A Robust and Reliability-Based Optimization Framework for Conceptual Aircraft Wing Design

by

Ricardo Miguel Paiva
B.Sc., Instituto Superior Técnico, Lisbon, Portugal, 2005
M.Sc., University of Victoria, Victoria, B.C., Canada, 2007

A Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Mechanical Engineering

A Robust and Reliability-Based Optimization Framework for Conceptual Aircraft Wing Design

by

Ricardo Miguel Paiva
B.Sc., Instituto Superior Técnico, Lisbon, Portugal, 2005
M.Sc., University of Victoria, Victoria, B.C., Canada, 2007

Supervisory Committee

Dr. Afzal Suleman, Supervisor
(Department of Mechanical Engineering)

Dr. Curran Crawford, Supervisor
(Department of Mechanical Engineering)

Dr. Zuomin Dong, Departmental Member
(Department of Mechanical Engineering)

Dr. Wu-Sheng Lu, Outside Member
(Department of Electrical Engineering)

Supervisory Committee

Dr. Afzal Suleman, Supervisor
(Department of Mechanical Engineering)

Dr. Curran Crawford, Supervisor
(Department of Mechanical Engineering)

Dr. Zuomin Dong, Departmental Member
(Department of Mechanical Engineering)

Dr. Wu-Sheng Lu, Outside Member
(Department of Electrical Engineering)

ABSTRACT

A robustness and reliability based multidisciplinary analysis and optimization framework for aircraft design is presented. Robust design optimization and Reliability Based Design Optimization are merged into a unified formulation which streamlines the setup of optimization problems and aims at preventing foreseeable implementation issues in uncertainty based design.

Surrogate models are evaluated to circumvent the intensive computations resulting from using direct evaluation in nondeterministic optimization. Three types of models are implemented in the framework: quadratic interpolation, regression Kriging and artificial neural networks. Regression Kriging presents the best compromise between performance and accuracy in deterministic wing design problems.

The performance of the simultaneous implementation of robustness and reliability is evaluated using simple analytic problems and more complex wing design problems, revealing that performance benefits can still be achieved while satisfying probabilistic constraints rather than the simpler (and not as computationally intensive) robust constraints. The latter are proven to to be unable to follow a reliability constraint as uncertainty in the input variables increases. The computational effort of the reliability analysis is further reduced through the implementation of a coordinate change in the respective optimization sub-problem.

The computational tool developed is a stand-alone application and it presents a user-friendly graphical user interface. The multidisciplinary analysis and design optimization tool includes modules for aerodynamics, structural, aeroelastic and cost analysis, that can be used either individually or coupled.

Contents

List of Tables

List of Figures

Acronynms

ADELINE	Adaptive Linear Element
AMV	Advanced Mean Value
ANN	Artificial Neural Networks
AOA	Angle of Attack
BLISS	Bi-Level Integrated System Synthesis
BWB	Blended Wing Body
CAD	Computer Assisted Design
CDF	Cumulative Density Function
CFD	Computational Fluid Dynamics
CO	Collaborative Optimization
DV	Design Variable
EPM	Equivalent Plate Model
FEM	Finite Element Model
FORM	First Order Reliability Method
GUI	Graphical User Interface
HSCT	High Speed Civil Transport
KKT	Karush Kuhn Tucker
LE	Leading Edge

MAC	Mean Aerodynamic Chord
MC	Monte Carlo
MDF	Multidisciplinary Feasible
MV	Mean Value
MDO	Multidisciplinary Design Optimization
MM	Method of Moments
MPP	Most Probable Point
NASA	National Aeronautics and Space Administration
PDF	Probability Density Function
PMA	Performance Measure Approach
RANS	Reynolds Averaged Navier-Stokes
RBDO	Reliability Based Design Optimization
RDO	Robust Design Optimization
RIA	Reliability Index Approach
R^{2}BDO	Robust and Reliability Based Design Optimization
SORM	Second Order Reliability Method
SP	Sigma Point
SQP	Sequential Quadratic Programming
TE	Trailing Edge
VLM	Vortex Lattice Method

ACKNOWLEDGEMENTS

First and foremost, and as is usual in these proceedings, I thank my family for always being there and for providing me with much needed support over the course of the past few years.

I am grateful to my mentors, Professor Afzal Suleman and Dr. Curran Crawford. Their encouragement and support during the course of the research and their input to improve the thesis document have been invaluable.

Many friends and fellow students have helped me along the way. I would like to thank my good friend André Carvalho for is invaluable help in programming issues. Also, my thanks to Luís Felix with whom I have developed and hope to keep developing software in the future.

Everyone in our research group would agree that Sandra Makosinski helped us maintain our sanity over the years by calling our attention to a life outside research. Nevertheless, her professionalism and promptness as the group's secretary are to be admired.

A special thanks goes to my friends and fellow graduate students Baris Ulutas, Bruno Rocha, Carlos Silva, Casey Keulen, Craig Bakker, Flavio Firmani, Jenner Richards, Joana Rocha, and Kerem Karakoc with whom I shared the hardships and thrills of research work. I would also like to acknowledge Pedro Gambôa for supplying the updated equivalent plate model code, and Daniel Filković for our discussions on panel methods. My thanks to Antonia Llobera, Federico Cutillas, Grzegorz Kawiecki, Yasser Essa, and all the current and former personnel at Aernnnova, that in one way or the other contributed with ideas for this project.

Finally I give my thanks to *Fundação para a Ciência e Tecnologia* for funding my doctoral studies in Canada under grant SFRH/BD/27863/2006.

DEDICATION

To my family.

Chapter 1

Introduction

1.1 Background

The increasing competitiveness in the aerospace industry has manufacturers searching for designs that are robust, in the sense that they still perform well in off design conditions - Robust Design Optimization (RDO), as well as reliable, in the sense that they present a low probability of failure. Moreover, in early stages of the design, many parameters are yet unknown or poorly characterized. The classical approach to structural design employing safety factors has frequently proved to be overly conservative, thus leaving room for improvement and achieving an edge over competitors. At times, however, it may also reveal itself to be too optimistic, as shown in the work of Sues *et al.* [1]. The implementation of Reliability Based Design Optimization (RBDO) techniques that allow the attainment of a specific probability of failure is therefore required in these situations. Additionally, in the conceptual to preliminary phases of aircraft design, the use of lower fidelity analysis tools because of their quick turnaround times means the resulting performance predictions present an appreciable degree of uncertainty. If robust optima are found at these stages, they are more likely to see their performance unaltered when re-evaluated using higher fidelity codes.

Additional concerns regarding aircraft usage and operations that also motivate the use of non-deterministic analysis during the design process include the following:

Unexpected events - turbulence, incidents during ground operations, in-flight collision with foreign objects lead to undesirable deformation of structures;

Payload estimation/fuel predictions - to be made shortly before the flight and may be vastly different from optimal design values;

Adverse weather, loiter times and flight altitudes set by air traffic control - significantly altering the mission profile/flight conditions an aircraft is optimized for;

Increasing costs of maintenance and downtime for inspection - to be avoided by producing more reliable structures/components from the onset.

In order to endeavor in RBDO, the otherwise deterministic optimization problems have to be augmented with a reliability analysis subproblem (more on this topic in section 2.3.2 of this thesis). Advanced reliability analysis techniques have been incorporated in computer programs such as CALREL [2], STRUREL [3], PROBAN [4] and RELSYS [5], while an example of the state of the art in RBDO tools is represented by ProFES [6], which can be coupled with commercial tools for Finite Element Analysis (FEA), for instance.

Other work in this area has been focused on the design for reliability of aircraft wings and stabilizers, while taking into account the uncertainty about allowable stresses in the materials [7]. In [8], a moment-based RBDO architecture is implemented, where the authors recognize the intrinsic shortcomings in trying to approximate the output of a probabilistic function directly (a topic that is closely related to the use of probabilistic objectives in RBDO that is further discussed in Chapter 2).

One of the main challenges in nondeterministic analysis and optimization is the additional computational effort required. Despite the advancements made in computer technology, it is a fact that some computational problems still remain quite cumbersome to solve and these difficulties are further accentuated when trying to solve them in a single or multidisciplinary design optimization environment where robustness/reliability analysis techniques are used. In comparison to the equivalent deterministic optimization problem, the solution of a RDO/RBDO problem easily requires a number of evaluations of objective and constraints that is one or even two orders of magnitude above. By replacing the computationally expensive analyzers with suitable surrogate models/metamodels, the optimization process can be greatly accelerated even if at the expense of a loss in fidelity [9]. An example of use of surrogate models in RDO is the case of the work of Bonte et al. on the optimization of metal forming processes [10].

Several applications of surrogates in aircraft design optimization problems are present in the literature. For instance, in a multiple fidelity approach, Giunta applied both polynomial models and Kriging to the optimization of a High Speed Civil Transport (HSCT) design [11]. Kriging metamodels, coupled with genetic algorithms, have also seen use when solving multiobjective and multidisciplinary optimization problems, as in the work of Kanazaki et al. and Kumano et al.[12, 13]. In a rare comparison of such different types of metamodels, Matias et al. used Kriging and ANN and suggested that Kriging has the advantage of not requiring as much tweaking as neural networks [14]. Willmes et al. go even further, comparing these approximations with standard use of evolutionary algorithms, effectively finding no advantages in their application to simple test problems [15].

In light of this evidence, one of the main objectives of the current work is to assess the benefits (or lack thereof) in using these approximations in conjunction with a Multidisciplinary Design Optimization (MDO) tool tailored for the conceptual design of aircraft wings. Therefore, an evaluation of some well known methods to create such surrogate models is performed, with applications based on simple aircraft design problems. The three approximation models implemented in this work are quadratic interpolation, Kriging and Artificial Neural Networks (ANN), which are covered in more detail in chapter 3. The modular nature of this tool, which is described in further detail in chapters 5 and 6, facilitates the integration of the code required for building the surrogate models. The result is an integrated approach which is coupled with an adaptive sampling strategy.

1.2 Optimization in Aircraft Design

The motivation behind aircraft design optimization has never been more clear. The constant fluctuations (normally upwards) of fuel prices and the demand for green, efficient aircraft cannot be ignored, and evidence of this fact has been put forward by NASA's Fundamental Aeronautics Program. At the end of this program, four industry teams presented reports that agreed on the following essential points [16]:

> **Slower cruising** - at about Mach 0.7, [...] which is 5 percent to 10 percent slower than today's aircraft and at higher altitudes, to save fuel.

Engines that require less power - on takeoff, for quieter flight.

Shorter runways - about 5,000 feet long, on average - to increase operating capacity and efficiency.

Smaller aircraft in the medium-size class of a Boeing 737, with cabin accommodations for no more than 180 passengers flying shorter and more direct routes, for cost-efficiency.

Reliance on promised advancements in air traffic management - such as the use of automated decision-making tools for merging and spacing enroute and during departure climbs and arrival descents.

These recommendations set goals for aircraft configurations by 2030 that would be hard or even impossible to achieve with current or foreseeable future technology unless aircraft design had not become an optimization process itself, no longer solely based on experience or intuition, but on the sound mathematical principles that support numerical optimization.

Computer models of aerospace structures for optimization have been in use ever since digital computers became powerful enough to efficiently solve linear systems of equations of relatively large dimension. The works of Reuther and Jameson [17] constitute some of the earliest examples of aerodynamic shape optimization using more complex numerical models. However, aerodynamic shape optimization is not particular to aerospace design, as it has also seen use in the automotive industry [18].

Multidisciplinary analysis has become commonplace in engineering design in the last couple of decades, aircraft design being no exception to that. The application of MDO to aircraft and more specifically, wing design, presents many challenges, since disciplines like aerodynamics and structural analysis have to be combined and interact. The level to which this interaction is implemented depends only on how much one is willing to pay in terms of computational cost. Fully coupled, high-fidelity Computational Fluid Dynamics (CFD)/Finite Element Model (FEM) approaches have been developed [19, 20, 21] and represent the state of the art in deterministic MDO.

An example of previous work on an MDO framework is the Bi-Level Integrated System Synthesis (BLISS) based application, developed by Sobieski et al. [22, 23], using a Collaborative Optimization (CO) implementation in MATLAB$^{®}$ and iSIGHT$^{®}$. Other MDO tools worthy of mention are Wakayama's WingMOD [24], specialized for the optimization of Blended Wing Body (BWB) aircraft, and the application developed by Ko also intended for BWB optimization but with provisions for a Distributed Propulsion concept [25]. WingOpt [26] is another example. Standalone, deployable MDO applications are, however, scarce. Another area of concern in MDO is the fact that for increasingly complex problems, the problem set up itself, as well as the post processing, can become too cumbersome for the regular user. Therefore, the introduction of a Graphical User Interface (GUI) is always welcome. Commercial CAD tools for visualization of the design optimization process are presented in [27]. Another interesting approach in the use of graphical interfaces in optimization tools is the real-time constraint visualization implemented by Deremaux [28].

1.3 Aircraft wing design

In this dissertation, the focus is put on aircraft wings since at early design stages they represent the most important design driver in terms of performance and flight qualities. This is not to say, however, that the methods and algorithms described in subsequent chapters could not be applied to full aircraft configurations. Since the advent of aeronautics in the early 20th century, aircraft wings have been built in various shapes and sizes, spanning almost every type of known structural material. In the particular case of civil aircraft wing structures, the evolution has been mostly in the materials employed since their geometry and the type of construction - stressed skin construction (of which examples are presented in Figs. 1.1 and 1.2) - has remained mostly unchanged since adoption of the jet engine by civil aviation in the 1950's.

Estimates of aerodynamic performance for the purpose of wing design optimization have been obtained from a variety of sources. Admittedly, the state of the art in CFD are full Navier-Stokes simulations. These are, however, more adequate for fine tuning at later stages of the design process as they are computationally intensive and significant changes to the baseline design require mesh adaptations that may be difficult to generate automatically. More common in high fidelity wing shape optimization problems is the use of Reynolds Averaged Navier-Stokes (RANS) or Euler solvers [20, 29, 30].

The wing assembly, composed of skin panels, spars, ribs and stringers has been modeled using various techniques over the years. A few examples:

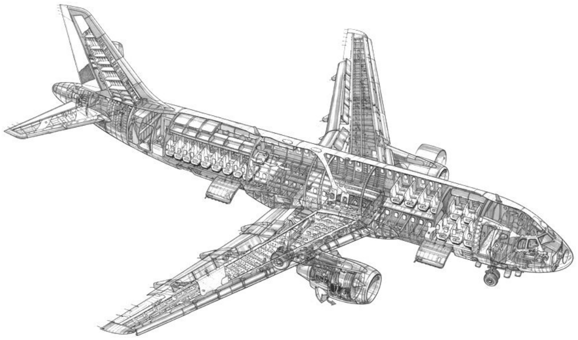


Figure 1.1: Airbus A320 cutaway view, 2006 Reed Business Information$^{©}$.

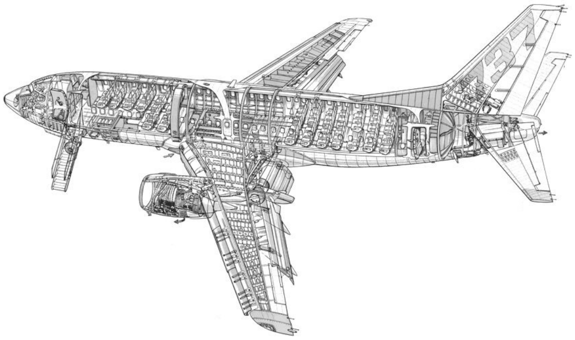


Figure 1.2: Boeing 737 cutaway view, 2006 Reed Business Information$^{©}$.

- analytical methods [31, 32]

- equivalent plate model [33, 34, 35], equivalent beam model [36]

- finite element models of variable complexity

These tools not only allow the determination of the structural feasibility of wing structures through stress analysis, but may also provide weight estimates. Although accurate results are obtained from finite element models, analytical methods can be used as well. One such example is the PDCYL tool developed at NASA [37].

In this work, the focus will be mainly upon aerodynamics and structural computational models, and interactions thereof, which essentially makes the analysis and optimization process a multidisciplinary one. The level of fidelity of these analyses is to be acceptable at the conceptual design stage. Nonetheless, some of the details eventually incorporated in the computational models developed also hint at a possible use at the preliminary design stage (a good distinction/definition of the various design stages in aircraft design may be found in [38]). A detailed description of the parameterizable wing model used for optimization is presented in Chapter 4.

1.4 Outline

This thesis is organized in eight chapters. Chapter 1 introduces the reader to the subject of multidisciplinary design optimization and the motivation of the proposed research.

In Chapter 2, robustness and reliability analysis for application in optimization problems (RDO and RBDO) are discussed, and analytic function problems are solved to illustrate the proposed methodology. Also, a hybrid architecture (R^2BDO) is introduced and tested.

Chapter 3 presents and evaluates surrogate model techniques and the underlying mathematical principles are discussed. Three models have been implemented in the framework: quadratic interpolation, regression Kriging and artificial neural networks. The use of surrogate models is necessary due to the high computational cost of nondeterministic analysis, as observed in Chapter 2. The examples presented in Chapter 2 are again used in Chapter 3 to evaluate the accuracy of the surrogate model in solving nondeterministic optimization problems.

Chapter 4 presents an overview of the various geometric, structural and aerodynamic parameters that define the aircraft wing model. This parameterizable model is used in both versions of the multidisciplinary design, analysis and optimization tool (low and medium fidelity) and this is described in more detail in the following two chapters.

Chapter 5 introduces the low fidelity version of the design and optimization framework which is used to evaluate the performance of the three surrogate models discussed in Chapter 3 when solving (deterministic) wing design problems. The latest version of the tool, its capabilities and validation are the focus of Chapter 7. Here, the graphical user interface for the framework is also presented.

Chapter 7 presents the evaluation and verification of the proposed low and high fidelity multidisciplinary design optimization tool using non deterministic optimization methods. Finally, Chapter 8 synthesizes the conclusions and possible improvements and further research are proposed.

1.5 Contribution to the state of the art

The main contribution of the current thesis lies in the development of a multidisciplinary analysis and design optimization tool for aircraft wing design which includes the capability of performing nondeterministic optimization using a hybrid algorithm. This novel architecture employs both RDO and RBDO techniques to search for robust optima while obeying reliability type of constraints. Named Robust and Reliability Based Design Optimization (R^2BDO), it is an effort to reconcile the search for designs that are simultaneously less sensitive to nonstandard conditions and present a preset measure of reliability/probability of failure. This methodology proves to be more versatile than RBDO at the same time that it is more accurate than RDO in estimating failure at nondeterministic limit functions. In addition, a new implementation of the classical RBDO subproblem, in which a change of coordinates (to hyperspherical coordinates) is performed, allows the reduction of the dimensionality of such subproblem while eliminating its equality constraint (as discussed in Chapter 2). The result is a much more computationally efficient reliability analysis.

Surrogate models are also used in an attempt to reduce the computational burden of direct evaluation in a gradient based optimization environment (with the added complexity of nondeterministic analysis) and are evaluated in aircraft design problems using a lower fidelity version of the current tool. Three different types of surrogate models are tested - Quadratic Interpolation, Kriging and Artificial Neural Networks - and from the results of this preliminary analysis, Kriging is chosen as the candidate to be incorporated into the final version of the R^2BDO framework.

The R^2BDO architecture is verified and evaluated using both analytic and more complex and realistic wing design problems and the results are compared with those of a gradient based deterministic optimization approach. Wing design problems are defined and solved using a purposely developed code which is capable of multidisciplinary analysis and optimization. Disciplines such as aerodynamics and structural analysis can be coupled in this fully standalone application.

Chapter 2

Design with uncertainty

2.1 Design uncertainty

An indirect definition of uncertainty can be found in [39]:

> Certainty, in the context of decision theory, is the condition in which a decision maker knows everything needed to select the most desirable outcome. Uncertainty is the gap of what the decision maker presently knows and certainty.

The taxonomy of uncertainty can be based on either its causes or its nature [39, 40]. Regarding the causes, uncertainty may fall into one of two categories:

Aleatory, stochastic or random - if it is related to the inherent variability in natural phenomena. Hence, it cannot be reduced, short of changing the phenomenon itself. It is *irreducible* even if more sample data/information is collected.

Epistemic or subjective - in which case the shortcomings of the models used to describe physical phenomena come into play. It stems from a lack of knowledge and is therefore *reducible* through obtaining additional information. It is also usually biased.

Additionally, this distinction can be made at different stages of the decision making process. If one considers the nature of uncertainty, the division between categories is not as universal since several authors have proposed different taxonomies. Zimmermann, for instance, distinguishes between lack of knowledge, complexity (awareness that models are simplified versions of reality), conflicting evidence, ambiguity, measurement error and human subjectivity. On another note, Ayyub elaborates on the lack of knowledge and whether it is acknowledged or not [39].

For the particular case of aeronautic structures, Yu and Du [41] elaborate on the origins of uncertainty :

Uncertainties in operations - e.g. aerodynamic loads, flight speed, altitude, angle of attack

Uncertainties in material properties - e.g. material tensile strength and Young's modulus

Uncertainties in manufacturing processes - e.g. tolerances on dimensions and shapes

Modeling uncertainties - e.g. simplifications of computational models, experimental determination of parameters, fidelity appropriate to the design stage

2.2 Random variable distributions

What follows is a brief introduction to the statistics nomenclature used in this work, together with some important definitions.

2.2.1 Probability density function and cumulative density function

The probability density function (PDF), p_X, of a continuous random variable X satisfies:

$$P(X \leq x) = \int_{-\infty}^{x} p_X(\xi)\, d\xi \tag{2.1}$$

where P signifies probability. Since $P(X \leq x) = F_X(x)$ is the cumulative density function (CDF), the PDF is the derivative of the CDF. Two important properties of the CDF are:

$$
\begin{aligned}
&P(-\infty < x < +\infty) = 1 \\
&P(x = a) = 0 \;\; \forall\, a \in \mathbb{R}
\end{aligned}
$$

2.2.2 Mean, Variance and higher order moments

The first moment of the distribution of a random variable X is the mean or expected value and is denoted by:

$$E\left(X\right) = \mu_X = \int_{-\infty}^{+\infty} t\, p_X(t)\; dt \tag{2.2}$$

where $p_X(t)$ is the probability density function of the random variable X.

The second central moment is the variance (standard deviation squared):

$$V\left(X\right) = \sigma_X^2 = \int_{-\infty}^{+\infty} \left(t - \mu_X\right)^2\, p_X(t)\; dt \tag{2.3}$$

Higher order moments (n^{th} order) are defined by:

$$M_n = \int_{-\infty}^{+\infty} \left(t - \mu_X\right)^n\, p_X(t)\; dt \tag{2.4}$$

The specification of the standard deviation may be done dynamically, by means of the coefficient of variation (c.o.v.), defined as $\frac{\sigma}{|\mu|}$.

Other notable quantities related to higher order moments are the skewness, γ_X ($M_3 = \gamma_X \sigma_X^3$) and the kurtosis, κ_X ($M_4 = \kappa_X \sigma_X^4$).

2.2.3 Multivariate distributions

The extension of Eqs. 2.2 to 2.4 for two or more random variables requires the characterization of a joint probability density function, $p_{X_1,\ldots,X_N}(\mathbf{t})$, so that:

$$P(X_1 < x_1, ..., X_n < x_n) = \int_{-\infty}^{x_1} \cdots \int_{-\infty}^{x_N} t\, p_{X_1,\ldots,X_N}(t_1, ..., t_N)\; dt_1 ... dt_N \tag{2.5}$$

$P\left(X_1 < x_1, ..., X_n < x_n\right) = F_{X_1,\ldots,X_N}\left(x_1, ..., x_N\right)$ being the joint cumulative distribution function.

The mean of the joint distribution is obtained from:

$$[\mu_{X_1}, ..., \mu_{X_N}] = \int_{-\infty}^{+\infty} ... \int_{-\infty}^{+\infty} [x_1, ..., x_N]\, p_{X_1,...,X_N}(x_1, ..., x_N)\; dx_1...dx_N \tag{2.6}$$

Similarly, for all central moments a general expression may be derived:

$$M_{m_1,...,m_N} = \int_{-\infty}^{+\infty} ... \int_{-\infty}^{+\infty} p_{X_1,...,X_N}(x_1, ..., x_N) \prod_{i=1}^{N} (x_i - \mu_{X_i})^{m_i}\; dx_1...dx_N \tag{2.7}$$

2.3 Robust and reliable design

To incorporate uncertainties in design optimization implies solving a suitably modified version of the deterministic design optimization problem. The methodologies to do so are divided into two main groups: RDO and RBDO. The (somewhat) subtle differences between RDO and RBDO are not mentioned often enough in the literature (an exception being [42]) and as such, a clarification is in order.

In an RDO formulation the goal is to optimize the response of a system about a mean value - maximizing robust performance while minimizing its sensitivity to random parameters, which is usually attained by minimizing standard deviation of the response, along with its mean. In this way, the performance in off-design conditions is also given consideration during the optimization process. The Monte Carlo (MC), Method of Moments (MM) and Sigma Point (SP) are some of the techniques that may be employed to compute the statistical characteristics of the system response, and will be discussed in subsequent sections.

On the other hand, RBDO approaches provide a way of designing while taking into account safety margins. In other words, the optimization can be performed while having a particular risk in mind - target reliability. Besides Monte Carlo, reliability may be estimated through the First or Second Order Reliability Methods (FORM and SORM), for instance.

While both architectures necessarily require more function evaluations than the equivalent deterministic optimization problem, RDO can be performed on an unconstrained problem while RBDO is by definition performed on constrained problems only.

In this work, when necessarily dealing with the probability distributions of multiple variables, two assumptions will be made. The first is that the uncertain variables are assumed to be normally distributed. This assumption implies no loss of generality since non-normal random variables whose probability distribution is known can be subjected to an appropriate transformation to make them normal (the Rosenblatt transformation, for instance [43]). The objective/constraints in an optimization problem are then obtained through function composition. The second assumption, however, is more restrictive, in the way that all the variables are assumed independent, in other words, no correlation can be set between them. Though problems where the covariance matrix is not diagonal can be solved through an equivalent subproblem where all variables are independent (by means of diagonalization of the covariance matrix and subsequent coordinate transformation) [44] and then all the methods developed henceforth can be applied, the impact on the sequence of operations is considerable, as may be the influence on the final results.

2.3.1 Robust Design Optimization

A generic statement for a deterministic optimization problem can be:

$$\begin{array}{lll} \min\limits_{\mathbf{x}} & f(\mathbf{x}) & \\ \text{subject to:} & g_i(\mathbf{x}) \leq 0 & i = 1, ..., n_g \\ & x_k^{LB} \leq x_k \leq x_k^{UB} & k = 1, ..., n_{DV} \end{array} \tag{2.8}$$

where $\mathbf{x}$ is the vector of design variables (DVs), $\mathbf{x}^{LB}$ and $\mathbf{x}^{UB}$ are the lower and upper bounds on the DVs, respectively.

Reformulation of the same problem in an RDO perspective would yield [45]:

$$\begin{array}{lll} \min\limits_{\mu_{\mathbf{x}}} & F(\mu_f(\mathbf{x}, \mathbf{r}), \sigma_f(\mathbf{x}, \mathbf{r})) & \\ \text{subject to:} & G_i(\mu_{g_i}(\mathbf{x}, \mathbf{r}), \sigma_{g_i}(\mathbf{x}, \mathbf{r})) \leq 0 & i = 1, ..., n_g \\ & P(x_k^{LB} \leq x_k \leq x_k^{UB}) \geq P_{bounds} & k = 1, ..., n_{DV} \end{array} \tag{2.9}$$

where μ and σ represent the mean and standard deviation of the quantities in the subscript (DVs, objective or constraints). These may be computed as per Eqs. 2.10 and 2.11. In this formulation, the otherwise deterministic parameters $\mathbf{r}$ are allowed a random distribution (in a typical design problem these may constitute material properties, for instance), and the design variables $\mathbf{x}$ can now be either deterministic or random. The robust objective and constraints are now functions of the mean and standard deviation of objective and constraints, which in turn depend on the probabilistic distribution of the variables. The bounds on the DVs are now themselves established in terms of their probability of residing inside the preset interval (although they can easily be reformulated as deterministic constraints back again since the probabilistic distribution of the variables is known).

$$\mu_f(\mathbf{x},\mathbf{r}) = \int_{-\infty}^{+\infty} ... \int_{-\infty}^{+\infty} f(\mathbf{t})\, p_{\mathbf{x},\mathbf{r}}(\mathbf{t})\; d\mathbf{t} \tag{2.10}$$

$$\sigma_f(\mathbf{x},\mathbf{r}) = \int_{-\infty}^{+\infty} ... \int_{-\infty}^{+\infty} \left[f(\mathbf{t})\; - \mu_f(\mathbf{x},\mathbf{r})\right]^2 p_{\mathbf{x},\mathbf{r}}(\mathbf{t})\; d\mathbf{t} \tag{2.11}$$

here f represents a function of interest and $p_{\mathbf{x},\mathbf{r}}$ is the joint probability density function (vector $\mathbf{t}$ has total dimension equal to that of $\mathbf{x}$ plus that of $\mathbf{r}$). The analytical evaluation of the integrals in Eqs. 2.10 and 2.11 is impossible in most practical cases, and for that reason a numerical procedure is required. Among the various techniques that may be used are Monte Carlo methods, the Taylor based Method of Moments [45] and the Sigma Point technique. These will be covered in the following sections in more detail.

An example of how a generic function, f, is affected by this formulation is represented in Fig. 2.1 (uncertainty is defined for the independent variable in the abscissa axis). In this instance, the robust function F is defined as $F = \mu_f + \sigma_f$. In essence, this transformation acts as a low pass filter which fills local minima located in areas where the function changes more abruptly.

Monte Carlo Sampling

Monte Carlo methods cover a wide range of algorithms dedicated to generating numerical samples with a predetermined statistical meaning. The name originated during World War II, and was related to the casino of Monte Carlo, whose roulette was known as the best random number generator of the time. Monte Carlo methods usually fare better in multidimensional problems where the fact they are mesh free wins over grid based methods and other deterministic types of designs. Nonetheless, the fact remains that for numerical integration, these methods converge very slowly in comparison with their deterministic counterparts. Most Monte Carlo methods hinge on a common principle which is the initial generation of a random uniform sample, and from there build a new sample with the desired characteristics. To

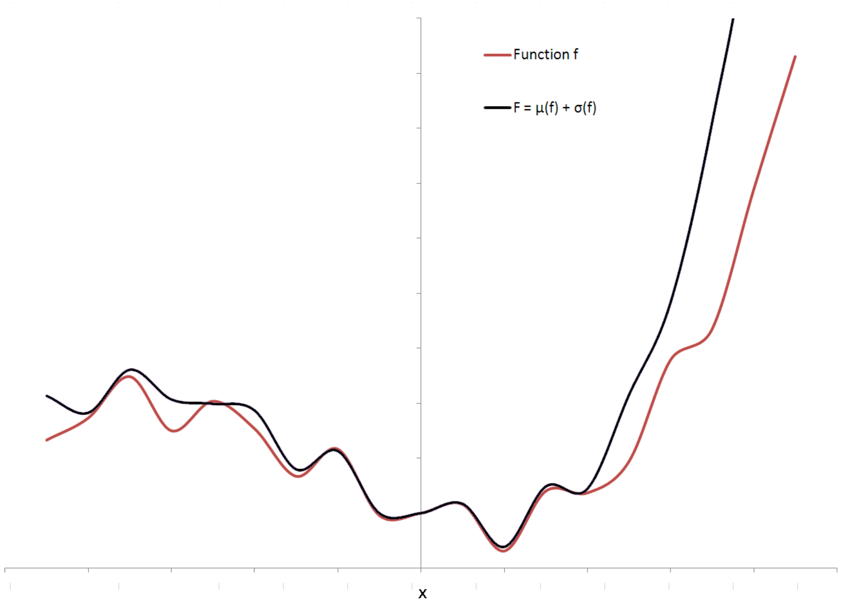


Figure 2.1: Example of robust function

accomplish this last step, Inverse Probability Transformation (the Box-Muller algorithm for generating normally distributed samples falls in this category) and Sampling Acceptance-Rejection are some of the methods that may be employed [46].

Taylor Based Method of Moments

As the name implies, the Taylor Based Method of Moments employs a Taylor series expansion of the function f of a random vector $\mathbf{x}$ about the mean of $\mathbf{x}$:

$$\begin{aligned} f\left(\mathbf{x}\right) &= f\left(\mu_{\mathbf{x}}\right)+\sum_{i=1}^{N_{RV}}\left(\frac{\partial f}{\partial x_i}\right)\left(x_i-\mu_{x_i}\right)+ \\ &\quad +\frac{1}{2}\sum_{i=1}^{N_{RV}}\sum_{j=1}^{N_{RV}}\left(\frac{\partial^2 f}{\partial x_i \partial x_j}\right)\left(x_i-\mu_{x_i}\right)\left(x_j-\mu_{x_j}\right)+ \\ &\quad +\frac{1}{3!}\sum_{i=1}^{N_{RV}}\sum_{j=1}^{N_{RV}}\sum_{k=1}^{N_{RV}}\left(\frac{\partial^3 f}{\partial x_i \partial x_j \partial x_k}\right)\left(x_i-\mu_{x_i}\right)\left(x_j-\mu_{x_j}\right)\left(x_k-\mu_{x_k}\right)+ \\ &\quad +\ldots \end{aligned} \tag{2.12}$$

where N_{RV} is the number of random variables.

Taking the mean of Eq. 2.12 and assuming all the entries in vector $\mathbf{x}$ are mutually independent and symmetrically distributed [45]:

$$E\left(x_i-\mu_{x_i}\right)=0 \tag{2.13}$$

$$E\left((x_i-\mu_{x_i})(x_j-\mu_{x_j})\right)=0,\ i\neq j \tag{2.14}$$

$$E\left((x_i-\mu_{x_i})(x_j-\mu_{x_j})\right)=\sigma_{x_i}^2,\ i=j \tag{2.15}$$

$$E\left((x_i-\mu_{x_i})(x_j-\mu_{x_j})(x_k-\mu_{x_k})\right)=0 \tag{2.16}$$

$$\mu_f=f\left(\mu_{\mathbf{x}}\right)+\frac{1}{2}\sum_{i=1}^{N_{RV}}\frac{\partial^2 f}{\partial x_i^2}\sigma_{x_i}^2+... \tag{2.17}$$

Similarly, taking the variance of f:

$$\begin{aligned}\sigma_f^2 &= \sum_{i=1}^{N_{RV}}\left(\frac{\partial f}{\partial x_i}\right)^2\sigma_{x_i}^2+\\ &+ \sum_{i=1}^{N_{RV}}\left[\frac{\partial^3 f}{\partial x_i^3}\frac{\partial f}{\partial x_i}\frac{\kappa_{x_i}}{3}+\left(\frac{\partial^2 f}{\partial x_i^2}\right)^2\frac{\kappa_{x_i}-1}{4}\right]\sigma_{x_i}^4+\\ &+ \sum_{i=1}^{N_{RV}}\sum_{j=1,\,i\neq j}^{N_{RV}}\left[\left(\frac{\partial^3 f}{\partial x_i^2\partial x_j}\right)\left(\frac{\partial f}{\partial x_j}\right)+\frac{1}{2}\left(\frac{\partial^2 f}{\partial x_i\partial x_j}\right)^2\right]\sigma_{x_i}^2\sigma_{x_j}^2+\\ &+ ...\end{aligned} \tag{2.18}$$

These estimates are then third order accurate and require information on high order derivatives. Notwithstanding, in many practical optimization cases, where only first order derivatives are computationally tractable, a first order series is employed, with considerable loss in accuracy, as will be shown in Section 2.4.1.

Sigma Point Method

The Sigma Point (SP) method is a derivative of the Taguchi method, used in statistical tolerancing since 1978 [47]. The idea behind SP is that it is easier to match an input distribution (SP is typically defined for a normal distribution) than to linearize (or in general, approximate) a nonlinear mapping [48, 49]. To compute the integrals in Eqs. 2.10 and 2.11 SP employs a procedure similar to Gaussian integration, but where the sample locations and respective weights are optimized to match the first moments of the input probability distribution.

For a given vector of random variables $(X_1, X_2, ..., X_{N_{RV}})$, representing the uncertainty about a certain point (mean is the zero vector), a candidate set of sigma points must respect the following [50]:

$$W_0 + \sum_{i=1}^{N_{SP}} W_i = 1 \tag{2.19}$$

$$\sum_{i=1}^{N_{SP}} W_i \, {}_1S_i^{m_1} {}_2S_i^{m_2} ... {}_{N_{RV}}S_i^{m_{N_{RV}}} = E\left(X_1^{m_1} X_2^{m_2} ... X_{N_{RV}}^{m_{N_{RV}}}\right)$$
$$\forall \left(m_1, ..., m_{N_{RV}}\right) : 1 \leq m_1 + ... + m_{N_{RV}} \leq k \tag{2.20}$$

where W_i are the weights, ${}_nS_i$ is the n^{th} coordinate of the i^{th} sigma point (coordinates are relative to the central point, $\mu_{\mathbf{x}}$, i.e, the the mean values of the design variables, where the weight is W_0), and N_{SP} is the number of sigma points minus the origin. The condition stated in Eq. 2.20 should be repeated for all integer values k up to the order required for the approximation (starting from 1). The order of the approximation also dictates the minimum number of points in a set that satisfies these conditions. Minimum point sets are relatively easy to find for multivariable problems with up to 3 random variables. Higher dimensionality sets usually require the usage of a nonlinear solver. Even then, the solution may not be unique [50].

For a set of independent random variables, SP can be reduced to sampling along the coordinate axis, at a distance from the point of interest that is only dependent on the standard deviation in that direction (hence the designation of the method as sigma point). For Gaussian random vectors, a deterministic procedure is available to determine the sigma point set for calculation of mean and standard deviation of a function of said vector [51, 48, 45]:

$$\chi_0 = \mu_{\mathbf{x}} \tag{2.21}$$

$$\chi_{i+} = \mu_{\mathbf{x}} + \sqrt{(N_{RV} + K)} \left(\sqrt{\Sigma_{\mathbf{x}}}\right)_i, \; i = 1, ..., N_{RV} \tag{2.22}$$

$$\chi_{i-} = \mu_{\mathbf{x}} - \sqrt{(N_{RV} + K)} \left(\sqrt{\Sigma_{\mathbf{x}}}\right)_i, \; i = 1, ..., N_{RV} \tag{2.23}$$

$$W_0 = \frac{K}{N_{RV} + K} \tag{2.24}$$

$$W_i = W_{i+} = W_{i-} = \frac{1}{2\left(N_{RV} + K\right)}, \; i = 1, ..., N_{RV} \tag{2.25}$$

where $\left(\sqrt{\Sigma_{\mathbf{x}}}\right)_i$ is the i^{th} row in the square root of the covariance matrix. In case the variables are independent this term reduces to the standard deviation in the i^{th} direction: σ_{x_i}.

The value of the real constant K should be set so that $N_{RV} + K = 3$, the kurtosis of the standard normal distribution. Based on this set, the estimators (denoted by ˆ) for the mean and standard deviation are [52]:

$$\hat{\mu}_f(\chi_0) = W_0 f(\chi_0) + \sum_{i=1}^{N_{RV}} W_i \left(f(\chi_{i+}) + f(\chi_{i-})\right) \tag{2.26}$$

$$\begin{aligned} 2\hat{\sigma}_f^2(\chi_0) &= \sum_{i=1}^{N_{RV}} W_i \left(f(\chi_{i+}) - f(\chi_{i-})\right)^2 + \\ &+ \sum_{i=1}^{N_{RV}} \left(W_i - 2W_i^2\right) \left(f(\chi_{i+}) + f(\chi_{i-}) - 2f(\chi_0)\right)^2 \end{aligned} \tag{2.27}$$

The derivatives of these estimates with respect to the design variable set:

$$\frac{\partial \hat{\mu}_f}{\partial x_k}(\chi_0) = W_0 \frac{\partial f(\chi_0)}{\partial x_k} + \sum_{i=1}^{N_{RV}} W_i \left(\frac{\partial f(\chi_{i+})}{\partial x_k} + \frac{\partial f(\chi_{i-})}{\partial x_k}\right) \tag{2.28}$$

$$\begin{aligned} \frac{\partial \hat{\sigma}_f^2}{\partial x_k}(\chi_0) &= \sum_{i=1}^{N_{RV}} W_i \left(\frac{\partial f(\chi_{i+})}{\partial x_k} - \frac{\partial f(\chi_{i-})}{\partial x_k}\right) \left(f(\chi_{i+}) - f(\chi_{i-})\right) + \\ &+ \sum_{i=1}^{N_{RV}} \left(W_i - 2W_i^2\right) \left(\frac{\partial f(\chi_{i+})}{\partial x_k} + \frac{\partial f(\chi_{i-})}{\partial x_k} - 2\frac{f(\chi_0)}{\partial x_k}\right) \left(f(\chi_{i+}) + f(\chi_{i-}) - 2f(\chi_0)\right) \end{aligned} \tag{2.29}$$

The set then contains $2N_{RV} + 1$ points so each uncertainty quantification carries an additional penalty in terms of function evaluations. If the number of evaluations required to compute sensitivities in a finite difference estimate for gradient based optimization is given by N_{DV}, a total of $(2N_{RV} + 1)(N_{DV} + 1)$ evaluations is required for each design point. This makes RDO problem solving a daring prospect in terms of computational resources, if finite differencing is used. In the case of independent random variables, it would certainly be beneficial if some of the sigma points could be superimposed with points used for finite differencing. However, in general, the distances from the central point at which the evaluations for sigma points and finite differencing are made are several orders of magnitude apart.

2.3.2 Reliability Based Design Optimization

An equivalent statement to Eq. 2.8 in RBDO is [53, 54]:

$$\begin{array}{lll} \min\limits_{\mathbf{x}} & f(\mathbf{x},\mathbf{r}) & \\ \text{subject to:} & g_i^{rc}(\mathbf{x},\mathbf{r}) \leq 0 & i=1,...,n_{rc} \\ & g_j^d(\mathbf{x}) \leq 0 & j=1,...,n_d \\ & x_k^{LB} \leq x_k \leq x_k^{UB} & k=1,...,n_{DV} \end{array} \tag{2.30}$$

The constraints set is now divided into reliability constraints, g_i^{rc}, and other design constraints, g_j^d (for which a reliability target is not established). Alternatively, the objective function may be defined in terms of the probability of the original function exceeding/not exceeding a certain target, which is to be minimized [42]:

$$\begin{array}{lll} \min\limits_{\mathbf{x}} & P(f(\mathbf{x},\mathbf{r}) - target \geq 0) \ or \ P(target - f(\mathbf{x},\mathbf{r}) \geq 0) & \\ \text{subject to:} & g_i^{rc}(\mathbf{x},\mathbf{r}) \leq 0 \quad i=1,...,n_{rc} & \\ & g_j^d(\mathbf{x}) \leq 0 \quad\quad\;\; j=1,...,n_d & \\ & x_k^{LB} \leq x_k \leq x_k^{UB} \;\; k=1,...,n_{DV} & \end{array} \tag{2.31}$$

The reliability constraints are of the form:

$$g_i^{rc} = P_{f_i} - P_{allow_i} = P(g_i(\mathbf{x},\mathbf{r}) \geq 0) - P_{allow_i} \tag{2.32}$$

$$P(g_i(\mathbf{x},\mathbf{r}) \geq 0) = \int_{g_i(\mathbf{x},\mathbf{r})\geq 0} p_{\mathbf{x},\mathbf{r}}(\mathbf{t}) \ d\mathbf{t} \tag{2.33}$$

effectively ensuring that the probability of the originally deterministic constraint g_i being violated is at the most P_{allow_i} - the allowable probability of failure. Determining the probability of failure, P_{f_i}, requires either sampling (again, Monte Carlo methods, used in a similar way as described in section 2.3.1) or techniques such as the First Order Reliability Method (FORM) and the Second Order Reliability Method (SORM) [53]. While FORM is widely used in reliability analysis, SORM has seen little practical use as it requires higher order information on the objective function and constraints [42, 53, 54, 55]. The next section is therefore dedicated to describing FORM in more detail.

Other methods that have been used in reliability analysis include the Mean Value (MV), Advanced Mean Value (AMV) and their derivatives [56]. These aim at approximating the probabilistic distribution of the function of interest given a known uncertainty distribution in the input. The procedure for obtaining these approximations bears some similarity with the Taylor Based Method of Moments described before. The probability of failure is then computed by evaluating the approximate CDF.

FORM

In essence, FORM consists of creating a linear approximation to the limit state function $g(\mathbf{r})$ ($\mathbf{r}$ now being a generalized set of random variables - which encompasses the uncertainties in both design variables and parameters). According to FORM the probability of failure is evaluated (approximately) as:

$$P_{f_i} = \Phi(-\beta) \tag{2.34}$$

where Φ is the cumulative distribution function of the standard normal distribution and β is the distance from the Most Probable Point (MPP) of failure to the current iterate (also called reliability index), measured in the standard normal space - $\mathbf{u}$ - (see Fig. 2.2). It is determined by solving the following optimization problem ($\beta = (\mathbf{u}^T\mathbf{u})^{\frac{1}{2}}$).

$$\begin{array}{ll} \min\limits_{\mathbf{u}} & (\mathbf{u}^T\mathbf{u})^{\frac{1}{2}} \\ \text{subject to:} & g\left(\mathbf{r}(\mathbf{u})\right) = 0 \end{array} \tag{2.35}$$

The vector $\mathbf{u}$, in standard normal coordinates, is obtained from $\mathbf{r}$ through a transformation (which is diagonal for the case of statistically independent variables):

$$\mathbf{u} = T(\mathbf{r}) \tag{2.36}$$

In this case though, the inverse transformation $T^{-1}(\mathbf{u})$, is the one that is most useful for the computation of the limit state function in Eq. 2.35. Examples of such transformations are listed in Table 2.1 [55] for various distributions of the random variable $\mathbf{r}$.

Realizing that, from the result of the MPP optimization problem, the reliability

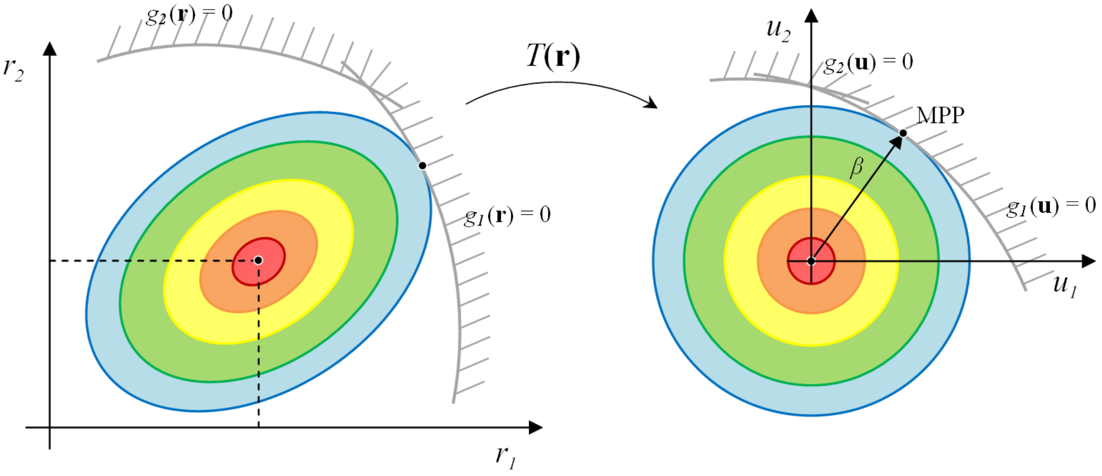


Figure 2.2: MPP determination (showing the contours of the joint PDF)

Table 2.1: Examples of transformations from **u**-space into **r**-space

Distribution type	Transformation, $r_k = T^{-1}(u_k)$
Normal (μ, σ)	$\mu + \sigma u_k$
Lognormal (μ, σ)	$e^{\mu + \sigma u_k}$
Uniform (a, b)	$a + (b - a)(0.5 + 0.5\,\mathrm{erf}(u_k\sqrt{2}))$
Gamma (a, b)	$ab\left(u_k \frac{1}{\sqrt{9a}} + 1 - \frac{1}{9a}\right)^3$

constraint may be written in terms of the reliability index β:

$$g_i^{rc} = \beta_{reqd} - \beta_i \tag{2.37}$$

the *Reliability Index Approach* (RIA) is obtained. β_{reqd} is the specified reliability index. The RIA is, however, problematic in cases where failure does not occur at all for a particular set of values of the design variables, or when the limit state surface is far from the origin. For that reason, the *Performance Measure Approach* (PMA) was devised [57]. In PMA the inverse problem of that one stated in Eq. 2.35 is solved instead:

$$\begin{aligned} \min_{\mathbf{u}} \quad & -g\left(\mathbf{r}(\mathbf{u})\right) \\ \text{subject to:} \quad & (\mathbf{u}^T\mathbf{u})^{\frac{1}{2}} - \beta_{reqd} = 0 \end{aligned} \tag{2.38}$$

This is not only a more robust formulation than RIA, it also immediately returns the required value of the reliability constraint ($g_i^{rc} = g^* (\mathbf{r}(\mathbf{u}))$). Another important advantage is that the MPP subproblem may be formulated in a minimax approach, effectively handling multiple constraints simultaneously, something that is not possible with RIA.

A proposed alternate strategy for solving the PMA problem stated in Eq. 2.38 consists in restricting the values of the vector $\mathbf{u}$ to lie on a hyper-spherical surface, effectively eliminating the need for the equality constraint $(\mathbf{u}^T\mathbf{u})^{\frac{1}{2}} - \beta_{reqd} = 0$. In addition, the dimensionality of this subproblem is reduced to $N_{RV} - 1$ which should allow for faster convergence. The problem statement is then simply:

$$\min_{\phi} \quad - g\left(\mathbf{r}(\mathbf{u}(\mathbf{\Phi}))\right) \tag{2.39}$$

where $\mathbf{\Phi}$ is the set of hyperspherical coordinates $\mathbf{\Phi} = \{\phi_1, \phi_2, ..., \phi_{N_{RV}-1}\}$.

Hence, the solution with hyper-spherical coordinates requires the usage of yet another transformation to obtain the $\mathbf{u}$ vector back from the ϕ_i coordinates.

$$\begin{aligned} u_1 &= \beta_{reqd} \cos\left(\phi_1\right) \\ &\vdots \\ u_{N_{RV}-1} &= \beta_{reqd} \prod_{i=1}^{N_{RV}-2} \sin\left(\phi_i\right) \cos\left(\phi_{N_{RV}-1}\right) \\ u_{N_{RV}} &= \beta_{reqd} \prod_{i=1}^{N_{RV}-1} \sin\left(\phi_i\right) \end{aligned} \tag{2.40}$$

The performance of this alternative approach is compared with that of the classical PMA in section 2.4.2.

Additional major considerations in RBDO problems include sensitivity analysis in the MPP subproblem (a derivation of sensitivities in both RIA and PMA is presented in [42]) as well as the manner in which the latter is coupled with the main optimization problem. In the class of double loop methods, the determination of the MPP is executed at each iteration of the main optimization algorithm, whereas in sequential methods the MPP subproblem is completely decoupled from the deterministic part of the optimization, the two problems being solved in turn until a convergence criteria is satisfied. Unilevel methods are the other extreme since the formulation of the MPP optimization problem is embedded into the original problem by means of its KKT conditions [41].

2.3.3 R^{2}BDO

A mix of robust objective/reliable constraints is believed to be better suited than the original RBDO formulation for general purpose optimization, given that probabilistic objective function targets are frequently problematic. If the initial guess for a target is too far off what an actual design can attain, the probability becomes either too close to zero or to one, slowly varying. As can be seen in Fig. 2.3, when the expected value of performance of the system (μ_f) exceeds the expectations set by the target by a large amount, the objective function becomes insensitive to change since the probability (accounting for the uncertainty in the design variables) is now being measured at either of the tails of the PDF. In practice, this means an optimizer would tend to stop prematurely, before finding a true candidate to local/global minimum.

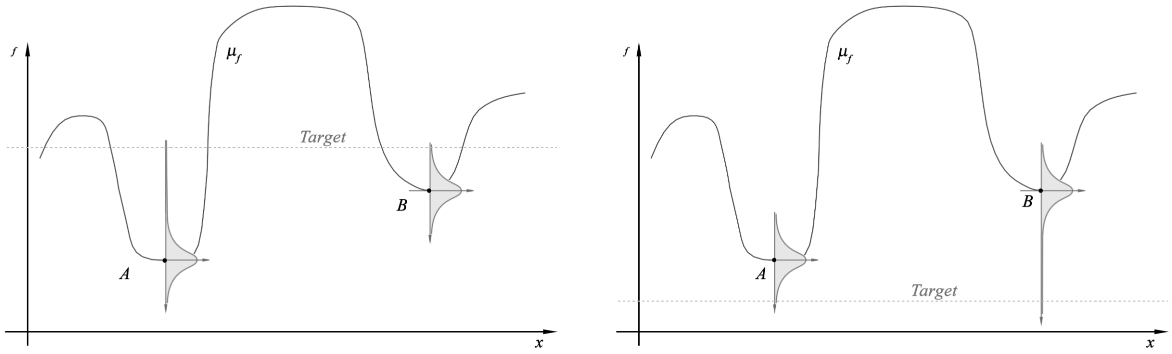


Figure 2.3: Possible issue with probabilistic objectives

On the other hand, constraint treatment in RDO is not transparent since the designer is left with a choice of weights that will ultimately define how far from the failure surface should the average optimum lie. This can be addressed by adapting the robust constraints so that the weights are calibrated in order to mimic a probabilistic constraint. This calibration is usually made using the PDF of the inputs distribution [51, 58]. Although a good approximation for very low input variances or quasi-linear constraints, in general, the constraint output distribution may greatly differ from that of the input, thus invalidating this type of analysis. The RBDO treatment of constraints is better suited for this task as it precisely quantifies the violation of the limit state by means of a probability of failure/reliability index.

Essentially, in this proposed hybrid formulation, the type of objective function used in RBDO (either deterministic or probabilistic, as mentioned in the previous section) is replaced by the type used in RDO.

$$\begin{array}{lll} \min\limits_{\mathbf{x}} & F(\mu_f(\mathbf{x},\mathbf{r}),\sigma_f(\mathbf{x},\mathbf{r})) & \\ \text{subject to:} & g_i^{rc}(\mathbf{x},\mathbf{r}) \leq 0 & i=1,...,n_{rc} \\ & g_j^{d}(\mathbf{x}) \leq 0 & j=1,...,n_{d} \\ & x_k^{LB} \leq x_k \leq x_k^{UB} & k=1,...,n_{DV} \end{array} \tag{2.41}$$

An example problem is solved in the next section, comparing not only R^{2}BDO with the other formulations, but also making the distinction between methods of implementation.

2.4 A numerical example

Using the classic Rosenbrock function, the following optimization problem can be formulated:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{x}) = 100\left(x_2 - x_1^2\right)^2 + q(1-x_1)^2 \\ \text{subject to:} & g(\mathbf{x}) = x_1^2 + x_2^2 - 1 \leq 0 \end{array} \tag{2.42}$$

which basically consists of the minimization of the function inside the unit circle. For q = 1, the solution to the deterministic optimization problem is: $x_1 = 0.7864$, $x_2 = 0.6177$ and $f = 0.0457$. This problem shall serve as a baseline for introducing robustness and reliability techniques in optimization in more detail. The starting point for the optimization is $\mathbf{x_0} = (0.1, 0.1)$ unless stated otherwise. Choosing this starting point, the reference solution is obtained after 54 function evaluations (using a Sequential Quadratic Programming - SQP - algorithm).

2.4.1 Example RDO problem

Making suitable modifications to the problem defined in Eq. 2.42, an RDO problem is defined, where the variable x_1 and the parameter q now carry an uncertainty (x_2 remains deterministic).

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & F(\mathbf{x}, q) = \mu_f + \sigma_f \\ \text{subject to:} & G(\mathbf{x}) = \mu_g + 2\sigma_g \leq 0 \end{array} \tag{2.43}$$

The objective is now to minimize the average value of the function at a design point, plus one standard deviation, to account for uncertainty. Similarly, a margin equal to twice the local standard deviation is applied to the constraint, acting as a safety buffer (if, as a first approximation, the output of g was considered to be normal, this would mean that approximately 98% of the realizations at a point where $G = 0$ would not violate the deterministic constraint). Tables 2.2 and 2.3 carry the results of the RDO optimization problem when using the Method of Moments and SP, for different combinations of the uncertainty in x_1 and q.

For the Method of Moments, the required derivatives are provided from the respective analytical expressions. In SP, the samples are taken according to Eqs. 2.21 to 2.25

The estimates of μ_f and σ_f are denoted by $\hat{\mu}_f$ and $\hat{\sigma}_f$, respectively, and are compared with a post-optimality analysis performed using Monte Carlo simulation (10^6 samples).

The first order Method of Moments proves to be inaccurate even for low values of input uncertainty, and is almost insensitive to the variation of σ_q. On the other hand, SP is remarkably accurate at predicting the desired statistical measures, comparing favorably with the much more computationally expensive Monte Carlo simulation. Not only does it manifest sensitivity to the hardly perceptible impact of σ_q, but also in an environment where finite differencing is used, SP takes exactly the same amount of function evaluations as the Method of Moments in return for a considerably higher accuracy.

As a side note, for the minima that lie on the robust constraint, the true value of the probability of failure/reliability index was determined. In some cases, although the factor of 2 for σ_g in the robust constraint approximates a reliability index of also 2 ($P_f \approx 0.0228$), the actual index, as measured by Monte Carlo, differs from this value ($\beta_{real} \approx 1.985$, for c.o.v $x_1 = 0.01$, $\sigma_q = 0.005$). Therefore, even if by a small margin, the system is not as reliable as expected. For higher input uncertainty values, the optimal solution becomes more conservative (due to the uncertainty in the objective), and the effective β values become very high as the probability of failure approaches zero.

Figure 2.4 illustrates how the uncertainty in the design variable x_1 and parameter q move the solution point away from the deterministic constraint (the contour plots are those of the deterministic Rosenbrock function and the constraint is shown as a bold line).

Table 2.2: RDO problem solutions - Method of Moments

c.o.v. x_1	σ_q	x_1	x_2	$\hat{\mu}_f$	ϵ_{μ_f} [%]	$\hat{\sigma}_f$	ϵ_{σ_f} [%]
0.005	0.005	0.7831	0.6119	0.0472	-7.4	2.36E-04	-96
	0.01	0.7831	0.6119	0.0472	-7.4	4.73E-04	-91
	0.05	0.7831	0.6120	0.0472	-7.4	0.0024	-60
	0.1	0.7831	0.6120	0.0472	-7.4	0.0047	-34
0.01	0.005	0.7797	0.6065	0.0487	-23	2.43E-04	-98
	0.01	0.7797	0.6065	0.0487	-23	4.86E-04	-98
	0.05	0.7796	0.6065	0.0487	-23	0.0024	-89
	0.1	0.7796	0.6065	0.0487	-24	0.0049	-78
0.05	0.005	0.7530	0.5653	0.0613	-84	3.05E-04	-99
	0.01	0.7530	0.5653	0.0613	-84	6.10E-04	-99
	0.05	0.7530	0.5653	0.0613	-84	0.0031	-99
	0.1	0.7530	0.5653	0.0613	-84	0.0061	-99
0.1	0.005	0.7221	0.5196	0.0776	-93	3.86E-04	-99
	0.01	0.7221	0.5196	0.0776	-93	7.72E-04	-99
	0.05	0.7221	0.5196	0.0776	-93	0.0039	-99
	0.1	0.7221	0.5196	0.0776	-93	0.0077	-99

Table 2.3: RDO problem solutions - Sigma Point Method

c.o.v. x_1	σ_q	x_1	x_2	$\hat{\mu}_f$	ϵ_{μ_f} [%]	$\hat{\sigma}_f$	ϵ_{σ_f} [%]
0.005	0.005	0.7830	0.6121	0.0510	$\|\varepsilon\| < 0.1$	0.0054	$\|\varepsilon\| < 0.5$
	0.01	0.7830	0.6121	0.0510	$\|\varepsilon\| < 0.1$	0.0054	$\|\varepsilon\| < 0.5$
	0.05	0.7830	0.6121	0.0510	$\|\varepsilon\| < 0.1$	0.0058	$\|\varepsilon\| < 0.5$
	0.1	0.7830	0.6121	0.0510	$\|\varepsilon\| < 0.1$	0.0071	$\|\varepsilon\| < 0.5$
0.01	0.005	0.7795	0.6067	0.0635	$\|\varepsilon\| < 0.1$	0.0210	$\|\varepsilon\| < 0.5$
	0.01	0.7795	0.6067	0.0635	$\|\varepsilon\| < 0.1$	0.0210	$\|\varepsilon\| < 0.5$
	0.05	0.7795	0.6067	0.0635	$\|\varepsilon\| < 0.1$	0.0211	$\|\varepsilon\| < 0.5$
	0.1	0.7796	0.6067	0.0635	$\|\varepsilon\| < 0.1$	0.0216	$\|\varepsilon\| < 0.5$
0.05	0.005	0.4777	0.2266	0.3260	$\|\varepsilon\| < 0.1$	0.0755	$\|\varepsilon\| < 1$
	0.01	0.4777	0.2266	0.3260	$\|\varepsilon\| < 0.1$	0.0756	$\|\varepsilon\| < 1$
	0.05	0.4798	0.2286	0.3247	$\|\varepsilon\| < 0.1$	0.0779	$\|\varepsilon\| < 1$
	0.1	0.4850	0.2338	0.3216	$\|\varepsilon\| < 0.1$	0.0843	$\|\varepsilon\| < 1$
0.1	0.005	0.3291	0.1054	0.4999	$\|\varepsilon\| < 0.1$	0.0716	≈ -2
	0.01	0.3292	0.1055	0.4998	$\|\varepsilon\| < 0.1$	0.0718	≈ -2
	0.05	0.3326	0.1080	0.4971	$\|\varepsilon\| < 0.1$	0.0776	≈ -2
	0.1	0.3399	0.1134	0.4916	$\|\varepsilon\| < 0.1$	0.0916	≈ -2

2.4.2 Example RBDO problem

The example problem for RBDO relies on the same optimization problem as a baseline:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & F(\mathbf{x}) = f(\mu_{\mathbf{x}}) \\ \text{subject to:} & g^{rc} \leq 0 \end{array} \tag{2.44}$$

For RIA g^{rc} is:

$$g^{rc} = \beta_{reqd} - \beta \tag{2.45}$$

where the reliability index, β, is the result of the optimization subproblem defined in Eq. 2.35 where the constraint is $x_1^2(\mathbf{u}) + x_2^2 - 1 = 0$ (requiring the appropriate transformation from **u**-space to the design variable space to be applied).

In the case of the PMA formulation, the reliability index is not computed directly. Instead, the constraint to be applied in the top level optimization problem is obtained directly from the solution of the subproblem stated in Eq. 2.38.

The objective function is defined using the mean values of the design variables as input, becoming virtually the same as if using the first order Method of Moments (which, as shown in the previous section, is not the best estimator of average function values as uncertainty in the input increases). Also, because of this and the fact that the parameter q is not involved in the constraint calculation, its uncertainty does not affect the solution of the problem. Therefore, rather than a range of values for the uncertainty in q, different starting starting points were used to assess the numerical robustness of the two formulations for reliability analysis. The value of the required reliability index (β_{reqd}) was set to both 2 and 3, which results in an admissible probability of failure of approximately 2.28% and 0.13%, repectively.

Tables 2.4 and 2.5 summarize the results, and Figs. 2.5 and 2.6 present the evolution of the solution point for increasing uncertainty in the variables when $\beta_{reqd} = 2$ and $\beta_{reqd} = 3$, respectively.

The errors (ϵ) are obtained by comparison of the real reliability with the target. The 'true' value of the probability of failure results from a post optimality Monte Carlo probability calculation using 2×10^7 normally distributed samples (normal distribution is centered on solution point). The actual value of the reliability index is obtained from these results and is then compared against the requirement.

The accuracy of both formulations in terms of the reliability index is acceptable, especially considering that FORM uses a first order approximation to the real failure hypersurface. In actuality, the errors are of the same order of magnitude to which

Table 2.4: RBDO problem solutions - RIA

$\mathbf{x}_0$	c.o.v. x_1	x_1	x_2	F	ϵ_β [%]
$\beta_{reqd} = 2$					
(0.1, 0.1)	0.005	0.7829	0.6122	0.0472	$\|\varepsilon\| < 0.1$
	0.01	0.7794	0.6067	0.0487	$\|\varepsilon\| < 0.1$
	0.05	0.7511	0.5634	0.0620	$\|\varepsilon\| < 0.1$
	0.1	0.7159	0.5118	0.0808	$\|\varepsilon\| < 0.1$
(-0.1, -0.1)	0.005	No feasible solution is found			
	0.01				
	0.05				
	0.1				
$\beta_{reqd} = 3$					
(0.1, 0.1)	0.005	0.7811	0.6094	0.0480	$\|\varepsilon\| < 0.1$
	0.01	0.7758	0.6012	0.0503	$\|\varepsilon\| < 0.1$
	0.05	0.7335	0.5372	0.0711	$\|\varepsilon\| < 0.1$
	0.1	0.6815	0.4637	0.1015	$\|\varepsilon\| < 0.1$
(-0.1, -0.1)	0.005	No feasible solution is found			
	0.01				
	0.05				
	0.1				

Table 2.5: RBDO problem solutions - PMA

$\mathbf{x}_0$	c.o.v. x_1	x_1	x_2	F	ϵ_β [%]
$\beta_{reqd} = 2$					
(0.1, 0.1)	0.005	0.7829	0.6122	0.0472	$\|\varepsilon\| < 0.1$
	0.01	0.7794	0.6067	0.0487	$\|\varepsilon\| < 0.1$
	0.05	0.7511	0.5634	0.0620	$\|\varepsilon\| < 0.1$
	0.1	0.7159	0.5118	0.0808	$\|\varepsilon\| < 0.1$
(-0.1, -0.1)	0.005	0.7829	0.6122	0.0472	$\|\varepsilon\| < 0.1$
	0.01	0.7794	0.6067	0.0487	$\|\varepsilon\| < 0.1$
	0.05	0.7511	0.5634	0.0620	$\|\varepsilon\| < 0.1$
	0.1	0.7159	0.5118	0.0808	$\|\varepsilon\| < 0.1$
$\beta_{reqd} = 3$					
(0.1, 0.1)	0.005	0.7811	0.6094	0.0480	$\|\varepsilon\| < 0.1$
	0.01	0.7758	0.6012	0.0503	$\|\varepsilon\| < 0.1$
	0.05	0.7335	0.5372	0.0711	$\|\varepsilon\| < 0.1$
	0.1	0.6815	0.4637	0.1015	$\|\varepsilon\| < 0.1$
(-0.1, -0.1)	0.005	0.7811	0.6094	0.0480	$\|\varepsilon\| < 0.1$
	0.01	0.7758	0.6012	0.0503	$\|\varepsilon\| < 0.1$
	0.05	0.7335	0.5372	0.0711	$\|\varepsilon\| < 0.1$
	0.1	0.6815	0.4637	0.1015	$\|\varepsilon\| < 0.1$

the MC analysis is converged, and therefore a distinction cannot be made between RIA/PMA formulation error and MC simulation error. Nevertheless, the same optima are reached through both formulations, but RIA fails to converge for one of the given starting points - which is not unexpected given what was said in section 2.3.2.

Alternative PMA Approach

To ascertain the performance impact of the proposed formulation for the FORM subproblem (Eq. 2.39), the RBDO problem defined above is augmented with another random variable (since q has no impact on the constraint). To that end, the constraint is changed to:

$$g\left(\mathbf{x}\right) = x_1^2 + \left(x_2 - h\right)^2 - 1 \leq 0 \tag{2.46}$$

where $\mu_h = 0.5$.

In this case, the deterministic minimum (solution for zero uncertainty) is $\mathbf{x}^* = (0.9306, 0.8659)$. As a reference, using starting point $\mathbf{x_0} = (0.1, 0.1)$, it takes 66 function evaluations to converge to the aforementioned solution. The results for different values of c.o.v. x_1 and σ_h are presented in Table 2.6 and Fig. 2.7. Because there are two random variables in the constraint, the PMA subproblem is only one-dimensional (reverts to polar coordinates with constant radius).

The performance advantage of the alternative approach is clear, reducing the number of function evaluations required by as much as 80%, while the achieved solution is exactly the same as if using conventional PMA. The errors in the final result are again computed by comparison against a Monte Carlo estimate. On another note, the effect of non-deterministic analysis on performance is nonetheless very clear: the required number of function evaluations is increased by up to two orders of magnitude.

If uncertainty is also considered for the previously deterministic variable x_2, bringing the total number of random variables to 3, the alternate PMA formulation maintains the lead, as shown in Table 2.7. In this case, the PMA subproblem requires the usage of spherical coordinates (with constant radius).

The increased complexity of this modified example problem certainly starts to show the limitations of FORM. Although the results of PMA and its alternative form still match, the error in the reliability index at the optimum now almost reaches 4% for the higher input variances.

Table 2.6: RBDO problem (2 RV) - PMA vs. alternative PMA approach

c.o.v. x_1	σ_h	x_1	x_2	F	ϵ_β [%]	#eval.PMA	#eval.PMA alt.
$\beta_{reqd} = 3$							
0.005	0.005	0.9248	0.8551	0.0057	$\lvert\varepsilon\rvert < 0.2$	2402	599
	0.01	0.9201	0.8465	0.0064	≈ -0.3	2750	630
	0.05	0.8931	0.7974	0.0114	$\lvert\varepsilon\rvert < 0.1$	2844	778
	0.1	0.8512	0.7241	0.0222	$\lvert\varepsilon\rvert < 0.1$	2560	787
0.01	0.005	0.9141	0.8354	0.0074	$\lvert\varepsilon\rvert < 0.1$	2493	500
	0.01	0.9132	0.8338	0.0075	$\lvert\varepsilon\rvert < 0.2$	2421	585
	0.05	0.8913	0.7941	0.0118	≈ -0.3	2966	795
	0.1	0.8506	0.7230	0.0224	$\lvert\varepsilon\rvert < 0.1$	2564	795
0.05	0.005	0.8483	0.7194	0.0230	$\lvert\varepsilon\rvert < 0.1$	2068	564
	0.01	0.8482	0.7193	0.0230	$\lvert\varepsilon\rvert < 0.1$	1723	472
	0.05	0.8450	0.7138	0.0240	≈ -1.1	1576	528
	0.1	0.8290	0.6868	0.0293	≈ -2.1	1822	716
0.1	0.005	0.7663	0.5871	0.0546	$\lvert\varepsilon\rvert < 0.1$	1850	491
	0.01	0.7663	0.5871	0.0546	$\lvert\varepsilon\rvert < 0.1$	1880	560
	0.05	0.7660	0.5866	0.0547	≈ -0.7	1836	570
	0.1	0.7647	0.5845	0.0554	≈ -2.7	1413	543

Table 2.7: RBDO problem (3 RV) - PMA vs. alternative PMA approach

c.o.v. x_1, x_2	σ_h	x_1	x_2	F	ϵ_β [%]	#eval.PMA	#eval.PMA alt.
$\beta_{reqd} = 3$							
0.005	0.005	0.9215	0.8489	0.0062	$\lvert\varepsilon\rvert < 0.2$	4083	975
	0.01	0.9198	0.8457	0.0064	≈ -0.3	4158	1197
	0.05	0.8930	0.7972	0.0115	$\lvert\varepsilon\rvert < 0.1$	3994	880
	0.1	0.8512	0.7240	0.0222	$\lvert\varepsilon\rvert < 0.1$	3959	840
0.01	0.005	0.9133	0.8339	0.0075	$\lvert\varepsilon\rvert < 0.2$	3872	681
	0.01	0.9124	0.8322	0.0077	≈ -0.3	4074	696
	0.05	0.8908	0.7933	0.0119	≈ -0.4	3516	891
	0.1	0.8503	0.7226	0.0224	$\lvert\varepsilon\rvert < 0.2$	4720	765
0.05	0.005	0.8467	0.7167	0.0235	≈ -0.6	3859	489
	0.01	0.8466	0.7165	0.0235	≈ -0.7	3277	534
	0.05	0.8430	0.7103	0.0247	≈ -1.6	3881	743
	0.1	0.8259	0.6816	0.0303	≈ -2.1	3566	806
0.1	0.005	0.7659	0.5864	0.0548	≈ -0.9	3394	210
	0.01	0.7659	0.5864	0.0548	≈ -0.9	3218	210
	0.05	0.7655	0.5859	0.0550	≈ -1.5	3307	520
	0.1	0.7637	0.5829	0.0559	≈ -3.9	3302	749

2.4.3 Example R^2BDO problem

Finally, the implementation of the hybrid R^2BDO formulation results in the following problem statement:

$$\begin{aligned} \min_{\mathbf{x}} \quad & F(\mathbf{x}) = \mu_f + \sigma_f \\ \text{subject to:} \quad & g^{rc} \leq 0 \end{aligned} \tag{2.47}$$

where $g^{rc} \leq 0$ is computed as in the first example of the alternative PMA formulation (section 2.4.2) - uncertainty is considered for x_1, q, and h. RIA is not used in this problem as the superiority of PMA was established (particularly the alternative PMA approach). The mean and standard deviation for the objective function are computed using SP. The aim of this example is to compare the accuracy of RDO and R^2BDO when performing optimization for robustness and reliability simultaneously. To that end, in the constraint for RDO:

$$G(\mathbf{x}) = \mu_g + K_\sigma \sigma_g \leq 0 \tag{2.48}$$

the standard deviation of the constraint is affected by a factor $K_\sigma = 3$, which aims at emulating a desired reliability index of 3 as well.

For the sake of readability of the results presented in Tables 2.8 and 2.9, the uncertainties in q and h take the same value for each test case. Once more, the errors refer to a post optimality MC analysis.

Comparison of these two tables shows that the initial assumption that RDO does not adequately model constraint reliability proves to be correct. Although it thrives for low input variances (requiring less function evaluations), the accuracy of the reliability index calculation suffers in comparison to the hybrid R^2BDO formulation as uncertainty increases. Even though the inputs are normal and both formulations share the same objective function, constraint nonlinearity precludes RDO from attaining the desired reliability (just as easily, the estimate could have been too conservative). For even higher input variances, the solution becomes independent of the constraint behavior, as the minimum moves to well within the feasible region due to the uncertainty in the objective ($\beta >> \beta_{reqd}$). In these situations, R^2BDO is dominated by RDO in terms of performance, due to the seemingly unnecessary reliability subproblem.

Table 2.8: R^{2}BDO problem solutions

c.o.v. x_1	σ_h and σ_q	x_1	x_2	$\hat{\mu}_f$	ϵ_{μ_f} [%]	$\hat{\sigma}_f$	ϵ_{σ_f} [%]	ϵ_β [%]	#eval.
$\beta_{reqd} = 3$									
0.005	0.005	0.9219	0.8497	0.0133	$\lvert\varepsilon\rvert < 0.1$	0.0103	$\lvert\varepsilon\rvert < 0.1$	$\lvert\varepsilon\rvert < 0.2$	966
	0.01	0.9202	0.8464	0.0136	$\lvert\varepsilon\rvert < 0.1$	0.0102	$\lvert\varepsilon\rvert < 0.1$	≈ -0.3	1096
	0.05	0.8932	0.7974	0.0178	$\lvert\varepsilon\rvert < 0.1$	0.0091	$\lvert\varepsilon\rvert < 0.1$	$\lvert\varepsilon\rvert < 0.2$	964
	0.1	0.8513	0.7240	0.0274	$\lvert\varepsilon\rvert < 0.1$	0.0078	$\lvert\varepsilon\rvert < 0.1$	$\lvert\varepsilon\rvert < 0.2$	926
0.01	0.005	0.8720	0.7601	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0328	$\lvert\varepsilon\rvert < 0.1$	$\beta >> \beta_{reqd}$	849
	0.01	0.8717	0.7596	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0328	$\lvert\varepsilon\rvert < 0.1$	$\beta >> \beta_{reqd}$	949
	0.05	0.8720	0.7602	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0328	$\lvert\varepsilon\rvert < 0.2$	$\beta \approx 4.4 > \beta_{reqd}$	1234
	0.1	0.8505	0.7231	0.0434	$\lvert\varepsilon\rvert < 0.1$	0.0298	$\lvert\varepsilon\rvert < 0.1$	$\lvert\varepsilon\rvert < 0.1$	894
0.05	0.005	0.4776	0.2265	0.3260	$\lvert\varepsilon\rvert < 0.1$	0.0755	≈ -0.7	$\beta >> \beta_{reqd}$	552
	0.01	0.4777	0.2267	0.3260	$\lvert\varepsilon\rvert < 0.1$	0.0756	≈ -0.6	$\beta >> \beta_{reqd}$	526
	0.05	0.4795	0.2284	0.3249	$\lvert\varepsilon\rvert < 0.1$	0.0778	≈ -0.7	$\beta >> \beta_{reqd}$	653
	0.1	0.4850	0.2338	0.3216	$\lvert\varepsilon\rvert < 0.1$	0.0844	≈ -0.6	$\beta >> \beta_{reqd}$	728
0.1	0.005	0.3291	0.1055	0.4998	$\lvert\varepsilon\rvert < 0.1$	0.0716	≈ -2.0	$\beta >> \beta_{reqd}$	532
	0.01	0.3292	0.1056	0.4998	$\lvert\varepsilon\rvert < 0.1$	0.0718	≈ -2.0	$\beta >> \beta_{reqd}$	554
	0.05	0.3326	0.1080	0.4971	$\lvert\varepsilon\rvert < 0.1$	0.0776	≈ -1.9	$\beta >> \beta_{reqd}$	481
	0.1	0.3398	0.1133	0.4917	$\lvert\varepsilon\rvert < 0.1$	0.0916	≈ -1.8	$\beta >> \beta_{reqd}$	602

Table 2.9: RDO problem solutions

c.o.v. x_1	σ_h and σ_q	x_1	x_2	$\hat{\mu}_f$	ϵ_{μ_f} [%]	$\hat{\sigma}_f$	ϵ_{σ_f} [%]	ϵ_β [%]	#eval.
$\beta_{reqd} = 3$									
0.005	0.005	0.9220	0.8498	0.0133	$\lvert\varepsilon\rvert < 0.1$	0.0103	$\lvert\varepsilon\rvert < 0.1$	≈ -0.7	432
	0.01	0.9203	0.8466	0.0136	$\lvert\varepsilon\rvert < 0.1$	0.0102	$\lvert\varepsilon\rvert < 0.1$	≈ -1.3	468
	0.05	0.8985	0.8070	0.0168	$\lvert\varepsilon\rvert < 0.1$	0.0093	$\lvert\varepsilon\rvert < 0.1$	≈ -13.7	396
	0.1	0.8717	0.7593	0.0223	$\lvert\varepsilon\rvert < 0.1$	0.0084	$\lvert\varepsilon\rvert < 0.1$	≈ -23.4	330
0.01	0.005	0.8717	0.7597	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0328	$\lvert\varepsilon\rvert < 0.1$	$\beta >> \beta_{reqd}$	384
	0.01	0.8719	0.7600	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0328	$\lvert\varepsilon\rvert < 0.1$	$\beta >> \beta_{reqd}$	468
	0.05	0.8721	0.7603	0.0396	$\lvert\varepsilon\rvert < 0.1$	0.0329	$\lvert\varepsilon\rvert < 0.1$	$\beta \approx 4.4 > \beta_{reqd}$	384
	0.1	0.8703	0.7572	0.0399	$\lvert\varepsilon\rvert < 0.1$	0.0326	$\lvert\varepsilon\rvert < 0.1$	≈ -22.6	390
0.05	0.005	0.4777	0.2265	0.3260	$\lvert\varepsilon\rvert < 0.1$	0.0755	≈ -0.7	$\beta >> \beta_{reqd}$	270
	0.01	0.4777	0.2266	0.3260	$\lvert\varepsilon\rvert < 0.1$	0.0755	≈ -0.6	$\beta >> \beta_{reqd}$	264
	0.05	0.4795	0.2283	0.3249	$\lvert\varepsilon\rvert < 0.1$	0.0778	≈ -0.7	$\beta >> \beta_{reqd}$	240
	0.1	0.4850	0.2338	0.3216	$\lvert\varepsilon\rvert < 0.1$	0.0844	≈ -0.6	$\beta >> \beta_{reqd}$	276
0.1	0.005	0.3291	0.1055	0.4998	$\lvert\varepsilon\rvert < 0.1$	0.0716	≈ -2.0	$\beta >> \beta_{reqd}$	258
	0.01	0.3293	0.1056	0.4997	$\lvert\varepsilon\rvert < 0.1$	0.0718	≈ -2.0	$\beta >> \beta_{reqd}$	246
	0.05	0.3326	0.1080	0.4971	$\lvert\varepsilon\rvert < 0.1$	0.0776	≈ -2.0	$\beta >> \beta_{reqd}$	186
	0.1	0.3399	0.1133	0.4917	$\lvert\varepsilon\rvert < 0.1$	0.0916	≈ -1.7	$\beta >> \beta_{reqd}$	252

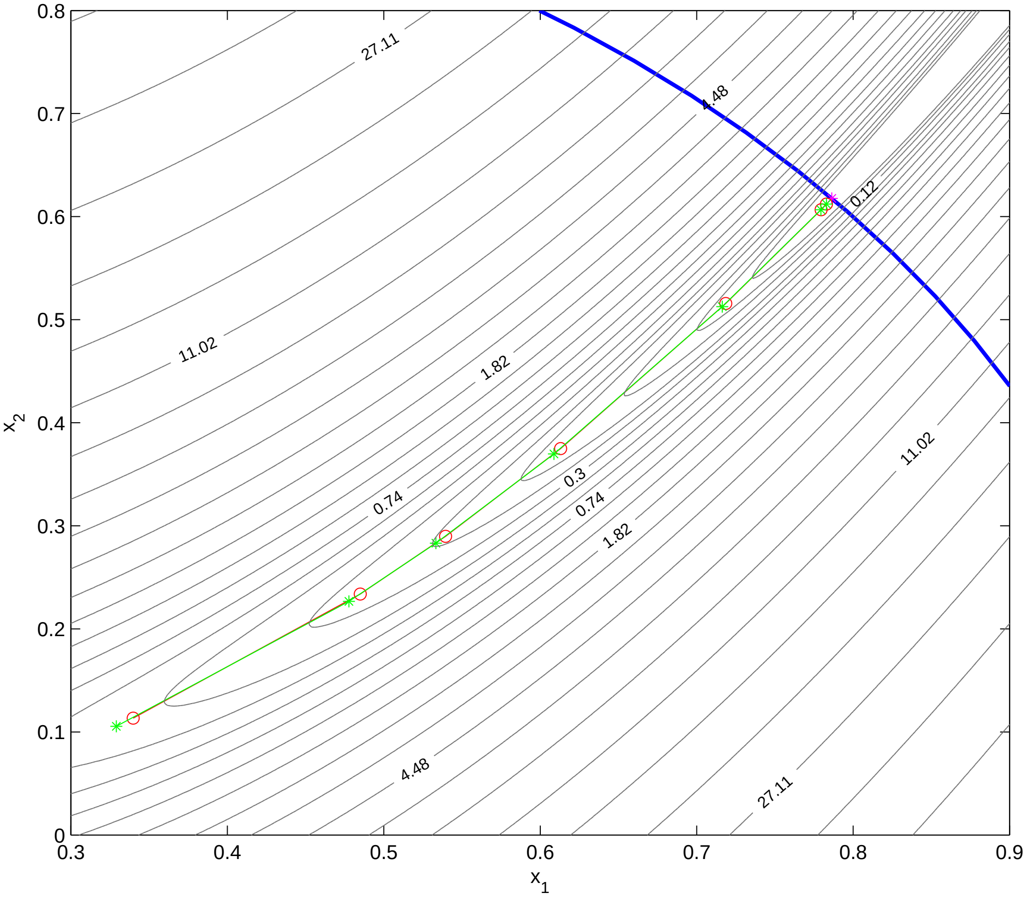


Figure 2.4: RDO example problem solution for values of the c.o.v. in x_1 ranging from 0.005 to 0.1, $\sigma_q = 0.005$ (green "*" symbol) and $\sigma_q = 0.1$ (red "o" symbol). Contour plot is that of the deterministic objective function.

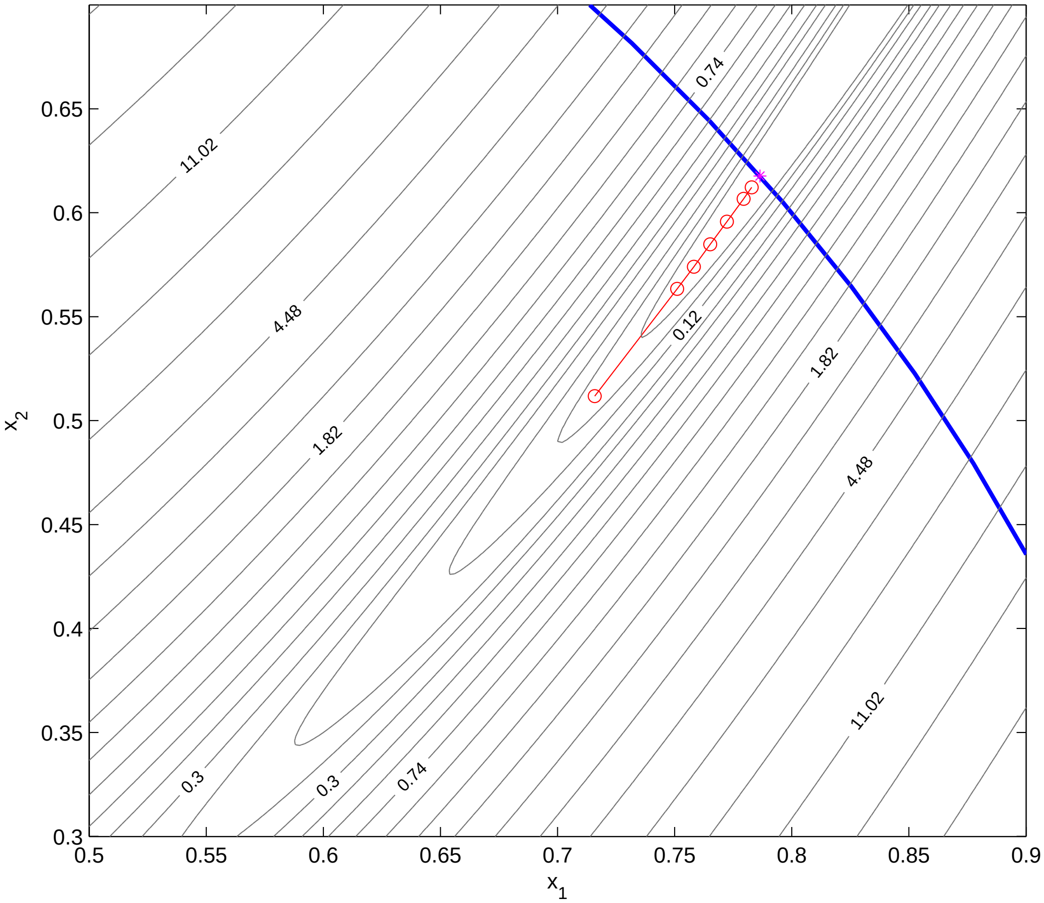


Figure 2.5: RBDO example problem solution for values of the c.o.v. in x_1 ranging from 0.005 to 0.1, $\beta_{reqd} = 2$. Contour plot is that of the deterministic objective function.

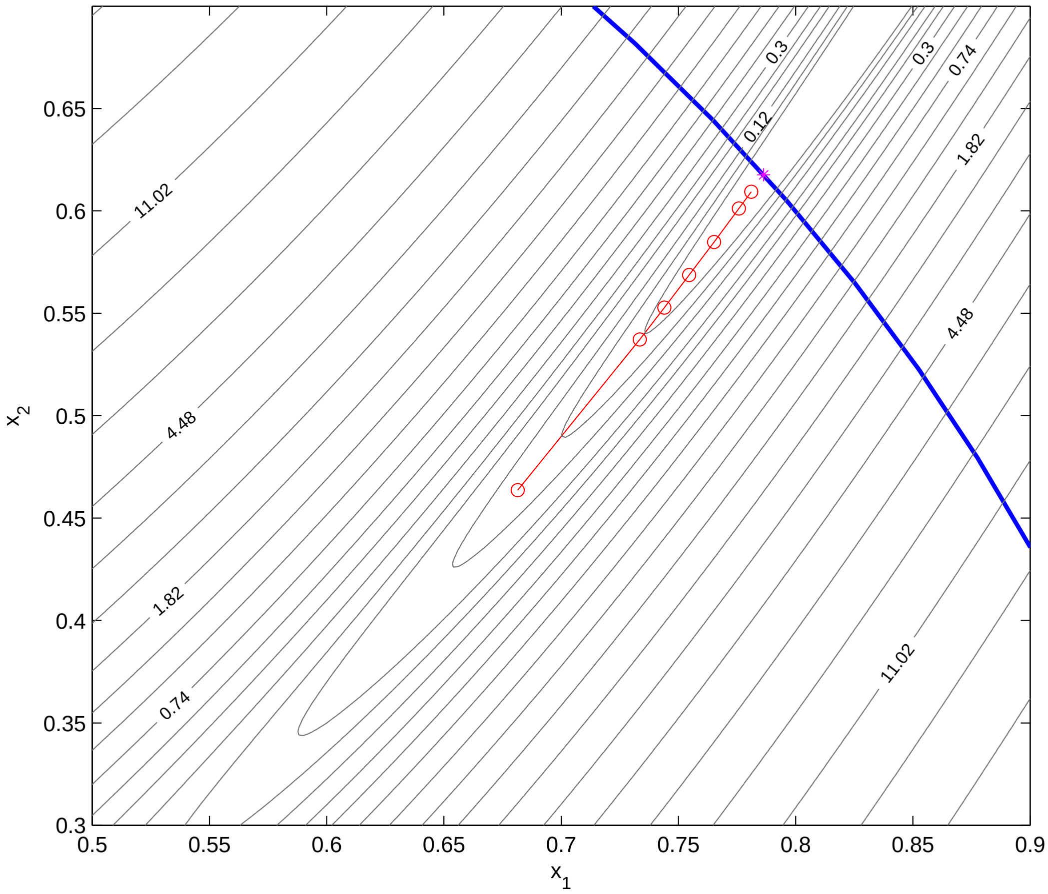


Figure 2.6: RBDO example problem solution for values of the c.o.v. in x_1 ranging from 0.005 to 0.1, $\beta_{reqd} = 3$. Contour plot is that of the deterministic objective function.

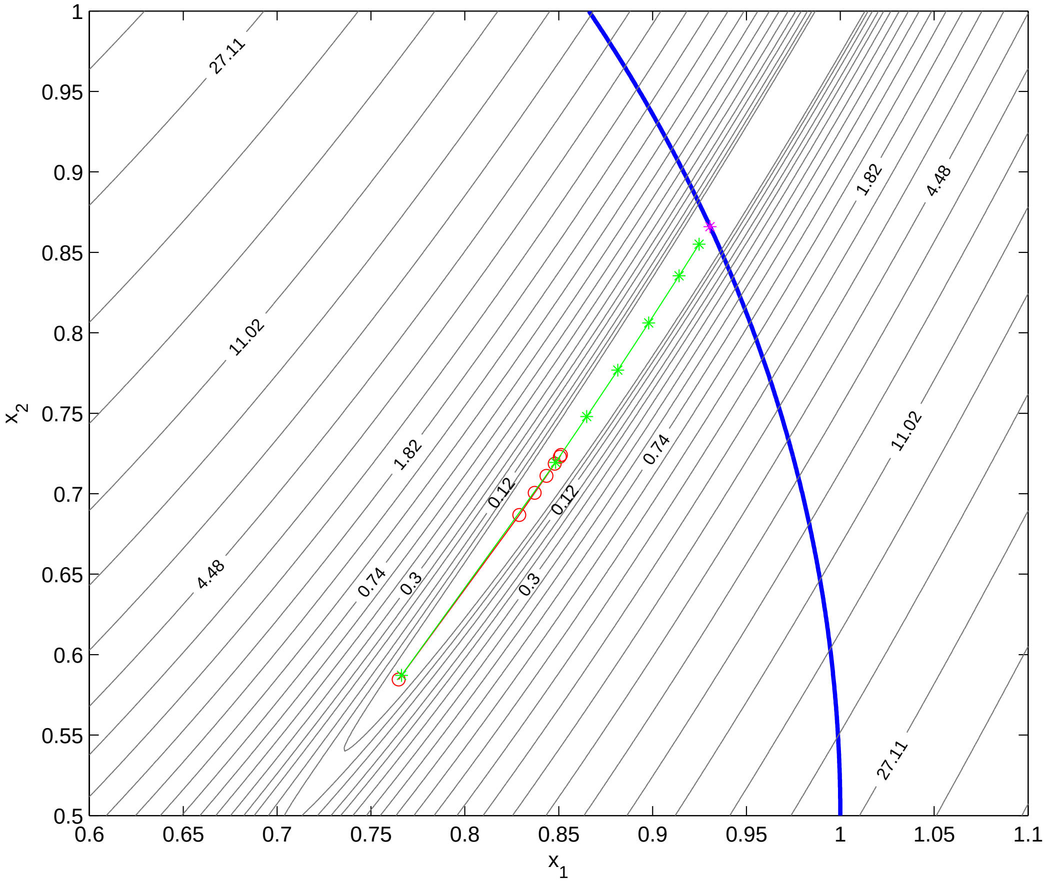


Figure 2.7: RBDO (2 RV) example problem solution for values of the c.o.v. in x_1 ranging from 0.005 to 0.1, $\sigma_h = 0.005$ (green “*” symbol) and $\sigma_h = 0.1$ (red “o” symbol). Contour plot is that of the deterministic objective function.

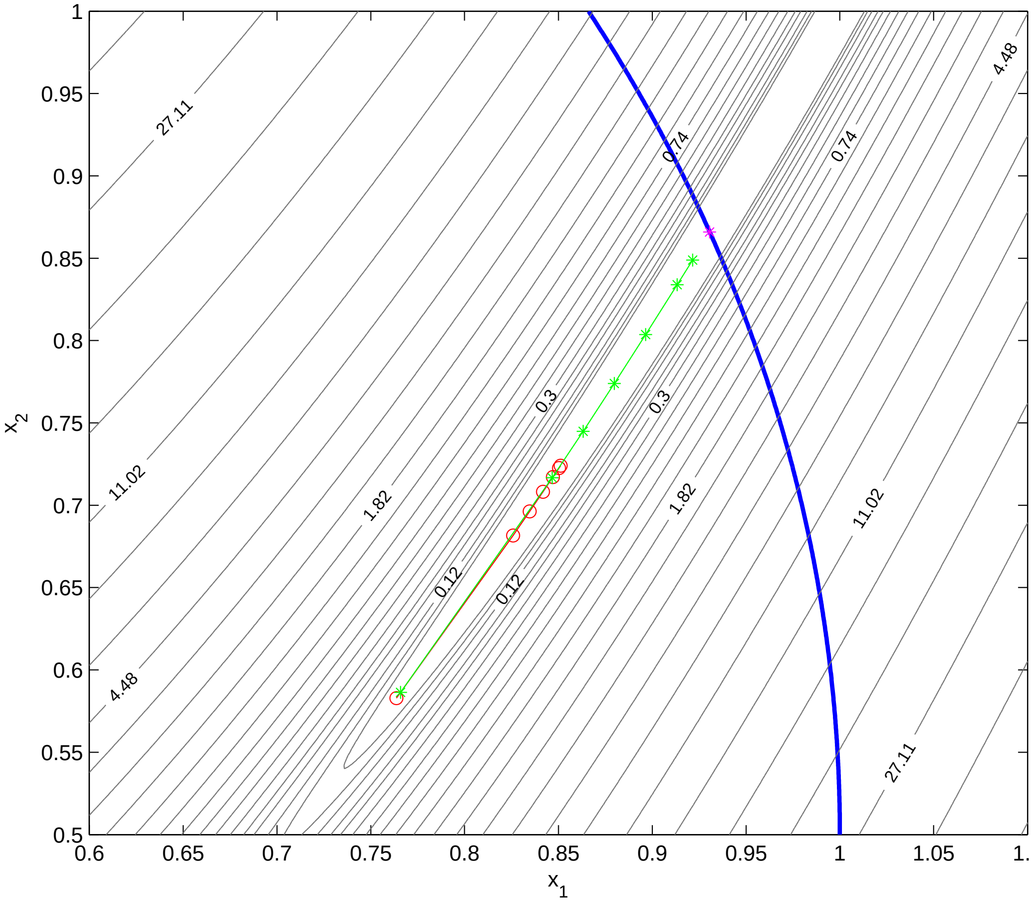


Figure 2.8: RBDO (3 RV) example problem solution for values of the c.o.v. in x_1/x_2 ranging from 0.005 to 0.1, $\sigma_h = 0.005$ (green "*" symbol) and $\sigma_h = 0.1$ (red "o" symbol). Contour plot is that of the deterministic objective function.

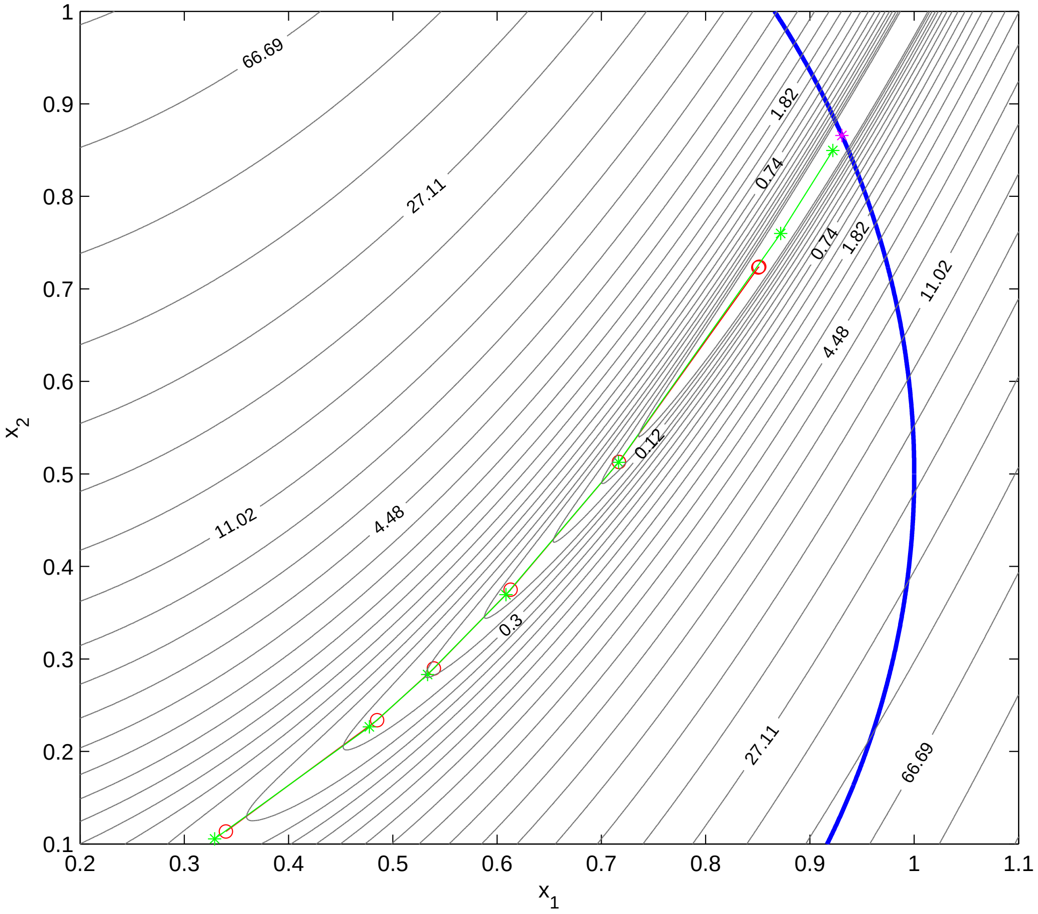


Figure 2.9: R^2BDO example problem solution for values of the c.o.v. in x_1 ranging from 0.005 to 0.1, $\sigma_h = \sigma_q = 0.005$ (green "*" symbol) and $\sigma_h = \sigma_q = 0.1$ (red "o" symbol), $K_\sigma = \beta_{reqd} = 3$. Contour plot is that of the deterministic objective function.

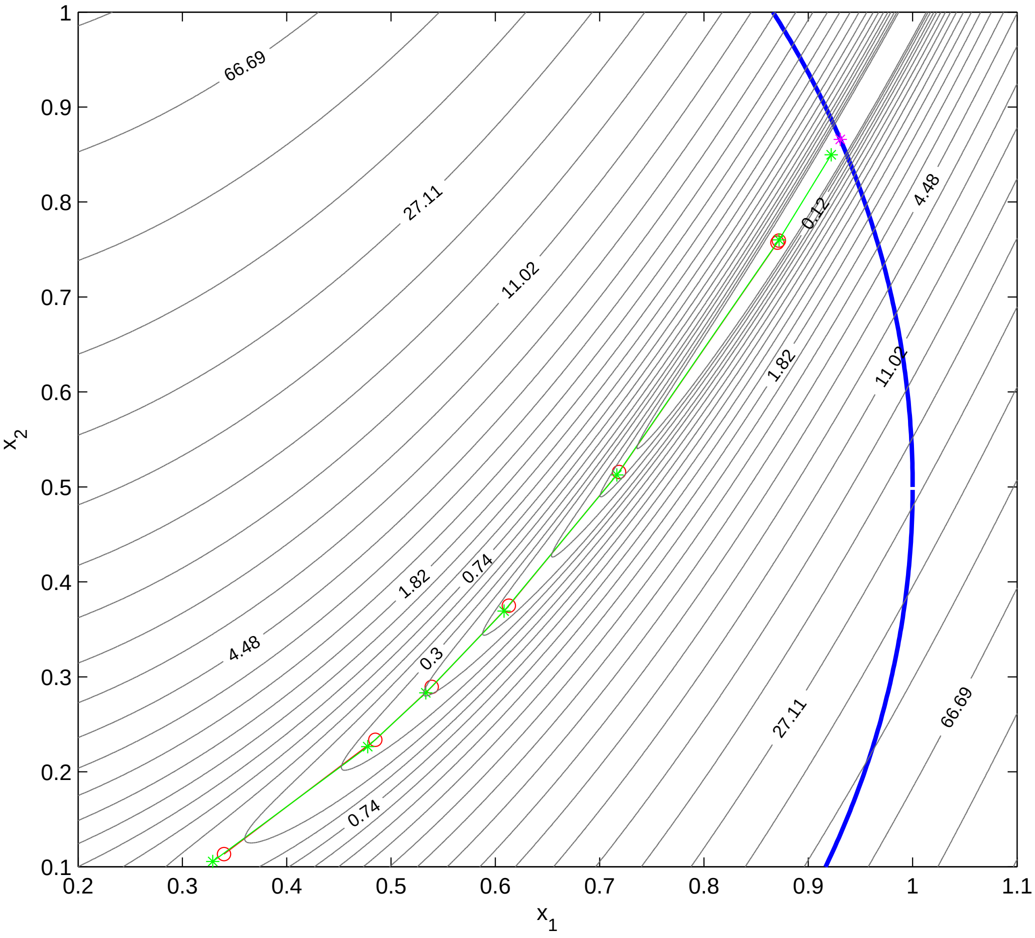


Figure 2.10: RDO example problem solution for values of the c.o.v. in x_1 ranging from 0.005 to 0.1, $\sigma_h = \sigma_q = 0.005$ (green "*" symbol) and $\sigma_h = \sigma_q$= 0.1 (red "o" symbol), $K_\sigma = \beta_{reqd} = 3$. Contour plot is that of the deterministic objective function.

Chapter 3

Surrogate models

As mentioned before, both RDO and RBDO are computationally much heavier than conventional deterministic optimization due to either the additional variables that are introduced in the problem or the required evaluation of probabilities. By replacing the computationally expensive analyzers (for aerodynamics, structural analysis, etc.) with suitable surrogate models/metamodels, the optimization process can be greatly accelerated even if at the expense of a loss in fidelity [9]. Much like with RDO and RBDO, in R^{2}BDO there should also be the possibility of using surrogate models to replace objective and constraint functions, especially in regards to limit state functions and the determination of the MPP.

Three response surface/metamodel generating methods were implemented: quadratic interpolation, Kriging and neural networks. The following sections serve as an introduction to each of them in some detail.

3.1 Quadratic interpolation

Quadratic interpolation based surrogate models are amongst the fastest available (in terms of execution speed at both creation and evaluation time) and their implementation is rather trivial [11]. As the name implies they involve the fitting of a second order polynomial to the function of interest (in some applications, incomplete versions of these polynomials are used, as some cross coupling terms are excluded). Although this approach may not be adequate should the interpolated function exhibit oscillatory behavior within its domain, it excels as an inexpensive global model.

The interpolating polynomial is therefore written as:

$$\hat{y} = c_0 + \sum_{j=1}^{n_{DV}} c_j x_j + \sum_{k=1}^{n_{DV}} \sum_{j=k}^{n_{DV}} x_k x_j c_{1+n_{DV}+\sum_{m=1}^{k-1}(n_{DV}-m+1)+j-k} \tag{3.1}$$

The total number of polynomial coefficients (n_t) is in this case equal to the the number of samples to be taken (n_s):

$$n_t = \frac{(n_{DV}+1)(n_{DV}+2)}{2} = n_s \tag{3.2}$$

The analytical derivative of the quadratic polynomial is readily calculated as:

$$\frac{d\hat{y}}{dx_i} = c_i + \sum_{\substack{k=1 \\ k\neq i}}^{n_{DV}} c_{F(i,\ k)} x_k + 2x_i c_{1+n_{DV}+\sum_{m=1}^{i-1}(n_{DV}-m+1)} \tag{3.3}$$

$$F(i,\ k) = \begin{cases} 1 + n_{DV} + \left(\sum\limits_{m=1}^{k-1} n_{DV} - m + 1\right) + i - k, & k < i \\ 1 + n_{DV} + \left(\sum\limits_{m=1}^{i-1} n_{DV} - m + 1\right) + k - i, & k > i \end{cases} \tag{3.4}$$

The fact that sensitivities may be computed analytically is an advantage of such models. Furthermore, the natural smoothness of the polynomial filters out local disturbances in the interpolated function (which may be due to numerical errors, mesh adaptation, etc.). This ensures that, in most cases, these sensitivities do represent the global design trends (though the method may fill in local extreme).

The polynomial coefficients ($\mathbf{c}$) are obtained by solving the linear system of equations which equates the values of the function at the sample points ($\mathbf{y_s}$) to the value of the polynomial at those same points, thereby employing exact interpolation.

$$\mathbf{y_s} = \mathbf{X}\ \mathbf{c} \tag{3.5}$$

$$\mathbf{X} = \begin{bmatrix} 1 & x_1^{(1)} & x_2^{(1)} & \cdots & \left(x_1^{(1)}\right)^2 & x_1^{(1)} x_2^{(1)} & \cdots & \left(x_{n_{DV}}^{(1)}\right)^2 \\ & \vdots & & \ddots & & \vdots & & \vdots \\ 1 & x_1^{(n_s)} & x_2^{(n_s)} & \ldots & \left(x_1^{(n_s)}\right)^2 & x_1^{(n_s)} x_2^{(n_s)} & \cdots & \left(x_{n_{DV}}^{(n_s)}\right)^2 \end{bmatrix} \tag{3.6}$$

$$\mathbf{c} = \begin{bmatrix} c_0 & c_1 & \cdots & c_{n_s-1} \end{bmatrix}^{\mathrm{T}} \tag{3.7}$$

$$\mathbf{c} = \mathbf{X}^{-1}\,\mathbf{y_s} \tag{3.8}$$

3.2 Kriging

Kriging is a statistical interpolation method, initially designed to predict the size of oil reserves in the mining industry. Originally introduced by Krige (hence the designation) in 1951, this field of geostatistics would not be established until a decade later [59].

Some recent examples of the use of Kriging to develop surrogates to be used in design optimization are the work of Huang [60], Lee and Park, [61] and Simpson et al. [62]. Several types of Kriging have been employed thus far. Some of the most well known variants are Simple Kriging (SK), Ordinary Kriging (OK), and Regression Kriging (RK). The latter is the one which was chosen for use in the present work. Regression Kriging is a hybrid approach between a (usually polynomial) regression and Ordinary Kriging models. It is based on the assumption that the value of a given multivariable function, $y(\mathbf{x})$, can be decomposed into deterministic and stochastic components:

$$y(\mathbf{x}) = f(\mathbf{x}) + z(\mathbf{x}) \tag{3.9}$$

$f(\mathbf{x})$ represents the deterministic model, $z(\mathbf{x})$ are the interpolated residuals and $\mathbf{x}$ is a location of interest, where no samples were taken. While the regression model describes the behavior of the function on a global level, the local discrepancies between such a model and the actual function are taken into account through the Kriging model. The global regression model takes the form:

$$f(\mathbf{x}) = \sum_{k=1}^{n_t} \beta_k \; q_k(\mathbf{x}) \tag{3.10}$$

q_k are a set of predictor functions, n_t is the total number of terms used in the regression, and β_k are their respective weights determined by means of fitting the sample data; in the present case generalized least squares is employed. Least squares fitting of the polynomial approximations in section 3.1 was not pursued because of diminishing returns of a non locally refined and non-interpolating approximation. The current implementation supports polynomials up to second order as the prediction functions.

On the other hand, $z(\mathbf{x})$ is assumed to have zero mean and covariance defined by:

$$E\left[z(\mathbf{x})\; z(\mathbf{w})\right] = \sigma^2\, \mathcal{R}\left(\mathbf{w}, \mathbf{x}, \theta\right) \tag{3.11}$$

where σ is the standard deviation and $\mathcal{R}\left(\mathbf{w}, \mathbf{x}, \theta\right)$ is a correlation function defined between points $\mathbf{x}$ and $\mathbf{w}$ (see table below), fitted to sample data by means of the parameters θ. In the Kriging approximation, the samples are weighted in the following manner:

$$\hat{y}(\mathbf{x}) = \sum_{k=1}^{n_s} \lambda_k\; y(\mathbf{s}_k) \tag{3.12}$$

where $y(\mathbf{s}_k)$ are the values of the function at the sample points $(\mathbf{s}_k)$, and λ_k are the Kriging weights which need to be estimated. For the purposes of implementing a Kriging approximation of the objective and constraint functions in the MDO problems, a MATLAB® toolbox - DACE - was converted to C#/.NET and hence designated DACE.NET. The detailed procedure of obtaining the weights in Regression Kriging escapes, however, the scope of this thesis, as it is thoroughly explained in the manual accompanying the toolbox [63] as well as in other publications [59, 64].

Table 3.1: Examples of correlation models for Regression Kriging

Name	Correlation model
Exponential	$e^{-\lvert w_j - x_j \rvert\, \theta_j}$
General Exponential	$e^{-\lvert w_j - x_j \rvert^p\, \theta_j}, \quad 0 < p \leq 2$
Gaussian	$e^{-(w_j - x_j)^2\, \theta_j}$
Spherical	$1 - 1.5\,\xi_j + 0.5\,\xi_j^3, \quad \xi_j = \min\{1,\ \theta_j\, \lvert w_j - x_j \rvert\}$
Linear	$\max\{0,\ 1 - \theta_j\, \lvert w_j - x_j \rvert\}$
Cubic	$1 - 3.0\,\xi_j + 2.0\,\xi_j^3, \quad \xi_j = \min\{1,\ \theta_j\, \lvert w_j - x_j \rvert\}$

Derivatives from the Kriging model are also easy to obtain since the derivatives from the regression and correlation functions are well known [63, 59].

3.3 Artificial Neural Networks

Artificial Neural Networks (ANN) were first theorized by Pitts and McCulloch in 1943 [65]. Based on their knowledge of the operation of organic brains, Pitts and McCulloch established several network configurations for logical neurons. In the years that followed, ANN were studied in great detail by the mathematical and computational analysis community. In time, this lead to major breakthroughs such as the development of the first learning rule by Hebb in 1949, the first *perceptron* by Rosenblatt in 1958, and the Adaptive Linear Element (ADELINE) by Widrow and Hoff in 1960 [66]. Nevertheless, in the following decades, there was a drastic decrease in interest in ANN since additional developments in the area would have required computational power not yet available at the time. This situation lasted until the early 1980s, when digital microprocessors began to see widespread use. In light of these technological achievements, ANN research regained some of the momentum it once had with the development of the associative memory network, by Hopfield, and with the Self-Organizing map, by Kohonen, both developed in 1982 [66, 67]. Nowadays there are over 20 different types of ANN used in a vast range of applications ranging from non-linear control to data mining.

The basic unit of ANN is the artificial neuron, which in its simpler form is referred to as a *perceptron*. The *perceptron* (see Fig. 3.1) is a functional with two components: a weighted summation of the inputs (Eq. 3.13), and an activation function (Eq. 3.14) [66, 68].

$$n = W_j x_j + \beta \tag{3.13}$$

$$y = \gamma\left(n\right) \tag{3.14}$$

where x_j are the inputs of the *perceptron*, β is a bias and W_j are the weights of each input. γ and y are the activation function and the result of the functional, respectively.

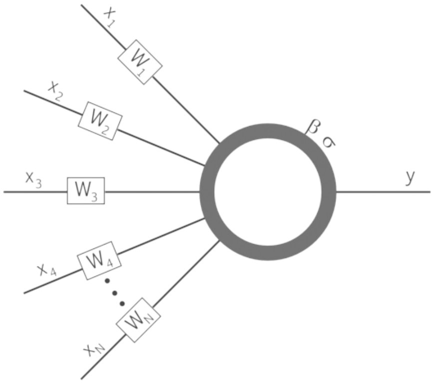


Figure 3.1: *Perceptron* - simplest form of the artificial neuron

The activation function can be any monotonic function, but the following are the most commonly used: origin crossing linear functions, hyperbolic tangents, logistic functions or the sign function. Usually, linear functions are used in the input and output layers of a network; the sign function is used for biological inspired applications or in digital networks; the hyperbolic tangent is preferentially used in system identification and control; and the logistic function in data mining [66, 68]. There are other, more complex, types of artificial neurons (e.g. Radial Basis Perceptrons) but in this work only the regular *perceptrons* were employed.

The Multi-layer *perceptron* (MLP) is one of the configurations proposed by Pitts and McCulloch and is the most versatile and simple network structure. As the name implies the MLP is comprised of a series of sequentially connected layers of *perceptrons*. A layer is defined as a group of *perceptrons* sharing a common set of inputs and, in the case of the MLP, there cannot be intra-layer connections between *perceptrons* (Fig. 3.2) [66].

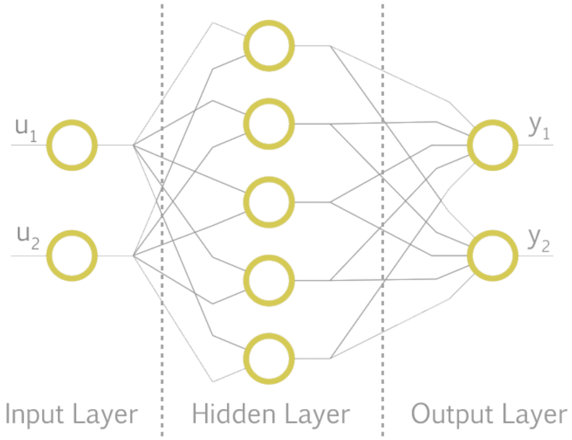


Figure 3.2: Multi-layer *Perceptron* showing the three types of layers

The main advantage of neural networks is their adaptive nature. The process by which an MLP learns is called the back-propagation algorithm. The back-propagation algorithm systematically applies a gradient-based optimization algorithm to each layer of the MLP. The back-propagation is divided in two parts: in the first the sensitivities for each layer are evaluated, starting from the output layer and ending on the first hidden layer; in the second part the weights are updated according to the computed sensitivities, this process starts in the first hidden layer and ends on the output layer.

The mathematical equations of the algorithm are highly dependent on the optimization method being used. For the steepest descent, which is one of the simplest and most common algorithms, Eqs. 3.15 and 3.16 represent the first and second parts, respectively:

$$\delta_i^m = \gamma'_{ij}\; W_{jk}^{m+1} \delta_k^{m+1} \tag{3.15}$$

$$\begin{cases} W_{ij}^m = W_{ij}^m + 2\,\eta\,\delta_i^m x_j^m \\ \beta_i^m = \beta_i^m + 2\,\eta\,\delta_i^m \end{cases} \tag{3.16}$$

where δ_i^m is the sensitivity for neuron i of layer m, γ'_{ij} is the Jacobian matrix of the activation functions of the layer, W_{ij}^m and β_i^m are the weight matrix and bias vector for layer m and η is the learning rate.

The implementation of the neural network model was done in C++. The neural network is enclosed in a class that has methods for training and testing the network. From these methods, one can obtain the results of the surrogate model and the corresponding derivative for a given input point. In order to be used with the MDO tool, the class was implemented in a dynamic library, thus maximizing portability.

3.4 Sampling

For the purposes of generating the samples required to build the surrogate models, both a random and a Latin Hypercube samplers were implemented. The process of creating the sample space, though customizable by the user, is oriented towards maximizing the number of samples lying within the constrained design space while minimizing the number of actual objective/constraints functions evaluations. To that end, a newly generated sample (which is already within the "hard" bounds) is submitted to the tests illustrated in Fig. 3.3, in which the distance to previously taken samples and linear constraints are computed first to avoid a possibly superfluous MDA (Multidisciplinary Analysis) evaluation. In the case of nonlinear constraints, after a number of failed tries the algorithm finally includes the constraint violating point into the sampling set. This situation frequently occurs near the intersection of two or more constraints as illustrated in Fig. 3.4 (the size of the clearance areas is exaggerated but the number of samples is reduced in order to favor understandability). To further ameliorate this issue, there is the option in the algorithm to include a tolerance for infeasible points (with respect to nonlinear constraints, i.e., $g(\mathbf{x}) \leq \epsilon, \epsilon > 0$, instead of $g(\mathbf{x}) \leq 0$).

Because an initial model will generally provide poor coverage and for that reason

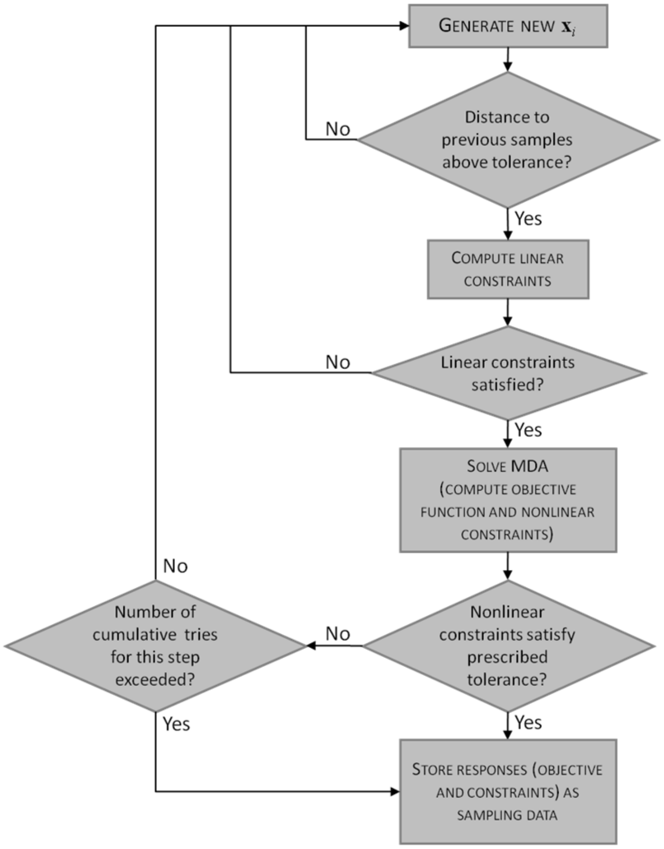


Figure 3.3: Sampling algorithm with constraint based filtering

the optimum found may still be far from a true optimum, an adaptive trust region can be defined around this point so that the sample is updated and further refined around this area of interest. Hence, the sampling procedure described above is executed only once on the overall design space and subsequently it is confined to the limits of the trust region, which is also illustrated on Fig. 3.4.

3.4.1 Latin Hypercube

An alternative approach to simply sampling at random is Latin Hypercube Sampling (LHS), which is a space filling technique, i.e., it ensures that at least one sample lies in a particular region of the design space. The method was originally developed by McKay, Conover, and Beckman in 1979 [69]. In LHS, the range of each variable to be sampled is divided into N intervals (bins) of equal probability and then one random value is taken from each of those intervals (based on the required proba-

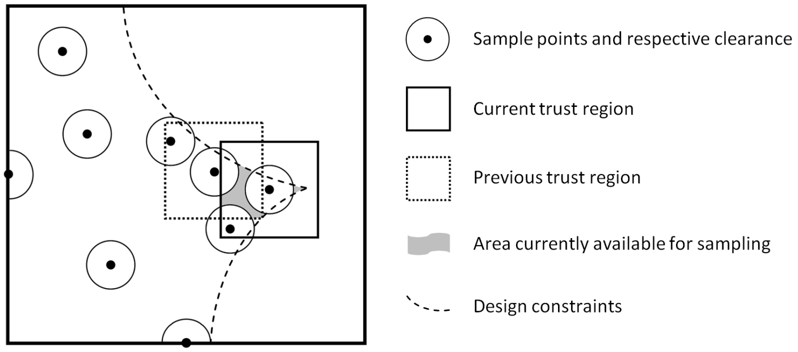


Figure 3.4: Sampling region in the vicinity of constraints

bility density). The N samples from each variable are randomly combined (random permutations) with similar samples from other variables in order to form the Latin Hypercube sample.

3.5 Application to R^2BDO

The R^2BMDO framework is to make use of surrogate models to expedite the evaluation of objective/constraint functions and their statistical dispersion measures if need be. For further clarification, a flowchart of the tasks involved in the R^2BMDO framework is presented in Fig. 3.5.

3.5.1 Kriging RDO/R^2BDO example problem

In the current example, based on the R^2BDO problem from the previous chapter, a regression Kriging model is used to perform the required objective/constraint function evaluations rather than the actual functions themselves. Kriging was selected given that it demonstrated superior performance/stability compared to the other two types of models hereby considered (see next chapter, section 5.2). There is a choice over how to create the Kriging model for robustness/reliability functions. The most straightforward manner in which to do this would be to approximate the robust objective/reliability constraints directly, based on samples taken from the results of the SP approach and FORM subproblem solutions, respectively. However, it was observed that, as suggested in [70], this leads to a higher modeling error than if SP and FORM are applied to surrogates of the original (deterministic) objective and constraints. As

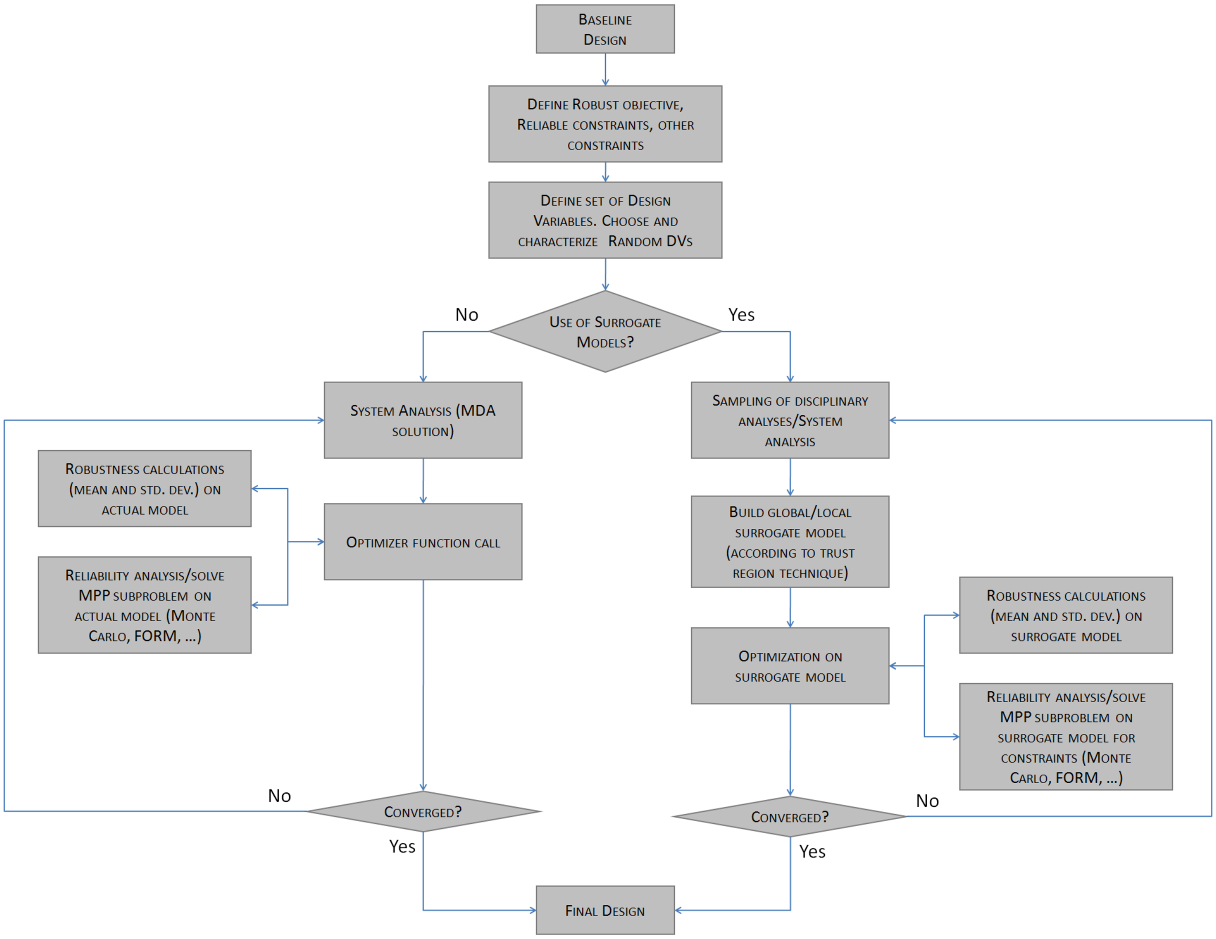


Figure 3.5: R^{2}BMDO framework layout

such, and referring back to the nomenclature used in the previous chapter, the Kriging model will approximate f and g, and only then are F and g^{rc} built. The results in Table 3.2 reflect the usage of a Kriging surrogate for various uncertainty levels in the example problem from section 2.4.3 (40 independent runs for each problem). The initial sample is comprised of 25 evaluations and from then on between 4 to 8 samples (depending on the level of uncertainty) are taken inside the moving trust region. The regression component of the Kriging model is a second order polynomial and a Gaussian correlation model is used.

For cases where the influence of the reliability constraint manifests itself, it is clear that the use of surrogate models to simulate such a constraint is advantageous. Considerable savings are observed, in terms of the number of function evaluations and, in these situations, the RMS error (measured with respect to the reference solu-

Table 3.2: R^{2}BDO problem solutions (Kriging approximation)

c.o.v. x_1	σ_h and σ_q	$\overline{x_1}$	$\overline{x_2}$	RMSE, $\mathbf{x_{opt}}$	ϵ_β (max.) [%]	# f. eval.	
$\beta_{reqd} = 3$						mean	std.dev.
0.005	0.005	0.9219	0.8497	1.2E-05	$\lvert\varepsilon\rvert < 0.2$	196	47
	0.1	0.8513	0.7240	1.2E-05	$\lvert\varepsilon\rvert < 0.2$	166	42
0.01	0.005	0.8741	0.7638	0.0067	$\beta >> \beta_{reqd}$	192	42
	0.1	0.8506	0.7230	4.0E-05	$\lvert\varepsilon\rvert < 0.2$	155	34
0.05	0.005	0.4776	0.2265	9.9E-04	$\beta >> \beta_{reqd}$	434	156
	0.1	0.4848	0.2336	0.0039	$\beta >> \beta_{reqd}$	307	125
0.1	0.005	0.3291	0.1054	0.0016	$\beta >> \beta_{reqd}$	266	86
	0.1	0.3397	0.1132	0.0015	$\beta >> \beta_{reqd}$	272	94

Table 3.3: RDO problem solutions (Kriging approximation)

c.o.v. x_1	σ_h and σ_q	$\overline{x_1}$	$\overline{x_2}$	RMSE, $\mathbf{x_{opt}}$	ϵ_β (max.) [%]	# f. eval.	
$\beta_{reqd} = 3$						mean	std.dev.
0.005	0.005	0.9220	0.8498	1.4E-05	≈ -0.8	173	43
	0.1	0.8717	0.7593	1.2E-05	≈ -23.4	181	41
0.01	0.005	0.8748	0.7649	0.0078	$\beta >> \beta_{reqd}$	134	28
	0.1	0.8703	0.7571	4.4E-05	≈ -22.6	146	23
0.05	0.005	0.4773	0.2262	0.0014	$\beta >> \beta_{reqd}$	169	47
	0.1	0.4848	0.2336	0.0023	$\beta >> \beta_{reqd}$	202	63
0.1	0.005	0.3289	0.1053	8.7E-04	$\beta >> \beta_{reqd}$	181	55
	0.1	0.3399	0.1133	2.1E-04	$\beta >> \beta_{reqd}$	186	52

tions from chapter 2) is also within reasonable bounds. The errors in the reliability index (where applicable) remain mostly unchanged which means the FORM constraint/robust constraint are being adequately captured by the model. On the other hand, when the level of uncertainty in the objective drives the optimum point inwards, towards the center of the feasible region, the Kriging model looses its edge. The solutions returned are then dispersed around the true optimum (some examples for both R^{2}BDO and RDO are presented in Figures 3.6 to 3.9), and take longer to be achieved. This happens both for RDO and R^{2}BDO, which means the robust objective is the problematic function in this case. The reason for this is linked with the fact that, for these uncertainty values, the function experiences extremely low gradient values in a wide area around the minimum making convergence of the optimization on the surrogate model difficult. Therefore, although these false optima are very

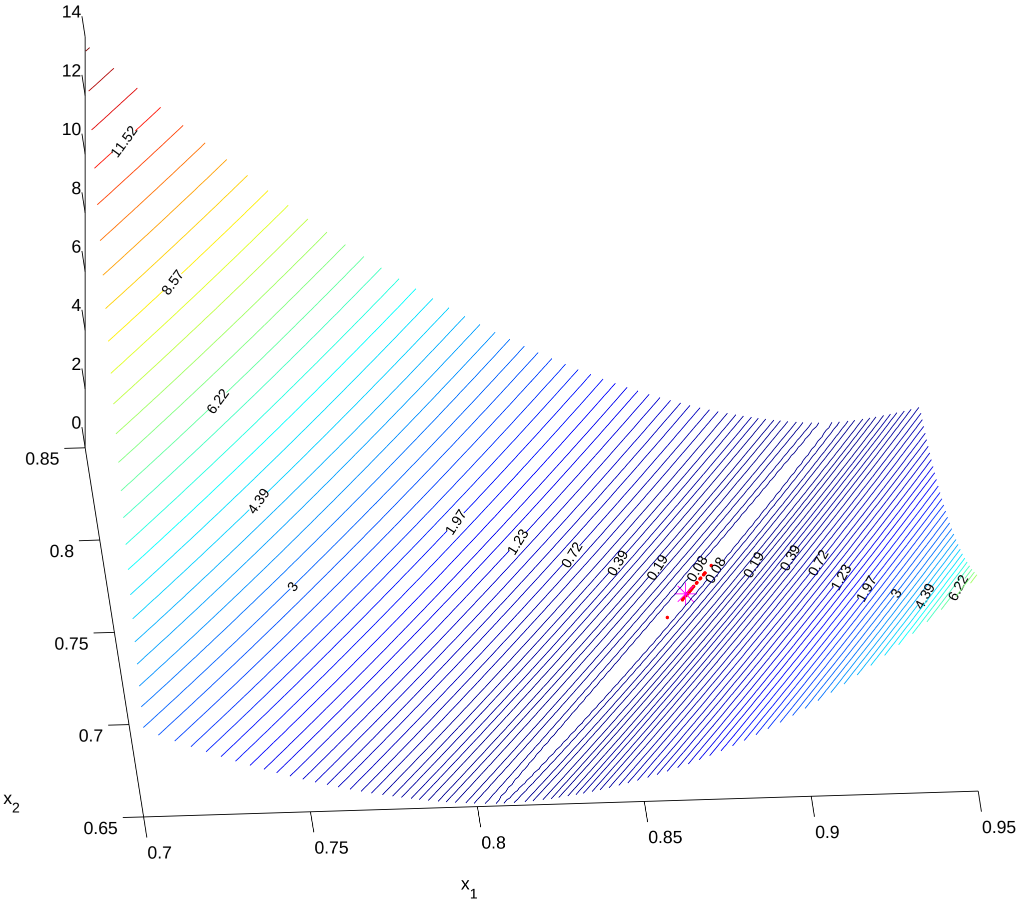


Figure 3.6: Kriging solutions of the R^{2}BDO example problem, c.o.v. x_1 = 0.01, σ_h = σ_q = 0.005 (reference solution represented by the '*' symbol). The contour plot is that of the robust objective function computed through SP.

close to the true solutions, if further refinement is needed, the framework is to revert to direct evaluation upon reaching the optimum on the surrogate. This is the strategy adopted when solving the aircraft design problems presented in chapter 7.

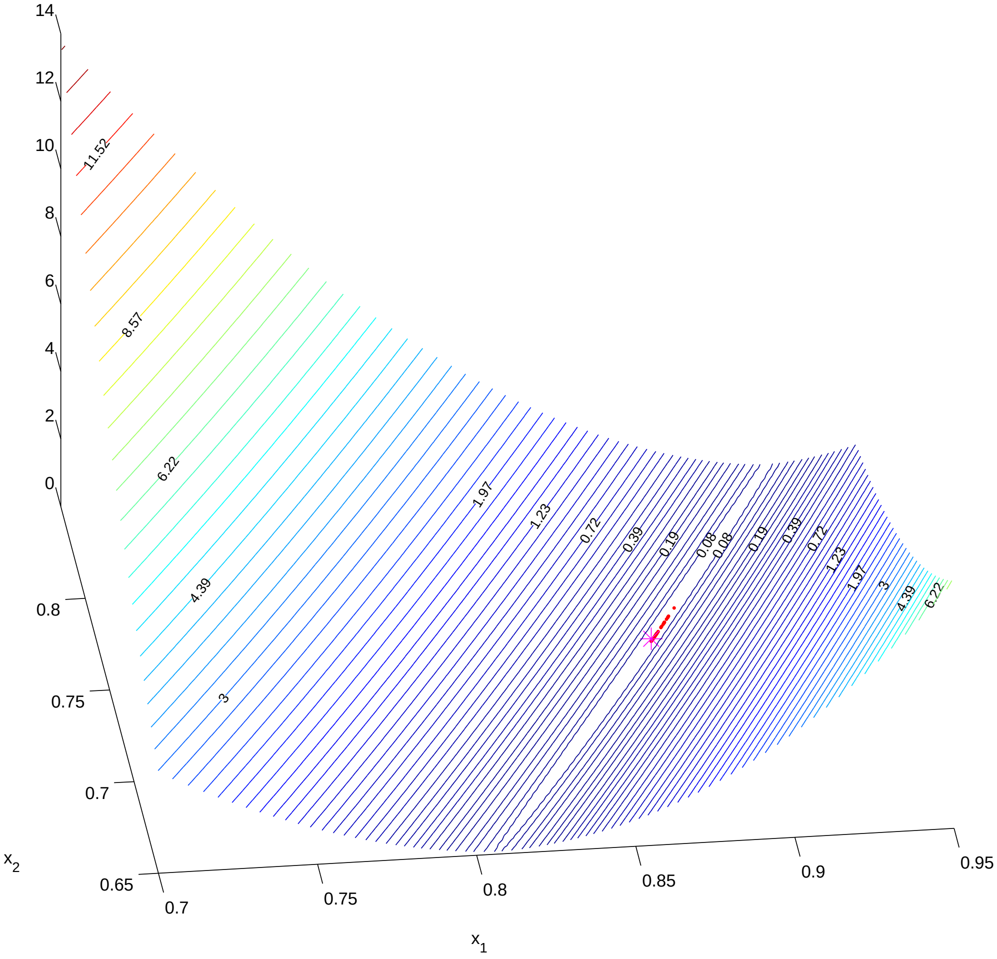


Figure 3.7: Kriging solutions of the RDO example problem, c.o.v. $x_1 = 0.05$, $\sigma_h = \sigma_q = 0.1$ (reference solution represented by the '*' symbol). The contour plot is that of the robust objective function computed through SP.

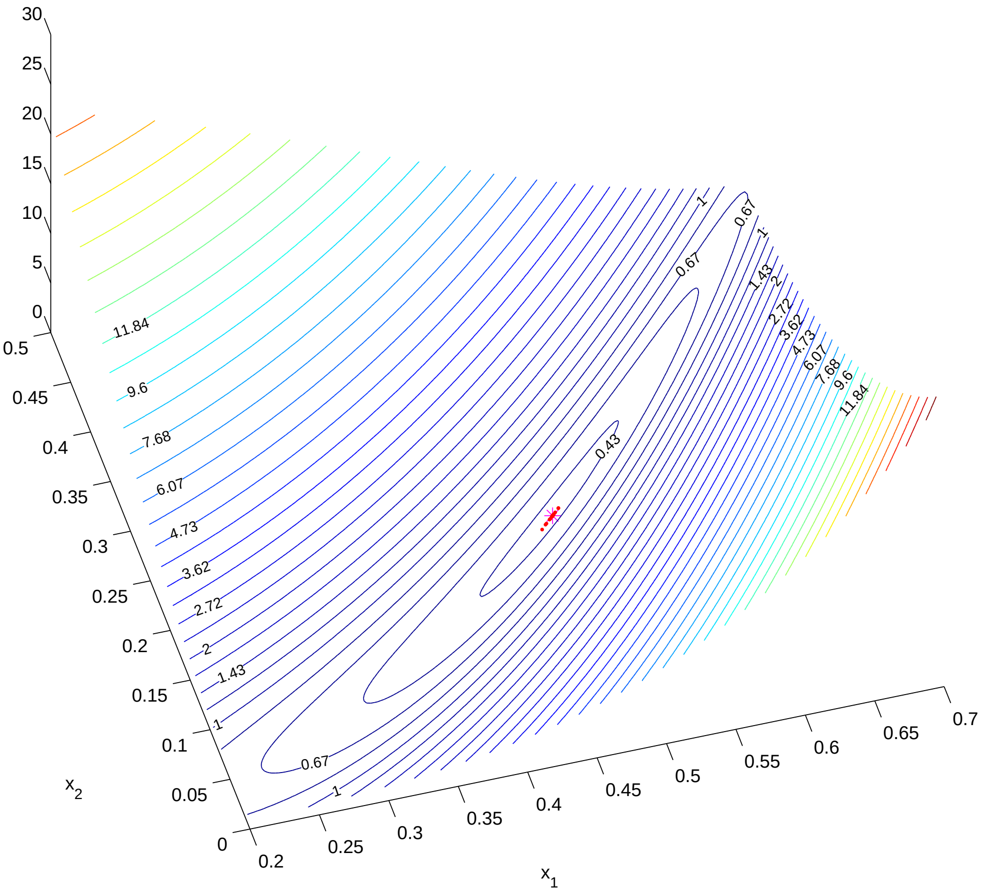


Figure 3.8: Kriging solutions of the R^{2}BDO example problem, c.o.v. $x_1 = 0.05$, $\sigma_h = \sigma_q = 0.1$ (reference solution represented by the '*' symbol). The contour plot is that of the robust objective function computed through SP.

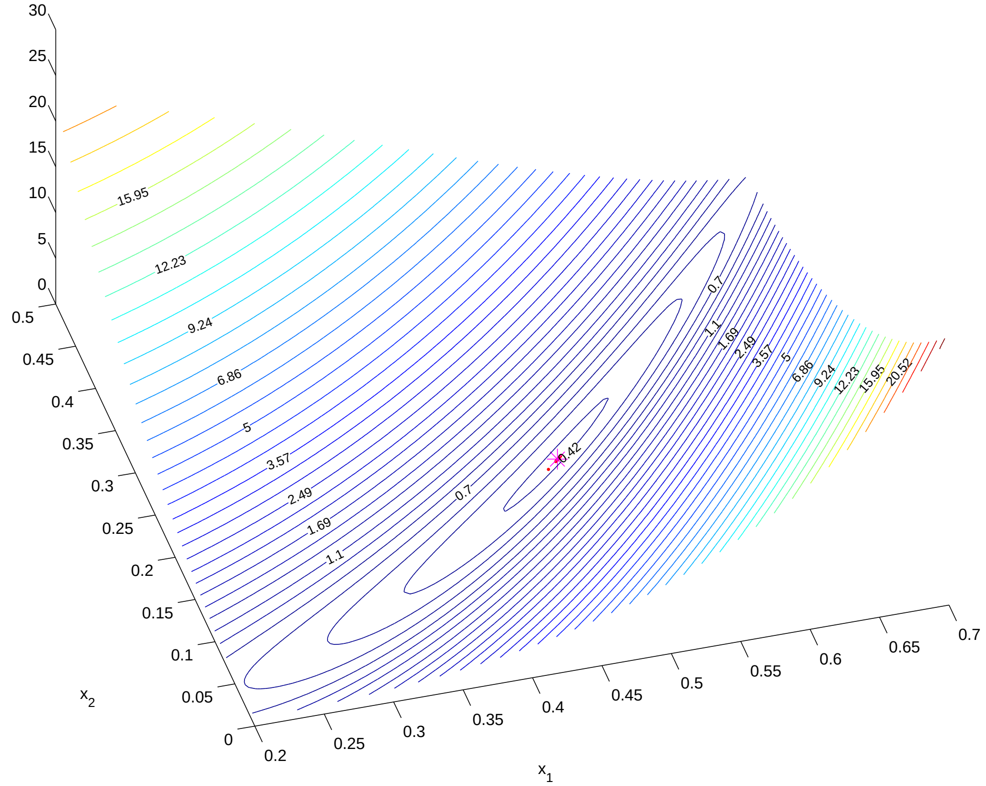


Figure 3.9: Kriging solutions of the RDO example problem, c.o.v. $x_1 = 0.05$, $\sigma_h = \sigma_q = 0.1$ (reference solution represented by the '*' symbol). The contour plot is that of the robust objective function computed through SP.

Chapter 4

Aircraft Wing Model

4.1 Planform design parameters

In this section, the variables considered to parametrize the wing model are covered. These parameters are, in general, common to both versions of the framework, described in Chapters 5 and 6.

Planform variables control the outer mold line of the wing. These are illustrated in Fig. 4.1. Optionally, the wing model may be provided with a pylon and nacelle installation to account for a wing mounted engine. The geometric parameters for these are shown in Fig 4.2.

The airfoil sections are also defined at the three main spanwise stations: root, breakstation and tip.

In terms of structural parameters, these generally comprise those defining the positioning of spars, ribs, skin panels and stringers, their thicknesses and section properties (breakstation quantities are a weighted average depending on the relative position of the breakstation in the wing). The following sections illustrate the parameters that define the geometry of each type of structural element.

4.2 Structural design parameters

4.2.1 Spars

Spar placements are determined by the chord fraction at the root and tip (Figure 4.3). For the current implementation, spars are treated as I-beams, their sections being

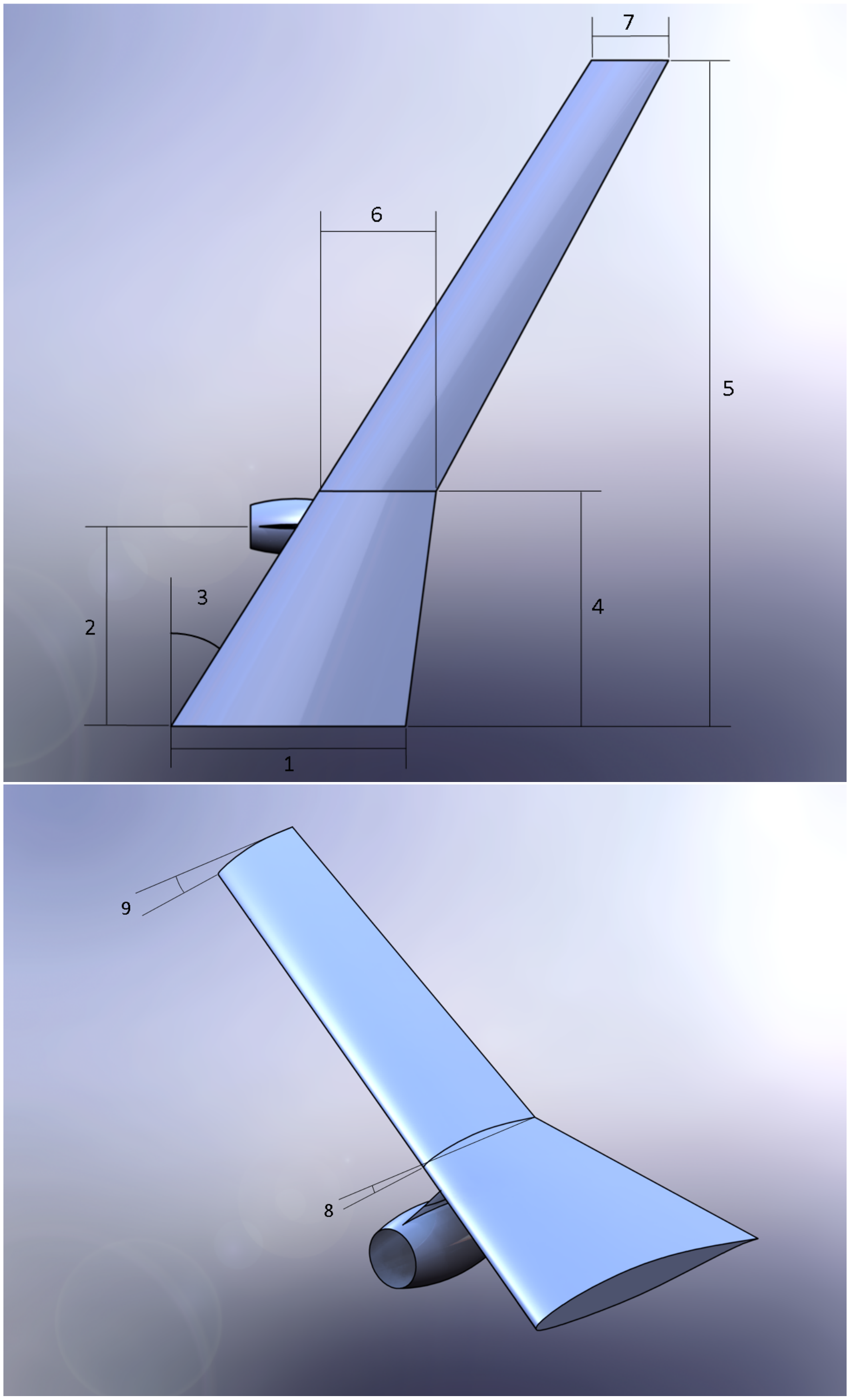


Figure 4.1: Wing planform variables: 1 - Root chord, 2 - Pylon spanwise position, 3 - Leading edge sweep angle, 4 - Break station, 5 - Semispan, 6 - Break station chord, 7 - Tip chord, 8 - Break station twist angle, 9 - Tip twist angle

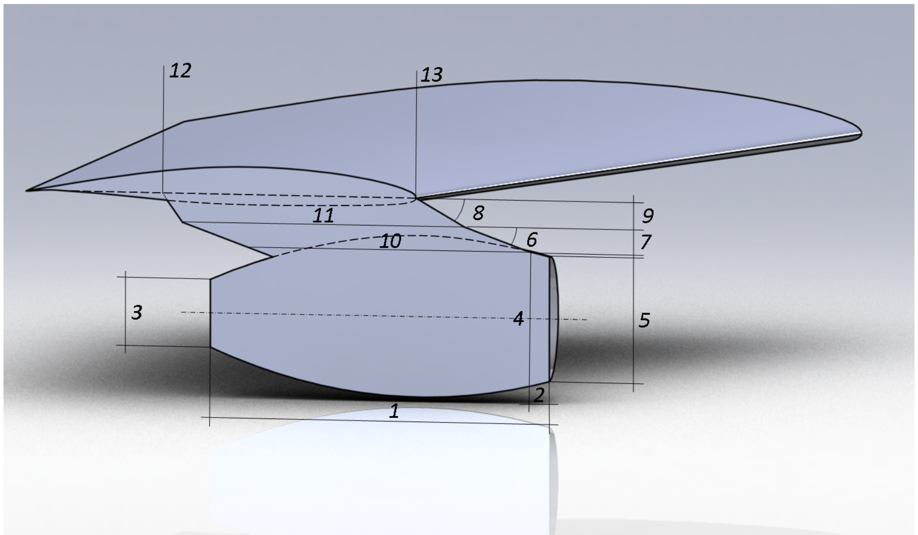


Figure 4.2: Pylon/Nacelle geometric description: 1 - Nacelle length, 2 - Distance to pylon apex, 3 - Nacelle exhaust diameter, 4 - Nacelle diameter at pylon apex, 5 - Nacelle inlet diameter, 6 - Pylon sweep angle (lower station), 7 - Pylon height (lower station), 8 - Pylon sweep angle (upper station), 9 - Pylon height (upper station), 10 - Pylon chord at apex, 11 - Pylon chord (between stations), 12 - Percent wing chord aft position, 13 - Percent wing chord forward position.

defined as seen in Figure 4.4. Different materials may be specified for the cap and web of a spar.

4.2.2 Ribs

Rib placements and dimensions are specified both at the leading and trailing edges (Figs. 4.5 and 4.6). Again, different material properties may be assigned to the cap and web.

4.2.3 Stringers

Stringers are placed in pairs (sharing the same chord fraction, one at the upper surface and another at the lower one), as shown in Figure 4.7. Each stringer's section is defined by its height and width as seen in Figure 4.8. A single material type is defined for each stringer.

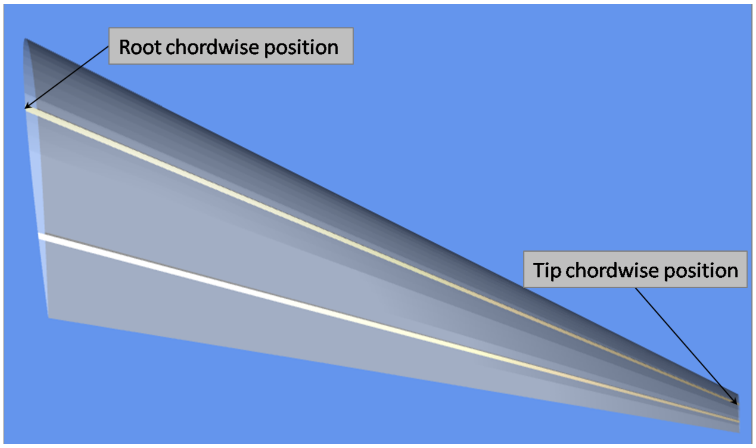


Figure 4.3: Spars positioning

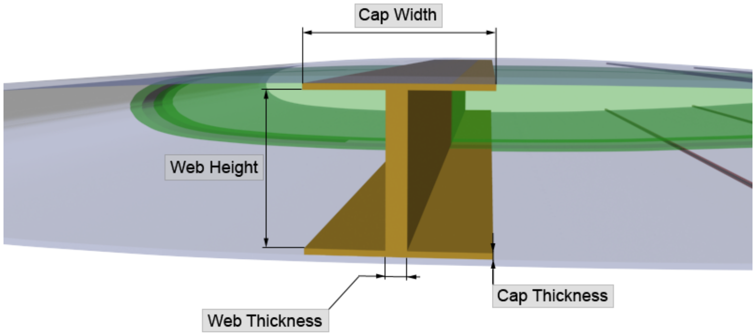


Figure 4.4: Spar section dimensions

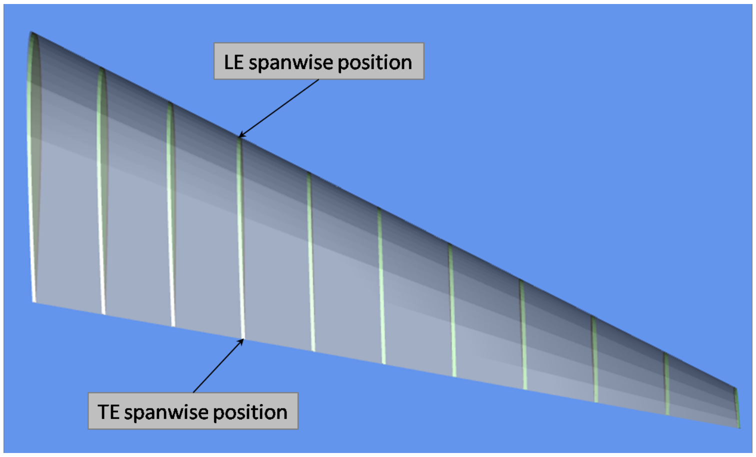


Figure 4.5: Ribs positioning

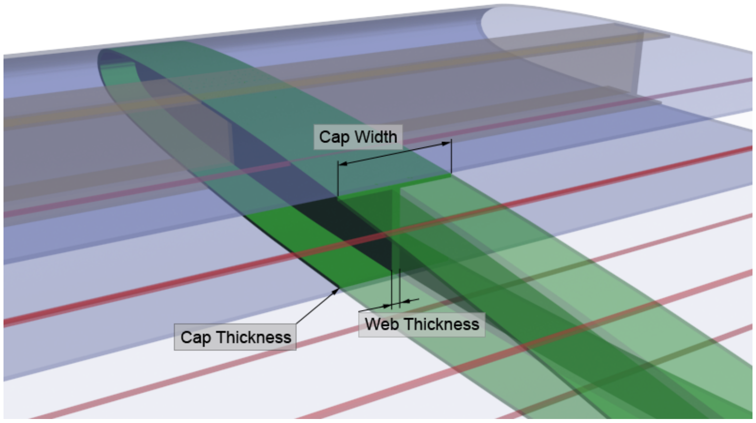


Figure 4.6: Rib section dimensions

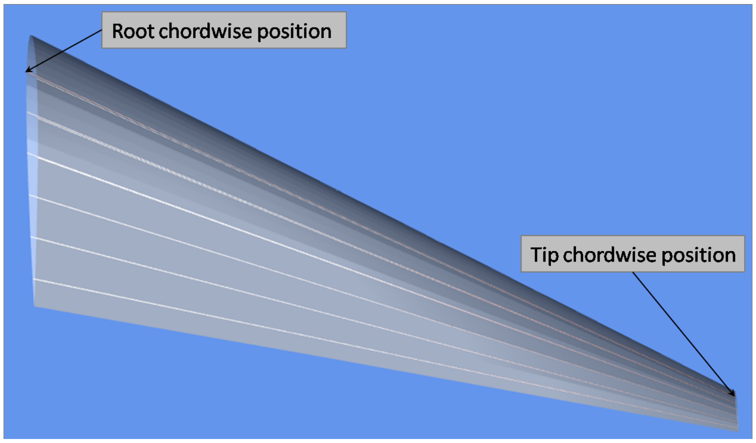


Figure 4.7: Stringers positioning

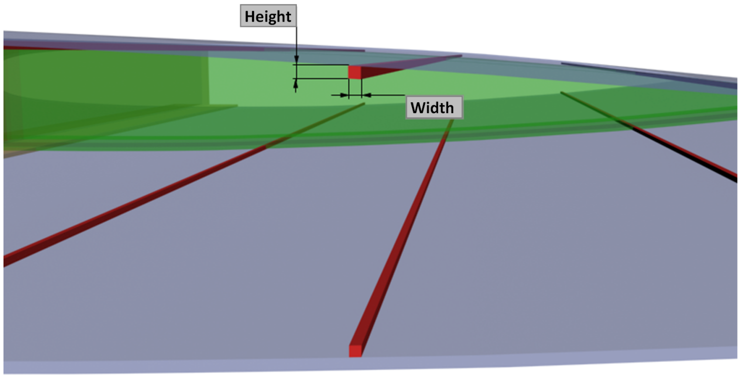


Figure 4.8: Stringer section dimensions

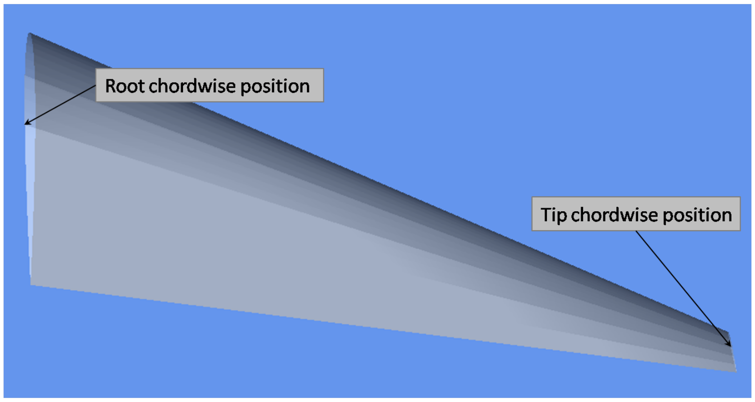


Figure 4.9: Placement of skin panels

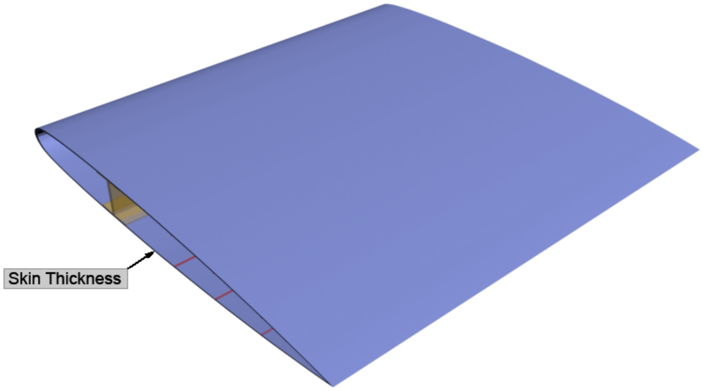


Figure 4.10: Definition of skin thickness

4.2.4 Skin panels

The skin panels' placement on the wing is defined by the front and aft chord fractions at root and tip (Fig. 4.9). Like stringers, they are defined in pairs. Fig. 4.10 shows the definition of skin thickness at wing root, while a tip/root thickness ratio may also be assigned.

4.3 Model constraints

The types of constraints that allow the creation of a feasible structural model (prior to analysis) during optimization can be either linear or nonlinear. They are automatically handled by the MDO framework (described in detail in the following chapters) and therefore require no user interaction. These constraints must be in obeyed so as to allow a successful meshing of the structural model. They mostly encompass constraints that prevent interference between the various structural components (e.g. prevent spars from overlapping by comparing their chord fractions at root and tip while taking into account the cap widths and position of other spanwise components such as stringers).

Chapter 5

MDO Framework (Low Fidelity) - Surrogate Model Evaluation

5.1 Low Fidelity Framework

A first version of the MDO framework, tailored for aircraft wing design, evolved from previous work by the author (MASc thesis [71]) and, in this chapter, it is used to evaluate the performance of the surrogate models described in Chapter 3. Components such as the aerodynamics module and structural/aeroelasticity module are updated versions. On the other hand, a new optimization code is implemented - CFSQP - which replaces a previous MATLAB$^{®}$ based optimizer. Two main disciplines are considered in this version of the framework: aerodynamics and structural mechanics. The interaction between the two is only one-way, however, in which aerodynamic loads are applied to the structural model.

5.1.1 Aerodynamics Module

The Vortex Lattice Method (VLM) code - *Zephyr* - was developed specifically for the first version of the tool. It includes compressibility corrections and a skin friction estimate which uses a combination of the Eckert Reference Temperature method for laminar flow and the van Driest II formula for turbulent flow [72] (more detail on these in Appendix A). This module was validated through comparison against other VLM codes and a RANS code - OVERFLOW [71, 73]. Support has also been added for the definition of control surfaces (flaps, differentially deflecting ailerons).

5.1.2 Structural Module

The structural code employed in this first version of the framework is also an updated version of the Equivalent Plate Model (EPM) used in previous work [71]. The main differences to previous versions are the capability to compute the stress field and the embedded optimizer. The main limitations are still the planform shape which must be strictly trapezoidal and that only one airfoil section may be defined.

5.1.3 Optimization Module

The optimizer, which controls the activity of the other modules, makes use of a gradient based algorithm - Feasible Sequential Quadratic Programming - to search for the minimum of a user defined objective function (and can be coupled with both direct analysis and the surrogate models that were the topic of chapter 3) [74]. The CFSQP (C version) algorithm implements both Armijo and nonmonotone line search types, for which a feasible point with respect to all but nonlinear equality constraints is achieved at each iteration.

Among geometric, aerodynamic, and structural parameters, there is a choice of over 300 different variables to be considered for optimization.

5.2 Surrogate Model Evaluation

The comparison between the surrogate model types introduced earlier is made by solving 4 different case studies, employing the tool described in the previous section (VLM/EPM version). The first three case studies aim at evaluating the performance impact of the number of design variables/complexity of the objective function (note the problems presented here are not truly scalable). The final case study, in which aerostructural optimization is performed, represents a more typical aircraft design problem.

The baseline planform used is trapezoidal and a schematic is presented in Fig. 5.1. The wing section airfoil is a symmetric NACA0009 at the three customizable sections (root, breakstation, and tip). Flight conditions are also the same for all optimization problems: M0.7@11000m (ISA), $\alpha = 2$ deg.

The initial wing structure possesses two spars, 11 ribs and 8 pairs of stringers (same position in lower and upper surfaces) built in aluminum, and a carbon fiber skin. The

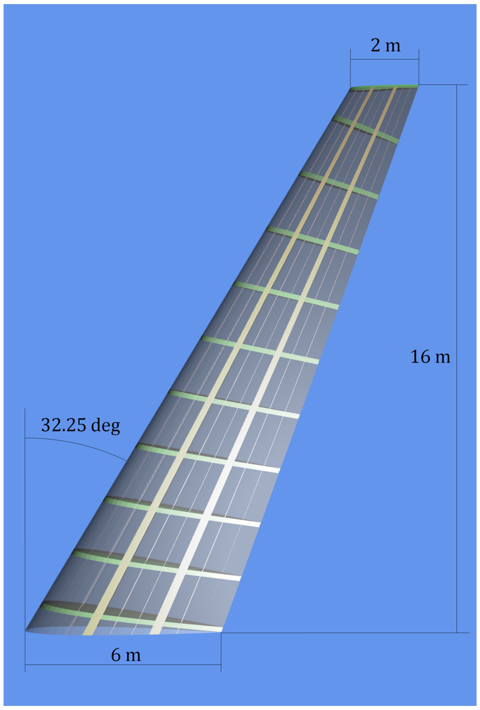


Figure 5.1: Baseline wing planform and internal structure

data does not attempt to portray any specific real life wing but is nonetheless a feasible design in terms of structural constraints.

Because of the stochastic nature of the methods being employed (due to the sampling scheme), several trial runs must be executed with each of them. More specifically, for each method, at least 30 independent runs (same starting point) were performed so the sample could be considered statistically significant. In most cases, a normal distribution is then assumed to fit the results and helps in determining whether in a probabilistic sense one method has an advantage over the other, and how well it fares against a conventional non-approximative approach. The comparison is performed using three main criteria: the number of function evaluations (actual analysis and not metamodel evaluations) until convergence, the final objective error (in terms of the percentage of the difference between initial and optimum objective function values), and the configuration error (percent error in the optimum design variables, measured as the norm of the difference in optimum $\mathbf{x}$ vectors). The last two are measured with respect to the conventional optimization results. The number of function evaluations, while not a continuous random variable, can be considered as such for the purposes of building a normal distribution plot, since the size of the discretization grid is generally small compared to its spread.

In terms of sampling, for both Kriging and ANN based methods the initial sample was limited to $(n_{DV}+1)(n_{DV}+2)/2$, whereas the update sample simply added n_{DV} successful samples (in the sense of what is described in section 3.4). The trust region for these models is initially set to the whole design space for the initial build and all subsequent sampling is performed in a hypercube centered on the current iterate and delimited at +/- 15% of the design variable range.

Because the implementations of both Kriging and ANN are highly customizable, a default parameter set that would offer a good compromise between speed and robustness had to be determined. With the Kriging metamodel, a Cubic correlation function was employed for all cases, except in section 5.2.3, where a Gaussian correlation is also used for comparison purposes. A quadratic regression model is in use at all times. Neural Networks were built with a single layer containing $4n_{DV}$ neurons, and the learning rate was set at 0.01.

5.2.1 Case Study I - Two design variables

The simplest problem being solved in this series deals with the aerodynamic shape optimization of a wing with respect to its semispan ($b/2$) and leading edge sweep angle (Λ_{LE}). The objective, to maximize the lift to drag ratio, is shared with case studies II and III, as well. The bounds for the DVs are presented in Table 5.1.

Table 5.1: Design variables and constraints for Case Study I

Design Variable	Min	Max
semispan, $b/2$	15 m	17 m
LE sweep, Λ_{LE}	22.25 deg	42.25 deg

The optimum point is, in this case, at $b/2 = 17$ m and $\Lambda_{LE} = 22.25$ deg (resulting in an L/D ratio of 28.64), and is captured precisely by all of the optimization strategies employed.

In terms of the number of function evaluations required to reach such an optimum, the quadratic response surface is the least costly - consistently converging with only 6 function evaluations, albeit with a certain amount of error in following the true value of the objective function. This fact is evidenced in Fig. 5.2, where a normal distribution was assumed for the objective error.

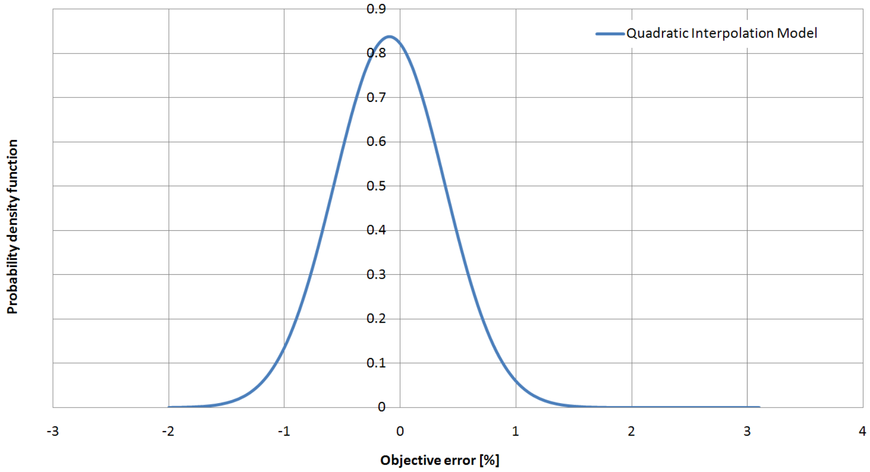


Figure 5.2: Distribution of the the error in optimum value of the objective function (Case Study I).

As for the other methods, both Kriging and ANN follow the reference set by the conventional optimization with absolutely no error in either objective or configuration. In terms of the number of function evaluations they average quite close to the reference (9 evaluations), with expected values of 9.4 and 9.2, respectively (the number of function evaluations alternates between 8 and 10 for these methods).

5.2.2 Case Study II - Five design variables

In this second optimization problem, the design variables from before are joined by the chords at the root and tip of the wing, as well as the chord at the breakstation (the breakstation spanwise location, y_{bs}, is in this case fixed at 8 m from the root, and hence the baseline value for the breakstation chord, c_{bs}, is 4 m). Table 5.2 presents the design variables limits and constraints.

The optimum design configuration was found to be $b/2 = 17$ m, $\Lambda_{LE} = 22.25$ deg, $c_{root} = 5.080$ m, $c_{bs} = 2.962$ m, $c_t = 1$ m. The lift to drag ratio is hence maximized to 31.01.

With this number of design variables, the quadratic interpolation response surface proved unfit for the task and consistently gave poor results (converged to completely

---

[1] Represents the total wing surface area of the aircraft.

Table 5.2: Design variables and constraints for Case Study II

Design Variable	Min	Max
semispan, $b/2$	15 m	17 m
LE sweep, Λ_{LE}	22.25 deg	42.25 deg
root chord, c_{root}	3 m	9 m
breakstation chord, c_{bs}	1 m	7 m
tip chord, c_{tip}	1 m	3 m
$S_{ref}{}^{1} \geq 100$ m^2		

different configurations than the optimum, and never the same), for which reason it was henceforth excluded from the comparison. Regarding the remaining methods, the results in Figs. 5.3, 5.4 and 5.5 indicate a clear performance advantage over the conventional approach can be attained with minimal cost in objective and configuration error. The benefits lead, on average, to about 50% reduction in the number of function calls both with Kriging and ANN, although the distribution for the latter reveals a higher deviation from the mean.

In these plots, since both the number of function evaluations and the configuration error are, by definition, positive quantities, a truncated normal distribution was employed (lower bound (a) of zero, for the configuration error, or the observed minimum in the case of the number of function evaluations, and upper bound (b) of $+\infty$):

$$f(x,a,b) = \frac{\frac{1}{\sigma}\,\phi\left(\frac{x-\mu}{\sigma}\right)}{\Phi\left(\frac{b-\mu}{\sigma}\right) - \Phi\left(\frac{a-\mu}{\sigma}\right)} = \frac{\frac{1}{\sigma}\,\phi\left(\frac{x-\mu}{\sigma}\right)}{1 - \Phi\left(-\frac{\mu}{\sigma}\right)} \tag{5.1}$$

where μ is the mean and σ the standard deviation of the fitted distribution, and, ϕ and Φ represent the probability density function and cumulative distribution function of the standard normal distribution, respectively.

5.2.3 Case Study III - Seven design variables

The final aerodynamic shape optimization problem has a slightly different set of variables, as illustrated by Table 5.3.

The reference optimum configuration for this problem is: $y_{bs} = 5$ m, $\Lambda_{LE} = 22.25$ deg, $\theta_{bs} = -2$ deg, $\theta_{tip} = -2$ deg, $c_{root} = 5.675$ m, $c_{bs} = 3.806$ m, $c_t = 1$ m (corresponding to an L/D value of 36.71). As mentioned before, two different correlation models are used with the Kriging model in order to illustrate the effect

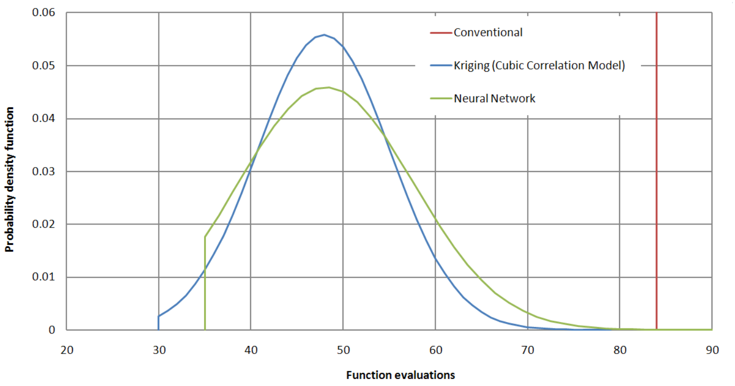


Figure 5.3: Distribution of the number of function evaluations (Case Study II).

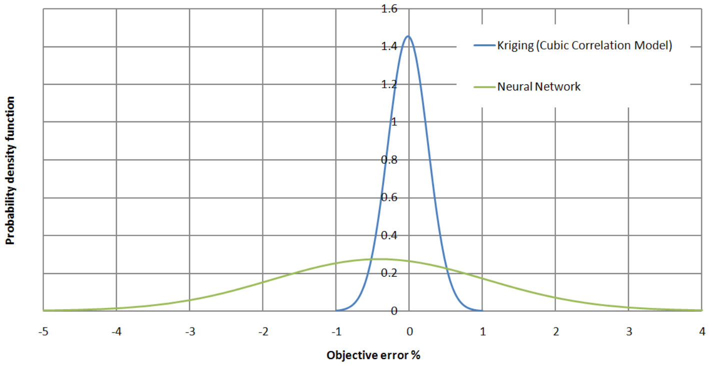


Figure 5.4: Distribution of the the error in optimum value of the objective function (Case Study II).

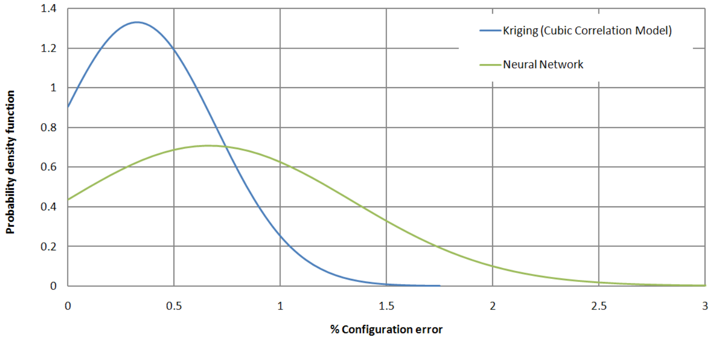


Figure 5.5: Distribution of the the error in optimum configuration (Case Study II).

Table 5.3: Design variables and constraints for Case Study III

Design Variable	Min	Max
breakstation, y_{bs}	5 m	11 m
LE sweep, Λ_{LE}	22.25 deg	42.25 deg
breakstation twist, θ_{bs}	-2 deg	2 deg
tip twist, θ_{tip}	-2 deg	2 deg
root chord, c_{root}	3 m	9 m
breakstation chord, c_{bs}	1 m	7 m
tip chord, c_{tip}	1 m	3 m
$S_{ref} \geq 100$ m^2		

their choice might have on the results.

A striking difference from the previous case study is the fact that the surrogate model based approaches are no longer irrevocably better than conventional optimization (112 function evaluations). For Kriging, the probability of improvement in terms of the number of function calls is now about 75% (obtained from integration of the probability density function up to the reference value) with the Cubic correlation model, dropping to around 54% with the Gaussian model (Fig. 5.6). The ANN approach performs worse than the reference most of the time since the probability of improvement is now below 50% and shows a tendency to miss the objective by a reasonable amount (as can be seen on Fig. 5.7).

It is therefore clear that both surrogate model types, but especially the ANN approach, would benefit from additional tuning of their respective parameters when solving this particular problem.

5.2.4 Case Study IV - aerostructural optimization, 6 design variables

With aerostructural optimization enabled, 4 geometry design variables are joined by 2 structural variables (skin thickness and skin thickness ratio between root and tip). Constraints are imposed on the maximum admissible stress[1] in each structural component (Kreisselmeier-Steinhauser constraint aggregation is used here to save on the number of nonlinear constraints being imposed. The constraint aggregation parameter, ρ, is determined as proposed in [75]), and a trapezoidal planform is enforced.

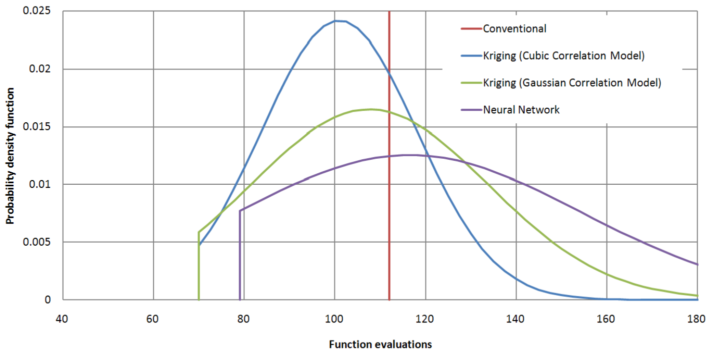


Figure 5.6: Distribution of the number of function evaluations (Case Study III).

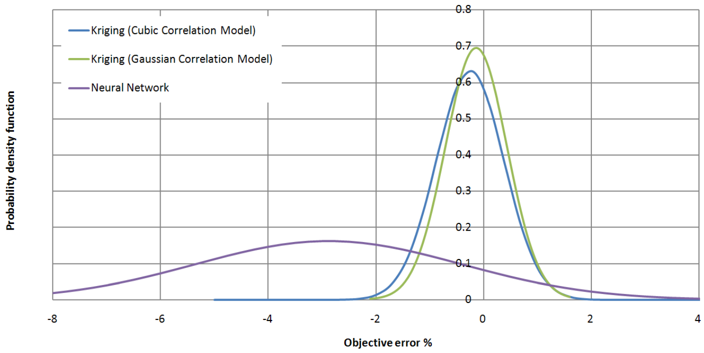


Figure 5.7: Distribution of the error in optimum value of the objective function (Case Study III).

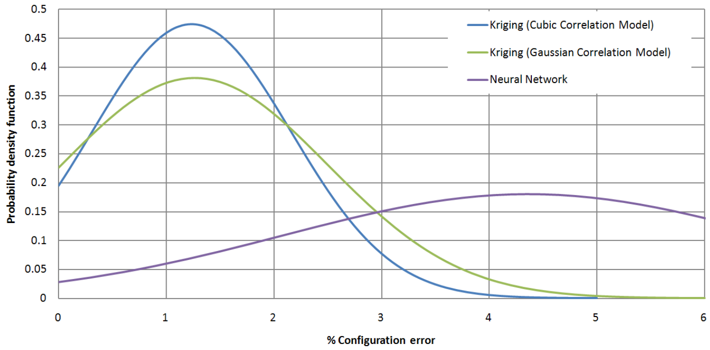


Figure 5.8: Distribution of the error in optimum configuration (Case Study III).

Table 5.4 summarizes the bounds on design variables and constraints.

Table 5.4: Design variables and constraints for Case Study IV

Design Variable	Min	Max
semispan, $b/2$	15 m	17 m
LE sweep, Λ_{LE}	22.25 deg	42.25 deg
root chord, c_{root}	3 m	9 m
tip chord, c_{tip}	1 m	3 m
skin thickness	0.003 m	0.009 m
skin thickness ratio	0.35	1.35
$S_{ref} \geq 100\ \mathrm{m}^2$		
$\frac{1}{\rho}\log\left(\sum_{i=1}^{N} e^{\rho g_i}\right) \leq 0,\ g_i = f\left(\sigma_i, \sigma_{\text{yield}}\right)$		

Additionally, an equality constraint, simulating a cruise flight condition is applied:

$$L - (2 \cdot m_{wing}\, g + W_{other}) = 0 \tag{5.2}$$

Where L is the total wing lift, $m_{wing}\, g$ represents the weight of one wing (output variable from the structural module) and W_{other} is the weight of all other aircraft components (fuselage, tail, engines) which is kept constant throughout the optimization process. Note the either positive or adverse lift of the tail is not being taken into account here, as a first approximation.

Changing from the previous case studies, the objective function is now:

$$f\left(\mathbf{x}\right) = 1000\, C_D + 0.001\, m_{wing} \tag{5.3}$$

which is weighting both wing drag and structural mass and corrected for the expected magnitudes of each. In this case, the surrogate model based optimization approaches are to be compared with a conventional Multidisciplinary Feasible (MDF) optimization architecture.

For this problem, multiple minima were detected while employing the Kriging approximation. These optima were readily validated using the conventional MDF approach (using as a starting point the average optimum configuration obtained with

---

[1] For isotropic materials, the Von Mises criterion is used. For composite structures, the Tsai-Hill criterion is employed instead.

the Kriging approximation) and are presented in Table 5.5 as well as in Fig. 5.9 (compared against the baseline). Beyond the four optima presented here, two others were found, but were not repeated often enough to become statistically significant. The optimum found with the conventional approach is also shown, for comparison purposes (starting point is the usual baseline wing). This effectively proves how this surrogate model approximation facilitates a more global search of the design space (albeit independent runs are required). Interestingly, the optimum found through regular MDF was not detected through the Kriging approximation which instead walks over the latter, yielding yet lower values of the objective function at the minima detected (the number of function evaluations for the conventional approach, presented in Fig. 5.10, refers only to the optimization run having the baseline wing as a starting point).

In line with what had been observed in the previous study, the ANN implementation would require more tuning of its parameters since it failed to regularly converge to any of the detected optima and instead frequently converged to suboptimal configurations (but never the same twice). Therefore, it has not been included in the results as that would require changing critical model parameters, preventing any direct comparison with the previous case studies.

Table 5.5: Local minima detected for Case Study IV

Design Variable	Optimum (MDF)	Optimum 1	Optimum 2	Optimum 3	Optimum 4
semispan, $b/2$	15.028 m	17 m	15.097 m	15.019 m	15.542 m
LE sweep, Λ_{LE}	35.31 deg	28.35 deg	40.86 deg	37.02 deg	35.63 deg
root chord, c_{root}	6.161 m	3.708 m	7.977 m	7.402 m	5.370 m
tip chord, c_{tip}	2.238 m	3.003 m	1.000 m	1.000 m	2.995 m
skin thickness (root)	0.003 m	0.003 m	0.003 m	0.003 m	0.003 m
skin thickness ratio	0.35	0.35	0.35	0.35	0.35
$f(\mathbf{x}_{optimum})$	8.495	8.467	8.483	8.460	8.484
C_D	0.007133	0.007200	0.007005	0.007063	0.007105
m_{wing}	1362 kg	1267 kg	1477 kg	1397 kg	1379 kg

It is duly noted that the design variables which are exclusive to the structural module (skin thickness and skin thickness ratio between root and tip) are brought to their lower limits by the optimizer in all of the optimum configurations detected. This indicates that there would be room from improvement should these limits be further relaxed.

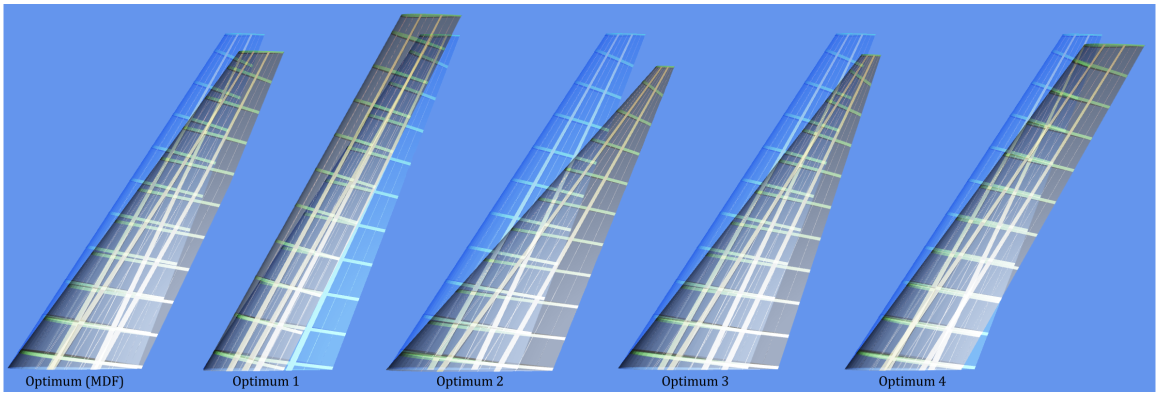


Figure 5.9: Optimum configurations (Case Study IV).

The results show how much the added complexity of objective and constraints functions in this case affect the conventional approach in terms of number of function evaluations until convergence (depending on the optimum point found). To the Kriging model, however, this added complexity is of little consequence, indicating that the number of design variables plays a more pivotal role.

Regarding the errors in the optimal value of the objective function and optimum configuration, their distribution was assessed for each of the detected optima (and compared with the reference MDF values). The results presented in Figs. 5.11 and 5.12 indicate that some optima are usually found with a smaller error relative to the reference than others. For instance, optimum 1 bounces against hard constraints on semispan and leading edge sweep angle, which helps in reducing the error in these design variables.

Additionally, these optimum points were not detected with the same frequency during the computational experiments. Their frequentist probabilities are: for optimum 1 - 26%, optimum 2 - 50%, optimum 3 - 6%, optimum 4 - 18%.

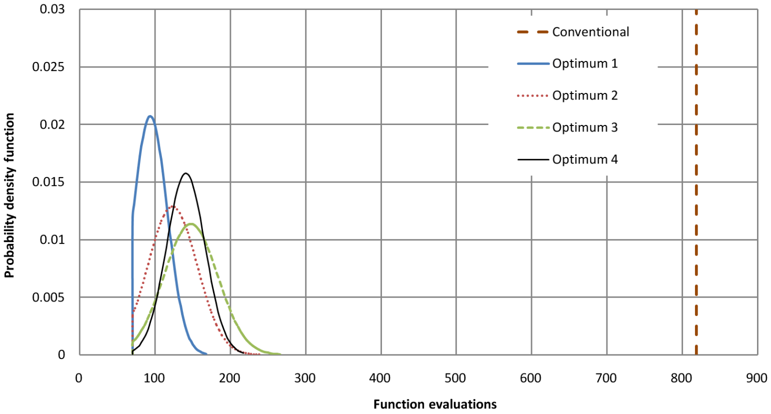


Figure 5.10: Distribution of the number of function evaluations (Case Study IV - starting point is baseline wing).

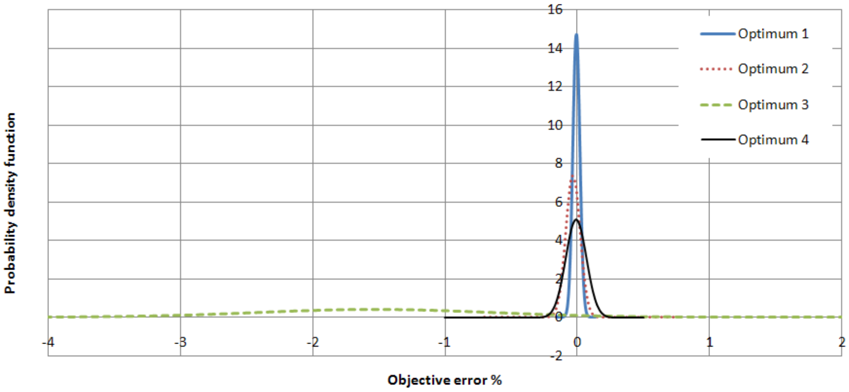


Figure 5.11: Distribution of the error in optimum value of the objective function (Case Study IV).

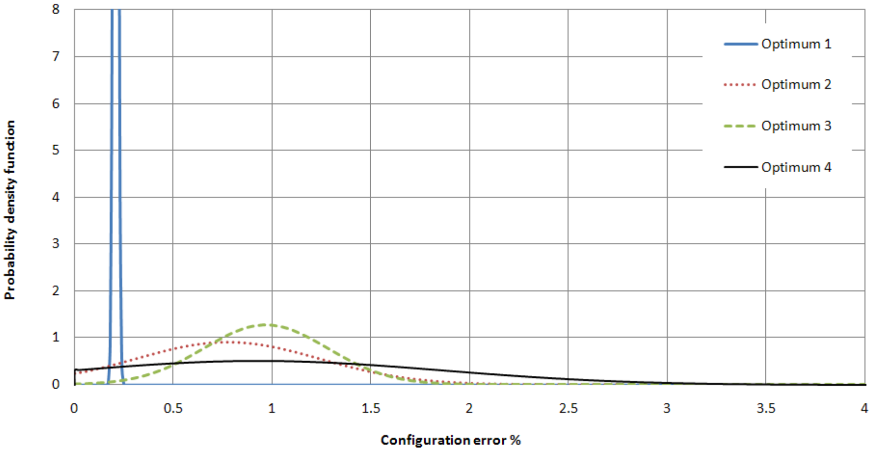


Figure 5.12: Distribution of the error in optimal configurations (Case Study IV).

Chapter 6

MDO Framework (Medium Fidelity)

6.1 Analysis Modules

The current version of the MDO framework is characterized by more capable and higher fidelity analysis modules. The new aerodynamics module is a 3D-panel code, while the structural module has been upgraded to a finite-element formulation. These are able to interact in a fluid structure interaction process that is the subject of section 6.3.2 in this chapter.

6.2 Doublet-Source Lattice Aerodynamics Module

A more capable aerodynamics module was required in order to allow a fully stressed model of the wing structure. Additionally, the new implementation should also support the analysis of wing mounted pylons and (powered) nacelles, further increasing the scope of configurations that can be analyzed. Therefore, a three dimensional panel code was developed based on the assumptions of inviscid, incompressible and irrotational flow (the same assumptions were made for the Vortex Lattice code).

The choice of a panel method over a full Navier-Stokes solver is not only due to the higher computational cost of the latter. As it will be shown in section 6.2.3, this decision proved to be adequate for the level of fidelity required at early design stages.

6.2.1 Theory and Implementation

Within the class of panel methods, the Doublet-Source Lattice method has seen wide usage in aerodynamics applications [76, 77, 78] because of its simplicity and its accuracy where viscous effects are not predominant (in general, this limits their usefulness to low angle of attack/sideslip angle simulations). It consists of a distribution of doublets and sources over the surface of the body whose pressure distribution, when immersed in a moving flow, is of interest. Additionally, for lifting bodies (as is the case with the wing), a wake, essentially a sheet of doublet panels, is built (later, it will be shown how a wake can also be used to compartmentalize the exhaust from nacelles). Panel methods then solve for the Laplace equation with respect to the velocity potential field:

$$\nabla^2 \Phi = 0 \tag{6.1}$$

where $\nabla \Phi = \mathbf{V}$ is the velocity vector field. The field is assumed to exist and satisfy the Laplace equation in two separate regions (inside and outside the body and infinitely thin wake) as seen in Fig. 6.1.

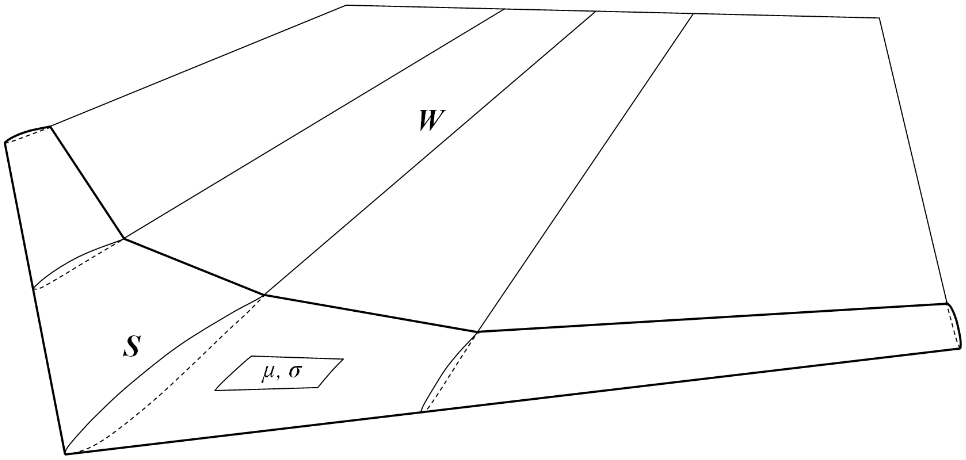


Figure 6.1: Domain regions for the panel method

If the wake is impermeable ($(\nabla \Phi_U - \nabla \Phi_L) \cdot \mathbf{n} = 0$, U and L referring to the regions immediately above and below the wake, respectively) the expression for the potential at an arbitrary point in the domain is given by [78]:

$$\begin{aligned}\Phi_P &= \frac{1}{4\pi}\iint\limits_S (\Phi - \Phi_i)\,\nabla\left(\frac{1}{r}\right)\cdot\mathbf{n}\,dS - \frac{1}{4\pi}\iint\limits_S \left(\frac{1}{r}\right)(\nabla\Phi - \nabla\Phi_i)\cdot\mathbf{n}\,dS \\ &+\frac{1}{4\pi}\iint\limits_W (\Phi_U - \Phi_L)\,\nabla\left(\frac{1}{r}\right)\cdot\mathbf{n}\,dW + \phi_\infty \end{aligned} \tag{6.2}$$

where Φ_i is the potential inside the body and ϕ_∞ the potential of the free stream flow.

Since the interest lies in points located on the surface, S, the potential for the particular situation where P resides on the surface is:

$$\begin{aligned}\Phi_P &= \frac{1}{4\pi}\iint\limits_{S-P} (\Phi - \Phi_i)\,\nabla\left(\frac{1}{r}\right)\cdot\mathbf{n}\,dS - \frac{1}{2}(\Phi - \Phi_i)_P \\ &-\frac{1}{4\pi}\iint\limits_S \left(\frac{1}{r}\right)(\nabla\Phi - \nabla\Phi_i)\cdot\mathbf{n}\,dS \\ &+\frac{1}{4\pi}\iint\limits_W (\Phi_U - \Phi_L)\,\nabla\left(\frac{1}{r}\right)\cdot\mathbf{n}\,dW + \phi_\infty \end{aligned} \tag{6.3}$$

in which a jump in the value of the potential occurs at P.

Because doublets and sources are already particular solutions to Eq. 6.1, the problem resides only in determining their strengths while obeying a boundary condition (BC) of no flow through the surface. The choice of internal potential, Φ_i, effectively determines the type of boundary condition. It may be applied directly to the velocity field at the surface (i.e. $\mathbf{V}\cdot\mathbf{n} = \mathbf{0}$), which constitutes a Neumann type of boundary condition, and is equivalent to setting the internal potential, Φ_i, to zero. Alternatively, a Dirichlet BC, prescribing the flow potential inside the surface, may be employed. This is the approach taken in the code as it leads to better numerical accuracy. The internal potential is chosen as $\Phi_i = \phi_\infty$ so that $\Phi - \phi_\infty = \phi$ is a perturbation potential, and $\Phi_P = \phi_\infty$. Applying this to Eq. 6.3:

$$\begin{aligned}&\frac{1}{4\pi}\iint\limits_{S-P} \phi\nabla\left(\frac{1}{r}\right)\cdot\mathbf{n}\,dS - \frac{1}{2}\phi_P - \frac{1}{4\pi}\iint\limits_S \left(\frac{1}{r}\right)(\nabla\phi)\cdot\mathbf{n}\,dS \\ &+\frac{1}{4\pi}\iint\limits_W (\Phi_U - \Phi_L)\,\nabla\left(\frac{1}{r}\right)\cdot\mathbf{n}\,dW = 0 \end{aligned} \tag{6.4}$$

The discrete equivalent of Eq. 6.4 for a number of panels N on S and N_W panels for the wake is (for the j$^{\text{th}}$ panel):

$$\sum_{k=1,\,k\neq j}^{N} A_{jk}\mu_k - 2\pi\mu_j + \sum_{k=1}^{N_W} A_{jk}\mu_{W_k} + \sum_{k=1}^{N} B_{jk}\sigma_k = 0 \tag{6.5}$$

where μ and σ are the doublet and source strengths, respectively, and A_{jk}/B_{jk} is the influence coefficient of the k$^{\text{th}}$ panel on the j$^{\text{th}}$ panel. μ_{W_k} is the doublet strength of the wake being shed at the trailing edge and is defined as the difference in doublet strengths of the two intersecting panels there. The wake itself is geometrically fixed as following the direction of free stream flow, upon departing from the trailing edge. The aerodynamic influence coefficients are obtained by evaluating the surface integrals in Eq. 6.4 for each panel and the respective formulas, dependent only on panel geometry and relative position, can be found in [79, 80]. From the boundary condition applied it also follows that:

$$\sigma = -\mathbf{n} \cdot \nabla\phi_\infty = -\mathbf{n} \cdot \mathbf{V}_\infty \tag{6.6}$$

i.e., for solid bodies, the source strength is set to be such that it matches the free stream component normal to the surface. The only unknowns to solve for are then the doublet strengths on the wing surface.

In the case where wing mounted engines are present, Eq. 6.6 takes a more general form at the inlet/outlet of the nacelle:

$$\sigma = V_N - \mathbf{n} \cdot \mathbf{V}_\infty \tag{6.7}$$

V_N is the inflow/outflow velocity at the boundary (positive for outflow and negative for inflow). These normal velocities are prescribed at simulation time (naturally, $V_{outlet} > V_{inlet} > V_\infty$). A crude approximation of engine thrust given by $T = \dot{m}(V_{outlet} - V_{inlet})$, where $\dot{m}$ is the air mass flow rate through the engine, helps in setting credible values for the velocities, so that $T > D$ (and that by a significant percentage, since the calculation of drag in the aerodynamics module does not include the fuselage or tail).

Additional considerations include the treatment of the exhaust plume, which is modeled as suggested in [76]. The exhaust jet is surrounded by its own wake as seen in Fig. 6.2 to simulate the defined shear zone between exhaust plume and freestream. This wake is initially normal to the nacelle exit, but mixing of what is in reality a hot, high energy massflow with the surroundings, cannot be directly modeled in an inviscid, irrotational formulation, and as such, a preset distance (L_{Jet}) defines where the exhaust will assume the same angle of attack/sideslip as the free stream. By default, it is the distance between the nacelle exit and the local wing trailing edge.

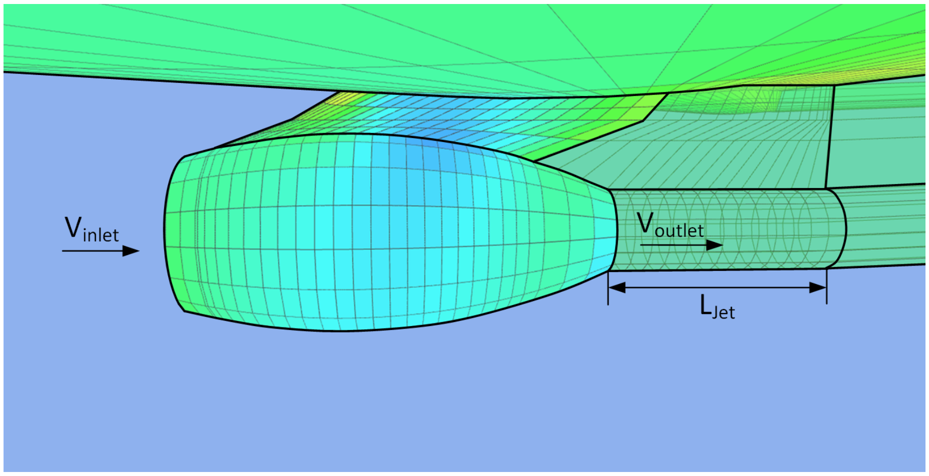


Figure 6.2: Pylon/Nacelle assembly and engine exhaust modelling

Upon the solution of the linear system represented in Eq. 6.5, the doublet distribution needs to be differentiated in order to obtain the tangential velocity at the control points located in the center of each panel. A second order central differentiation scheme is used where possible, reverting to left or right schemes if close to a geometric edge. The pressure on each panel may then be directly determined from the formula for the incompressible coefficient of pressure:

$$C_{p_0} = 1 - \frac{\mathbf{V} \cdot \mathbf{V}}{V_\infty^2} \tag{6.8}$$

The individual panel pressures are then integrated over the whole surface (excluding nacelle inlet/exit faces, if present) in order to obtain total body pressure forces. Compressibility corrections and friction and form drag calculations complement the inviscid force calculation and are the subject of the next section.

6.2.2 Compressibility correction and friction and form drag

Due to the assumption of incompressible flow in the solution of the flow equations, a compressibility correction should be applied according to the freestream Mach number (M_∞) [81], otherwise the accuracy of the solution will suffer. Usually, compressibility effects come into play beyond $M_\infty = 0.3$. The implementation of the Prandtl-Glauert rule, which consists of applying a transformation to the geometry along the x-axis, is similar to the one made in the VLM code. More specifically, the equivalent wing in incompressible flow is a lengthened version of the original one, stretched by a factor $\beta = \sqrt{1 - M_\infty^2}$ [81]. In practice, this materializes in the following transformation, applied to the incompressible C$_p$ distribution:

$$C_p = \frac{C_{p_0}}{\sqrt{1 - M_\infty^2}} \tag{6.9}$$

The friction and form drag model is also the same as in the VLM code:

$$C_{D_f} = C_f FF \frac{S_{wet}}{S_{ref}} \tag{6.10}$$

where C_f is the coefficient of friction, accounting for a laminar/turbulent boundary layer flow developing across the wing[1], and FF is the form factor (here, the implementation differs from that of the VLM code in the way that the form factor also needs to be computed for pylons and nacelles). The computation of the friction coefficient also takes compressibility into account. One of the main assumptions is that this type of drag does not vary significantly for small angles of attack. The calculations involved are detailed in Appendix A.

6.2.3 Validation

A sample verification of the accuracy of the aerodynamics module is accomplished through comparison with VSAERO® and commercial package ANSYS CFX®.

Comparison with VSAERO

A test case is available for comparison in [78]. The test wing has a simple trapezoidal planform with $b/2 = 6$ m (semispan), $c_{root} = 3$ (root chord), $c_{tip} = 1$ (tip chord), a mid-chord sweep angle of 30 deg, and a NACA0002 airfoil section. The angle of attack is 5 deg. A comparison of the results is presented in Fig. 6.3 (test section is located at 54.9% of the wing semispan).

Being a panel method of a similar kind, it is not surprising that the results of VSAERO® are closely matched by the present code.

---

[1] The value of the expected transition Reynolds number - Re_{trans} - should be provided. Typical values range between 5×10^5 and 1×10^6.

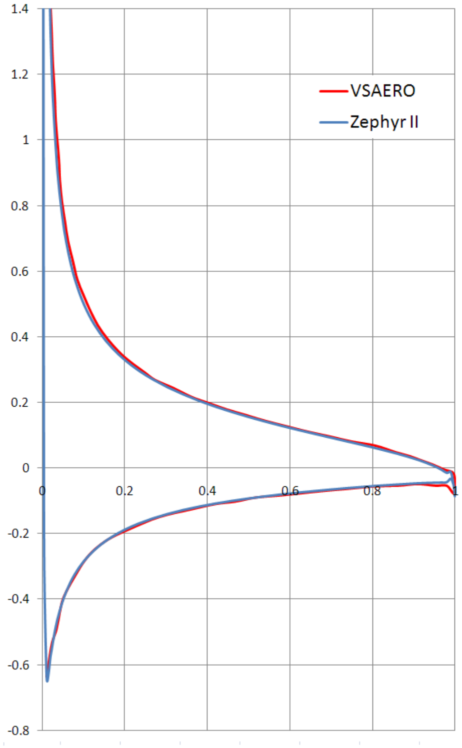


Figure 6.3: C_p plot at station $2y/b = 0.549$, $b/2 = 3$ m, $c_{root} = 3$, $c_{tip} = 1$, $\Lambda_{0.5c} = 30$ deg, NACA0002 airfoil, $\alpha = 5$ deg.

Comparison with CFX

A second test case was deemed necessary in order to assess the behavior of the analysis module when analyzing more complex configurations. Also, it is an opportunity to verify the implementation of the pylon/nacelle model presented earlier. As such, the geometry portrayed in Fig. 6.4 was devised. The geometry is then analyzed in two distinct flight conditions. The first, a purely incompressible case, where $V_\infty = 60$m/s, $\alpha = 5$ deg, at standard sea level; inlet/outlet velocities for the engines are set at 80 m/s and 200 m/s, respectively. The second run, employing the Prandtl-Glauert compressibility correction embedded in the code, is executed at $M_\infty = 0.7$ at an altitude of 11000 m (ISA). The inlet/outlet velocities are now 220 m/s and 400 m/s - this value for the outlet velocity means there is supersonic flow at the exit, if the temperature is kept at ambient (at this altitude, T = 216.77 K). To prevent supersonic flow and consequent shock waves due to the interference with the freestream flow, a more realistic (and higher) exhaust gas temperature - EGT - should be enforced.

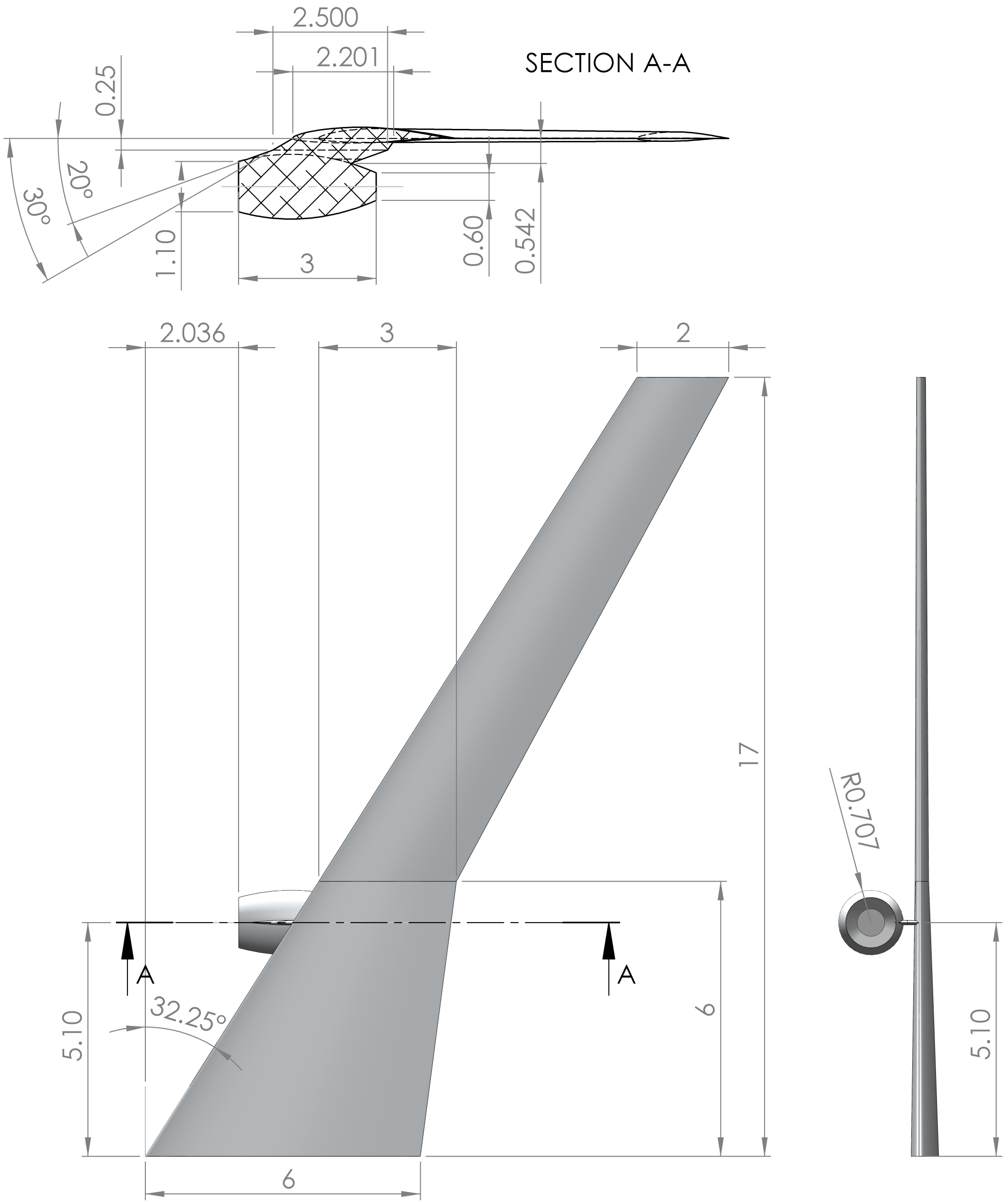


Figure 6.4: Wing with mounted engine for CFX analysis (wing airfoil: NACA65410, pylon airfoil: NACA0004)

However, thermal energy transfer cannot be modeled in the panel code and therefore, for comparison purposes, the same ambient temperature was used for the engine exhaust boundary condition in CFX (and so the massflows are the same). In order to assess the influence of this simplification on the results a third run was prepared, enforcing an EGT of 900K at the engine outlet. The results are affected mostly in terms of the drag force - the error in the drag coefficient rises to 29.8%. More details on this paricular example, including C_p plots at various spanwise sections, are available in Appendix B.

The analysis performed in CFX required a mesh of over 10 million cells for convergence, local refinement being applied in order to adequately capture boundary layer behavior in the proximity of the body surfaces. The results are presented below in terms of both total aerodynamic coefficients (Table 6.1) and C_p plots at various stations across the span.

Table 6.1: Comparison with CFX

	Incompressible, $V_\infty = 60$m/s, $\alpha = 5$ deg			$M0.7, h = 11km, \alpha = 0$ deg		
	CFX	***Zephyr II***	**Error[%]**	**CFX**	***Zephyr II***	**Error[%]**
C_L	0.6071	0.6092	0.3	0.2844	0.2861	0.6
C_D	0.0314	0.0312	-0.6	0.0165	0.0183	10.6
C_M	-1.0208	-1.0300	-0.9	-0.5476	-0.5655	-3.3

From the pressure plots in Figs. 6.5 to 6.8, it is apparent how the presence of the nacelle is not perfectly captured in the panel code (stations at y = 4.40 m and y = 6.95 m). Also, the results usually differ near the trailing edge due to the prescribed wake methodology employed (in contrast, a free wake would require a computationally heavier iterative procedure to calculate).

Overall, the quality of the incompressible results presented in Table 6.1 is well within the acceptable range for a conceptual design stage. For the high subsonic case, the drag force is overestimated by a substantial margin, although the lift coefficient retains a small error. Hence, drag prediction capability at high Mach numbers, in the presence of pylons/nacelles, is limited, with the current code.

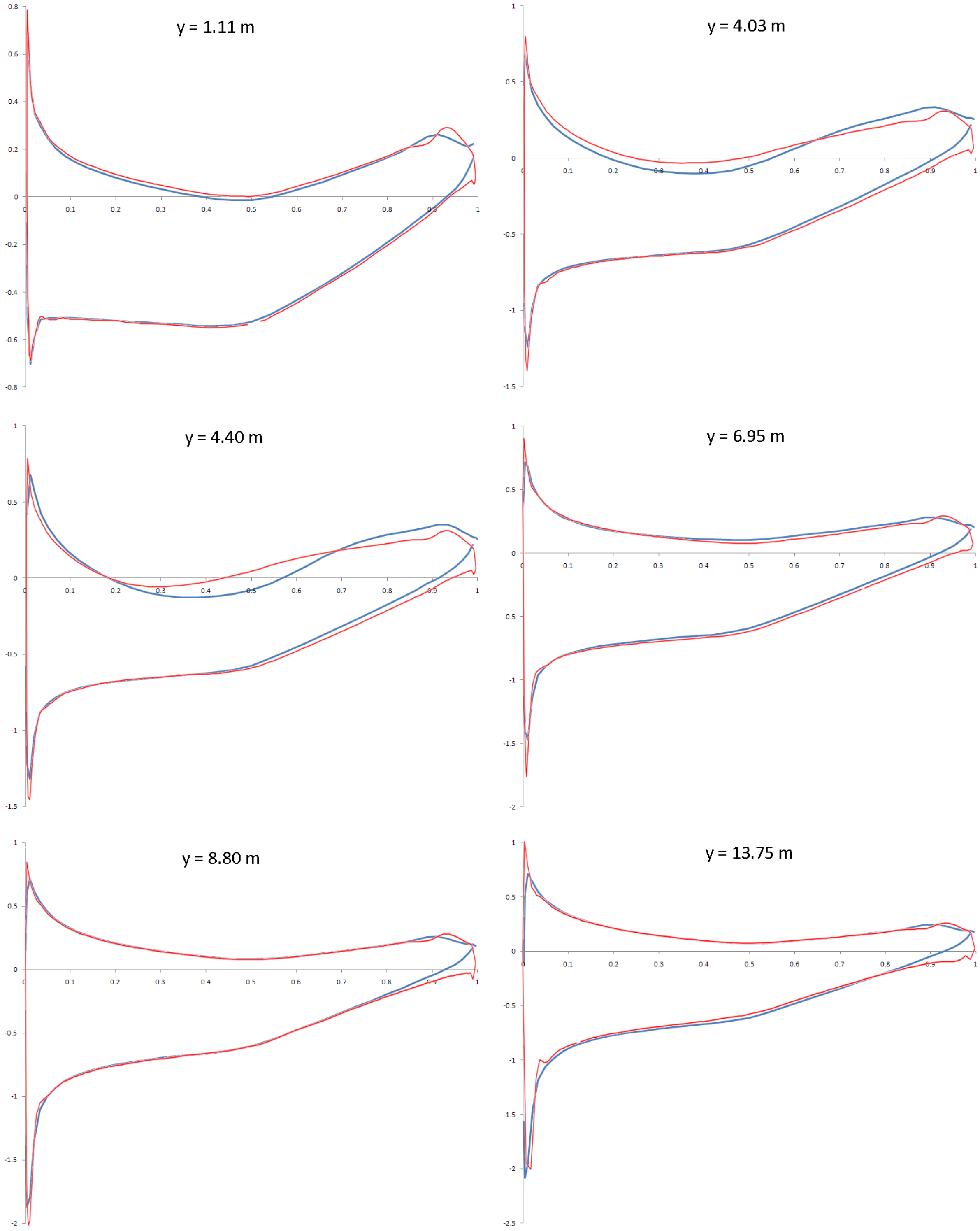


Figure 6.5: Streamwise C_p distributions, Incompressible case, $\alpha = 5$ deg

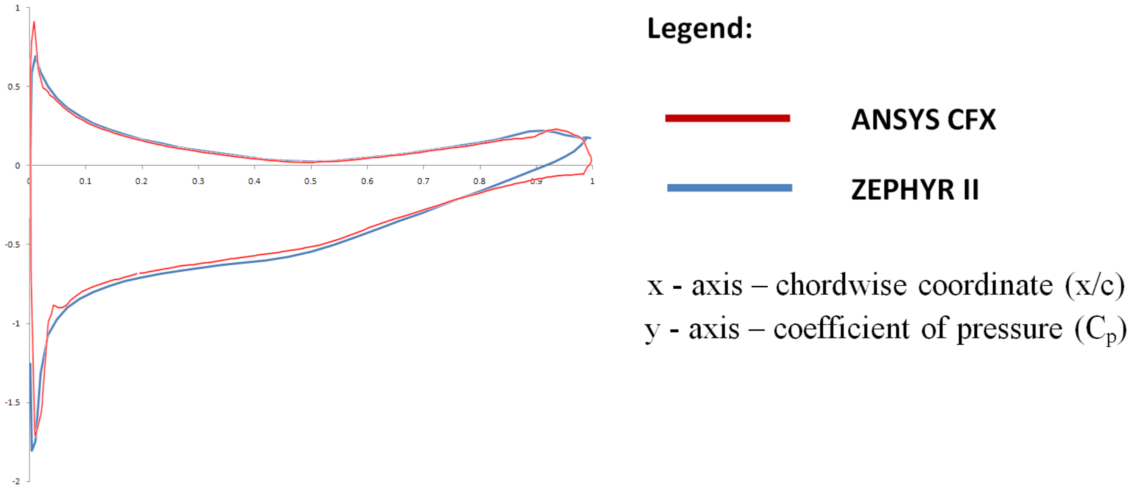


Figure 6.6: Streamwise C_p distributions, Incompressible case, $\alpha = 5$ deg (cont.)

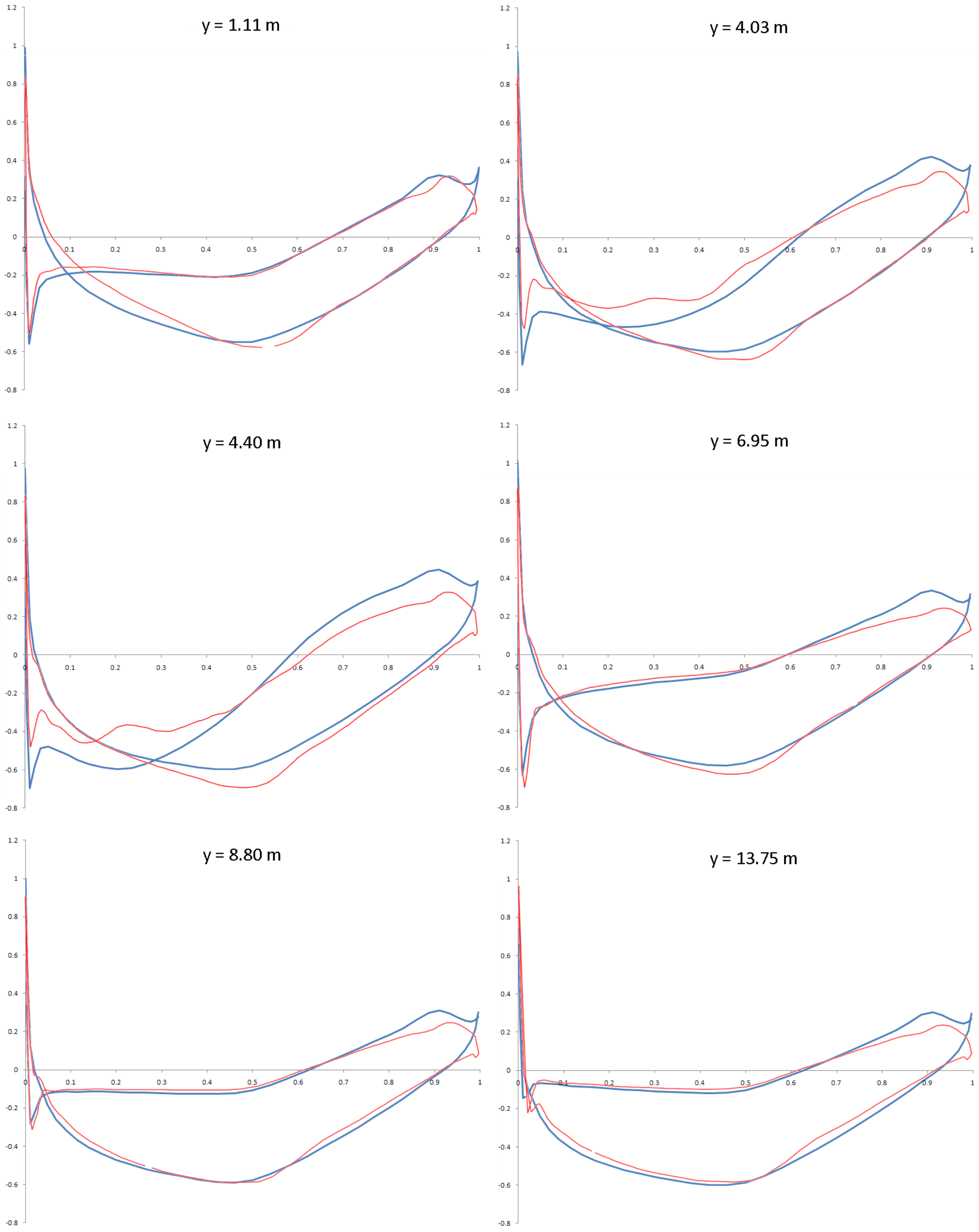


Figure 6.7: Streamwise C_p distributions, M0.7, h=11 km, $\alpha = 0$ deg

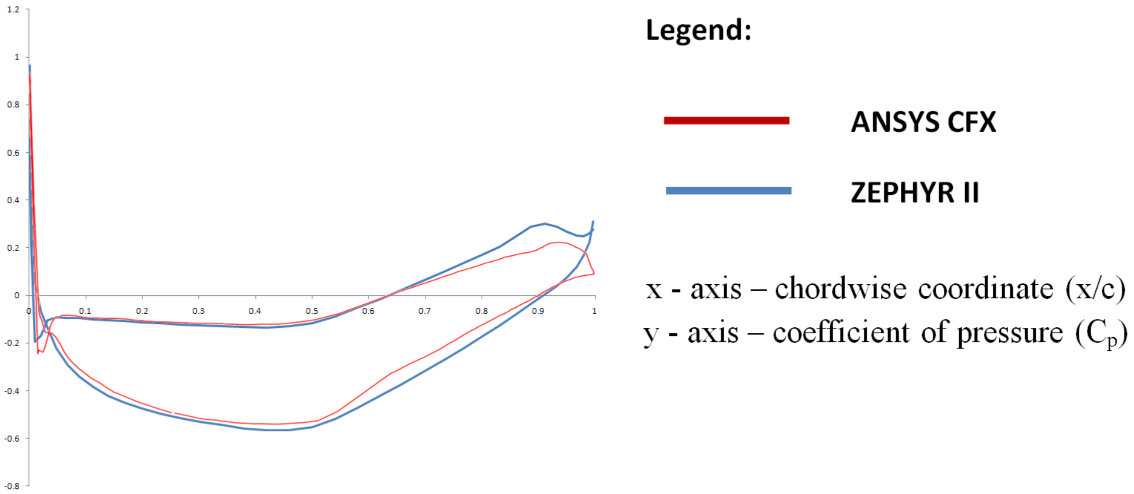


Figure 6.8: Streamwise C_p distributions, M0.7, h=11 km, $\alpha = 0$ deg, (cont.)

6.3 Finite Element Structural Analysis Module

6.3.1 Theory and Implementation

For a versatile and accurate representation of the wing structure, a finite element code is used to perform the structural analysis - *FEAP* - developed at the University of California [82]. This finite element package is capable of solving multiphysics problems: from electromagnetics to heat transfer, and of course, structural mechanics.

The wing structure is assumed to behave linearly in the elastic domain of the materials it is composed of (maximum displacements are required to be no more than $\approx$10% of the maximum dimension of the bounding box enclosing the wing), therefore, the interest lies in linear elasticity, which means the lumped contributions of every element may be assembled in a system of equations of the form:

$$\mathbf{K}\mathbf{u} = \mathbf{f} \tag{6.11}$$

where $\mathbf{K}$ is the global stiffness matrix (the superposition of all the elemental stiffness matrices according to element connectivities), $\mathbf{u}$ is the displacement field (both nodal linear displacements and rotations), and $\mathbf{f}$ is the vector of nodal loads. Boundary conditions are then applied either in terms of setting appropriate values in the load vector or by specifying displacements at particular nodes (e.g. fixed boundary at the wing root station).

Element stresses are also usually required to assess the performance/feasibility of the structural configuration under the given load. These are computed in a post processing step in which strains are first computed by applying the strain-displacement relation - $\mathbf{B}_\alpha$ (essentially a differential form of the element shape functions), particular to the type of element in use.

$$\epsilon = \mathbf{B}_\alpha \mathbf{u}^\alpha \tag{6.12}$$

where the nodal displacements $\mathbf{u}^\alpha$ are already transformed from the element local coordinate system to the global coordinate system.

Following the computation of strains, the constitutive law of the material is used (matrix $\mathbf{D}$) to obtain the stress field:

$$\sigma = \mathbf{D}\epsilon \tag{6.13}$$

For the aircraft wing structure, the discretization is performed using shell elements for all components with the exception of the stringers, which are modeled with beam elements.

In light of the fact that the meshing capabilities of *FEAP* are somewhat limited, especially for complex geometries, such as the wing model in use, an automated mesher code was written. This provides FEAP with an input where both geometry and mesh are already defined. The meshing procedure for skin panels, stringers and spar/rib caps is illustrated in Fig. 6.9, consisting of essentially four steps:

Step 1 Calculation of an interference pattern between structural components (also accounting for geometric discontinuities, e.g. breakstation).

Step 2 From the results of the previous step, build a mesh of convex polygons which become patches for meshing.

Step 3 Determine the mesh size on the sides of each patch based on reference element size and local value of the stretching function - Eq. 6.14 (currently, stretching is only applied in the chordwise direction).

Step 4 Apply Delaunay triangulation to each patch in order to obtain a mesh of triangular elements.

The chordwise stretching function is (for a uniformly distributed parameter η) [83]:

$$s = P\eta + (1 - P)\left(1 - \frac{\tanh(Q(1-\eta))}{\tanh(Q)}\right) \tag{6.14}$$

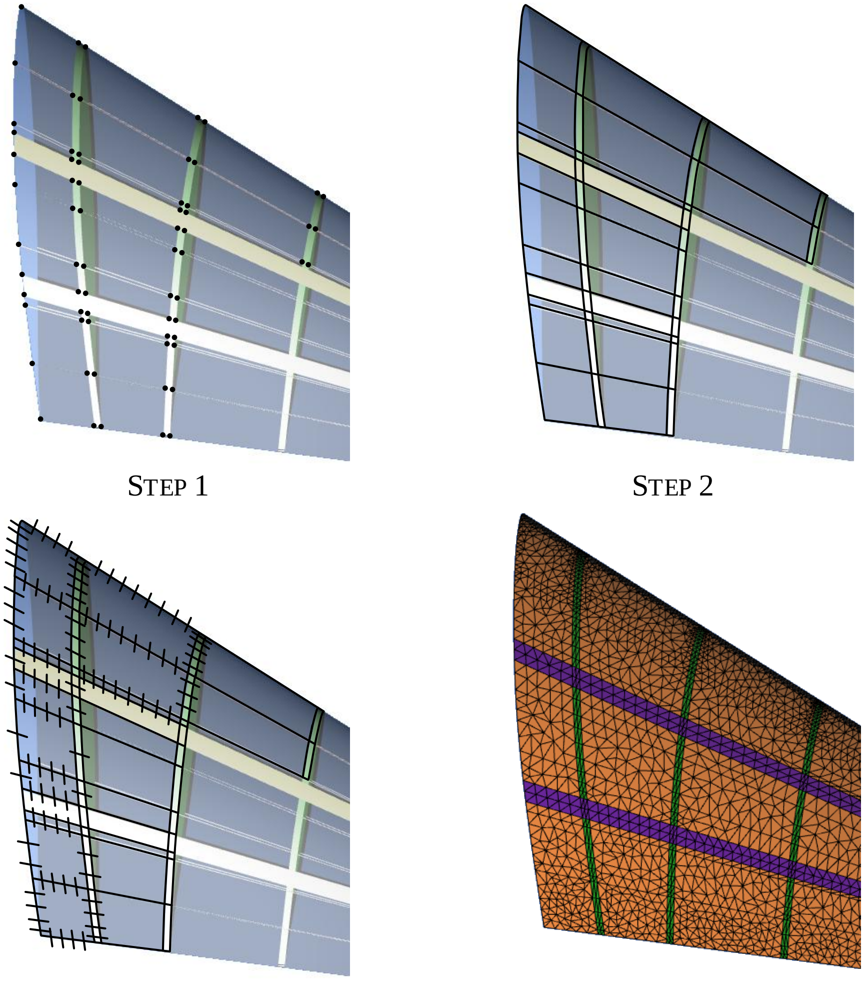
where s is the position along the line being meshed, and P and Q are parameters that control the stretching.

Quadrangular elements are used in the discretization of spar webs, as well as in the central region of rib webs. In the latter, Delaunay triangulation is applied to the leading and trailing edges (Fig. 6.10).

Fig. 6.11 displays a few examples of different configurations that have been successfully automatically meshed.

6.3.2 Load and Displacement Transfer

For true aeroelastic solutions, a Fluid-Structure Interaction (FSI) procedure is required. To that end, two-way FSI is implemented in the framework, where the loads



Step 1

Step 2

Step 3

Step 4

Figure 6.9: Automatic meshing procedure steps

obtained in the aerodynamics module are relayed to the finite element code, and conversely, structural displacements are applied to the aerodynamics mesh. This iterative process is however faced with one important obstacle: aerodynamics and structural surface meshes do not generally match. Therefore, in intermediate steps, interpolation of load and displacement fields at suitable locations must be performed. The algorithm presented herein is based on that described in [84].

Step 1 Set $i = 0$. Initial flow solution to obtain undeformed wing aerodynamic loading. Store the undeformed aerodynamics mesh (panels array).

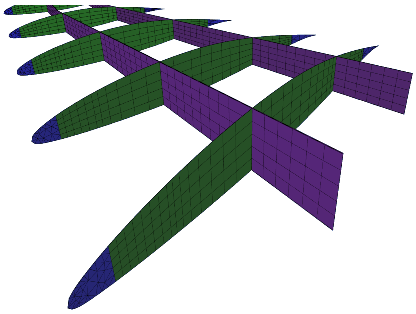


Figure 6.10: Mesh detail of spar and rib webs

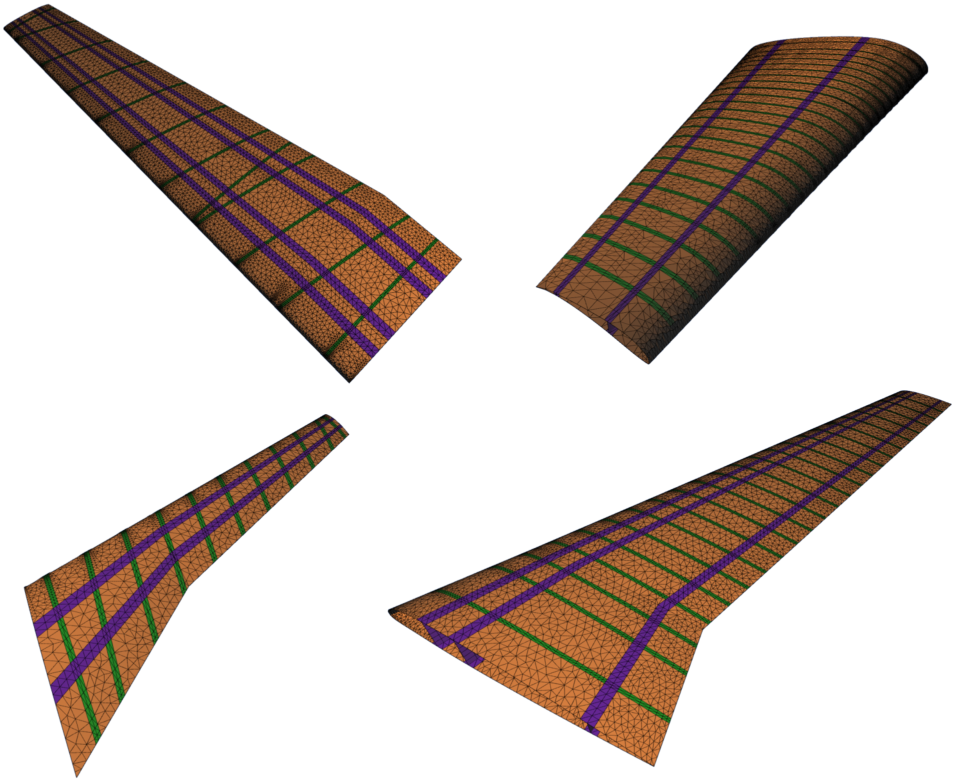


Figure 6.11: Automatic meshing examples

Step 2 *Load transfer*. The pressures on aerodynamics mesh panels are converted into concentrated loads at panel centers ($\mathbf{F_{AM}}$). These control points are then projected along the element normals onto the structural mesh (Fig. 6.12). The relative position of this projection in the underlying element determines how the load is distributed among the nodes (Eq. 6.15).

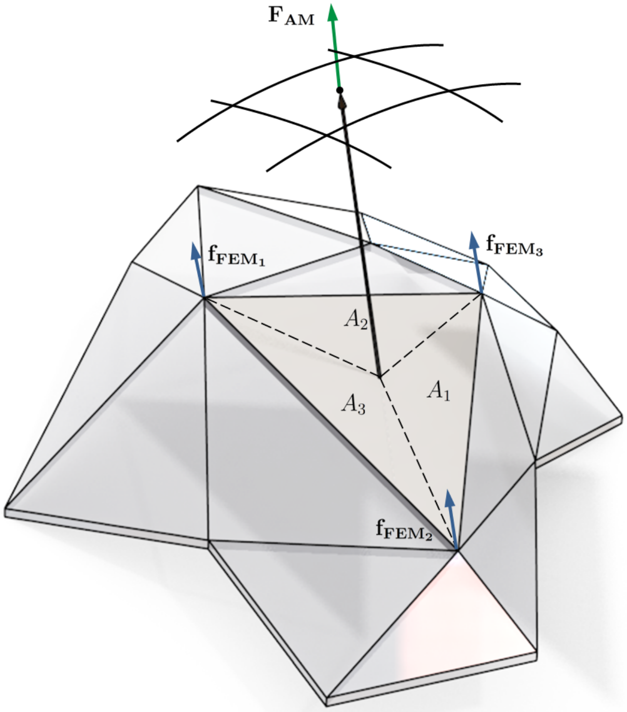


Figure 6.12: Load transfer

$$\begin{aligned}\mathbf{f_{FEM_1}} &= A_1\mathbf{F_{AM}}\\ \mathbf{f_{FEM_2}} &= A_2\mathbf{F_{AM}}\\ \mathbf{f_{FEM_3}} &= A_3\mathbf{F_{AM}}\end{aligned} \tag{6.15}$$

Step 3 Run a structural analysis and obtain the displacement field.

Step 4 *Displacement transfer*. In order to displace the aerodynamics mesh, each undeformed aerodynamics mesh point (panel corners) must be paired with a structural element (a single element may be paired with more than one panel corner but not vice versa). Again, this is accomplished by a similar procedure to the one undertaken in Step 2 for panel centers.

Because a general displacement field comprises not only translational degrees of freedom (DOF), but also rotational DOF, these have to be taken into consideration whilst deforming the aerodynamics mesh. $\mathbf{\Delta P} = \mathbf{P'} - \mathbf{P}$ represents the total displacement of an arbitrary aerodynamics mesh point (Fig. 6.13).

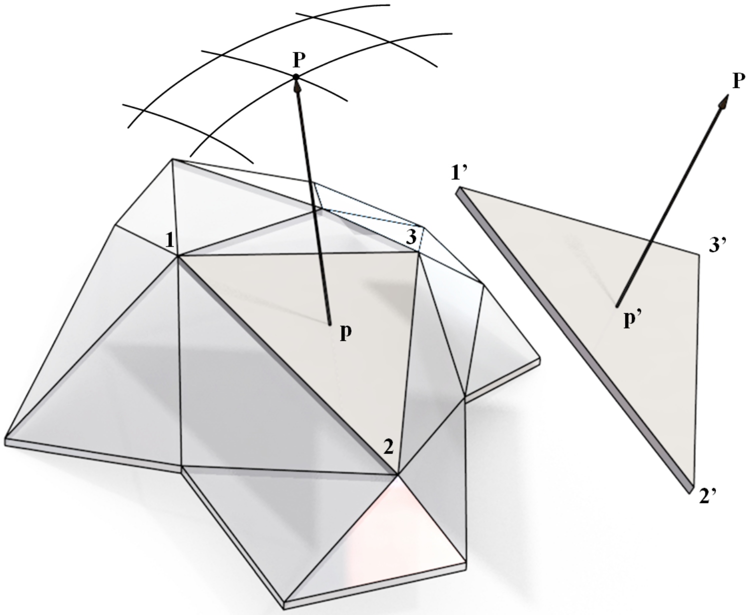


Figure 6.13: Displacement transfer

$$\Delta\mathbf{P} = \mathbf{P}' - \mathbf{P} = \mathbf{u_{tr}}\left(\mathbf{p}\right) + \left(\mathbf{p} - \mathbf{P}\right) \times \mathbf{u_{rot}}\left(\mathbf{p}\right) \tag{6.16}$$

$\mathbf{u_{tr}}$ and $\mathbf{u_{rot}}$ are subsets of the displacement vector $\mathbf{u}(\mathbf{p}) = \left[\mathbf{u_{tr}}(\mathbf{p})\ \mathbf{u_{rot}}\left(\mathbf{p}\right)\right]^T$. The value of the displacements at the projection point, $\mathbf{u}(\mathbf{p})$, is obtained from a wighted average of the nodal values:

$$\mathbf{u}(\mathbf{p}) = A_1\mathbf{u_1} + A_2\mathbf{u_2} + A_3\mathbf{u_3} \tag{6.17}$$

Step 5 Relaxation is applied to the new displacements before they are applied to the aerodynamic mesh proper. This is done in order to encourage a smoother convergence.

$$\Delta\mathbf{P}' = \omega\Delta\mathbf{P}_i + (1-\omega)\Delta\mathbf{P}_{i-1} \tag{6.18}$$

A typical value for ω is 0.6.

Step 6 Apply displacements to the aerodynamic mesh and perform the aerodynamic analysis, thereby obtaining new panel pressures.

Step 7 Convergence check - either aerodynamic coefficients or tip displacement may be used as criteria. If failed, continue to Step 8, otherwise stop.

Step 8 Proceed to Step 2 ($i = i + 1$).

Note that the pairing between undeformed aerodynamic panel points and structural elements needs to be done only once.

Additional loads applied to the FE model include the weight of the structure (mass of each element is evenly distributed across its nodes - all elements are constant thickness/section) and engine installation loads. Engine loads comprise the aerodynamic loads as well as weight and thrust (the last two are user provided and applied at the center of the nacelle), for which an equivalent force and moment is computed at the center of the pylon/wing intersection. As an approximation, this force and moment are then evenly distributed across the intersection area. The wing structure is tested without any fuel loads.

The procedure described above ensures the coupling between aerodynamics and structural disciplines and is used in MDO architectures such as MDF. In certain architectures, however (e.g. Individual Discipline Feasible (IDF), not covered in this thesis), this type of procedure is controlled by the optimizer itself by means of penalty/constraint functions which force the solution to converge to an aerostructural equilibrium state (not necessarily attained at each optimizer iteration).

In order to illustrate the impact of structural flexibility on aerodynamic performance, an aeroelastic analysis is performed on the test wing that will be used in section 6.3.3. The results are summarized in Table 6.2.

Table 6.2: Rigid vs. non-rigid aerodynamic analysis

M0.7, h = 11 km, $\alpha = 0$ deg			
	Rigid	**Non-Rigid**	**Change[%]**
C_L	0.2861	0.1941	-32.2
C_D	0.0183	0.0174	-4.9
C_M	-0.5655	-0.3832	-32.2

The bending and twisting of the structure reduces overall lift by more than 30%, which is not surprising given the flexibility of the test wing. More typical reductions are usually in the 10-20% range [85]. Nevertheless, aeroelastic effects are shown to have an impact on the design. Fig. 6.14 illustrates the deformed wing in true scale and the resulting pressure field.

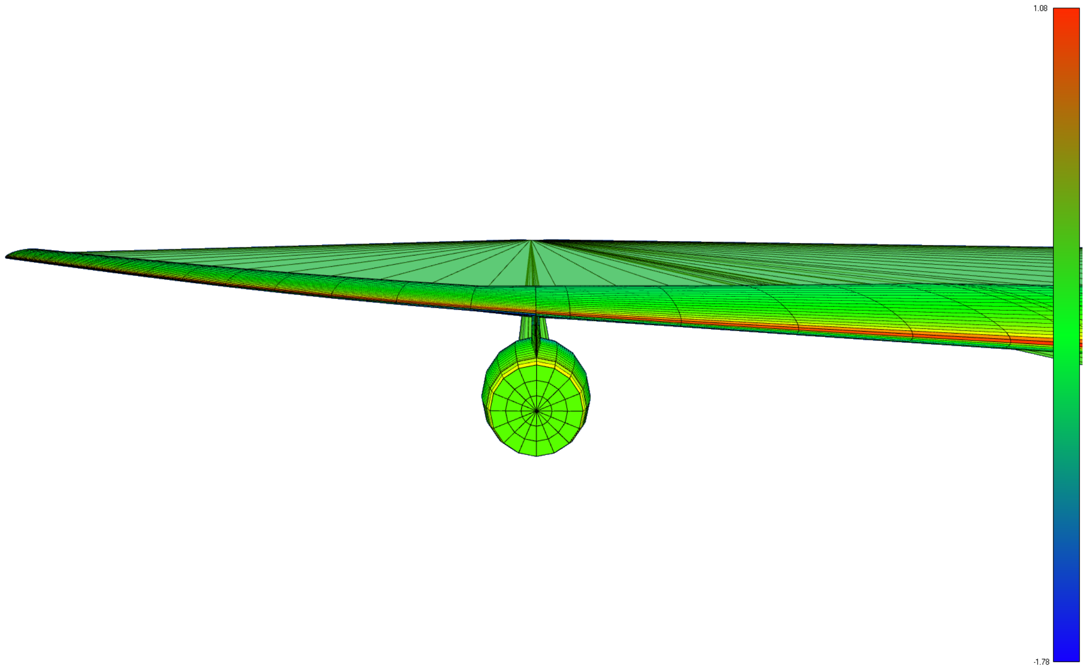


Figure 6.14: Deformed wing (true scale) and respective pressure distribution (C_p)

6.3.3 Validation

For validation of the structural module, the geometry and loading from the compressible case in section 6.2.3 is reused. The internal structure is illustrated in Fig. 6.15. The material used is aluminum (E = 70 GPa, ν = 0.33, ρ = 2700 Kg.m^{-3}) and for simplicity, all thicknesses are set to 0.005 mm. An automatic mesh of 27319 elements is then built for the analysis. Engine thrust is set at 11 kN and pylon/nacelle/engine weight is taken as 9 kN.

The results obtained with FEAP are compared against those of ANSYS$^{®}$ (the same mesh and loads are used by parsing an ANSYS Parametric Design Language (APDL) input file that is written by the mesher) in Table 6.3. The positions for stress probing are shown in the contour plots of the upper and lower surfaces of the test wing (Fig. 6.16).

Beyond the results presented in Table 6.3, the RMS value of the relative error in the displacement field was determined to be 3.6%. Overall, the results show an agreement between the two codes, the existing differences being only explained by different implementations of the element types. Noticeably, these differences are most prominent where the stresses are lower (position 4 and 8). Also worthy of note is the

Table 6.3: Comparison with ANSYS®

	Units	**ANSYS**	***FEAP***	**Error[%]**
Maximum displacement (norm)	[m]	1.372	1.378	0.4
Overall maximum equivalent stress	[MPa]	443.28	421.38	-4.9
Equivalent stress at position:		(shell top/bottom)		
1 (X = 4.082m, Y = 2.778m, Z = 0.321m)	[MPa]	95.89/107.16	101.27/102.21	5.6/-4.6
2 (X = 5.193m, Y = 5.583m, Z = 0.220m)	[MPa]	64.06/75.69	66.29/77.70	3.5/2.7
3 (X = 7.854m, Y = 10.590m, Z = 0.183m)	[MPa]	65.11/68.57	64.13/71.12	-1.5/3.7
4 (X = 10.909m, Y = 15.689m, Z = 0.147m)	[MPa]	3.23/3.76	3.11/4.68	-3.7/24.5
5 (X = 4.082m, Y = 2.778m, Z = -0.118m)	[MPa]	86.29/75.69	82.73/79.83	-4.1/4.1
6 (X = 5.193m, Y = 5.583m, Z = -0.080m)	[MPa]	54.94/82.38	57.31/80.06	4.3/-2.8
7 (X = 7.854m, Y = 10.590m, Z = -0.071m)	[MPa]	55.50/54.50	56.32/54.06	1.5/-0.8
8 (X = 10.909m, Y = 15.689m, Z = -0.0551m)	[MPa]	3.05/3.01	3.41/3.05	11.8/1.3

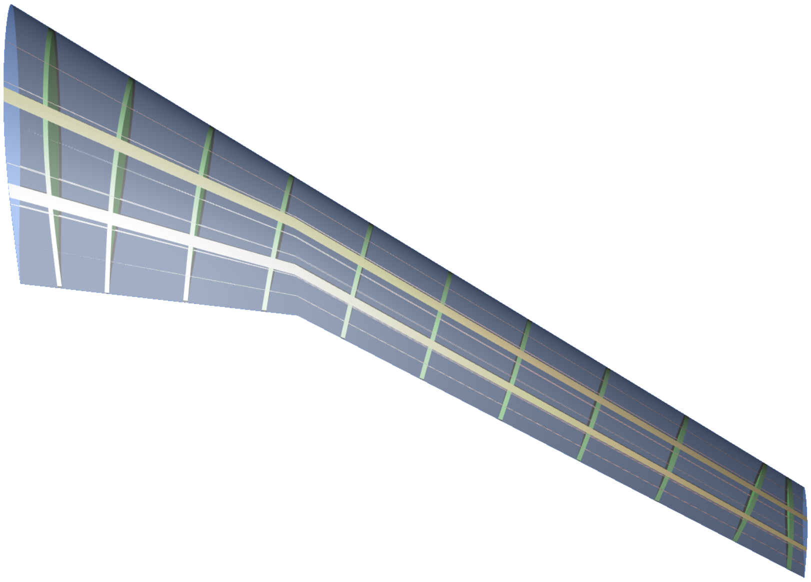


Figure 6.15: Wing structure used for validation

fact that if a typical yield stress for high strength aluminum is considered to be 400 MPa, the structure would be failing under this load (maximum equivalent stress occurs in the lower surface, at the pylon support). Therefore, as it is desirable to start an optimization based on a feasible design, the baseline wing structure used for optimization purposes in the next chapter will be a modified version of the one presented here.

6.4 Framework Layout and User Interface

The framework, coded in C# object oriented programming, handles the definition of the optimization problems by managing arrays of design variables, objective and constraints, as well as the data structures containing input/output to/from the analyzer modules. It also allows for the interactive user interface that will be described in more detail further ahead. The analyzer modules are coded in C++ for the sake of performance, and are encapsulated as dynamic link libraries (.dll), meaning they can be replaced/updated with relative ease, independently of the main application. The result is a standalone application which does not require an encapsulating environment to function. In Fig. 6.17, an informal class diagram is presented, showing the main classes and methods.

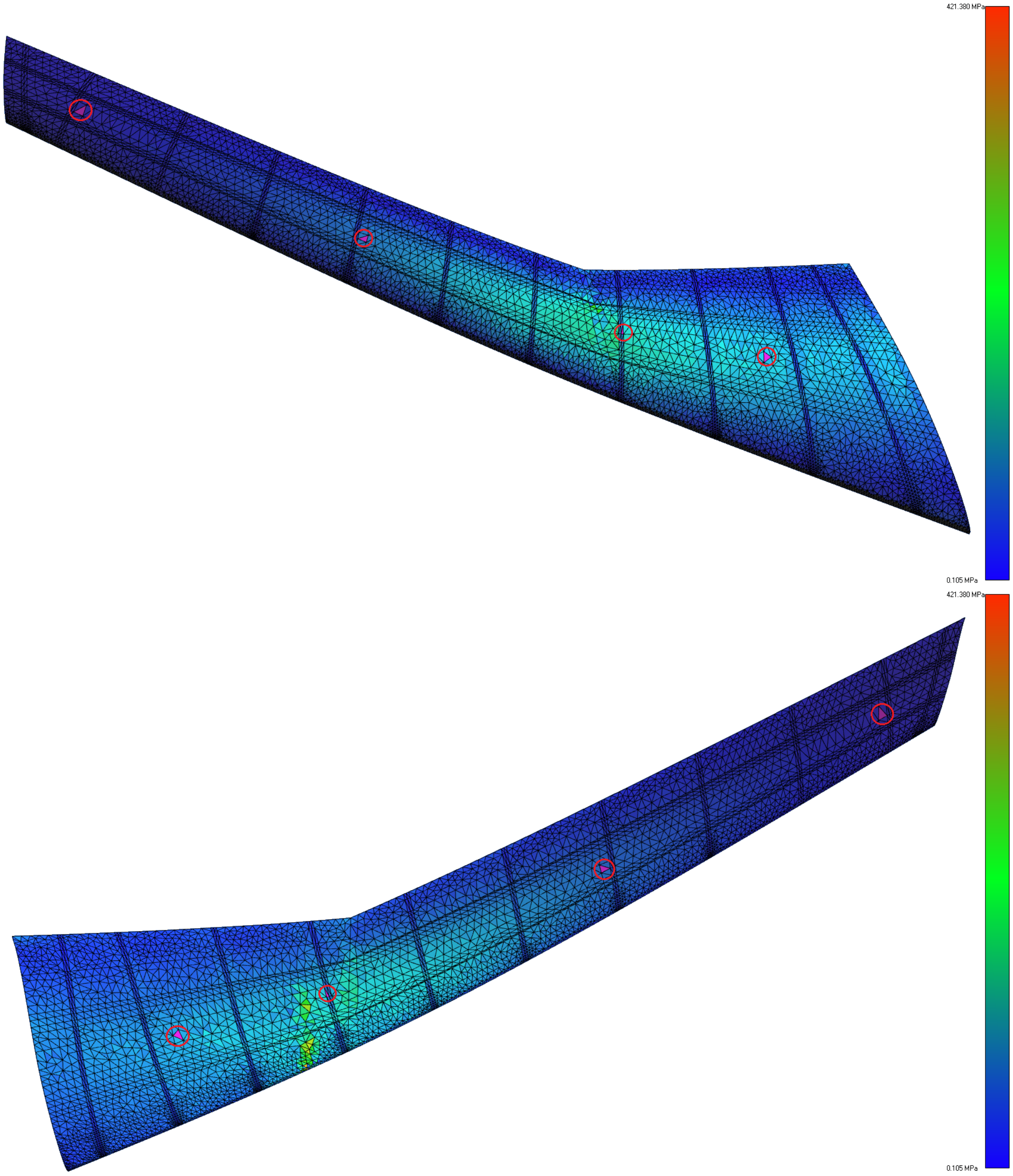


Figure 6.16: Von Mises stress contour plots for upper and lower wing surfaces (shell top results). Highlighted elements indicate stress probes.

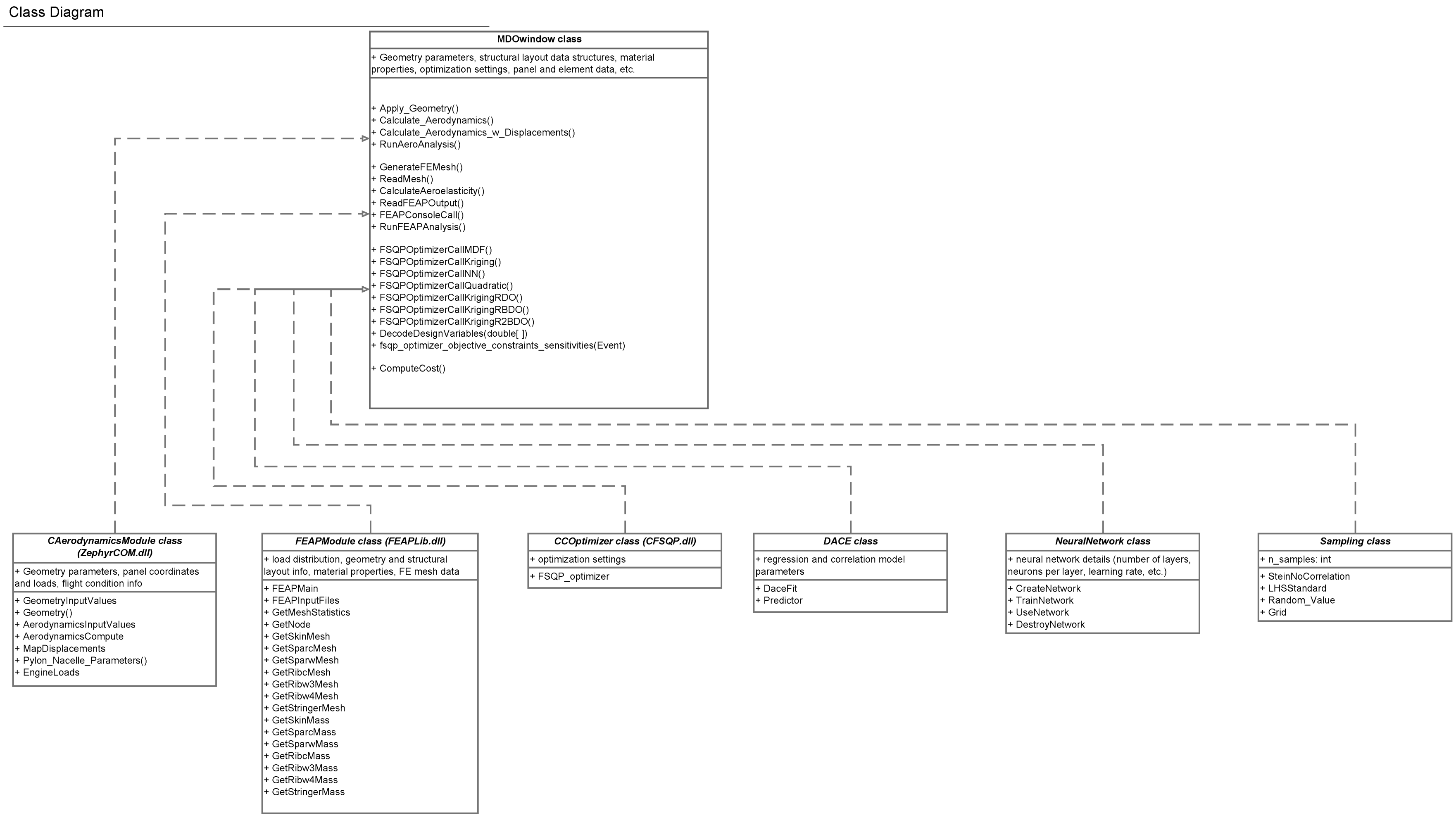
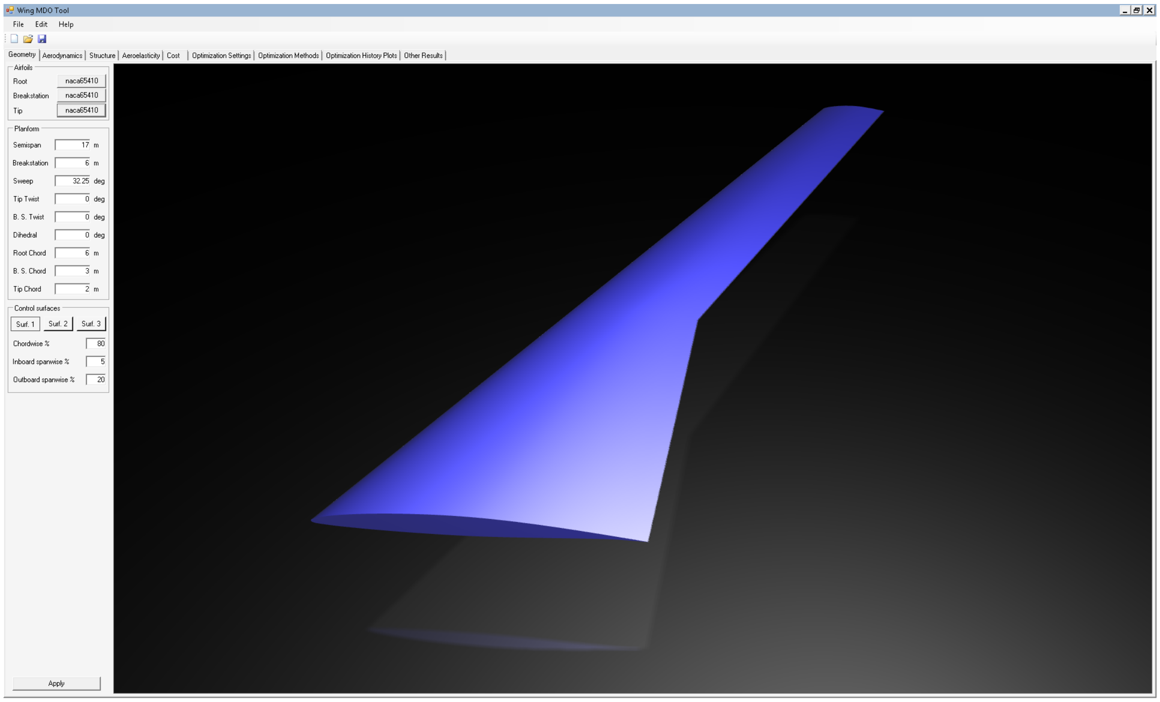


Figure 6.17: Class diagram for the framework (showing only main classes and methods)

The interface that was developed for this framework is divided into several sections. Their purpose is to guide the user in first creating a baseline design and analyzing it if needed, an then to engage into deterministic/non-deterministic optimization using either direct evaluation or surrogate models, as explained in previous chapters.

In the first tab, *Geometry* (Fig. 6.18), the general shape of the wing is defined based on its geometric dimensions and user defined airfoils at three spanwise sections (see Chapter 5). The geometry engine interpolates airfoil points by means of either cubic or Akima splines, after which the wing surface is created by a bilinear interpolation loft of these sections.



Root naca65410
Breakstation naca65410
Tip naca65410

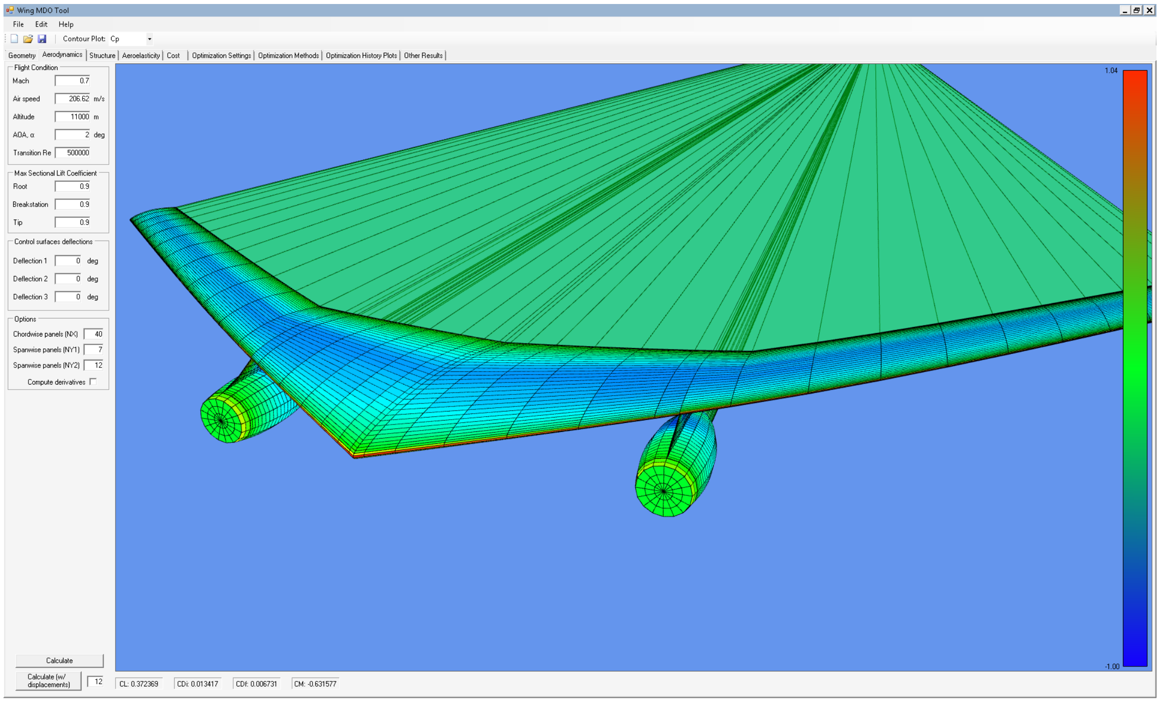
Semispan 17 m
Breakstation 6 m
Sweep 32.25 deg
Tip Twist 0 deg
B. S. Twist 0 deg
Dihedral 0 deg
Root Chord 6 m
B. S. Chord 3 m
Tip Chord 2 m

Chordwise % 80
Inboard spanwise % 5
Outboard spanwise % 20

Figure 6.18: Geometry module

In the *Aerodynamics* tab (Fig. 6.19), a prediction of wing aerodynamic performance can be obtained for a custom flight condition trough Zephyr II, the embedded analyzer module. It employs the Intel® Math Kernel Library, which enables multithreading support. The window then displays a three dimensional model of the panel meshed geometry, color coded for either coefficient of pressure or doublet strength values.

The *Structure* (Fig. 6.20) tab performs a similar function to *Geometry* but fo-



Contour Plot: Cp

Mach 0.7
Air speed 206.62 m/s
Altitude 11000 m
AOA, α 2 deg
Transition Re 500000

Root 0.9
Breakstation 0.9
Tip 0.9

Deflection 1 0 deg
Deflection 2 0 deg
Deflection 3 0 deg

Chordwise panels (NX) 40
Spanwise panels (NY1) 7
Spanwise panels (NY2) 12
Compute derivatives ☐

12

CL: 0.372369
CDi: 0.013417
CDf: 0.006731
CM: -0.631577

Figure 6.19: Aerodynamics module

cused on the internal structure of the wing. Here, the data structure containing dimensions and material properties details for the various structural elements can be easily manipulated through a *Treeview* control.

Upon the definition of the structure the automatic mesher may be activated in the fourth tab (meshing parameters such as default element size or stretching functions are, as of this version, still defined by keying them directly in the code and are not yet available in the interface). The finite element solver powered by FEAP is also called in this tab where the results of the structural analysis may be viewed in terms of contour plots of either the displacement or stress fields (Fig. 6.21). A 2 GB memory limitation is in effect for the stiffness matrices created by FEAP, which in practice means a maximum of around 40-50000 elements may be defined.

Setting up optimization problems and controlling the usage of the CFSQP embedded optimizer is the function of the last couple of tabs. Here, the design space is defined (design variables and their range), nonlinear constraints are set - linear constraints are handled automatically based on which variables were chosen for optimization -, and optimizer settings can be tweaked. The second tab in this group allows the customization of deterministic or robustness/reliability based optimization

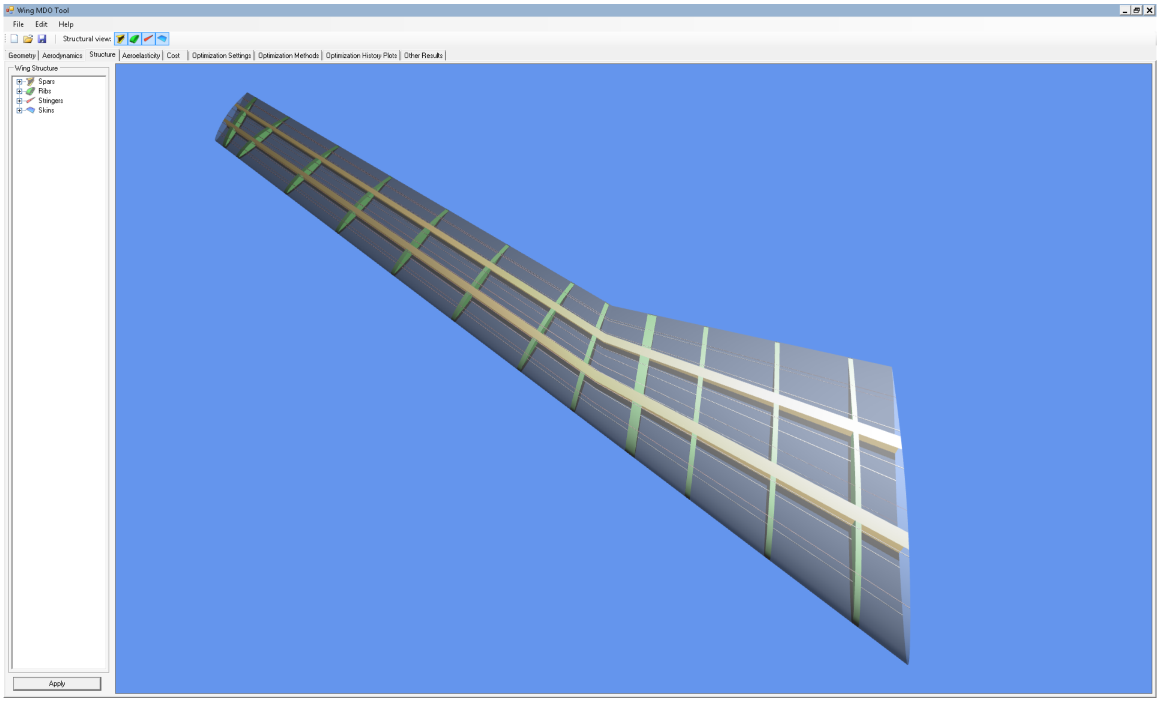
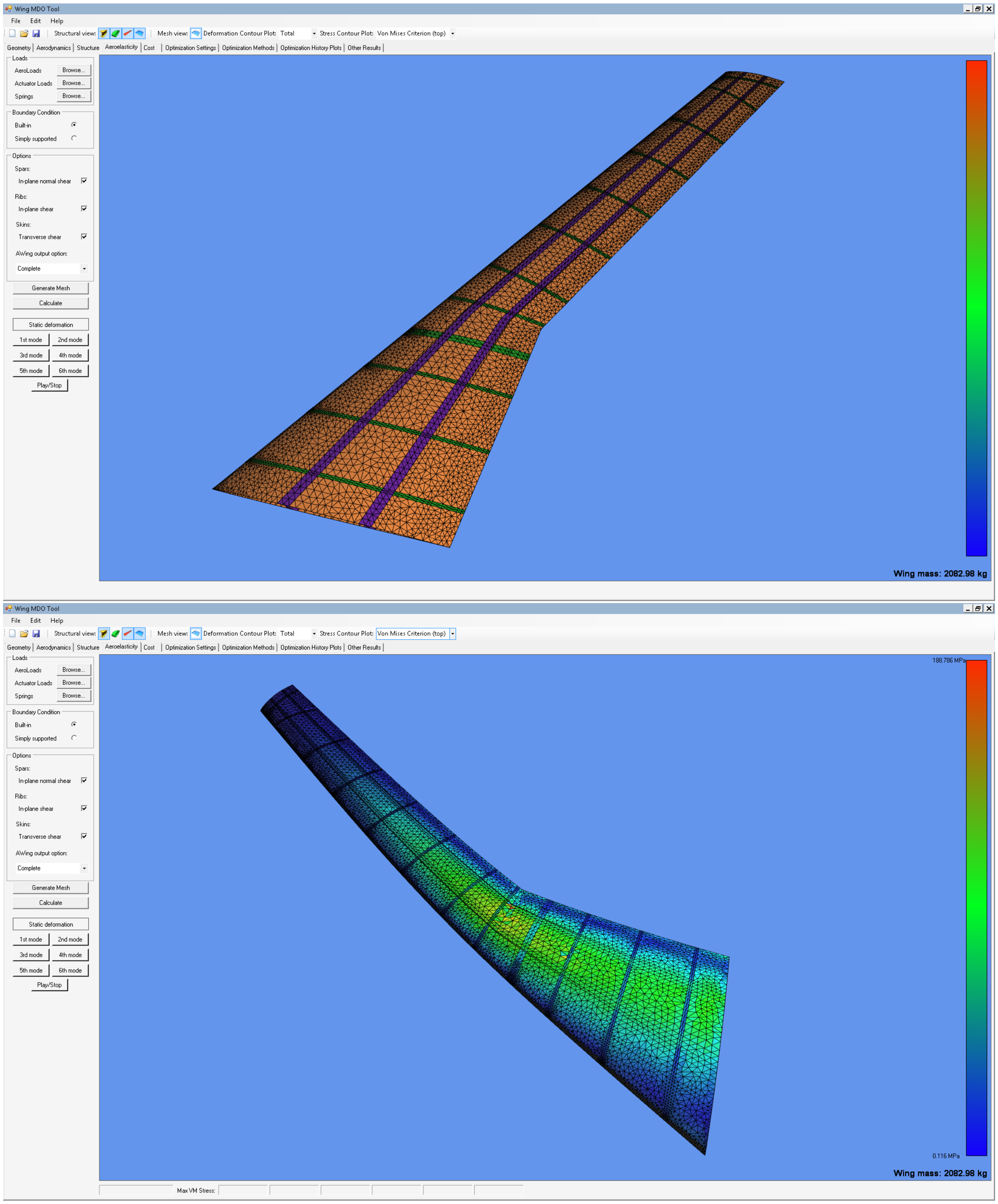


Figure 6.20: Structure module

runs and surrogate model parameters (Fig. 6.22). The analysis modules are linked to the optimizer in an MDF type of architecture.



Deformation Contour Plot: Total
Stress Contour Plot: Von Mises Criterion (top)

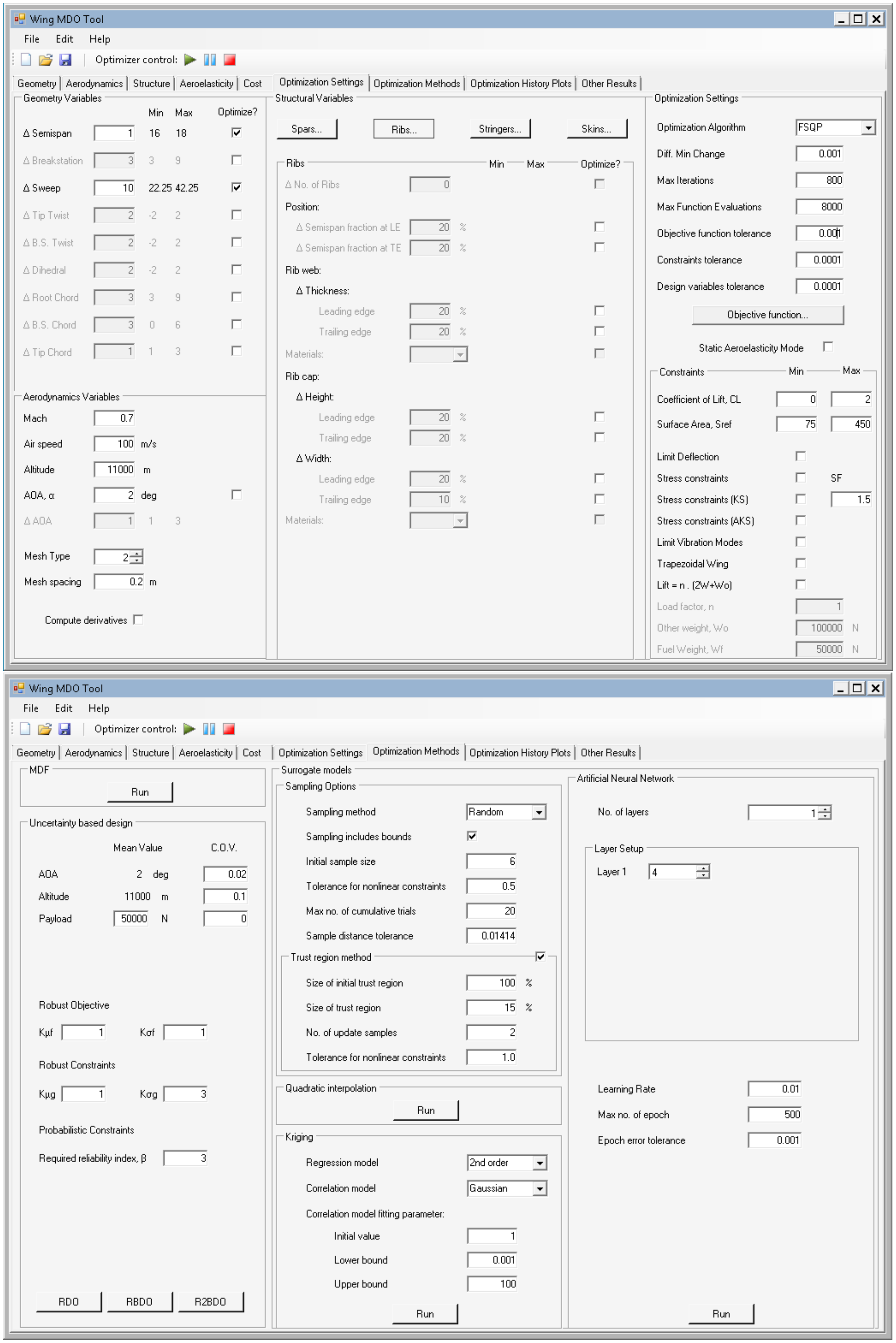
Built-in ●
Simply supported ○

In-plane normal shear ☑
In-plane shear ☑
Transverse shear ☑

AWing output option: Complete

Wing mass: 2082.98 kg

Figure 6.21: Aeroelasticity module



Geometry Variables

		Min	Max	Optimize?
Δ Semispan	1	16	18	☑
Δ Breakstation	3	3	9	☐
Δ Sweep	10	22.25	42.25	☑
Δ Tip Twist	2	-2	2	☐
Δ B.S. Twist	2	-2	2	☐
Δ Dihedral	2	-2	2	☐
Δ Root Chord	3	3	9	☐
Δ B.S. Chord	3	0	6	☐
Δ Tip Chord	1	1	3	☐

Aerodynamics Variables

Mach 0.7
Air speed 100 m/s
Altitude 11000 m
AOA, α 2 deg ☐
Δ AOA 1 1 3
Mesh Type 2
Mesh spacing 0.2 m
Compute derivatives ☐

Structural Variables

Ribs			Min	Max	Optimize?
Δ No. of Ribs	0				☐
Position:					
Δ Semispan fraction at LE	20	%			☐
Δ Semispan fraction at TE	20	%			☐
Rib web:					
Δ Thickness:					
Leading edge	20	%			☐
Trailing edge	20	%			☐
Materials:					☐
Rib cap:					
Δ Height:					
Leading edge	20	%			☐
Trailing edge	20	%			☐
Δ Width:					
Leading edge	20	%			☐
Trailing edge	10	%			☐
Materials:					☐

Optimization Settings

Optimization Algorithm FSQP
Diff. Min Change 0.001
Max Iterations 800
Max Function Evaluations 8000
Objective function tolerance 0.001
Constraints tolerance 0.0001
Design variables tolerance 0.0001
Static Aeroelasticity Mode ☐

Constraints	Min	Max
Coefficient of Lift, CL	0	2
Surface Area, Sref	75	450
Limit Deflection ☐		
Stress constraints ☐		SF
Stress constraints (KS) ☐		1.5
Stress constraints (AKS) ☐		
Limit Vibration Modes ☐		
Trapezoidal Wing ☐		
Lift = n . (2W+Wo) ☐		
Load factor, n	1	
Other weight, Wo	100000	N
Fuel Weight, Wf	50000	N

MDF

Uncertainty based design

	Mean Value		C.O.V.
AOA	2	deg	0.02
Altitude	11000	m	0.1
Payload	50000	N	0

Robust Objective
Kμf 1 Kσf 1

Robust Constraints
Kμg 1 Kσg 3

Probabilistic Constraints
Required reliability index, β 3

Surrogate models

Sampling Options
Sampling method Random
Sampling includes bounds ☑
Initial sample size 6
Tolerance for nonlinear constraints 0.5
Max no. of cumulative trials 20
Sample distance tolerance 0.01414

Trust region method ☑
Size of initial trust region 100 %
Size of trust region 15 %
No. of update samples 2
Tolerance for nonlinear constraints 1.0

Quadratic interpolation

Kriging
Regression model 2nd order
Correlation model Gaussian
Correlation model fitting parameter:
Initial value 1
Lower bound 0.001
Upper bound 100

Artificial Neural Network
No. of layers 1
Layer Setup
Layer 1 4
Learning Rate 0.01
Max no. of epoch 500
Epoch error tolerance 0.001

Figure 6.22: Optimization module

Chapter 7

Robust and Reliability-Based Design: Simulation and Evaluation

In this chapter, practical application is given to the MDO framework in the form of wing design problems. While uncertainty is considered in flight condition parameters (such as angle of attack or altitude), no uncertainty is considered in geometric dimensions or material properties. This is deemed sensible since at the conceptual design stage, manufacturing tolerances are most likely unknown (and even if specified, they would be of such magnitude that their implications on the design would not be sensed by the analyzer modules in use). Uncertainties in material properties are also a topic to be more thoroughly analyzed in later stages of the design process as they are dependent not only on the material itself, but also on the manufacturing process and even the supplier.

Departures from theoretical flight conditions can have both direct and indirect influences on the performance of a cruising aircraft. In a representative set of uncertain parameters chosen, direct influences are portrayed by uncertainty in the angle of attack (linear effect on wing loading due to lift). Other indirect sources of uncertainty, such as differences in pressure altitude, cause changes in the properties of air (namely viscosity, density and, by consequence, local speed of sound), which affect both wing loading due to lift and drag.

The remaining uncertain parameter is the payload (cargo or passengers). Typical allowances for per-passenger weights at the conceptual design stage may be violated by a significant margin in regular operations. In addition, passenger load varies according to several factors: season of the year, fares, etc. Although payload estimation is a routine procedure in airliner operations (some aircraft models even possess weight sensors in their undercarriage), the fact is that this serves only as a means to determine fuel load and ensure weight balance (for stability purposes). The question of whether an aircraft remains a cost effective platform (i.e. maintains its competitiveness) for a wide range of payload values should be addressed by the designer.

The normal probability distributions used to characterize the random parameters mentioned above are presented in Fig. 7.1. The PDF for the angle of attack in cruise is shown centered around a single value, although in Case Study II it becomes a design variable and, as such, its standard deviation is not constant.

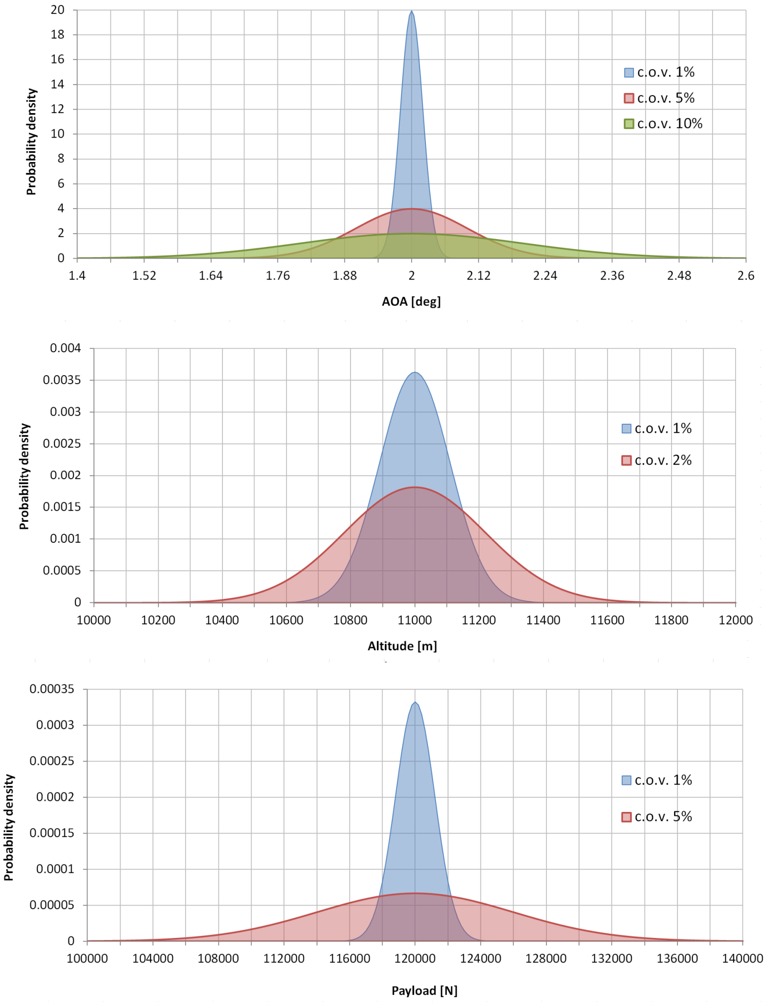


Figure 7.1: Probability distributions for uncertain variables in Case Studies I and II

Essentially, two design problems will be covered: one with 3 geometric design variables and 2 uncertain parameters and another one with 8 design variables, 2 uncertain parameters and one uncertain design variable. Altitude, $\mu_{Alt} = 11000$ m, angle of attack (AOA), $\mu_{AOA} = 2$ deg and a Mach number of 0.7 characterize the expected flight conditions for all problems.

The baseline wing geometry and structure are inherited from the previous section, albeit with some modifications to accommodate for the stress concentration that was verified at the pylon-wing intersection. Therefore, a reinforced rib is added at the location where the pylon is supported, displacing neighboring ribs. In addition, stringers are moved so that more configurations are possible for spar locations in the second case study (otherwise these would be tightly constrained). The new arrangement is shown in Fig. 7.2.

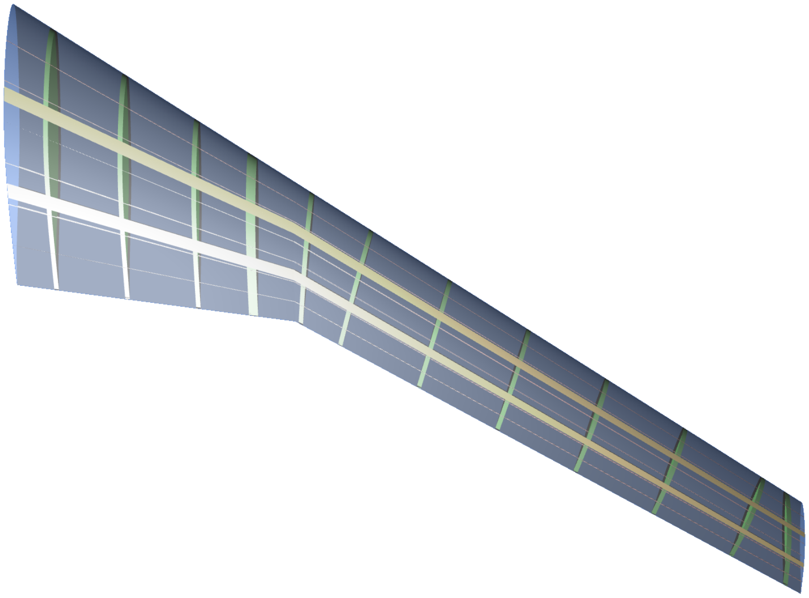


Figure 7.2: Baseline wing structure for optimization problems

7.1 Case Study I: three design variables, two random variables

The design variable bounds and constraints for this first set of problems are shown in Table 7.2. Altitude and AOA are the uncertain parameters.

The objective is to maximize the lift to drag ratio, which is an important measure

Table 7.1: Design variables and constraints for Case Study I

Design Variable	Baseline	Min	Max
semispan, $b/2$	17 m	16 m	18 m
root chord, c_{root}	6 m	4.5 m	9 m
tip chord, c_{tip}	2 m	1.5 m	2.5 m
$\frac{1}{\rho}\ \log\left(\sum_{i=1}^{N} e^{\rho g_i}\right) \leq 0,\ g_i = 1 - \frac{\sigma_i}{\sigma_{\text{yield}}}$			

of how efficient a wing is at producing lift (it directly affects the maximum range of an aircraft - see section 7.2). Even small changes in L/D can represent significant savings in aircraft operating costs. In this case, for RDO and R^{2}BDO the robust objective is:

$$F(\mathbf{x}, \mathbf{r}) = -\mu_{L/D} + \sigma_{L/D} \tag{7.1}$$

The required reliability index is set to 3 ($\beta_{reqd} = 3$) which means that, as before (sections 2.4.3 and 3.5.1), $K_\sigma = 3$ in the robust constraint for RDO. The deterministic optimum used for comparison is obtained by performing direct evaluation of the objective/constraint functions at mean values of the parameters. The uncertainty based design problems employ a Kriging surrogate initially and then revert to direct evaluation to refine the result. The values presented for the required number of function evaluations are, nevertheless, only indicative of performance because the problems were only repeated to ensure the result was a minima, rather than to fully evaluate numerical performance. Statistically characterizing the computational cost as in Chapters 3 and 5 would require several runs for each problem, which is not feasible in this case given the cost of each individual analysis. For comparison purposes only, the results for the optimized wing, obtained using two-way FSI, are also shown (the * next to the number of function evaluations indicates the latter corresponds to fully coupled aeroelastic evaluations which themselves include several individual aerodynamics and structural evaluations).

The results in Table 7.2 show the expected general trend where as the uncertainty in the angle of attack increases, the performance of the optimal configurations diminishes (as measured by $\mu_{L/D}$) and at the same time its standard deviation increases. This is explained by the decrease in the aspect ratio of the wing to maintain the design feasible with respect to the reliability-based stress constraint (in short, smaller wings are both more robust and more reliable). The configurations obtained through

Table 7.2: Results for Case Study I (c.o.v. values are for angle of attack and altitude, respectively)

	Baseline	Deterministic	c.o.v. 1%, 1%		c.o.v. 5%, 1%		c.o.v. 10%, 1%		Aeroelastic (deterministic)
Variable			RDO	R^2BDO	RDO	R^2BDO	RDO	R^2BDO	
semispan	17 m	17.988 m	17.865 m	17.995 m	17.765 m	17.352 m	17.782 m	17.206 m	17.086 m
root chord	6 m	5.779 m	5.636 m	5.735 m	5.360 m	6.139 m	5.376 m	5.791 m	5.640 m
tip chord	2 m	1.5 m	1.508 m	1.500 m	1.684 m	1.501 m	1.553 m	1.684 m	1.992 m
F	-22.807	-23.697	-23.489	-23.636	-23.113	-22.943	-22.834	-22.450	-23.531
$\mu_{L/D}$	-	23.697	23.547	23.694	23.375	23.185	23.364	22.939	-
$\sigma_{L/D}$	-	0	0.058	0.058	0.262	0.242	0.530	0.489	-
# eval.	-	260	524	758	1718	617	965	662	321 (*)

RDO are invariably less conservative. In Fig. 7.4, a qualitative comparison of wing shapes obtained for RDO, R^2BDO and deterministic optimization is presented.

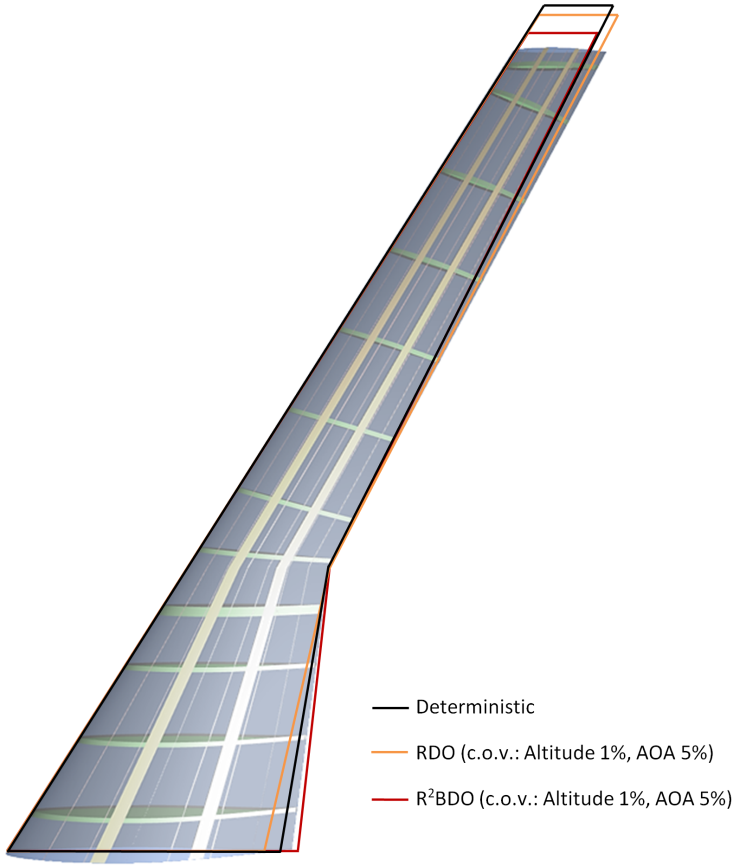


Figure 7.3: Comparison of results for Case I, c.o.v.: Altitude 1%, AOA 5%.

7.2 Case Study II: nine design variables, three random variables

In this second problem, 9 DVs are used, among which is the angle of attack, which doubles as an uncertain variable, together with altitude and payload.

Table 7.3: Design variables and constraints for Case Study II

Design Variable	Baseline	Min	Max
angle of attack, AOA	2 deg	1 deg	3 deg
semispan, $b/2$	17 m	16 m	18 m
sweep, Λ_{LE}	32.25 deg	22.25 deg	42.25 deg
root chord, c_{root}	6 m	4.5 m	9 m
tip chord, c_{tip}	2 m	1.5 m	2.5 m
spar 1 root chord fraction	0.3333	0.2933	0.3733
spar 1 tip chord fraction	0.3333	0.2933	0.3733
spar 2 root chord fraction	0.6667	0.6267	0.7067
spar 2 tip chord fraction	0.6667	0.6267	0.7067
$\frac{1}{\rho}\ \log\left(\sum_{i=1}^{N} e^{\rho g_i}\right) \leq 0,\ g_i = 1 - \frac{\sigma_i}{\sigma_{\text{yield}}}$			

Besides the KS stress constraint, a trimmed flight constraint is also imposed. This accounts for the balance of forces at the start of the cruise phase (full fuel load). Eq. 7.2 represents this equality constraint:

$$\frac{L}{n(W_{other} + W_{payload} + W_{fuel} + 2W_{wing})} - 1 = 0 \tag{7.2}$$

where n represents the load factor ($n = 1$ during cruise), W_{other} is the empty weight of the aircraft excluding the wing structure, $W_{payload}$ is the total payload weight (crew/passengers/cargo), W_{fuel} is the fuel weight at the start of the cruise phase, and W_{wing} is the weight of each wing. The values used for the various weight components are: $W_{other} = 172500$ N, $W_{payload} = 120000$ N (average value, since it is considered an uncertain parameter), and $W_{fuel} = 110000$ N.

In this instance, the goal is to maximize the cruise range defined as:

$$R = \frac{V_\infty}{gSFC}\frac{C_L}{1.2C_D}log\frac{W_1}{W_2} \tag{7.3}$$

with V_∞ being the flight speed, g the gravitational acceleration, SFC the specific fuel consumption (engine dependent - the value of 2×10^{-5} is used), C_L is the lift coefficient, C_D is the total wing drag coefficient (the factor of 1.2 accounts for other, not measured, drag sources, such as fuselage and tail). W_1 and W_2 are the aircraft weight at the beginning and end of the cruise phase, respectively ($W_2 = W_1 - W_{fuel}$).

The robust objective function is then established as:

$$F(\mathbf{x}, \mathbf{r}) = 10^{-6}(-\mu_R + \sigma_R) \tag{7.4}$$

The results are displayed in Table 7.4. The deterministic optimum is obtained through the use of Kriging surrogate model approach.

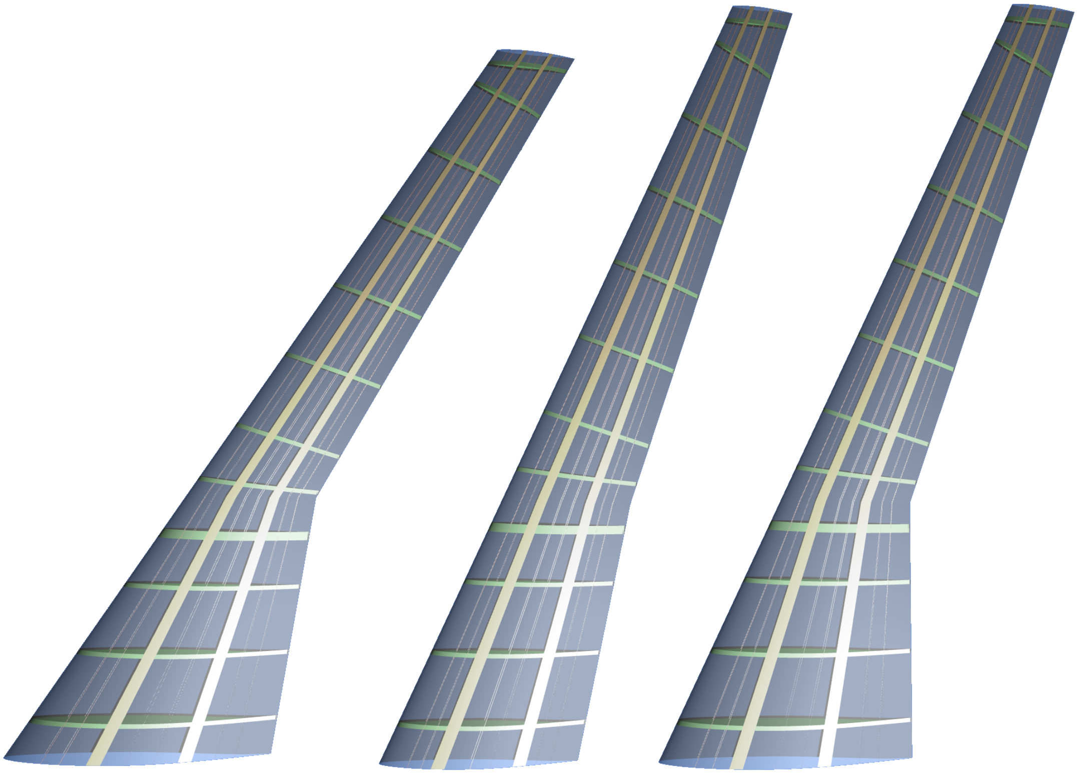


Figure 7.4: Comparison of results for Case II. From left to right: Baseline, deterministic optimization, R^2BDO (c.o.v.: AOA 5%, Altitude 2%, Payload 5%).

The values in parentheses in Table 7.4 refer to robustness analyses of the particular configurations at either the lower or higher uncertainty levels. Performing this type of analysis on the baseline configuration reveals that the average range is 5723/5737 km, for the low/high uncertainty problem. The respective standard deviations are

47 km and 114 km (note the output distributions are not normal, though). So, the optimized configurations are comparatively an improvement. A similar analysis to the deterministic optimum reveals that the expected cruise range is 6100/6115 km with standard deviations of 52/127 km (low/high uncertainty, respectively). Hence, the robust optima are not necessarily better performing than the deterministic one. Notwithstanding, this indicates that the reliability constraint is in effect (lower aspect ratios on the configurations obtained with R^2BDO), and the deterministic configuration does not satisfy the required reliability requirement.

An issue that becomes even more relevant in these higher dimensionality problems is the fact that there may exist several local minima, and as such a global solution cannot be guaranteed. Due to the random nature of the sampling scheme, it would be possible to capture different optima as was the case in Chapter 5.

Table 7.4: Results for Case Study II

Variable	Baseline	Deterministic	R^{2}BDO (AOA 2%, Alt. 1%, PL. 2%)	R^{2}BDO (AOA 5%, Alt. 2%, PL. 5%)
angle of attack, AOA	2 deg	2.03 deg	1.99 deg	1.74 deg
semispan, $b/2$	17 m	18.000 m	18.000 m	18.000 m
sweep, Λ_{LE}	32.25 deg	22.25 deg	22.25 deg	22.25 deg
root chord, c_{root}	6 m	4.5 m	4.512 m	5.799 m
tip chord, c_{tip}	2 m	1.5 m	1.593 m	1.559 m
spar 1 root chord fraction	0.3333	0.2933	0.2933	0.2998
spar 1 tip chord fraction	0.3333	0.3111	0.3702	0.2933
spar 2 root chord fraction	0.6667	0.6953	0.6347	0.6394
spar 2 tip chord fraction	0.6667	0.6306	0.6345	0.6621
F	-5.710	-6.087	-6.020	-5.841
μ_R	(5723/5737 km)	6087 km (6100/6115 km)	6071 km	5961 km
σ_R	(47/114 km)	0 km (52/127 km)	52 km	120 km
# eval.	-	178	2219	2632

Chapter 8

Conclusions

The main objective of this work was to develop an integrated framework capable of robustness and reliability based analysis and optimization for aircraft conceptual design. The framework also became a test bench for a hybrid Robust and Reliability Based Design Optimization (R^2BDO) architecture with the objective of including a combination of robustness and reliability considerations into otherwise deterministic optimization problems.

Meeting such goals required the preliminary testing of RDO and RBDO techniques using sample problems. For the computation of statistical moments to be used in robust objective/constraint functions, the Sigma Point Technique was employed as it revealed good accuracy when validated against Monte Carlo sampling. On the reliability side, PMA was favorably compared against RIA, when using Monte Carlo as a validation tool for the probability of failure and reliability index. The most important conclusion drawn in Chapter 2 was the realization that RDO does not allow a precise quantification of the probability of failure (at least not when the first two moments are used - estimating higher order moments would result in a unacceptable performance penalty for the practical application envisaged). R^2BDO, on the other hand, proved to be a good compromise in finding robust optima while satisfying precise reliability requirements.

Also, it was determined that nondeterministic optimization in an aircraft wing design environment would be too costly if direct evaluation was employed, and therefore the ability to deliver solutions within a reasonable computational cost would be impaired. In order to overcome this, a combination of two solutions was devised: the use of surrogate models augmented by the adoption of an alternative form of PMA for the reliability subproblem employing hyperspherical coordinates. Among the three surrogate model types tested using a low fidelity version of the framework, regression Kriging proved to be the most consistent approach in terms of accuracy and computational efficiency. It was therefore adopted as the surrogate model of choice in nondeterministic optimization. The combination of these measures led to significant reductions in computational effort, making the more complex wing design problems tractable.

The comparative results of RDO and R^{2}BDO in aircraft design problems show the same trend that was observed in the simpler analytical problems - RDO invariably returns results that are more optimistic than those predicted by R^{2}BDO. Much like what was observed in the Rosenbrock examples, while sharing a common origin, which converges to the deterministic optimum (zero uncertainty), with increasing uncertainty the RDO and R^{2}BDO solutions start to diverge. This is due to the enforcement of the reliability constraint on the yield stress in R^{2}BDO, which in this case turns out to be more conservative than the measure of constraint robustness used in RDO. It is noted that in these situations, validation through probabilities of failure is impracticable given the computational effort required by a single evaluation of the wing aerostructural analysis (plus MCS would require in the order of 10^6 samples surrounding each solution point).

Finally, a modular and user friendly multidisciplinary analysis and design optimization software tool has been developed and tested. The modular nature of the framework allows an easy extension to other disciplines in aircraft design optimization such as propulsion, stability and control. Topics that could be further developed include, on the framework side:

Parallelization - a surrogate model approach based on independent samples is a suitable candidate for massive parallelization. Undoubtedly, sampling is the rate limiting step in the surrogate model building process and, as such, parallel solving of the wing analysis problems in N processors would essentially translate into a speedup by a factor of N.

Multi-fidelity optimization - given that different versions exist for each of the main analyzer modules, surrogate models drawing information from each of these could be built in a multiple fidelity approach to uncertainty based design (data can be merged using, for instance, Hafka's scaling function [86]). There is also the possibility of testing other surrogate model types to achieve this (e.g. Multi Adaptive Regression Splines (MARS), high order polynomial interpolations).

Support for complete configurations - capitalizing on the modularity of the framework, efforts may be directed to allow design optimization of complete and novel aircraft configurations. This includes the parametrization of the more conventional fuselage and tail structures, or even unconventional configurations, such as Blended Wing Body and Joined Wing configurations.

Other topics of interest, on a more broad spectrum:

New measures of robustness - the possibility is open for the exploration of novel methods for assessing system robustness. For instance, using the critical complexity concept [87] which is based on a distinct philosophy from the usual statistical concept of robustness.

Merging with different MDO architectures - robustness and reliability calculations are computationally expensive, and therefore it could be desirable to decompose multidisciplinary problems using MDO architectures other than MDF. The particularities of reliability analysis add an interesting dimension to any developments proposed in this area, as reliability can be either considered at a system or subsystem (possibly disciplinary) level.

Appendices

Appendix A

Skin friction and form drag calculation

The method as described in [72], uses the *Eckert Reference Temperature* method for laminar flow and the *van Driest II formula* for turbulent flow to compute skin friction. For each of these, some assumptions are made, the most relevant of which being that $\frac{T_w}{T_{aw}} = 1$, i.e., the surface (wall) temperature is such that it is equal to the adiabatic wall temperature (hence, no heat transfer). The form factor (FF) to be used in Eq. (6.10) is taken from [88]. All air data (T, ρ) is computed from user supplied information of the flight altitude and is based on the International Standard Atmosphere (ISA) charts.

In the current approach, the wing is divided into spanwise strips. At each of those, the values of the chord ($c\left(y\right)$), maximum thickness to chord ratio ($\left(\frac{t}{c}\right)_{\max}\left(y\right)$) and chordwise location of the maximum thickness to chord ratio ($\left(\frac{x}{c}\right)_{\left(\frac{t}{c}\right)_{\max}}\left(y\right)$). A local Re is then computed:

$$Re_L = \frac{\rho V_\infty \cos\left(\Lambda_{LE}\right) c\left(y\right)}{\mu} \tag{A.1}$$

in which the dynamic viscosity (μ), is computed according to Sutherland's law:

$$\mu = 1.458 \times 10^{-6} \sqrt{\frac{T}{1 + 110.4T}} \tag{A.2}$$

Based on the supplied Re_{trans} (see footnote in Sec. 6.2.2), the chordwise location of the laminar-turbulent boundary layer transition is calculated as:

$$\frac{x_{trans}}{c} = \frac{Re_{trans}}{Re_L} \tag{A.3}$$

in case that $Re_{trans} > Re_L$ then $Re_{trans} = Re_L$.

In order to compute the laminar friction coefficient $C_{f_{lam}}$, the following values are assumed for compressible flow: $\gamma = 1.4$, $Pr = 0.72$, $r = \sqrt{Pr} = \sqrt{0.72}$ and the edge temperature, $T_e = T$. The procedure for a given value of Re is then:

$$\frac{T_w}{T_e} = \frac{T_w}{T_{aw}} \left(\frac{1 + r\,(\gamma - 1)\,M^2}{2} \right) \tag{A.4}$$

$$\frac{T^*}{T_e} = 0.5 + 0.039M^2 + 0.5\frac{T_w}{T_e} \tag{A.5}$$

$$C^* = \sqrt{\frac{T^*}{T_e}} \left(\frac{1 + \frac{T_k}{T_e}}{\frac{T^*}{T_e} + \frac{T_k}{T_e}} \right) \tag{A.6}$$

So that:

$$C_{f_{lam}} = 0.664\sqrt{\frac{C^*}{Re}} \tag{A.7}$$

For the turbulent friction coefficient: $\gamma = 1.4$, $r = 0.88$ and again $T_e = T$.

$$x_m = (\gamma - 1)\,\frac{M^2}{2} \tag{A.8}$$

$$F = \frac{T_w}{T_e} = \frac{T_w}{T_{aw}}\frac{T_{aw}}{T_e} \tag{A.9}$$

$$\frac{T_{aw}}{T_e} = 1 + r\,x_m \tag{A.10}$$

$$T_w = F\,T_e \tag{A.11}$$

$$A = \sqrt{\frac{r\,x_m}{F}} \tag{A.12}$$

$$B = \frac{1 + r\,x_m - F}{F} \tag{A.13}$$

$$\alpha = \frac{2A^2 - B}{(4A^2 + B^2)^{\frac{1}{2}}} \tag{A.14}$$

$$\beta = \frac{B}{(4A^2 + B^2)^{\frac{1}{2}}} \tag{A.15}$$

$$F_c = \begin{cases} \frac{r\,x_m}{(\csc(\alpha)+\csc(\beta))^2}, & M_\infty > 0.1 \\ \left(\frac{1+\sqrt{F}}{2}\right)^2, & M_\infty \leq 0.1 \end{cases} \tag{A.16}$$

$$F_\theta = \frac{\mu_e}{\mu_w} = \sqrt{\frac{1}{F}} \left(\frac{1 + \frac{122}{T_w} \times 10^{-\frac{5}{T_w}}}{1 + \frac{122}{T_e} \times 10^{-\frac{5}{T_e}}} \right) \tag{A.17}$$

$$F_x = \frac{F_\theta}{F_c} \tag{A.18}$$

For a specified value of Re we define:

$$\overline{Re} = F_x \; Re \tag{A.19}$$

$$\frac{0.242}{\sqrt{\overline{C_f}}} = \log_{10}\left(\overline{Re}\;\overline{C_f}\right) \tag{A.20}$$

To solve Eq. A.20, the Newton method is employed:

$$C_f\;(initial\ guess) = \frac{0.074}{\left(\overline{Re}\right)^{0.2}} \tag{A.21}$$

$$\left(\overline{C_f}\right)^{i+1} = \left(\overline{C_f}\right)^{i} \left(1 + \frac{0.242 - \sqrt{\left(\overline{C_f}\right)^{i}} \log_{10}\left(Re \left(\overline{C_f}\right)^{i} \right)}{0.121 + \frac{\sqrt{\left(\overline{C_f}\right)^{i}}}{\log(10)}} \right) \tag{A.22}$$

$$C_{f_{turb}} = \frac{\overline{C_f}}{F_c} \tag{A.23}$$

Using Eqs. A.7 and A.23 we can compute (N is the spanwise number of strips):

$$C_{D_f} = \sum_{i=1}^{N} FF \frac{S_{wet}}{S_{ref}} \left(C_{f_{turb}}\left(Re_L\right) - \frac{x_{trans}}{c}\left(C_{f_{turb}}\left(Re_{trans}\right) - C_{f_{lam}}\left(Re_{trans}\right)\right)\right) \tag{A.24}$$

Where the form factor is (for each wing/pylon strip) [88]:

$$FF = 1 + \frac{0.6}{\left(\frac{x}{c}\right)_{\left(\frac{t}{c}\right)_{\max}}} \left(\frac{t}{c}\right)_{\max} + 100 \left(\left(\frac{t}{c}\right)_{\max} \right)^4 \tag{A.25}$$

For nacelles (or other bodies), a fineness ratio is defined ($\frac{l}{d}$, ratio between length and diameter, which for nacelles requires a total streamlined body length to be extrapolated) . The formula for the form factor is then:

$$FF = 1 + 1.5 \left(\frac{d}{l}\right)^{3/2} + 50 \left(\frac{d}{l}\right)^{5} \tag{A.26}$$

And the wetted area, S_{wet} is defined as (for VLM, only. In the doublet-source lattice code, wetted area is directly available for each individual panel):

$$S_{wet} = \begin{cases} \frac{2.003\, S_{ref}}{\cos\varphi} & \left(\frac{t}{c}\right)_{\max} < 0.05 \\ \left(1.977 + 0.52\frac{t}{c}\right)\frac{S_{ref}}{\cos\varphi} & \left(\frac{t}{c}\right)_{\max} \geq 0.05 \end{cases} \tag{A.27}$$

Appendix B

Comparison with CFX

Table B.1: Comparison with CFX - EGT = 900K

	M0.7, h = 11 km, $\alpha = 0$ deg		
	CFX	***Zephyr II***	**Error**[%]
C_L	0.2965	0.2861	-3.5
C_D	0.0141	0.0183	29.8
C_M	-0.5658	-0.5655	-0.04

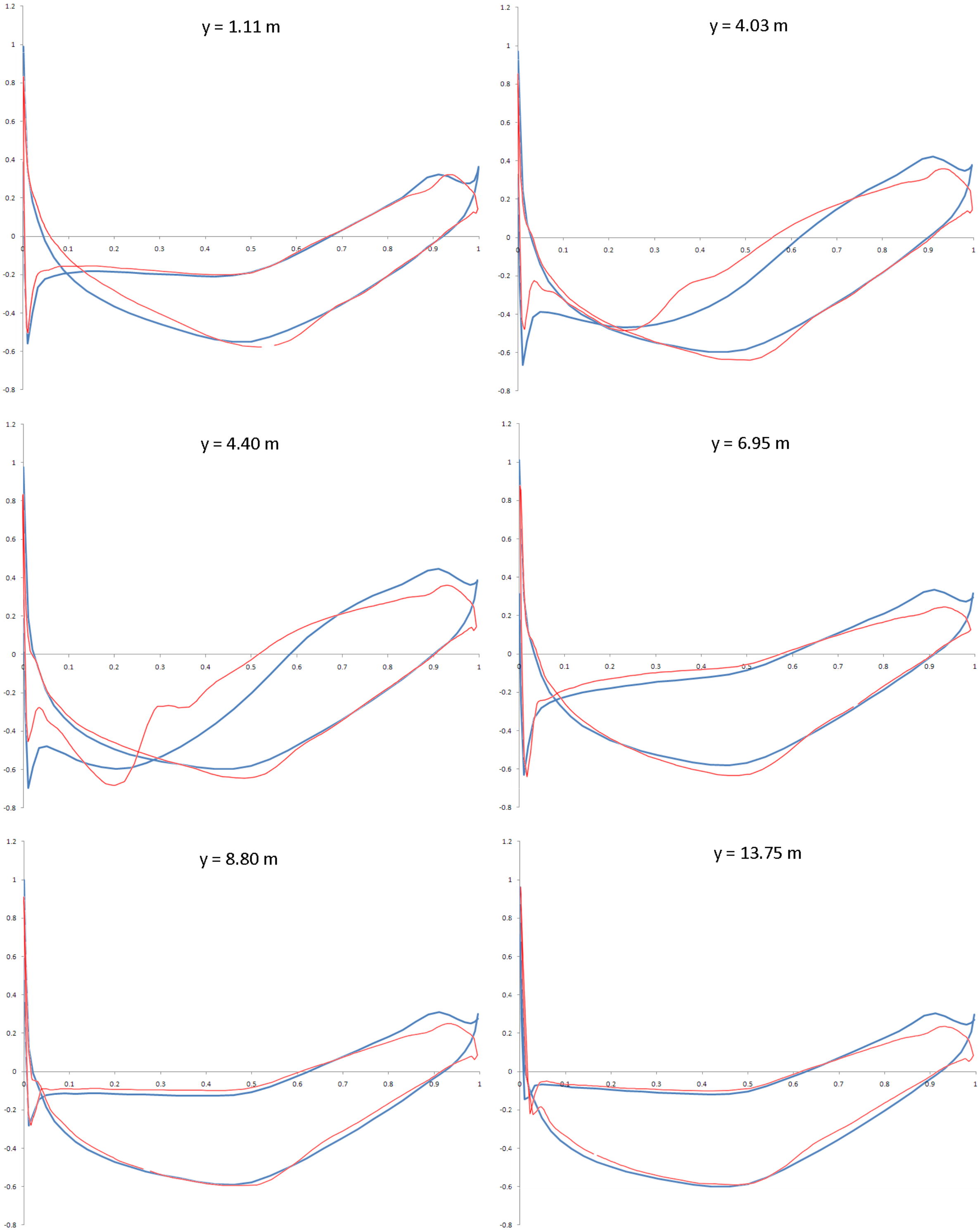
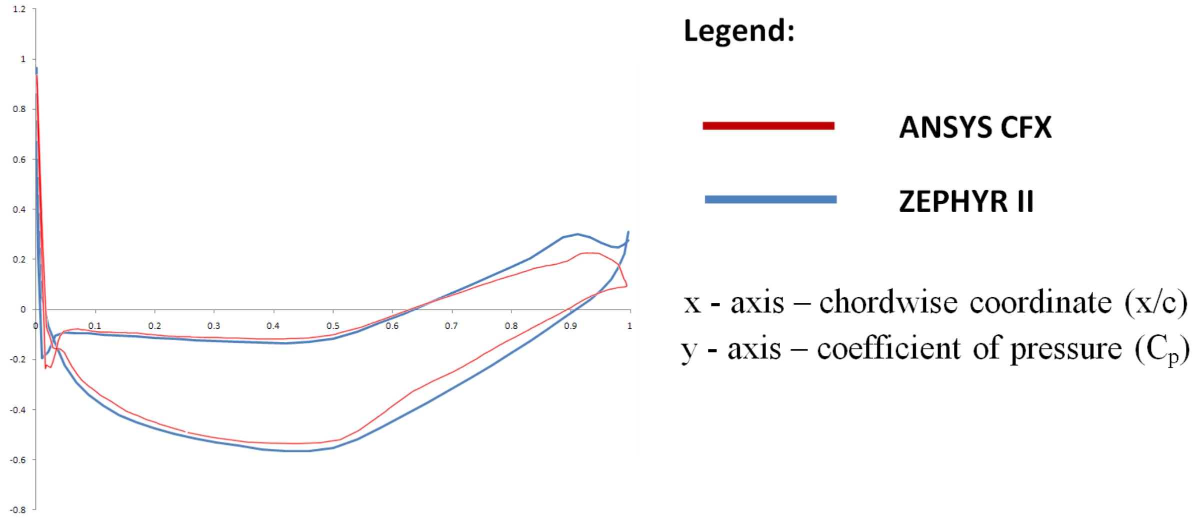


Figure B.1: Streamwise C_p distributions, M0.7, h = 11 km, $\alpha = 0$ deg, EGT = 900K



x - axis – chordwise coordinate (x/c)
y - axis – coefficient of pressure (C_p)

Figure B.2: Streamwise C_p distributions, M0.7, h = 11 km, $\alpha = 0$ deg, EGT = 900K (cont.)

Bibliography

[1] Sues, R. H., Shin, Y., and Wu, J. Y.-T., "Applications of Reliability-Based Design Optimization," *Engineering Design Reliability Handbook*, CRC Press, 2005.

[2] Liu, P.-L., Lin, H.-Z., and Kiureghian, A. D., "CALREL User Manual," Tech. Rep. UCB/SEMM-89/18, Langley Research Center, Berkeley, California, 1989.

[3] Rackwitz, R., "The Program Package STRUREL," *Structural Reliability Methods*, Chichester: John Wiley, 1996, pp. 360–367.

[4] Bjerager, P., "The Program System PROBAN," *Structural Reliability Methods*, Chichester: John Wiley, 1996, pp. 347–360.

[5] Estes, A. C. and Frangopol, D. M., "RELSYS: A Computer Program for Structural System Reliability Analysis," *Struct Engng Mech*, Vol. 6, No. 8, 1998, pp. 901 – 919.

[6] Wu, Y.-T., Shin, Y., Sues, R. H., and Cesare, M. A., "Probabilistic Function Evaluation System (ProFES) for Reliability-Based Design," *Structural Safety*, Vol. 28, 2006, pp. 164–195.

[7] Acar, E. and Haftka, R. T., "Reliability Based Aircraft Structural Design Optimization with Uncertainty about Probability Distributions," *6th World Congresses of Structural and Multidisciplinary Optimization*, Rio de Janeiro, Brazil, 2005.

[8] Ju, B. H. and Lee, B. C., "Reliability-Based Design Optimization Using a Moment Method and a Kriging Metamodel," *Engineering Optimization*, Vol. 40, No. 5, Feb. 2008, pp. 421–438.

[9] Rijpkema, J. J. M., Etman, L. F. P., and Schoofs, A. J. G., "Use of Design Sensitivity Information in Response Surface and Kriging Metamodels," *Optimization and Engineering*, Vol. 2, 2002, pp. 469 – 484.

[10] Bonte, M., van den Boogaard, A., and Huétink, J., "Deterministic and Robust Optimisation Strategies for Metal Forming Processes," *Forming Technology Forum 2007 Application of Stochastics and Optimization Methods*, Zurich, Switzerland, 2007.

[11] Giunta, A. A., "Aircraft Multidisciplinary Design Optimization Using Design of Experiments Theory and Respons Surface Modeling Methods," Tech. Rep. MAD Center Report 97–05–01, Multidisciplinary Analysis and Design Center for Advanced Vehicles, Virginia Polytechnic Institute & State University, Blacksburg, VA, May 1997.

[12] Kanasaki, M., Tanaka, K., Jeong, S., and Yamamoto, K., "Multi-objective Aerodynamic Optimization of Elements' Setting for High-lift Airfoil Using Kriging Model," *44th Aerospace Sciences Meeting and Exhibit*, AIAA, 2006.

[13] Kumano, T., Jeong, S., Obayashi, S., Ito, Y., Hatanaka, K., and Morino, H., "Multidisciplinary Design Optimization of Wing Shape for a Small Jet Using Kriging Model," *44th Aerospace Sciences Meeting and Exhibit*, AIAA, 2006.

[14] Matias, J. M., Vaamonde, A., J.Taboada, and Gonzalez-Manteiga, W., "Comparison of Kriging and Neural Networks With Application to the Exploitation of a Slate Mine," *Mathematical Geology*, Vol. 36, No. 4, 2004, pp. 463 – 486.

[15] Willmes, L., Baeck, T., Jin, Y., and Sendhoff, B., "Comparing Neural Networks and Kriging for Fitness Approximation in Evolutionary Optimization," *Proceedings of IEEE Congress on Evolutionary Computation*, 2003, pp. 663–670.

[16] "Beauty of Future Airplanes is More than Skin Deep," http://www.nasa.gov/topics/aeronautics/features/future_airplanes.html, September 2010.

[17] Reuther, J. and Jameson, A., "Control Theory Based Airfoil Design for Potential Flow and a Finite Volume Discretization," *AIAA 32nd Aerospace Sciences Meeting and Exhibit*, AIAA, Reno, Nevada, Jan. 1994.

[18] Dumas, Laurent, Herbert, Vincent, and Muyl, F., "Comparison of Global Optimization Methods for Drag Reduction in the Automotive Industry," *Computational Science and Its Applications ICCSA 2005*, edited by O. Gervasi, M. L. Gavrilova, V. Kumar, A. Lagan, H. P. Lee, Y. Mun, D. Taniar, and C. J. K. Tan, Vol. 3483 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, 2005, pp. 948–957, 10.1007/11424925-99.

[19] Chiba, K., Obayashi, S., and Morino, H., "High-Fidelity Multidisciplinary Design Optimization of Aerostructural Wing Shape for Regional Jet," *23rd AIAA Applied Aerodynamics Conference*, AIAA, Westin Harbour Castle, Toronto, Ontario, Canada, 2005.

[20] Martins, J. R. R. A., Alonso, J. J., and Reuther, J. J., "High-Fidelity Aerostructural Design Optimization of a Supersonic Business Jet," *Journal of Aircraft*, Vol. 41, No. 3, 2004, pp. 523 – 530.

[21] Gumbert, C. R., Hou, G. J.-W., and Perry, A. N., "High-Fidelity Computational Optimization for 3-D Flexible Wings: Part I - Simultaneous Aero-Structural Design Optimization (SASDO)," *Journal of Optimization and Engineering*, Vol. 6, 2005, pp. 117 – 138.

[22] Sobieszczanski-Sobieski, J., Agte, J. S., and Jr., R. R. S., "Bi-Level Integrated System Synthesis(BLISS)," Tech. Rep. NASA/TM-1998-208715, Langley Research Center, Hampton, Virginia, Aug. 1998.

[23] Sobieszczanski-Sobieski, J., Emiley, M. S., Agte, J. S., and Jr., R. R. S., "Advancement of Bi-level Integrated System Synthesis (BLISS)," *38th Aerospace Sciences Meeting and Exhibit*, AIAA, Reno, Nevada, 2000.

[24] Wakayama, S., "Blended-Wing-Body Optimization Problem Setup," *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, AIAA, Long Beach, California, 2000.

[25] Ko, Y.-Y. A., *The Multidisciplinary Design Optimization of a Distributed Propulsion Blended-Wing-Body Aircraft*, Ph.D. thesis, Virginia Polytechnic Institute, Blacksburg, Virginia, April 2003.

[26] Bhatia, M., Ajmera, H. C., Abhyankar, S. N., Mujumdar, P. M., and Sudhakar, K., "WingOpt - An MDO Tool for Concurrent Aerodynamic Shape and

Structural Sizing Optimization of Flexible Aircraft Wings," *International Conference on Modeling, Simulation, Optimization of Multidisciplinary Engineering Systems*, Goa, India, 2003.

[27] Sobester, A. and Keaney, A. J., "Multidisciplinary Design Optimization of UAV Airframes," *47^{th} AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, Newport, Rhode Island, 2006.

[28] Deremaux, Y., *Physically-Based, Real-Time Visualization and Constraint Analysis in Multidisciplinary Design Optimization*, Master's thesis, Massachussets Institute of Technology, Boston, Massachussets, June 2003.

[29] Shenoy, A., Heinkenschloss, M., and Cliff, E. M., "Airfoil Design by an All-At-Once Method," Tech. Rep. CRPC-TR-97703-S, Center for Research on Parallel Computation, Rice University, 1997.

[30] Barnewitz, H., "Flexible Wing Optimisation Based on Shapes and Structures," *MEGADESIGN and MegaOpt - German Initiatives for Aerodynamic Simulation and Optimization in Aircraft Design*, edited by N. Kroll, D. Schwamborn, K. Becker, H. Rieger, and F. Thiele, Vol. 107 of *Notes on Numerical Fluid Mechanics and Multidisciplinary Design*, Springer Berlin / Heidelberg, 2009, pp. 287–305, 10.1007/978-3-642-04093-1-20.

[31] Megson, T. H. G., *Aircraft Structures*, Edward Arnold, 2nd ed., 1990.

[32] Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., *Aeroelasticity*, Dover Publications, Mineola, New York, 2nd ed., 1996.

[33] Kapania, R. K. and Liu, Y., "Static and Vibration Analyses of General Wing Structures Using Equivalent-Plate Models," *AIAA Journal*, Vol. 38, No. 7, 2000, pp. 1269 – 1277.

[34] Giles, G. L., "Equivalent Plate Analysis of Aircraft Wing Box Structures with General Planform Geometry," *Journal of Aircraft*, Vol. 23, No. 11, 1986, pp. 859 – 864.

[35] Livne, E., "Equivalent Plate Structural Modeling for Wing Shape Optimization Including Transverse Shear," *AIAA Journal*, Vol. 32, No. 6, 1994, pp. 1278 – 1288.

[36] Thoman, D. C., *Structural Optimization of Joined-Wing Beam Model with Bend-Twist Coupling Using Equivalent Static Loads*, Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio, June 2009.

[37] Ardema, M. D., Chambers, M. C., Patron, A. P., Hahn, A. S., Miura, H., and Moore, M. D., "Analytical Fuselage and Wing Weight Estimation of Transport Aircraft," Tech. Rep. 110392, National Aeronautics and Space Administration, Ames Research Center, Moffett Field, California 94035-1000, May 1996.

[38] Raymer, D. P., *Enhancing Aircraft Conceptual Design Using Multidisciplinary Optimization*, Ph.D. thesis, Department of Aeronautics, Royal Institute of Technology, Stockholm, Sweden, 2002.

[39] Nikolaidis, E., "Types of Uncertainty in Design Decision Making," *Engineering Design Reliability Handbook*, CRC Press, 2005.

[40] Cafeo, J. A., Donndelinger, J. A., Lust, R. V., and Mourelatos, Z. P., "The Need for Nondeterministic Approaches in Automotive Design: A Business Perspective," *Engineering Design Reliability Handbook*, CRC Press, 2005.

[41] Yu, X. and Du, X., "Reliability-Based Multidisciplinary Optimization for Aircraft Wing Design," *Structure and Infrastructure Engineering*, Vol. 2, No. 3-4, 2006, pp. 277–289.

[42] Allen, M. and Maute, K., "Reliability-Based Shape Optimization of Structures Undergoing Fluid Structure Interaction Phenomena," *Computer Methods in Applied Mechanics and Engineering*, Vol. 194, No. 30-33, 2004, pp. 3472–3495.

[43] Rosenblatt, M., "Remarks on a multivariate transformation," *The Annals of Mathematical Statistics*, Vol. 23, No. 3, Sept. 1952, pp. 470–472.

[44] Kiureghian, A. D., "First and Second-Order Reliability Methods," *Engineering Design Reliability Handbook*, CRC Press, 2005.

[45] Padulo, M., Forth, S. A., and Guenov, M. D., "Robust Aircraft Conceptual Design Using Automatic Differentiation in MATLAB," *Lecture Notes in Computational Science and Engineering*, Vol. 64, 2008, pp. 271–280.

[46] Ghiocel, D. M., "Stochastic Simulation Methods for Engineering Predictions," *Engineering Design Reliability Handbook*, CRC Press, 2005.

[47] D'Errico, J. R. and Nicholas A. Zaino, J., "Statistical Tolerancing Using a Modification of Taguchi's Method," *Technometrics*, Vol. 30, No. 4, 1988, pp. 397–405.

[48] Bertuccelli, L. F. and How, J. P., "Robust Markov Decision Processes Using Sigma Point Sampling," *2008 American Control Conference*, AACC, Seattle, Washington, June 2008.

[49] Julier, S. J. and Uhlmann, J. K., "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, Vol. 92, No. 3, March 2004, pp. 401–422.

[50] de Menezes, L., Ajayi, A., Christopoulos, C., Sewell, P., and Borges, G., "Efficient computation of stochastic electromagnetic problems using unscented transforms," *IET Science, Measurement and Technology*, Vol. 2, No. 2, March 2008, pp. 88–95.

[51] Steiner, G., Watzenig, D., Magele, C., and Baumgartner, U., "Statistical Robust Design Using the Unscented Transformation," *The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, Vol. 24, No. 2, 2005, pp. 606 – 619.

[52] Padulo, M., Campobasso, M. S., and Guenov, M. D., "Comparative Analysis of Uncertainty Methods for Robust Engineering Design," *International Conference on Engineering Design*, Paris, France, Aug. 2007.

[53] Agarwal, H., Renaud, J. E., Lee, J. C., and Watson, L. T., "A Unilevel Method for Reliability Based Design Optimization," *45th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, AIAA, Palm Springs, California, 2004.

[54] Frangopol, D. M. and Maute, K., "Life-cycle Reliability-Based Optimization of Civil and Aerospace Structures," *Computers and Structures*, Vol. 81, 2003, pp. 397–410.

[55] Frangopol, D. M. and Maute, K., "Reliability-Based Optimization of Civil and Aerospace Structural Systems," *Engineering Design Reliability Handbook*, CRC Press, 2005.

[56] Cruse, T., Wu, Y.-T., Dias, B., and Rajagopal, K., "Probabilistic structural analysis methods and applications," *Computers and Structures*, Vol. 30, No. 1/2, 1988, pp. 163–170.

[57] Tu, J., Choi, K. K., and Park, Y. H., “A New Study on Reliability-Based Design Optimization,” *Journal of Mechanical Design*, Vol. 121, 1999, pp. 557–564.

[58] P.N. Koch, R.-J. Y. and Gu, L., “Design for Six Sigma Through Robust Optimization,” *Journal of Structural and Multidisciplinary Optimization*, Vol. 26, 2004, pp. 235 248.

[59] Hengl, T., “A Practical Guide to Geostatistical Mapping of Environmental Variables,” Tech. Rep. EUR 22904 EN, Joint Research Centre, Institute for Environment and Sustainability, Ispra, Italy, Sept. 2007.

[60] Huang, D., *Experimental Planning and Sequential Kriging Optimization Using Variable Fidelity Data*, Ph.D. thesis, Ohio State University, Ohio, USA, 2005.

[61] Lee, K.-H. and Park, G.-J., “A Global Robust Optimization Using Kriging Based Approximation Model,” *JSME International Journal*, Vol. 49, No. 3, 2006, pp. 779 – 788.

[62] Simpson, T. W., Mauery, T. M., Korte, J. J., and Mistree, F., “Kriging Models for Global Approximation in Simulation-Based Multidisciplinary Design Optimization,” *AIAA Journal*, Vol. 39, No. 12, 2001, pp. 2233 – 2241.

[63] Lophaven, S. N., Nielsen, H. B., and Søndergaard, J., “DACE - A MATLAB Kriging Toolbox,” Tech. Rep. IMM-TR-2002-12, Informatics and Mathematical Modelling, Technical University of Denmark, DK-2800 Kgs. Lyngby - Denmark, Aug. 2002.

[64] Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P., “Design and Analysis of Computer Experiments,” *Statistical Science*, Vol. 4, No. 4, 1989, pp. 409 – 423.

[65] McCulloch, W. S. and Pitts, W., “A Logical Calculus of the Ideas Immanent in Nervous Activity,” *Bulletin of Mathematical Biology*, Vol. 52, No. 1-2, 1990, pp. 99 – 115.

[66] Suykens, J. A. K., Vandewalle, J. P. L., and Moor, B. L. R. D., *Artificial Neural Networks for Modeling and Control of Non-linear Systems*, Kluwer Academic Publishers, 1996.

[67] Rojas, R., *Neural Networks*, Springler-Verlag, 1996.

[68] Nguyen, D. H. and Widrow, B., “Neural networks for self-learning control systems,” *International Journal of Control*, Vol. 54, No. 6, 1991, pp. 1439 – 1451.

[69] Eldred, M. S., Giunta, A. A., van Bloemen Waanders, B. G., Steven F. Wojtkiewicz, J., Hart, W. E., and Alleva, M. P., “DAKOTA, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis - Version 3.1 Users Manual,” Tech. rep., Sandia National Laboratories, Albuquerque, New Mexico, USA, April 2003.

[70] Giunta, A. A., Eldred, M. S., and Castro, J. P., “Uncertainty Quantification Using Response Surface Approximations,” *9th ASCE Specialty Conference on Probabilistic Mechanics and Structural Reliability*, Albuquerque, New Mexico, USA, 2004.

[71] Paiva, R. M., *Development of a Modular* MDO *Tool for Preliminary Wing Design*, Master’s thesis, University of Victoria, British Columbia, Canada, Dec. 2007.

[72] Mason, W. H., “Program FRICTION - Virginia Tech Aerospace Engineering Aerodynamics and Design Software Collection,” Tech. rep., Department of Aerospace and Ocean Engineering, Virginia Polytechnic Institute and State University, Jan. 2006.

[73] Chao, D. D. and van Dam, C. P., “Wing Drag Prediction and Decomposition,” *Journal of Aircraft*, Vol. 43, No. 1, 2006, pp. 82 – 90.

[74] Lawrence, C. T., Zhou, J. L., and Tits, A. L., “User’s Guide for CFSQP Version 2.5: A C Code for Solving (Large Scale) Constrained Nonlinear (Minimax) Optimization Problems, Generating Iterates Satisfying All Inequality Constraints,” Tech. rep., Electrical Engineering Department and Institute for Systems Research, University of Maryland, April 1997.

[75] Martins, J. R. R. A. and Poon, N. M. K., “On Structural Optimization Using Constraint Aggregation,” *6th World Congress on Structural and Multidisciplinary Optimization*, Rio de Janeiro, Brazil, June 2005.

[76] Saaris, G. R., "A502I User's Manual - PAN AIR Technology Program for Solving Problems of Potential Flow about Arbitrary Configurations," Tech. Rep. D6-54703, Boeing Co., 1992.

[77] Woodward, F., "USSAERO Computer Program - Development, Versions B and C," Tech. Rep. 3227, National Aeronautics and Space Administration, 1980.

[78] Maskew, B., "Program VSAERO Theory Document - A Computer Program for Calculating Aerodynamic Characteristics of Arbitrary Configurations," Tech. Rep. 4023, National Aeronautics and Space Administration, 1987.

[79] Katz, J. and Plotkin, A., *Low Speed Aerodynamics, Second Edition*, Cambridge University Press, 2001.

[80] Filković, D., "Diplomski Rad (Graduation Thesis)," Faculty of Mechanical Engineering and Naval Architecture, University of Zagreb, Croatia, 2008.

[81] Bertin, J. J., *Aerodynamics for Engineers, Fourth Edition*, Prentice Hall, 2002.

[82] Taylor, R. L., "FEAP - A Finite Element Analysis Program," Tech. rep., Department of Civil and Environmental Engineering, University of California at Berkeley, March 2008.

[83] Ferziger, J. H. and Peric, M., *Computational Methods for Fluid Dynamics, Third Edition*, Springer, 2002.

[84] Doi, H., *Fluid/Structure Coupled Aeroelastic Computations for Transonic Flows in Turbomachinery*, Ph.D. thesis, Stanford University, Stanford, California, Aug. 2002.

[85] Hodges, D. H. and Pierce, G. A., *Introduction to Structural Dynamics and Aeroelasticity*, Cambridge University Press, 2002.

[86] Gano, S. E., Renaud, J. E., Agarwal, H., and Tovar, A., "Reliability-Based Design Using Variable-Fidelity Optimization," *Structure and Infrastructure Engineering*, Vol. 2, No. 3/4, 2006, pp. 247–260.

[87] Formica, A. and Marczyk, J., "Integrated Multiscale Knowledge-Based ScienceEngineering Framework," Tech. rep., ONTONIX - Complexity Management, Como, Italy, April 2009.

[88] Corke, T. C., *Design of Aircraft*, Prentice Hall, 2002.