal to 3 at all times and only three methods were used (one deterministic and two stochastic). Based on the overall performance of all the methods in this thesis, the SP and PMA methods were chosen to be used in this part of the test case. The SP method was chosen for its ability to provide very good results at lower computational costs. As for PMA, it was chosen because it often converges to similar results to SP, and because it produces good results in terms of reliability as well, provided that there is not a large number of non-critical design variables.

After all the optimizations had been completed, some things that were not exactly expected happened. The first one was the inability of the PMA method to achieve the target reliability, while optimizing configuration 1 with 1% $c.o.v.$. This probably happened because of some numerical error that may have had to do with the MPP or the boundaries of the surrogates. It was just one case, but this proves that although the method provides good results, sometimes it cannot converge to the optimum. The other thing that happened (this one more than once) was the same inability to achieve the target reliability but this time by the SP method. It is known that robust methods are not the best when it comes to ensuring that a certain reliability is met. Nevertheless, in the previous tests, SP proved to be more than able to do so. In this case, however, the same did not happen for configurations 1 to 4. Somehow for these configurations, the SP method was not conservative enough and the reliabilities were lower than the target. This proves how for different cases, these methods can have different performances.

Since it is not practical to present all the results that were obtained, only the deterministic (with a Safety Factor of 1.5) and PMA ones (with $c.o.v.$ 5% and 8%) were chosen to be shown, as an example, in Tab.6.8. By looking at this table, some conclusions can be drawn.

In terms of the differences between the deterministic and stochastic optimizations, it can be seen that, unlike in the previous test case, the deterministic objective function is lower than the stochastic one. Despite uncertainty being considered in the stochastic methods, this happens because of the more conservative Safety Factor that is used in the deterministic optimization. This is a clear advantage that the stochastic methods are known to have over the deterministic ones. Provided that the uncertainty is well quantified, performances can really be improved.

Another thing that is interesting about Tab.6.8 is the fact that, for different levels of uncertainty, the optimizations provide us with different "best" configurations. Even though configuration 6 has higher C_L/C_D when compared to configuration 5, it is configuration 5 that has a slightly higher objective function in the cases of $c.o.v.$ 5% and $c.o.v.$ 8%. This happens because the lower weight of configuration 5 has a bigger influence in the objective function than the slightly higher C_L/C_D of configuration 6. However, in the deterministic optimization case, where there is no uncertainty, the opposite happens. This just serves to emphasize the differences between deterministic and stochastic optimization. As uncertainty grows, the "best" configuration changes from configuration 6 to 5.

In terms of the evolution of the design variables (thicknesses) with the increase of uncertainty, the results were all pretty straightforward. While one of the thicknesses remained practically unaltered throughout the whole process (t_w/c), as its minimum did not cause any failures, the other one (t_s/t_{max}) increased with the uncertainty, in order to maintain the target reliability.

The comparison between the weights and the values of the objective function, that were obtained with deterministic optimization and PMA ($c.o.v.$ 5% and 8%) is shown in Fig.6.9. There are two things that must be noted before analyzing this figure. First, the results belong to configuration 5, and second, the results shown are a percentage of the maximum obtained for all the configurations and $c.o.v.$ of 5% and 8% (of both weight and objective function). As can be seen, the deterministic optimization resulted in the heavier winglet (more conservative), to which corresponds the lowest objective function. In terms of stochastic optimization, since one of the thicknesses increases with uncertainty, the higher it gets, the

Config #		1	2	3	4	5	6	7	8	9
Deterministic	Objective	5.8661	5.8989	5.9197	5.8655	5.9476	5.9479	5.9170	5.9212	5.8098
	$\frac{t_w}{c}$	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	3.60E-03
	$\frac{t_s}{t_{max}}$	8.164E-03	8.287E-03	8.303E-03	8.496E-03	8.747E-03	8.850E-03	8.724E-03	9.179E-03	9.200E-03
	Weight (N)	7.282E+05	7.285E+05	7.285E+05	7.290E+05	7.297E+05	7.300E+05	7.297E+05	7.308E+05	7.451E+05
PMA ($c.o.v.$ 5%)	Objective	5.9074	5.9426	5.9624	5.9131	5.9967	5.9954	5.9659	5.9627	5.9704
	$\frac{t_w}{c}$	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04
	$\frac{t_s}{t_{max}}$	6.135E-03	6.153E-03	6.227E-03	6.164E-03	6.379E-03	6.565E-03	6.358E-03	7.174E-03	7.558E-03
	Weight (N)	7.237E+05	7.237E+05	7.239E+05	7.238E+05	7.244E+05	7.248E+05	7.244E+05	7.263E+05	7.272E+05
	β	3.0041	3.0226	3.2351	3.0059	3.0036	3.0032	3.0243	3.0372	3.0075
PMA ($c.o.v.$ 8%)	Objective	5.8926	5.9278	5.9490	5.8981	5.9809	5.9792	5.9504	5.9454	5.9517
	$\frac{t_w}{c}$	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04	5.40E-04
	$\frac{t_s}{t_{max}}$	6.860E-03	6.873E-03	6.874E-03	6.896E-03	7.135E-03	7.341E-03	7.103E-03	8.005E-03	8.454E-03
	Weight (N)	7.253E+05	7.254E+05	7.254E+05	7.255E+05	7.261E+05	7.266E+05	7.260E+05	7.281E+05	7.293E+05
	β	3.002	2.9991	3.0052	2.9933	2.9818	2.976	2.9696	2.9813	3.0036

Table 6.8: Test case 2 - Optimization results for the different configurations

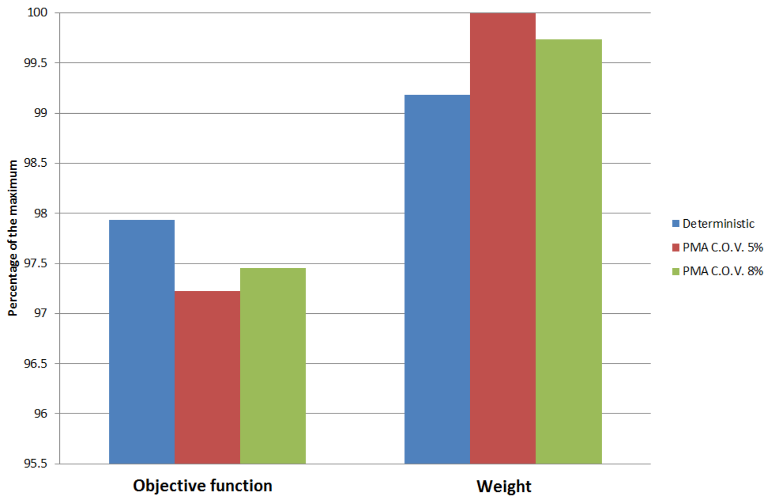


Figure 6.9: Test case 2 - Evolution of the results

heavier the winglet (still lighter than the deterministic winglet for $c.o.v.$ 8%) and the lower the objective (still higher than the deterministic objective function for $c.o.v.$ 8%).

6.4.3 Winglet Performance Comparison

Finally, with all the results obtained, it is now possible to compare all the winglet configurations. The different geometric parameters and angle combinations, result in different aerodynamic forces that require different thicknesses to achieve the target reliability. These thicknesses in turn result in different weights and, when all things are considered, different objective function results are obtained. All of this is showed in Fig.6.10. In this figure, the results that are shown were obtained with PMA and, like in Fig.6.9, the

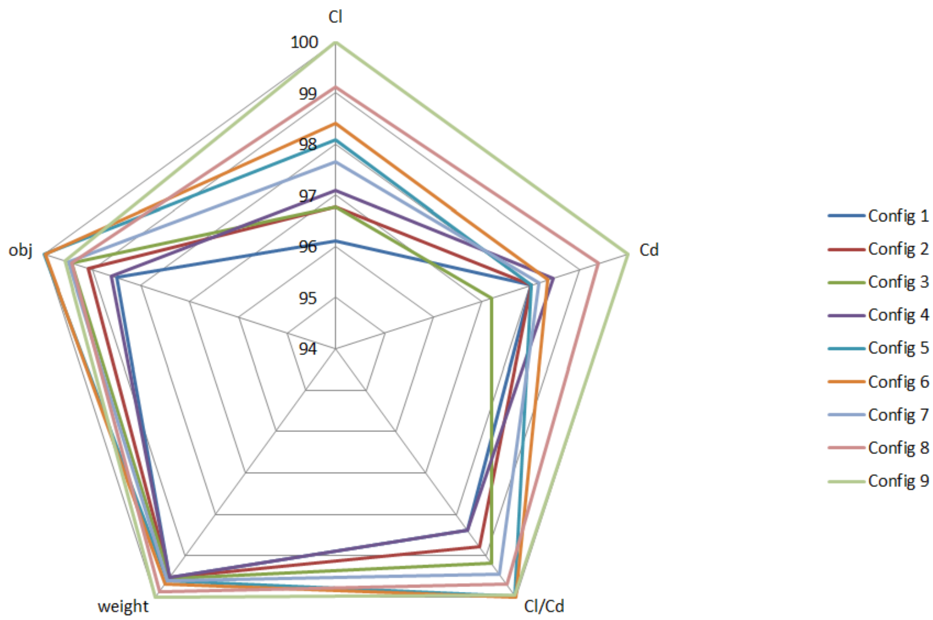


Figure 6.10: Test case 2 - Winglet comparison chart

values of each category are also a percentage of the maximum obtained, in that same category, among all configurations, while using a $c.o.v.$ of 5%.

It is possible to see that every winglet configuration is unique, not only geometrically but also in the results they produce. Although not clearly, it can also be seen that two configurations stand out from the rest. Configurations 5 and 6 have similar objective function results (higher than all the other configurations). Even though they are not the ones that produce the higher aerodynamic forces, the combination of the other factors make them the ”best” winglet configurations for this specific problem.

Chapter 7

Conclusions

7.1 Achievements

The main objective of this thesis was to assess the performance of the current stochastic optimization methods, particularly in aircraft design problems.

Some of the most common and promising methods were chosen among those that are used across the different stochastic optimization formulations (RDO, RBDO and the hybrid R^2BDO). Depending on the formulations, the methods had different objectives, were based on different principles and had different implementation complexities.

After using an analytical sample problem to test these methods and formulations, some conclusions were drawn as to their suitability and efficiency. Methods wise, in the RDO category the SP method proved to be superior to MM in terms of accuracy and to MC in terms of computational cost. In RBDO, even though PMA confirmed to be numerically more robust than both RIA and MC, it also demonstrated to be computationally more expensive, when compared to RDO approaches. It was also verified that methods like the alternative PMA, SORA and RDS were all able to maintain the same accuracy and numerical robustness as PMA, but at lower computational cost. As for the formulations, RDO struggled to keep up with both RBDO and R^2BDO when it came to obtaining a target reliability for the design. Even though RDO obtained some good results for lower uncertainty levels, as these got higher, RDO started obtaining results that were too conservative.

Some of these methods were chosen to be tested in two more realistic aircraft design problems as well. This way, not only it would be possible to test their performances in more realistic situations, but also to see whether using them in spite of deterministic methods had any advantages at all. The methods were chosen based on the stochastic optimization formulation, performance and implementation complexity. The chosen methods ended up being SP (RDO), PMA (RBDO), SORA (RBDO) and SP+SORA (R^2BDO).

The first test case consisted of a cruise optimization problem for maximum aircraft range. The main focus was on assessing each method performance, where several cases comprising different levels of uncertainty, target reliabilities and number of random variables were tested. Surrogate models were used to reduce the overall computational cost. For the most part, the conclusions drawn from this test case were all in accordance to those of the analytical problem, both in terms of function evaluations and reliability errors. Yet, one thing that was not expected was the fact that SP was able to achieve good reliability results despite the high uncertainty level. Other thing worth mentioning was the fact that for this problem, the accuracy of both SORA and SORA+SP was slightly worse than PMA's.

In the second test case, where baseline+winglet optimization was performed, in terms of function evaluations, everything stayed the same compared to the first test case, with SP being the method that required the least and PMA the most. As for the error, while SP once again proved to be a good alternative to the more computationally expensive RBDO methods, PMA struggled because of some numerical errors due to the MPP problem being solved too close to the surrogate bounds. Another thing worth mentioning is the fact that, in this test case, not only stochastic optimization proved to obtain better results than deterministic optimization, but it also showed how accounting for uncertainties in the optimization process can lead to the discovery of some novel more efficient aircraft configurations.

To sum up, the efficiency of each method is tied to the problem being solved. There is no such thing as the best method or best formulation, since every method and formulation performed poorly in at least one test case. Despite that, the method that stood out the most was SP, by being able to accurately achieve the reliability target, at reasonable cost. Its robust formulation did not allow it to perform better in the analytical test case but in the end, it proved to be more than capable to solve simpler problem, thus confirming why robust optimization is still widely used when uncertainties are taken into account.

It was also seen that stochastic optimization has some advantages over deterministic optimization. Not only stochastic optimization proved to be less conservative (while still achieving the target reliabilities), but it also demonstrated its potential when it comes to finding new and more efficient aircraft configurations. It should be noticed that, because the uncertainties were not properly quantified, the comparisons were only qualitative.

7.2 Future Work

Despite all the work that has been done in this thesis, there are still a few things that are worth being further studied. Since the numerical test cases only considered target reliabilities, a study that focus more on the actual robustness of the design points obtained with RDO and R^{2}BDO needs to be conducted. Also, either more complex surrogates should be created or the actual disciplinary modules should be used, so that these methods can be applied to even more complex problems so that one can assess how the methods perform when there are more constraints and more complicated objective functions. For example, morphing problems including several flight phases could be studied, as a means to find out the different configurations that stochastic optimization would produce and compare them to the deterministic optimal solution.

Finally, it would also be interesting to be able to test stochastic optimization in a case where uncertainties were well quantified, so that a quantitative comparison could be made with the results of deterministic optimization.

Bibliography

H. Agarwal. *Reliability based design optimization: formulations and methodologies*. PhD thesis, University of Notre Dame, Notre Dame, Indiana, USA, 2004.

M. Allen and K. Maute. Reliability-based design optimization of aeroelastic structures. *Structural and Multidisciplinary Optimization*, 27:228–242, 2004.

ANSYS, Inc. ANSYS Mechanical. www.ansys.com, 2010. Release 12.0.

L. F. Bertuccelli and J. P. How. Robust Markov decision processes using sigma point sampling. In *American Control Conference*, Seattle, WA, USA, 2008.

U. Cella and D. G. Romano. Assessment of optimization algorithms for winglet design: design optimization of 3D winglets lead to performance improvement of a piaggio aero business jet. In *International CAE Conference*, 2012.

Z. Chen, H. Qiu, L. Gao, L. Su, and P. Li. An adaptive decoupling approach for reliability-based design optimization. *Computers & Structures*, 117:58–66, February 2013.

K. Chiba, S. Obayashi, and K. Nakahashi. High-fidelity multidisciplinary design optimization of aerostructural wing shape for regional jet. In *23rd AIAA Applied Aerodynamics Conference*, 2005.

E. de Rocquigny. Structural reliability under monotony: properties of FORM, simulation or response surface methods and a new class of monotonous reliability methods (MRM). *Structural Safety*, 31: 363–374, 2009.

X. Du. *First Order and Second Reliability Methods*, chapter 7. Missouri University of Science and Technology, 2005. Course notes ME 360 - Probabilistic Engineering Design.

X. Du and W. Chen. Sequential optimization and reliability assessment method for efficient probabilistic design. In *Probabilistic Design," ASME Design Engineering Technical Conferences*, pages 225–233, 2002.

X. Du and W. Chen. Collaborative reliability analysis for multidisciplinary systems design. *Optim. Eng.*, 6:63–84, 2005.

D. M. Frangopol and K. Maute. Life-cycle reliability-based optimization of civil and aerospace structures. *Computers and Structure*, 81:397–410, 2003.

D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing, 1989.

C. R. Gumbert, G. J.-W. Hou, and P. A. Newman. High-fidelity computational optimization for 3-D flexible wings: Part I - simultaneous aero-structural design optimization (SASDO). *Optimization and Engineering*, 6(1):117–138, 2005a.

C. R. Gumbert, P. A. Newman, and G. J.-W. Hou. High-fidelity computational optimization for 3-D flexible wings: Part II - effect of random geometric uncertainty on design. *Optimization and Engineering*, 6(1): 139–156, 2005b.

B. Hou, W. Wang, S. Li, Z. Lin, and Z. C. Xia. Stochastic analysis and robust optimization for a deck lid inner panel stamping. *Materials and Design*, 31(3):1191–1199, 2010.

H.-Z. Huang, X. Zhang, Y. Liu, D. Meng, and Z. Wang. Enhanced sequential optimization and reliability assessment for reliability-based design optimization. *Journal of Mechanical Science and Technology*, 26(7):2039–2043, 2012a.

M. F. Huang, C. M. Chan, and W. J. Lou. Optimal performance-based design of wind sensitive tall buildings considering uncertainties. *Computers & Structures*, 98-99:7–16, 2012b.

L. Huyse. Solving problems of optimization under uncertainty as statistical decision problems. In *AIAA-2001-1519*, 2001.

P. W. Jansen, R. E. Perez, and J. R. R. A. Martins. Aerostructural optimization of nonplanar lifting surfaces. *Journal of Aircraft*, 47(5):1490–1503, 2010.

M. Kaufmann, D. Zenkert, and P. Wennhage. Integrated cost/weight optimization of aircraft structures. *Structural and Multidisciplinary Optimization*, 41(2):325–334, 2010.

P. N. Koch, R.-J. Yang, and L. Gu. Design for six sigma through robust optimization. *Structural and Multidisciplinary Optimization*, 26:235–248, 2004.

C. T. Lawrence and A. L. Tits. A computationally efficient feasible sequential quadratic programming algorithm. *SIAM Journal on Optimization*, 11:1092–1118, 2001.

F. Li, T. Wu, A. Badiru, M. Hu, and S. Soni. A single-loop deterministic method for reliability-based design optimization. *Engineering Optimization*, 45(4):435–458, 2013.

R. P. Liem, G. K. W. Kenway, and J. R. R. A. Martins. Multi-point, multi-mission, high-fidelity aerostructural optimization of a long-range aircraft configuration. In *14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Indianapolis, Indiana, USA, 2012.

S. N. Lophaven, H. B. Nielsen, and J. Søndergaard. DACE – a matlab kriging toolbox – version 2.0, 2002.

A. C. Marta. Multidisciplinary design optimization of aircrafts. Course notes, IST, 2013. 754 pages.

J. R. R. A. Martins, J. J. Alonso, and J. J. Reuther. High-fidelity aerostructural design optimization of a supersonic business jet. *Journal of Aircraft*, 41(3):523–530, 2004.

F. Mastroddi and S. Gemma. Analysis of pareto frontiers for multidisciplinary design optimization of aircraft. *Aerospace Science and Technology*, 28(1):40–55, 2013.

D. N. Mavris and O. Bandte. A probabilistic approach to multivariate constrained robust design simulation, 1997.

M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21:239–245, 1979.

M. D. Mckay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 42(1):55–61, 2000.

W. Meng and D. Ma. Robust design optimization method for aircraft. In *3rd International Symposium on Systems and Control in Aeronautics and Astronautics (ISSCAA)*, Harbin, China, 2010.

M. Menshikova. *Uncertainty Estimation Using the Moments Method Facilitated by Automatic Differentiation in Matlab*. PhD thesis, Cranfield University, Cranfield, Bedfordshire, England, 2010.

R. H. Myers and D. C. Montgomery. *Response Surface Methodology: Process and Product Optimization Using Designed Experiments*. Wiley, 1995.

T. H. Nguyen, J. Song, and G. H. Paulino. Single-loop system reliability-based design optimization using matrix-based system reliability method: Theory and applications. *ASME Journal of Mechanical Design*, 132, 2010.

H. Niederreiter. Quasi-Monte Carlo methods and pseudo - random numbers. *Bulletin of the American Mathematical Society*, 84(6):957–1041, 1978.

NOVEMOR. NOvel Air VEhicle Configurations: From Fluttering Wings to MORphing Flight. `http://www.novemor.eu`, 2013. Collaborative research project under the EU 7_{th} Framework Program.

D. Padmanabhan, H. Agarwal, J. E. Renaud, and S. M. Batill. A study using Monte Carlo simulation for failure probability calculation in reliability-based optimization. *Optimization and Engineering*, 7: 297–316, 2006.

M. Padulo, M. S. Campobasso, and M. D. Guenov. Comparative analysis of uncertainty propagation methods for robust engineering design. In *Proceedings of ICED 2007*, 2007.

M. Padulo, S. A. Forth, and M. D. Guenov. *Robust Aircraft Conceptual Design using Automatic Differentiation in Matlab*, pages 271–280. Springer, 2008.

R. M. Paiva. *A Robust and Reliability-Based Optimization Framework for Conceptual Aircraft Wing Design*. PhD thesis, University of Victoria, Victoria, British Columbia, Canada, 2010.

S. N. Patnaik, S. S. Pai, and R. M. Coroneos. Reliability based design for a raked wing tip of an airframe. In *2nd International Conference on Engineering Optimization*, 2010.

N. V. Queipo, R. T. Haftka, W. Shyy, T. Goel, R. Vaidyanathan, and P. K. Tucker. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41:1–28, 2005.

S. S. Rao. *Engineering optimization : theory and practice*. John Wiley & Sons, 2009.

S. Shan and G. G. Wang. Reliable design space and complete single-loop reliability-based design optimization. *Reliability Engineering & System Safety*, 93:1218–1230, 2008.

S. M. J. Spence and M. Gioffre. Large scale reliability-based design optimization of wind excited tall buildings. *Probabilistic Engineering Mechanics*, 28:206–215, 2012.

A. Suleman, F. Lau, J. Vale, and F. Afonso. Multidisciplinary performance based optimization of morphing aircraft. In *SciTech (Accepted)*. AIAA, 2014.

K. Svanberg. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM Journal on Optimization*, pages 555–573, 2002.

J. Tu, K. K. Choi, and Y. H. Park. A new study on reliability-based design optimization. *ASME Journal of Mechanical Design*, 121(4):557–564, 1999.

H. Wang. An efficient dimension-adaptive uncertainty propagation approach. *Applied Mathematics and Computation*, 218(7):3230–3237, 2011.

S. F. Wojtkiewicz, M. S. Eldred, R. V. Field, A. Urbina, and J. R. Red-horse. Uncertainty quantification in large computational engineering models. In *In Proceedings of the 42rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, number AIAA-2001-1455*, Seattle, Washington, USA, 2001.

F. Xiong, B. Xue, Z. Yan, and S. Yang. Polynomial chaos expansion based robust design optimization. In *International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (ICQR2MSE)*, Xi'an, China, 2011.

H. Xu and S. Rahman. A generalized dimension-reduction method for multidimensional integration in stochastic mechanics. *International Journal for Numerical Methods in Engineering*, 61(12): 1992–2019, 2004.

O. P. Yadav, S. S. Bhamareb, and A. Rathore. Reliability-based robust design optimization: a multi-objective framework using hybrid quality loss function. *Quality and Reliability Engineering International*, 26:27–41, 2010.

W. Yao, X. Chen, W. Luo, M. van Tooren, and J. Guo. Analysis of pareto frontiers for multidisciplinary design optimization of aircraft. *Progress in Aerospace Sciences*, 47:450–479, 2011.

X. Yin and W. Chen. Enhanced sequential optimization and reliability assessment method for probabilistic optimization with varying design variance. *Journal of Mechanical Science and Technology*, 2(3): 261 – 275, 2006.

H. Yu, F. Gillot, and M. Ichchou. Reliability based robust design optimization for tuned mass damper in passive vibration control of deterministic/uncertain structures. *Journal of Sound and Vibration*, 332: 2222–2238, 2013.

X. Yu and X. Du. Reliability-based multidisciplinary optimization for aircraft wing design. *Structure and Infrastructure Engineering: Maintenance, Management, Life-Cycle Design and Performance*, 2:277–289, 2006.

T. A. Zang, M. J. Hemsch, M. W. Hilburger, S. P. Kenny, J. M. Luckring, P. Maghami, S. L. Padula, and W. J. Stroud. Needs and opportunities for uncertainty - based multidisciplinary design methods for aerospace vehicles. Technical report, National Aeronautics and Space Administration, 2002.