

Cranfield University

Jeremy Maginot

Sensitivity Analysis for Multidisciplinary Design
Optimisation

School of Engineering

PhD

Cranfield University

School Of Engineering
Department of Aerospace Engineering

PhD Thesis

Academic Year 2007-2008

Jeremy Maginot

Sensitivity Analysis for Multidisciplinary Design
Optimisation

Supervisor: Professor Marin D. Guenov

November 2007

This thesis is submitted in partial fulfilment of the requirements for the Degree of
Doctor of Philosophy

© Cranfield University, 2007. All rights reserved. No part of this publication may be
reproduced without the written permission of the copyright holder.

Abstract

When designing a complex industrial product, the designer often has to optimise simultaneously multiple conflicting criteria. Such a problem does not usually have a unique solution, but a set of non-dominated solutions known as Pareto solutions. In this context, the progress made in the development of more powerful but more computationally demanding numerical methods has led to the emergence of multi-disciplinary optimisation (MDO). However, running computationally expensive multi-objective optimisation procedures to obtain a comprehensive description of the set of Pareto solutions might not always be possible.

The aim of this research is to develop a methodology to assist the designer in the multi-objective optimisation process. As a result, an approach to enhance the understanding of the optimisation problem and to gain some insight into the set of Pareto solutions is proposed. This approach includes two main components. First, global sensitivity analysis is used prior to the optimisation procedure to identify non-significant inputs, aiming to reduce the dimensionality of the problem. Second, once a candidate Pareto solution is obtained, the local sensitivity is computed to understand the trade-offs between objectives. Exact linear and quadratic approximations of the Pareto surface have been derived in the general case and are shown to be more accurate than the ones found in literature. In addition, sufficient conditions to identify non-differentiable Pareto points have been proposed. Ultimately, this approach enables the designer to gain more knowledge about the multi-objective optimisation problem with the main concern of minimising the computational cost.

A number of test cases have been considered to evaluate the approach. These include algebraic examples, for direct analytical validation, and more representative test cases to evaluate its usefulness. In particular, an airfoil design problem has been developed and implemented to assess the approach on a typical engineering problem.

The results demonstrate the potential of the methodology to achieve a reduction of computational time by concentrating the optimisation effort on the most significant variables. The results also show that the Pareto approximations provide the designer with essential information about trade-offs at reduced computational cost.

Acknowledgements

My greatest thanks goes to my parents and my brothers who always supported me through this experience and did everything they could to help me.

All my gratitude goes to Professor Marin D. Guenov, for giving me the opportunity to work with him, for his advice and knowledge. Special thanks also go to my colleagues: Helen, Libish, Marco, Mattia, Paolo, and Yves.

I particularly want to express my gratitude to Professor Sergei Utyuzhnikov for his precious help and expertise.

My gratefulness goes also to the people from Airbus UK, Bristol whom I had the privilege to work with on the VIVACE Project.

I also wish to thank all my friends who have made the completion of this thesis possible.

Finally, a very special thanks goes to Daniela who gave me determination and supported me during the whole writing up process.

Contents

1	Introduction	1
1.1	Multidisciplinary design optimisation	1
1.2	MDO main issues	3
1.2.1	Concurrent engineering	3
1.2.2	Optimisation and decision-making.....	4
1.3	Research aim and objectives.....	6
1.4	Overview of thesis.....	6
2	Literature review	8
2.1	Sensitivity analysis	8
2.1.1	Objectives of sensitivity analysis	10
2.1.2	Sensitivity analysis procedure	10
2.1.3	Graphical methods.....	12
2.1.4	Derivatives.....	13
2.1.5	Regression analysis	14
2.1.6	Response surface methodology	15
2.1.7	ANOVA.....	18
2.1.8	Screening sensitivity analysis: Morris method.....	18
2.1.9	Global sensitivity analysis: variance-based methods	21
2.2	Multi-objective optimisation problem.....	31
2.2.1	Formulation	31
2.2.2	Pareto concept.....	32
2.2.3	Obtaining Pareto solutions	32
2.3	Pareto frontier exploration and sensitivity analysis for decision making.....	34
2.3.1	Obtaining the sensitivity of Pareto solutions.....	34
2.3.2	Articulation of preferences	35
2.3.3	Understanding local trade-off with Pareto approximation	39
2.4	Summary.....	43

3	A sensitivity analysis approach for an efficient multi-objective optimisation	44
3.1	Reducing the dimensionality of the optimisation problem.....	44
3.1.1	Identifying non-significant inputs	45
3.1.2	Obtaining the sensitivity analysis indices.....	46
3.1.3	Freezing non-significant variables to a specific value.....	47
3.2	Pareto sensitivity analysis.....	52
3.2.1	Understanding the trade-offs on the Pareto surface	52
3.2.2	Requirements for sensitivity analysis on the Pareto surface	52
3.3	Summary.....	53
4	Obtaining local sensitivity on the Pareto surface	54
4.1	Multi-objective optimisation problem formulation and assumptions	55
4.2	Discussion on an existing technique.....	57
4.2.1	Obtaining the sensitivity coefficients on the Pareto surface.....	57
4.2.2	Deriving a linear approximation from the sensitivity coefficients	61
4.2.3	Deriving a quadratic approximation from the sensitivity coefficients ...	63
4.2.4	Limitations.....	67
4.3	New approximations of the Pareto surface.....	69
4.3.1	Derivation of linear and quadratic approximations in the general case .	69
4.3.2	Local trade-off study	76
4.3.3	Extent of validity of the local Pareto approximations	77
4.4	Identification of non-differentiable Pareto points	78
4.4.1	Non-differentiability concept on the Pareto surface.....	79
4.4.2	Derivation of a sufficient condition.....	79
4.4.3	Use of the sufficient condition for non-differentiability	87
4.5	Procedure for local Pareto sensitivity analysis	88
4.6	Limitations of the Pareto sensitivity analysis.....	90
4.7	Numerical examples	91
4.7.1	Example 1: Non linear three-objective optimisation.....	91
4.7.2	Example 2: Non-differentiable Pareto point when design and criterion spaces coincide	93

4.7.3	Example 3: Non-differentiable Pareto point when design and criterion space are different.....	96
4.8	Conclusions	99
5	Beam optimisation problem.....	101
5.1	Description	101
5.2	Pareto surface	103
5.3	Local Pareto sensitivity analysis	104
5.4	Conclusions	114
6	Airfoil design optimisation.....	115
6.1	Description of the problem and model implementation	115
6.1.1	PARSEC-11 parameterization.....	116
6.1.2	VGK	118
6.1.3	Implementation for optimisation	120
6.2	Optimisation strategy.....	125
6.2.1	Initial range of variation and PARSEC-11 issues	125
6.2.2	Definition of a suitable range of variation for input variables	129
6.2.3	Kriging adaptive sequential training	131
6.3	Formulation of optimisation problem.....	135
6.4	Reducing the dimensionality of the problem.....	136
6.4.1	Identifying non-significant inputs using VBM indices	136
6.4.2	Formulation of the reduced optimisation problem	139
6.4.3	Choosing values for non-significant variable using SOM.....	139
6.4.4	Obtaining the different Pareto fronts.....	141
6.5	Local Pareto sensitivity analysis	142
6.6	Conclusions	150
7	Summary and conclusions	151
7.1	Reducing the dimensionality of the design problem	152
7.2	Pareto sensitivity analysis.....	153
7.2.1	Investigation into available methods.....	153

7.2.2	Derivation of new local Pareto approximations	153
7.2.3	Development of a new method for identifying non-differentiable Pareto points	154
7.2.4	Limitations of the proposed approach	155
7.3	Test results	155
7.4	Further work	156
Appendix A.....		157
Appendix B.....		158
Appendix C.....		159
Appendix D.....		161
References		166
Bibliography.....		176

List of figures

Figure 1.1: Coupling between aerodynamic and structure	2
Figure 1.2: Multidisciplinary design optimisation issues	3
Figure 2.1: Representation of a computer model: the black box.....	9
Figure 2.2: Typical sensitivity analysis procedure	11
Figure 2.3: Representation of the p-level grid for p=5	19
Figure 2.4: Typical output variance decomposition	24
Figure 2.5: Representation of information vs cost for model-independent sensitivity analysis	31
Figure 3.1: Pareto fronts obtained with full optimisation and reduced problem for $x_3 =$ 0.5	49
Figure 3.2: Self-organizing map for test optimisation problem	50
Figure 3.3: Pareto fronts obtained with full optimisation, reduced problem for $x_3 = 0.5$ and $x_3 = 0.75$	50
Figure 3.4: Global sensitivity analysis approach flowchart	51
Figure 4.1: Gradient projection method	57
Figure 4.2: Unit-sphere optimisation problem – Erroneous linear approximation.....	61
Figure 4.3: Unit-sphere optimisation problem – Linear approximation derived from tangent directions.....	62
Figure 4.4: Unit-sphere optimisation problem – Parametric curves.....	64
Figure 4.5: Unit-sphere optimisation problem – Quadratic approximation derived from parametric curves.....	66
Figure 4.6: Modified unit-sphere optimisation problem – Limitations of the method...	68
Figure 4.7: Unit-sphere optimisation problem – New linear approximation	74
Figure 4.8: Unit-sphere optimisation problem – New quadratic approximation.....	74
Figure 4.9: Unit-sphere optimisation problem – Relative error in predicting f_3	74
Figure 4.10: Non-differentiable Pareto point concept.....	79
Figure 4.11: Pareto analysis flowchart	89
Figure 4.12: Global and local Pareto fronts.....	90
Figure 4.13: Example 1 – Non linear three objectives problem – linear approximation	92

Figure 4.14: Example 1 – Non linear three objectives problem – Quadratic approximation	92
Figure 4.15: Example 1 – Non linear three objectives problem – Relative error in predicting f_3	92
Figure 4.16: Example 2 – Relative position of the tangent cone and its polar at a non-differentiable Pareto point	93
Figure 4.17: Example 2 – Existence of vector $\nabla\Psi$	94
Figure 4.18: Example 3 – Design space / Pareto set	96
Figure 4.19: Example 3 – Objective space / Pareto set	97
Figure 4.20: Example 3 – Relative position of the tangent cone and its polar at a non-differentiable Pareto point	97
Figure 4.21: Example 3 – Existence of vector $\nabla\Psi$	98
Figure 5.1: The I-beam design problem	102
Figure 5.2: Pareto front for the I-beam design problem	104
Figure 5.3: Candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$) in criterion space	105
Figure 5.4: Linear and quadratic approximation at candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$)	106
Figure 5.5: Extent of validity of approximation at candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$)	108
Figure 5.6: Beams associated with the limits of validity of the approximation at candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$)	108
Figure 5.7: Extent of validity of approximation at candidate Pareto solution ($f_1 = 128.718 \text{ cm}^2, f_2 = 0.0472 \text{ cm}$)	112
Figure 5.8: Beams associated with the limits of validity of the approximation at candidate Pareto solution ($f_1 = 128.718 \text{ cm}^2, f_2 = 0.0472 \text{ cm}$)	113
Figure 6.1. PARSEC-11 airfoil parameterization	116
Figure 6.2: Overall configuration of the model	121
Figure 6.3: Postprocessing checks flowchart	124
Figure 6.4: PARSEC-11 erroneous parameterization – Inconsistent maximum thickness location	127

Figure 6.5: PARSEC-11 erroneous parameterization – Intersection between upper and lower surfaces	128
Figure 6.6: SOM of erroneous PARSEC-11 profiles obtained on initial input range..	130
Figure 6.7: Adaptive sequential strategy for Kriging model training.....	133
Figure 6.8. Error history of Kriging model for C_d during adaptive training for different regression models	134
Figure 6.9. Error history of Kriging model for C_m during adaptive training for different regression models	134
Figure 6.10. Area of inner trapezoid box defined by front and rear spar	135
Figure 6.11: Variance-based sensitivity analysis – Main and interaction effect for C_d and C_m	138
Figure 6.12: Variance-based sensitivity analysis – Main and interaction effect for area, $t_{12\%}$ and $t_{60\%}$	138
Figure 6.13: SOM with feasible regions for Δz_{te} , β_{te} and all outputs	140
Figure 6.14: Pareto surface obtained for the two-objective airfoil optimisation problem	142
Figure 6.15: A few characteristic solutions in the criterion space.....	143
Figure 6.16: A few characteristic Pareto solution airfoils.....	143
Figure 6.17: Linear approximation at candidate Pareto solution ($C_d = 0.00807$, area = 0.04339).....	145
Figure 6.18: Airfoils associated with the limits of validity of the approximation at candidate Pareto solution ($C_d = 0.00807$, area = 0.04339).....	146
Figure 6.19: Linear approximation at candidate Pareto solution ($C_d = 0.00837$, area = 0.04512).....	148
Figure 6.20: Airfoils associated with the limits of validity of the approximation at candidate Pareto solution ($C_d = 0.00837$, area = 0.04512).....	149

List of tables

Table 2.1: Number of model evaluation for model-independent sensitivity analysis methods.....	30
Table 3.1: Main and total sensitivity indices of objectives f_1 and f_2 and constraint g for test example	48
Table 3.2: Computational cost comparison between full optimisation and reduced problem for $x_3 = 0.5$ and $x_3 = 0.75$	51
Table 5.1: Candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$)	104
Table 5.2: First and second order derivative on the Pareto surface at candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$)	105
Table 5.3: Inputs, objectives and constraint values of candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$) and approximate Pareto solutions associated with the limits of validity of the approximation	107
Table 5.4: First and second order derivative on the Pareto surface for strength constraint at candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$).....	109
Table 5.5: Candidate Pareto solution ($f_1 = 128.718 \text{ cm}^2, f_2 = 0.0472 \text{ cm}$)	110
Table 5.6: First and second order derivative on the Pareto surface at candidate Pareto solution ($f_1 = 128.718 \text{ cm}^2, f_2 = 0.0472 \text{ cm}$)	111
Table 5.7: Inputs, objectives and constraints values of candidate Pareto solution ($f_1 = 128.718 \text{ cm}^2, f_2 = 0.0472 \text{ cm}$) and approximate Pareto solutions associated with the limits of validity of the approximation	112
Table 6.1: Description of model inputs	116
Table 6.2: VGK settings for flow calculation about an airfoil	122
Table 6.3: Initial input range of variation.....	125
Table 6.4: Example of PARSEC-11 input values giving an airfoil with inconsistent maximum thickness location	127
Table 6.5: Example of PARSEC-11 input values giving an airfoil where upper and lower surface intersect	128
Table 6.6: Reduced input range of variation	131
Table 6.7: Max error and MSE of Kriging models with different regression functions after adaptive training.....	135

Table 6.8: Description of model outputs	136
Table 6.9: First order sensitivity indices for C_d , C_m , area, $t_{12\%}$ and $t_{60\%}$	137
Table 6.10: Total order sensitivity indices for C_d , C_m , area, $t_{12\%}$ and $t_{60\%}$	137
Table 6.11: Performance comparison between different three optimisation runs.....	141
Table 6.12: First and second order derivative on the Pareto surface at candidate Pareto solution ($C_d = 0.00807$, area = 0.04339).....	145
Table 6.13: Inputs, objectives and constraints values of candidate Pareto solution ($C_d = 0.00807$, area = 0.04339) and approximate Pareto solutions associated with the limits of validity of the approximation	147
Table 6.14: First and second order derivative on the Pareto surface at candidate Pareto solution ($C_d = 0.00837$, area = 0.04512).....	147
Table 6.15: Inputs, objectives and constraints values of candidate Pareto solution ($C_d = 0.00837$, area = 0.04512) and approximate Pareto solutions associated with the limits of validity of the approximation	149

Notations

C	Polyhedral cone of objective gradients
$E(X)$	Expectation of random variable X
\mathbf{f}	Objective function vector
f_i	Objective function
\mathbf{G}	Active constraint vector
\mathbf{g}	Inequality constraint vector
g_i	Inequality constraint
h_i	Equality constraint
I	Number of active constraints
\mathbf{J}	Matrix of linearly independent gradients of active constraints
K	Constraint cone
L	Number of constraints
M	Number of output/objective functions
N	Number of input/design variables
\mathbf{n}_f	Local normal vector to Pareto surface
n_f	Number of linearly independent columns of $\mathbf{P}\nabla\mathbf{f}$
\mathbf{P}	Projection matrix
Q	Number of active inequality constraints
\mathbf{S}	Descent vector
S_i^I	First order sensitivity index of input variable x_i
S_i^T	Total sensitivity index of input variable x_i
V_I	Subspace generated by the gradients of active constraints
$V(X)$	Variance of random variable X
X	Random variable
\mathbf{X}	Design space
\mathbf{x}	Design vector

\mathbf{x}^*	Point under study
\mathbf{x}_{red}	Reduced design vector of significant input variables
x_i	Design variable
Y	Random variable
\mathbf{Y}	Objective space
\mathbf{y}	Objective vector
y_i	Objective
$\boldsymbol{\lambda}$	Lagrange multiplier vector
μ	Mean
Φ	Aggregate function
Ψ	Test function
σ	Standard deviation

Abbreviations

DACE	Design and analysis of computer experiments
DOE	Design of experiments
FAST	Fourier amplitude sensitivity test
GSA	Global sensitivity analysis
MDO	Multidisciplinary design optimisation
MSE	Mean square error
OAT	One-at-a-time sensitivity analysis
RS	Response surface
RSM	Response surface methodology
SA	Sensitivity analysis
SOM	Self-organizing map
VBM	Variance-based method

Chapter 1

Introduction

Development of complex systems is the result of a large set of considerations on a variety of engineering disciplines. During this process, the designer searches for combinations of values for the parameters describing the system that will improve its characteristics and satisfy a set of predefined requirements. Typically, one attempts to minimise cost and/or maximise performance levels subject to engineering or regulations constraints. Over the past few decades, designers have spent a considerable effort to integrate design techniques from different disciplines. The integration is motivated by the idea that better designs can be achieved through concurrent engineering and the commercial imperative of reducing design time and cost (Bartholomew, 1998). This trend has been accompanied by the emergence of a new field in the area of complex system design: Multidisciplinary Design Optimisation (MDO).

1.1 Multidisciplinary design optimisation

In the aerospace industry, designing an aircraft requires taking into account contributions of different disciplines, which might interact and exhibit strong coupling. For an aircraft, aerodynamics and structure are intimately coupled. The pressure

distribution over the wing results in a deformation of geometry, which in return affects the aerodynamics. This interaction is illustrated in Figure 1.1. Other common interactions that might be involved in the design of an aircraft include flight dynamics, payload distribution, fuel capacity and so forth.

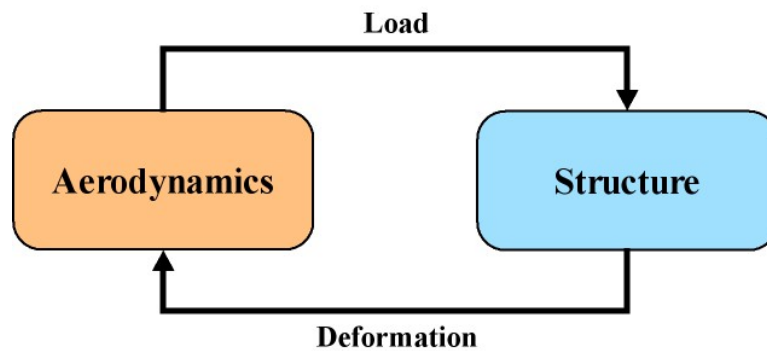


Figure 1.1: Coupling between aerodynamic and structure

In the traditional way of designing a complex system, each team of experts optimises its specific discipline and forwards the resulting design to the team in charge of the next analysis. In practice, since some analysis tools might require results from others, a few iterations of the optimisation procedure have to be completed to achieve a final consistent design between all disciplines. Such design process has the main disadvantage that each discipline analysis is carried out one after the other. Therefore, when several disciplines with possible interactions are taken into account into the design, each team of experts has to wait for the results of the previous step before they can carry out their optimisation of the system and propose further modifications. This approach can result in a very long design procedure without the guarantee that a final consistent design is obtained. Such practice has been extensively used for designing complex systems.

The MDO concept has been present in designers mind for the past decades (Sobieszczanski-Sobieski *et al.*, 1972, 1984; Fulton *et al.*, 1974; Kroo *et al.*, 1994). However, its implementation in the industry was not possible because of enormous computational effort required and limited communication capabilities between numerical tools. Nowadays, together with the development of computer power, high-fidelity numerical methods, parallel programming and integration frameworks (Scott,

2001), MDO has become a reality and attracts growing interest in the aerospace industry (Bowcutt, 2003; Kim *et al.*, 2002; Walsh, 2000, Giunta *et al.*, 1996).

MDO can be described as an environment, which integrates the knowledge and creativity of human designer with the power and capability of mathematics and computers in making design decision. The MDO concept is achieved through a set of methods, which provides a means of coordinating efforts and performing the optimisation of the design.

1.2 MDO main issues

1.2.1 Concurrent engineering

The recent developments in MDO have come together with new challenges for the designer as shown in Figure 1.2.

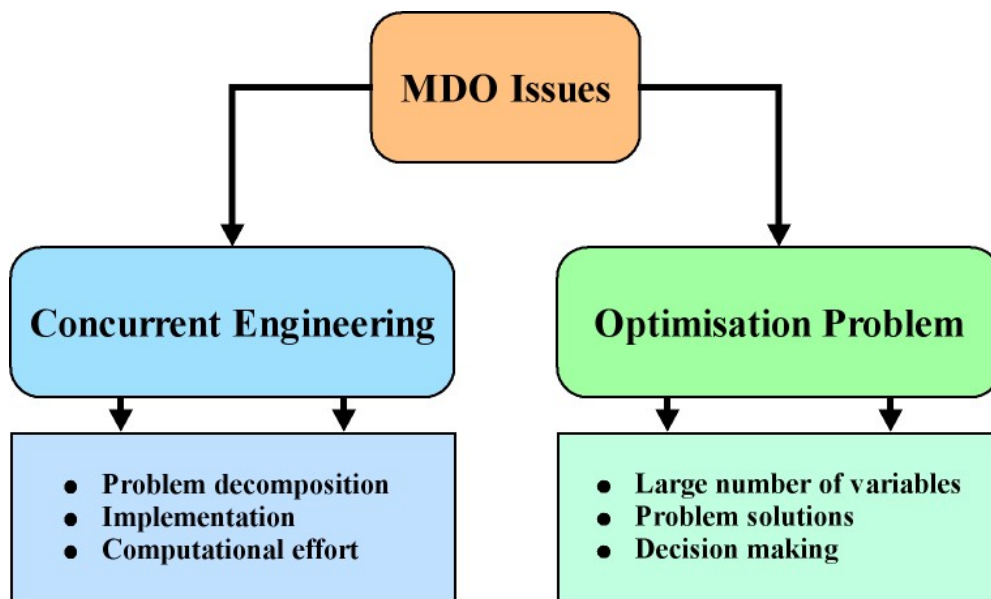


Figure 1.2: Multidisciplinary design optimisation issues

The first main challenge is intrinsically related to the multidisciplinary aspect of MDO and the way the interactions and coupling between disciplines are handled. The implementation of concurrent engineering in the design optimisation procedure has required the development of integration frameworks to facilitate the communication between the different numerical tools called during the discipline analysis. They allow

the connection of existing engineering software and pieces of code which can be executed separately or in parallel from different locations. A lot of effort has also been spent on improving existing numerical tools and developing new capabilities for MDO. An essential component of an MDO process is the product model generator, the role of which is to provide each analysis tool with the appropriate information about the system to compute the various discipline outputs. In the case of an aircraft, a parametric geometry generator provides the information (external surfaces, structure layout...) required by each tool (CFD, FEA...) at each step of the optimisation process (La Rocca *et al.*, 2002). Another issue related to the integration of various disciplines in the design process concerns the decomposition and arrangement of the problem into contributing analyses, which perform single tasks. This aspect of MDO is particularly important when there exists a strong coupling between the disciplines involved in the design. In such situation, there is no hierarchical organisation of the disciplines and the contributing analyses cannot be run sequentially. An iterative procedure must be used to achieve a consistent design. Researchers have come up with a range of algorithms to handle this complex situation and ensure that all disciplines are consistent for the final optimum design (Balling and Sobieszczanski-Sobieski, 1996). The deployment of MDO capabilities making use of high fidelity tools, complex problem decomposition and large number of design variables implies large computational times.

1.2.2 Optimisation and decision-making

The second main challenge of MDO concerns the design optimisation and decision-making. Beside, the computational burden associated with the integration and implementation of an MDO procedure, there is an inherent difficulty to achieve a suitable solution according to the designer formulation.

In the context of MDO, the design process of a complex system frequently results in optimising multiple criteria for a large number of design variables. In the context of the design of an aircraft, a designer might want to maximise its range while minimising its life-cycle cost. The number of parameters required to describe an aircraft depends on the level of maturity of the design and can vary from tens to a few thousand. Such large number makes the exploration of the design space difficult. Choosing appropriate parameters to describe the system to be optimised is not straightforward. It

might depend on the level of accuracy required to describe the system, on the computational effort that can be afforded or previous design experience. On the one hand, if design variables with non-significant impact on the results of the optimisation are considered, computational effort is unnecessarily spent to set the value of these parameters. On the other hand, if only a reduced set of input variables is considered, it is possible that some parameters that can improve the results of the optimisation are omitted. The complex internal mechanisms of each analysis together with the strong interactions between disciplines make the choice for appropriate parameters to the optimisation a difficult task.

In the multi-objective optimisation formulation of the MDO problem, the criteria involved are often conflicting and cannot be improved together. There is no single optimum solution to such a problem but a set of non-dominated solutions also referred as Pareto solutions (Miettinen, 1999). Each Pareto solution represents an alternative feasible design for a particular trade-off between performance criteria considered in the optimisation formulation. They all satisfy the requirement expressed in the multi-objective optimisation problem and none is better with respect to all criteria. Therefore, additional preferences on the solutions that are not taken into account in the initial optimisation problem formulation must be articulated to help the designer select the most suitable solution. A common approach is to define preferences representing the relative importance the designer assigns to each objective before performing the optimisation. These preferences correspond to a specific trade-off between the objectives and are included in the optimisation formulation. For each set of preferences, a different Pareto solution is obtained. This method is referred as *a priori articulation of preferences* (Marler and Arora, 2004). The main disadvantage associated with this method is the difficulty for the designer to postulate reasonable preferences without prior knowledge on the possible solutions of the problem. An alternative method is to generate Pareto solutions corresponding to various combinations of trade-off between objectives and choose among the available solutions. Such method is referred as *Generate First – Choose Later* (GFCL) (Balling and Richard, 2000) or *a posteriori articulation of preferences* (Marler and Arora, 2004). This method seems more attractive since no prior knowledge of the design problem is required and the designer can make use of the solutions available to define a set of preferences. Moreover, if no

solution from the reduced set of Pareto points suits the designer's preferences, he/she can decide to carry out further exploration of a particular part of the Pareto front and its corresponding region in the design space. Such iterative procedure can help him/her refine the set of preferences for choosing the final trade-off solution.

1.3 Research aim and objectives

The aim of this research is to propose an approach to improve the efficiency of the optimisation and decision-making aspects in multidisciplinary design. This involves providing the designer with appropriate information about the problem. In this thesis, two aspects are considered.

The first aspect is to provide the designer a means to understand prior to the optimisation which inputs will have the most important impact on the final solution. Only significant input variables are retained to reduce the computational effort and make the optimisation procedure more efficient.

The second aspect of the method concerns the multi-objective optimisation and occurs after Pareto solutions are obtained. The objective is to provide the designer with a means to obtain a better insight into the set of solutions the problem. For complex designs, the Pareto surface cannot be obtained analytically and its description is often limited to a reduced number of points. It is therefore essential for the designer to make the most out of any solution obtained without having to run more optimisation procedures. Additional valuable information about the solution set would enhance the understanding of the problem and benefit the designer.

Each aspect of the approach corresponds to a sensitivity analysis (SA) and provides the designer a better insight into the problem, at two different steps of the MDO procedure.

1.4 Overview of thesis

In chapter 2, a literature review of relevant SA techniques is presented. It includes a description of SA methods used on a model to understand relationships between its input and output variables. Chapter 2 also introduces the multi-objective optimisation problem together with the Pareto concept and briefly describes some of the methods used in the field to generate solutions. Finally, it describes various techniques

developed to approximate the set of Pareto solutions and understand the trade-offs between objectives. In chapter 3, the approach proposed to assist the decision-maker in the optimisation process is presented. In chapter 4, a new method for approximating the Pareto set and understanding local trade-offs in the criterion space is derived. This technique also provides a means to relate the new approximate solution to the design space without carrying out more optimisation runs. In chapter 5, the new method for approximating the Pareto surface is used the optimisation of an I-beam. The whole approach described in chapter 3 is comprehensively tested in chapter 6 in the context of an airfoil optimisation for minimum drag and maximum wing volume. Finally, in chapter 7, a summary of the research is given and conclusions are drawn.

Chapter 2

Literature review

Over the last few decades, a large number of methods for sensitivity analysis have been developed to obtain a better insight in their models and facilitate the decision-making process in the science and engineering field. The choice of a sensitivity analysis method to adopt is not easy to make and must be based on the problem the designer is trying to address, on the characteristics of the model under study and on the computational cost that can be afforded. In this chapter, the most relevant sensitivity analysis techniques are reviewed. Then, the multi-objective optimisation formulation together with the Pareto concept are introduced and methods for generating solutions of such a problem are briefly outlined. Finally, attention is focused on methods and techniques developed to help the designer obtain sensitivity of the Pareto solution and understand the trade-offs as part of the decision-making process.

2.1 Sensitivity analysis

Sensitivity analysis (SA) is the study of how the outputs of a model can be apportioned, qualitatively or quantitatively, to the different inputs.

Models have been developed in every area to approximate or mimic systems and processes of different nature and different complexity. Experimental results are often expensive, time consuming or impossible to obtain. As a result, investigation of the

process or the system is to be carried out through a mathematical or computational model. Depending on the complexity and level of development, these models enable results to be obtained in a reduced amount of time. Despite that time and cost to obtain these results have been considerably condensed, the modelling of complex systems still remains time consuming. Therefore, performing intensive model simulation to explore systems or processes over the whole range of variation of input parameters is not always a suitable strategy.

Any complex model can be represented as a black box with input parameters coming in and output results going out as represented in Figure 2.1.

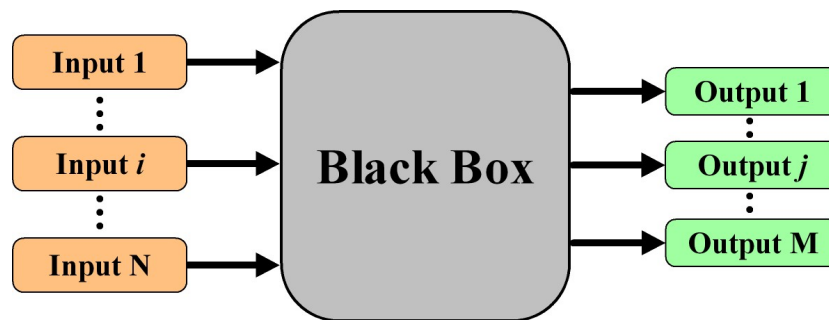


Figure 2.1: Representation of a computer model: the black box

A mathematical model is defined as a set of equations, inputs and outputs aimed to describe the process being investigated. In this thesis, the terms *factors*, *parameters* and *variables* are all used to refer to the *inputs* of a model. In the same way, the term *response* refers to a model *output*. Absence of information or poor understanding of the internal mechanisms inside the model has led the modellers to develop methods to understand how inputs and outputs are related. A definition of sensitivity analysis is given in Saltelli et al. (2000a):

“Sensitivity analysis studies the relationship between information flowing in and out a model.”

Originally, sensitivity analysis was developed to deal simply with uncertainties in the input variables and model parameters but it has been extended by providing an

understanding of how the model response variables respond to changes in the inputs (Saltelli et al., 2000a).

2.1.1 Objectives of sensitivity analysis

Use of sensitivity analysis is made in many different fields for different purposes. In engineering, SA is mainly used for design reliability and robustness, in chemistry and the social sciences, for model validation and confirmation, in economics and risk assessment to determine the most influential parameters. Sensitivity analysis methods are based on different mathematical or statistical tools, depending on the problem and the information they provide. These different types of analyses share a common aim to investigate how a given model responds to variations in its inputs. A review of the main applications of sensitivity analysis is given by Saltelli et al. (2004):

- *Model validation*
- *Robustness of the model results*
- *Identification of most influential input parameters* to focus attention on a small set of inputs
- *Identification of non-significant input parameters* which can be eliminated from the final model to reduce the problem dimension
- *Determination of optimal regions* where specific values for output can be achieved
- *Identification of interactions between subsets of factors.* This is particularly relevant when factors have effects which cannot be reduced to the sum of the individual ones

2.1.2 Sensitivity analysis procedure

Information on models used to describe a complex system cannot be obtained analytically because the internal mechanisms are not known. This is particularly true in MDO as complicated coupling between disciplines must be taken into account. In such a situation, only numerical results can be obtained. A way to get a better understanding of the model is to perform a sample-based sensitivity analysis.

A sample-based sensitivity analysis is a procedure in which the model is executed repeatedly for a particular set of input values. Such a set of input values is also referred as design of experiments (DOE) and corresponds to a sampling of input space. A variety of DOE strategies has been developed to address different problems: input interactions studies, polynomial curve fitting, input space exploration. Many references on design of experiments can be found in the literature (Mc Kay *et al.*, 1979; Myers and Montgomery, 1995; Kleijnen, 2004a, 2004b). As described in Saltelli *et al.* (2000a) and illustrated in Figure 2.3, the following steps can be identified in a sample-based sensitivity analysis:

- *Design the experiment*: the question which the sensitivity analysis should answer is defined and an appropriate strategy for sampling the input factors concerned with the analysis is decided
- *Assign the probability distribution or range of variations to each input factor*
- *Generate a sample of input factors*
- *Execute the model to obtain the output for each input factor generated*
- *Perform sensitivity analysis to obtain the relevant information about the model*

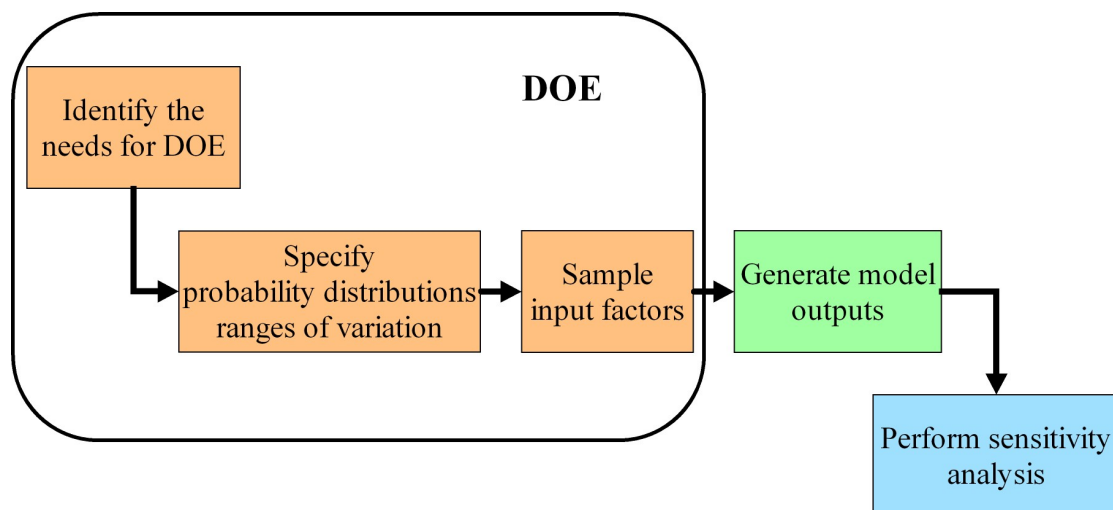


Figure 2.2: Typical sensitivity analysis procedure

Note that when partial derivatives are used as sensitivity analysis methods for complex systems, they are often evaluated numerically by finite differences. In this case, a uniform distribution is assumed over the small interval selected for the evaluation.

Another important point is that when sample-based methods are used, both one-at-a-time (OAT) sensitivity analysis and multivariate sensitivity analysis can be performed. With sample-based techniques, input factors can be varied together to study interaction between sets of variables and their effect on the outcome of the model.

When implementation of sensitivity analysis is concerned, many different approaches can be followed and a large number of techniques is available, which makes it difficult to find out the appropriate method.

2.1.3 Graphical methods

Graphical methods are the most intuitive and simple techniques for sensitivity analysis (Saltelli et al., 2000a). They are used to identify visually the influence of inputs on a particular output. They may reveal relationship between model inputs and outputs such as non-linear relationship and thresholds, which could be used as prior knowledge of the model to decide which sensitivity analysis to apply for further investigations.

Graphical methods are model-independent and can be considered as global measures of importance because the input vector can be varied over the whole range of variation. A key advantage of these methods is that they allow for the identification of potentially complex dependencies. An understanding of the nature of the dependencies between inputs and outputs can guide the selection of other appropriate sensitivity analysis methods. Popular graphical methods for sensitivity include scatter plots, radar graphs, cobwebs also known as parallel coordinates plots, (Saltelli, et al., 2000a; Holden, 2004; Holden and Keane, 2004). Scatter plots enable to identify correlations between input and output variables. They can also be arranged in a matrix form to enhance the comparison between all variables. Radar graphs and cobwebs share the same principle: the variables are represented on different coordinate axes. In the former, the radial axes are arranged evenly while in the latter the axes are parallel. These graphical methods are used to identify groups of design vectors or particular behaviour in the data sample. Permutation of axes can sometime facilitate the visualization. These

basic graphical methods are often used as a first step in other sensitivity analyses such as regression analysis and response surface methods. A clear disadvantage of graphical methods is that they become inconvenient for large number of inputs and large data samples.

Another interesting technique for understanding relationship between inputs and outputs is the Self-organizing map (SOM) developed by Kohonen (2000). SOM is an efficient technique for visualizing multidimensional data. The SOM consists of neurons positioned on a low dimensional regular grid and described by an N -dimensional prototype vector in the high dimensional space. These neurons are trained on a set of data and their prototype vectors are iteratively updated to best describe the set of data. In the high dimensional space, the SOM can be seen as a net that unfolds onto the training set. The nodes of the SOM in the high dimensional space correspond to a fixed map unit in the component plane. That is, the SOM allows to project a high dimensional space onto a two dimensional components map. In the component plane, the position of the map units is arbitrary but the relative position between these maps is meaningful. This representation allows the visualisation of many inputs at once and to get an insight into their relationships (Holden, 2004; Holden and Keane, 2004). SOM can also be used for classifying large amount of data and identify possible clusters and similarities between data points (Vesanto, 1999).

2.1.4 Derivatives

In MDO problems, local SA is often performed through computation of partial derivatives. For the designer, local sensitivity analysis is of great use with regard to an MDO problem. The aim is to provide an insight into the model behaviour in the vicinity of a point of interest and give directions in the design space where optimisation should be carried out to obtain a better design. For complex mathematical models, these partial derivatives must be evaluated numerically. In the case of high fidelity models with large number of variables, obtaining the outputs gradient can be very expensive. However, when automatic differentiation can be used, obtaining the derivative for SA becomes more efficient (Frey and Patil, 2000). Many sensitivity analysis techniques based on calculation of partial derivatives are developed for specific problems and applications (Gumbert *et al.* 2001; Sobieszczanski-Sobieski, 1990).

Local sensitivity analysis performed with partial derivative can be viewed as a particular case of one-factor-at-a-time approach. When one factor is varied, all the others are held constant. The local sensitivity index S_j can be expressed as follows:

$$S_j = \frac{\partial Y}{\partial X_j} \quad (2.1)$$

where Y is the response output, X_j is the input factor of interest and $x^0 = (x_1^0, x_2^0, \dots, x_k^0)$ is the nominal value for the input vector. However, the relative contribution is more useful to compare the effects between all inputs (Saltelli et al. 2000a, 2000b). This can be achieved by multiplying the above sensitivity index (2.1) by a fraction of the mean value for the factor of interest on the mean value for the output response, resulting in:

$$S_j = \frac{x_j^0}{y^0} \frac{\partial Y}{\partial X_j} \quad (2.2)$$

or by multiplying the index by a fixed fraction of the standard deviation:

$$S_j = \frac{\sigma(X_j)}{\sigma(Y)} \frac{\partial Y}{\partial X_j}$$

Saltelli et al. (2000a) advise not to use such an approach when the model is known to be highly non-linear.

2.1.5 Regression analysis

In regression analysis (Hines and Montgomery, 1980; Kennedy and Neville, 1986; Mason et al., 2003; Saltelli et al. 2000a), the user typically tries to fit a linear relationship between input and output variables of the following form:

$$y_j = b_0 + \sum_i b_i x_{ij} + \varepsilon_j, \quad i = 1, \dots, N \quad (2.3)$$

where j corresponds to the number of runs, b_i are the coefficients of the regression and ε_j is the error that is due to the linear approximation of the model. A least square analysis is commonly used to compute the b_i . The coefficient of determination is defined by:

$$R^2 = \frac{\sum_{i=1}^k (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^k (y_i - \bar{y})^2} \quad (2.4)$$

where \bar{y} is the mean of the sample and \hat{y}_i represents the estimates of y_i given by the linear regression. It indicates how well the model outputs can be predicted by the linear regression. A value close to one attests good performance. Linear regression can be used for SA in different ways. One of them is to use it to predict new values for the model under consideration. Another one is to investigate the coefficients to determine their statistical significance and the intervals of confidence of the regression. When the inputs are statistically independent, the study of the coefficients enables to understand the importance of the effect of each input on the output. Indeed, the regression coefficients b_j can be used for sensitivity analysis since they quantify the effect of each input factor on the output response. In some cases, it might be necessary to rewrite the linear regression in a standardized form to facilitate the comparison between inputs. Other variants of the linear regression analysis can be obtained through variables transformations or output response transformations. Regression analysis is based on the assumption that inputs are statistically independent and that the residuals are normally distributed. If these assumptions are not valid, the regression analysis will provide erroneous quantitative information about the sensitivity of the model (Frey and Patil 2002).

2.1.6 Response surface methodology

In the response surface methodology (RSM), the response of the model is approximated with an analytical function:

$$y = f(\mathbf{x}) + \varepsilon \quad (2.5)$$

where ε represent the error term and correspond to the output part that cannot be described by f . Linear regression can be considered as linear response surface (Myers and Montgomery, 1995). The error term is modelled as a random variable and therefore the response itself can be considered as a random variable, even if the model under consideration is strictly deterministic. RSM is often employed in optimisation when the current model response is expensive to compute.

One popular form for the analytical function is a quadratic polynomial (Myers and Montgomery, 1995; Mason *et al.* 2003). The response surface, referred as second-order, is then represented by:

$$y = \beta_0 + \sum_{i=1}^N \beta_i x_i + \sum_{i=1}^N \beta_{ii} x_i^2 + \sum_{i=1}^{N-1} \sum_{j=i+1}^N \beta_{ij} x_i x_j + \varepsilon. \quad (2.6)$$

The $(N+1)(N+2)/2$ coefficients of the second order polynomial are determined using the least square method over a number of sampled points. The entire method for obtaining the polynomial coefficients is described in detail in Myers and Montgomery (1995) and Mason *et al.* (2003). The presence of second order terms enables the description of interactions between input variables. As for linear regression, statistical test can be performed on the coefficients of the quadratic response to study their significance and effects on the output response (Myers and Montgomery, 1995; Frey and Patil 2002).

Another popular response surface methodology is the DACE (Design and Analysis for Computer Experiments) methodology (Sacks *et al.*, 1989a, 1989b) also known as Kriging in geostatistics. A Kriging model is an interpolation model which expresses the realization of a deterministic process as the sum of the realization of a regression model and random function (stochastic process):

$$\hat{y} = F(\mathbf{x}) + z(\mathbf{x}), \quad (2.7)$$

where F is a low order polynomial and the random process z is assumed to have mean zero and covariance:

$$\text{Cov}(Z(\mathbf{x}^i), Z(\mathbf{x}^j)) = \sigma^2 \mathbf{R}(\mathbf{x}^i, \mathbf{x}^j) \quad (2.8)$$

where σ^2 is the process variance. The correlation model \mathbf{R} is defined by:

$$\mathbf{R}(\mathbf{x}^i, \mathbf{x}^j) = \prod_{l=1}^N \mathbf{R}_l(\theta_l, x_l^i - x_l^j). \quad (2.9)$$

A variety of correlation functions is available but the most popular correlation model remains the Gaussian function:

$$\mathbf{R}_l(\theta_l, x_l^i - x_l^j) = \exp(-\theta_l (x_l^i - x_l^j)^2). \quad (2.10)$$

The hyperparameters θ_l are determined by fitting the Kriging model to a training sample. This is performed by maximising the likelihood of the training sample (Jones et al., 1998; Lophaven et al., 2002a). In a Kriging model, the correlation component accounts for interactions between variables that cannot be captured by the regression model.

For both types of response surface presented, the ability to predict a good model response is restricted to the input ranges of variation used to obtain the data sample on which the function was fitted.

The main disadvantage with this type of methodology is that depending on the number of inputs, the range of variation and the complexity of the model to approximate, obtaining an appropriate training sample might be very expensive. A clear advantage is that once a suitable response surface is obtained, a computationally expensive model is reduced to a more affordable one which makes this method very attractive for MDO purposes.

Sensitivity information can be obtained by studying the values of the parameters or by simply performing other types of sensitivity analysis on the reduced model (Frey and Patil 2002).

2.1.7 ANOVA

ANOVA is a statistical method performed on a sample of data, to test hypothesis on the influence of parameters on the output response (Hines and Montgomery, 1980; Kennedy and Neville, 1986). Note that it is different from the regression analysis as no model is given to explain the relationships between inputs and outputs (Frey and Patil 2002). ANOVA is a non-parametric and model independent method used for sensitivity analysis to determine if the values of the output vary in a statistically significant manner associated with variations in values for one or more inputs. The exact nature of the relations is not determined by ANOVA. The significance of the output response to changes in inputs is usually assessed with the F-test (Kennedy and Neville, 1986).

The main advantage of ANOVA is that no assumption is needed regarding the type of underlying model and both continuous and discrete inputs can be analyzed. Only assumptions for normal distribution of input factors and output responses are required. However, its main drawback is that it can become computationally expensive if the number of inputs is very large.

2.1.8 Screening sensitivity analysis: Morris method

Screening methods are used with computationally expensive models to identify the most influential input parameters in order to freeze or screen out insensitive variables, resulting in a reduction of the problem dimension and complexity. The knowledge gained about the model is only used to simplify the problem for further investigation. Screening methods are mainly sampled-based methods where the contribution of each factor to the response variation can be assessed. A good approach is to use a design of experiments where all factors are varied. It allows the evaluation of the main effect of each factor on the response and the effect of the interaction between factors. As a drawback, methods used exclusively for screening provide only qualitative sensitivity measures, allowing to rank the input factors in order of importance but do not

give useful quantitative information on the importance of the contribution of each factor to the output variation.

A popular screening method is the Morris method (Morris, 1991; Campolongo, 1999). This method determines which input factors may be considered to have effects which are:

- Negligible
- Linear and additive
- Non-linear or involved in interaction with other factors

The experimental plan proposed by Morris is composed of individually randomised one-at-a-time experiments designed to evaluate the impact of changing one factor at a time: its elementary effect. Consider a computational model for which an output y is a deterministic function of an input vector \mathbf{x} of N inputs denoted by $x_1, x_2, x_3, \dots, x_N$ where $0 \leq x_i \leq 1$ for $i=1, \dots, N$. The design space is defined by the N -dimensional hypercube $K^N = (x \mid 0 \leq x_i \leq 1, i=1, \dots, N)$. In the method of Morris, each input factor can only take a discrete number of values, called levels, which are chosen within the set $\{0, 1/(p-1), 2/(p-1), \dots, (p-2)/(p-1), 1\}$. The region of experimentation Ω is then represented by an N -dimensional p -level grid as illustrated in Figure 2.3.

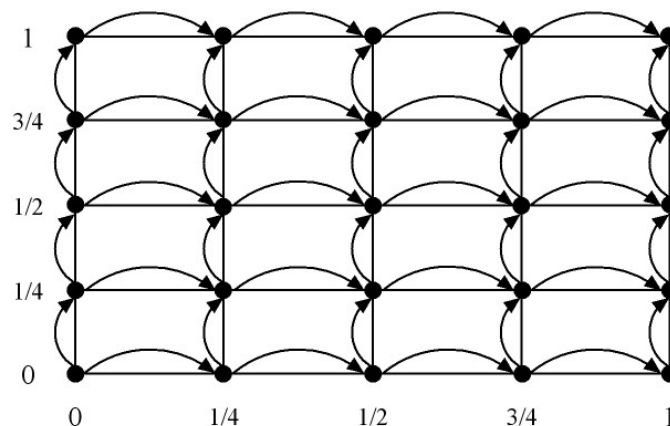


Figure 2.3: Representation of the p -level grid for $p=5$

The elementary effect for the i^{th} input is defined by:

$$d_i(x) = \frac{[y(x_1, \dots, x_{i-1}, x_i + \Delta, x_{i+1}, \dots, x_N) - y(x)]}{\Delta} \quad (2.11)$$

where Δ is a predetermined multiple of $1/(p-1)$ and $x = (x_1, x_2, \dots, x_N)$ is any selected value of the k -dimensional p -level grid Ω such that the transformed point $(x + e_i \Delta)$, where e_i is a vector of zeros with a unit as its i^{th} component is still in Ω for each index $i = 1 \dots N$. The finite distribution of elementary effects associated with the i^{th} input factor, is obtained by randomly sampling different x from Ω and is noted F_i . The total number of elements of F_i that can be computed is given by:

$$p^{N-1}[p - \Delta(p-1)] \quad (2.12)$$

The distribution of the absolute value of the elementary effects namely G_i is also considered as it enables to assess the contribution of each input in the case where the elementary effects can take both positive and negative values and therefore cancel each other, leaving the experimenter with no real insight into the model sensitivity. In his original work, Morris (1991) proposes to use both μ and σ , respectively the mean and the standard deviation of F_i as measures of sensitivity. However, because of the adverse cancellation effect, μ on its own is not reliable for ranking inputs according to the importance of their contribution to the output variation. Instead, the mean value for G_i , noted μ^* , is a better index to detect input factors with an important overall influence on the output. σ is used to detect factors involved in interaction with other factors or whose effect is non-linear.

The method of Morris is economic in the sense that it requires a number of model evaluations that is linear in the number of factors because input factors are varied one at a time. As a drawback, the method relies on a sensitivity measure, called the *elementary effect*, which uses incremental ratios and is apparently a local measure. However, the final measures μ and μ^* , are obtained respectively by averaging

elementary effects and their absolute values computed at different points of the design space, so as to lose the dependence on the specific point of the design space at which the elementary effects are computed. In this sense, since the average is made over the whole sample of points from the design space, the method can be regarded as a global sensitivity analysis method. In his paper, Morris (1991) compares his factorial sampling method with classical experimental designs and reports that the latter are much more expensive when the number of input factors is important. Added to its cheap computational cost, another advantage of the method of Morris is that its design does not rely on any assumptions of the model and is therefore model independent.

The Morris method is a qualitative method in the sense that the information given by (μ, σ) or (μ^*, σ) is only used to rank the input factors according to the importance of the contribution they make to the output. It is then used as a screening method to identify and discard non-significant input variables from the model under study. The information obtained with the Morris method is more general with respect to an elementary OAT method. Indeed, even if the sensitivity results are based on the evaluation of an elementary effect computed by varying each factor one at a time, the averaging on different values obtained over sampling grid gives a picture of the whole design space and makes the method global. As an advantage of its one-at-a-time heritage, the method of Morris remains relatively cheap and the number of runs required is given by:

$$r(N+1) \tag{2.13}$$

where N is the number of inputs and $5 \leq r \leq 15$ is the number of local measures of the elementary effect that are averaged.

2.1.9 Global sensitivity analysis: variance-based methods

Variance-based methods (VBM) are rigorous and theoretically sound approaches for total sensitivity calculation (Saltelli et al., 1993, 1999, 2002; Chan et al., 1997; Homma and Saltelli, 1996; Saisana et al., 2005). Variance-based approach for sensitivity analysis decomposes the output variance into partial variances of increasing dimensionality:

$$V(Y) = \sum_i V_i + \sum_{i \neq j} V_{ij} + \dots + V_{1,2,3,\dots,k} \quad (2.14)$$

where

$$V_i = V[E(Y / X_i = x_i^*)] \quad (2.15)$$

$$V_{ij} = V[E(Y / X_i = x_i^*, X_j = x_j^*)] - V_i - V_j \quad (2.16)$$

and so on.

In the decomposition of the variance, the term V_{ij} is the interaction effect between x_i and x_j . V_{ij} represents the part of the output variation due to input parameters x_i and x_j which cannot be explained by the sum of the first order effects of parameter x_i and x_j . Similar considerations can be made for higher order terms.

In the expression $V[E(Y / X_i = x_i^*)]$, the expectation is calculated over all variables X_j with $i \neq j$ and the variance is calculated over the values of X_i . $E(Y / X_i = x_i^*)$ is a function of X_i and for an influential factor its value will vary a lot with X_i resulting in a large value for the variance $V[E(Y / X_i = x_i^*)]$. Therefore, the usefulness of $V[E(Y / X_i = x_i^*)]$ as a measure for sensitivity analysis is easy to grasp.

The signification of $V[E(Y / X_i = x_i^*)]$ is derived from the note-worthy equation:

$$V(Y) = V[E(Y / X_i = x_i^*)] + E[V(Y / X_i = x_i^*)] \quad (2.17)$$

which can be rewritten as

$$E[V(Y / X_i = x_i^*)] = V(Y) - V[E(Y / X_i = x_i^*)] \quad (2.18)$$

The term $E[V(Y / X_i = x_i^*)]$ is the expectation of the variance of Y when the input X_i is frozen. It results that $V[E(Y / X_i = x_i^*)]$ is the actual reduction of the output variance that one can expect if one fixes the input X_i to a particular value. The term V_{ij}

represents the part of $V[E(Y / X_i = x_i^*, X_j = x_j^*)]$ that cannot be explained by the sum of the first-order terms V_i and V_j .

Let X_{-i} be the vector made up of all x_j with $j \neq i$ and let x_{-i}^* be a particular value of X_{-i} . $V[E(Y / X_{-i} = x_{-i}^*)]$ is the contribution to the output variance that is not due to i . It means that the difference $V(Y) - V[E(Y / X_{-i} = x_{-i}^*)]$ is equal to the sum of all terms in the variance decomposition that include X_i . Two coefficients of main interest for sensitivity can be defined:

- The main effect index: $S_i^1 = \frac{V[E(Y / X_i = x_i^*)]}{V(Y)}$. This gives the first-order contribution of X_i to the output response.
- The total effect index: $S_i^T = \frac{V(Y) - V[E(Y / X_{-i} = x_{-i}^*)]}{V(Y)}$. This gives the total contribution of X_i to the output response.

The first term gives the direct contribution of each variable to the output and the second term gives the total contribution. For each input, the difference between its total effect and its main effect gives an indication of the importance the contribution to the output due to interactions with other inputs. Therefore, both the main effect indices and total effect indices are necessary to obtain information about the non-additivity of the model and on the relative importance of variable interactions. Together, the main effect terms and total effect terms give a good description of the sensitivity of the model under investigation. An illustration of the use of variance-based method is given in Figure 2.4 where a model with three input variables and one output is considered and its variance is decomposed into the different contributions.

Variance-based methods are thought to be good candidates for potential global sensitivity analysis in MDO. Indeed, they present desirable properties for sensitivity analysis of complex models:

- These methods are model independent. No assumption on linearity or additivity of the model is required to apply them. It is particularly interesting for MDO problems where models are usually very complex and highly non-linear
- Very few assumptions on variables are necessary. They must be independent. In an MDO problem, variables are independent and could be varied on the whole design space.
- These methods can deal with both discrete and continuous variables which add to their flexibility
- Give a better insight into the design as input contributions are considered all together and not one-at-a-time as in a traditional sensitivity analysis
- Give quantitative information on the influence of each input factor to help the designer identify the most influential variables, on which the computational effort can be concentrated, and the non-sensitive variables which can be discarded or frozen to a specific value

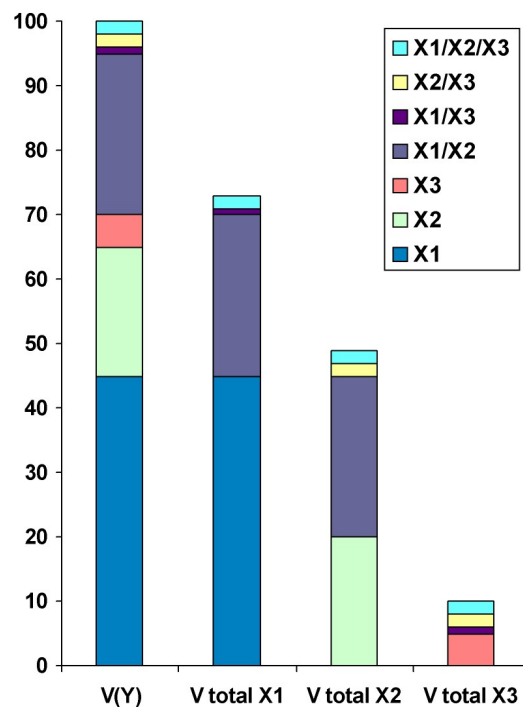


Figure 2.4: Typical output variance decomposition

Two main methods, Fourier Amplitude Sensitivity Test (FAST) and Sobol, have been developed to compute the different terms of the variance decomposition and are described in the two next sections. Based on two different approaches, they enable to compute both main and total effect indices and lead to the same predictions.

FAST and FAST Extended

The FAST method enables the estimation of the first order partial variances V_i as well as the total output variance V and hence first order sensitivity indices S_i . The key idea underlying the method is to compute an integral in a N -dimension space through a non-dimensional integral (Cukier et al., 1978; Saltelli and Bolado, 1998). Let the function

$$y = f(\mathbf{x}) = f(x_1, x_2, \dots, x_N) \quad (2.19)$$

be defined in the N -dimensional unit hypercube:

$$K^N = (\mathbf{x} \mid 0 \leq x_i \leq 1, i = 1, \dots, N) \quad (2.20)$$

Consider the set of transformations:

$$x_i = g_i(\sin(\omega_i \cdot s)), i = 1, \dots, N \quad (2.21)$$

This set of parametric transformations allows each factor to be varied in its range, as the new parameter s is varied from $-\infty$ to $+\infty$. The parametric equations define a curve that systematically explores the whole input parameter space. Note that the output function y is now expressed as a function of the parameter s only:

$$y = f(x_1(s), x_2(s), \dots, x_N(s)) = f(s) \quad (2.22)$$

If ω_i is chosen so that the set of frequencies $\{\omega_1, \omega_2, \dots, \omega_N\}$ is linearly independent, i.e. no ω_i may be obtained as a linear combination of the other frequencies with integer coefficients, summary statistics on the output can be computed by integrating either over the design space or along the curve itself. A set of k points usually equally spaced along the line is necessary for the evaluation of the set of main effect indices. A Fourier analysis of the output function f considered as a function of s is performed. The spectrum $\Lambda^2(\omega)$ of $f(s)$ at each frequency ω is computed by:

$$\Lambda^2 = A^2 + B^2 \quad (2.23)$$

where

$$A(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(s) \cos(\omega s) ds \quad (2.24)$$

$$B(\omega) = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(s) \sin(\omega s) ds \quad (2.25)$$

are integrals numerically evaluated over s . Finally, the S_i 's are obtained by computing the ratio between V_i and V , which are estimated according to:

$$V_i = 2 \sum_{p=1}^{+\infty} \Lambda^2(p\omega_i) \quad (2.26)$$

$$V = 2 \sum_{j=1}^{+\infty} \Lambda^2(p\omega_j) \quad (2.27)$$

An extended version of this method, FAST Extended, enables to obtain both S_i and S_i^T by choosing a very high value for the frequency ω_i and low values for all other frequencies ω_{-i} 's. By evaluating the Fourier spectrum for low frequencies only, the total effect index S_i^T can be estimated, whereas the first order effect index is evaluated as in the classical FAST.

Sobol Indices

Let a mathematical model output y described by a function $y = f(\mathbf{x})$ where $\mathbf{x} = (x_1, \dots, x_N)$, and is defined in a unit N -dimensional hypercube:

$$K^N = (\mathbf{x} \mid 0 \leq x_i \leq 1, i = 1, \dots, N) \quad (2.28)$$

Under certain assumptions (Sobol, 1993, 2001), it is possible to decompose $f(\mathbf{x})$ into $2^N - 1$ summands of increasing dimensions:

$$f(x_1, \dots, x_N) = f_0 + \sum_{i=1}^N f_i(x_i) + \sum_{i=1}^{N-1} \sum_{i < j}^N f_{ij}(x_i, x_j) + \dots + f_{1,2,\dots,N}(x_1, x_2, \dots, x_N) \quad (2.29)$$

where f_0 is a constant defined as $f_0 = \int_{K^N} f(\mathbf{x}) d\mathbf{x}$ and represents the mean value of f over K^N . The integral of every summand over any of its own variables is zero:

$$\int_0^1 f_{i_1, \dots, i_s}(x_{i_1}, \dots, x_{i_s}) dx_{i_n} = 0, \quad 1 \leq n \leq s \quad (2.30)$$

Note that the decomposition into summands can be easily generalised to groups of input factors with no intersection. The total variance of $f(x)$ can be written as :

$$V = \int_{K^N} f^2(\mathbf{x}) d\mathbf{x} - f_0^2 \quad (2.31)$$

while

$$V_{i_1, \dots, i_s} = \int_0^1 \dots \int_0^1 f_{i_1, \dots, i_s}^2(x_{i_1}, \dots, x_{i_s}) dx_{i_1} \dots dx_{i_s} \quad (2.32)$$

At this point, the sensitivity estimates can be introduced by:

$$S_{i_1, \dots, i_s} = \frac{V_{i_1, \dots, i_s}}{V} \quad (2.33)$$

The terms S_{i_1, \dots, i_s} 's can be considered as natural sensitivity estimates, as they give the fraction of the total variance of $f(x)$ which is due to any individual parameter or combination of parameters. The applicability of these estimates is related to the possibility of evaluating the multi-dimensional integrals using Monte-Carlo methods. Estimates of sample mean and variance are given by:

$$\hat{f}_0 = \frac{1}{k} \sum_{m=1}^k f(\bar{x}_m) \quad (2.34)$$

$$\hat{V} + \hat{f}_0^2 = \frac{1}{k} \sum_{m=1}^k f^2(\bar{x}_m) \quad (2.35)$$

where \bar{x}_m is a sampled point in the space K^k and N the size of the sample. The partial variances V_i needed to compute the main effects are evaluated by:

$$\hat{V}_i + \hat{f}_0^2 = \frac{1}{k} \sum_{m=1}^k f(\bar{u}_{im}, \bar{x}_{im}) \cdot f(\bar{v}_{im}, \bar{x}_{im}) \quad (2.36)$$

where u and v denote different realisations of the vector $(x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_N)$. Basically all variables are re-sampled but x_i . As

$$V = V_i + V_{-i} + V_{i,-i} \quad (2.37)$$

$$V_i^T = V_i + V_{i,-i} = V - V_{-i} \quad (2.38)$$

Similarly, the total effect index of x_i can be computed by evaluating

$$\hat{V}_{-i} + \hat{f}_0^2 = \frac{1}{k} \sum_m f(x_{1m}, \dots, x_{i-1m}, x_{im}, x_{i+1m}, \dots, x_{Nm}) \cdot f(x_{1m}, \dots, x_{i-1m}, x'_{im}, x_{i+1m}, \dots, x_{Nm}) \quad (2.39)$$

where only x_i is re-sampled.

Comparison between FAST/FAST Extended and Sobol

Comparisons have been conducted in the literature to assess their advantages and drawbacks (Homma and Saltelli, 1996; Campolongo et al., 1999; Saltelli and Bolado, 1998; Saltelli et al., 1999). Evaluation of both methods for computing both main and total indices S_i and S_i^T is summarised as follow:

Pros of FAST Extended:

- FAST Extended is cheaper than Sobol' to predict both main and total effects
- Relatively independent of the sample size

Cons of FAST Extended:

- Computationally expensive as a large sample is required for computing the integrals
- Problem of interference when defining the frequencies ω_i . If poorly chosen, when the main order effect of one input is evaluated, a fraction of the contributions of other inputs might be added
- Poor ability to compute high order indices

Pros of Sobol' method:

- Stable predictions if sample large enough
- Efficient at predicting total effects
- Possibility to compute high order effect indices

Cons of Sobol' method:

- Very dependent on the sample size
- Computationally more expensive than FAST Extended

The main feature of variance-based methods is that they give a means to obtain quantitative information about the importance of the contribution of each input to the

model output. FAST Extended and Sobol' method used to compute variance-based first order and total indices are "large sample-based" methods which may render this approach non-viable for computationally expensive models with large number of inputs. A large number of model evaluations is required to compute numerically the different integrals derived in the previous section. A summary of the amount of model evaluations required to carry out the different sensitivity analysis is given in Table 2.1.

	SA Method	Number of model evaluations (Model with N inputs)
Local SA	OAT	N
Global SA	Morris	$r(N+1)$ where $5 \leq r \leq 15$ number of local measures
	FAST (to obtain S_i 's only)	kN where $100 \leq k \leq 10000$ sample size
	FAST Extended (to obtain S_i and S_i^T)	kN where $100 \leq k \leq 10000$ sample size
	Sobol' (to obtain S_i and S_i^T)	$2(N+1)k$ where $100 \leq k \leq 10000$ sample size

Table 2.1: Number of model evaluation for model-independent sensitivity analysis methods

Compared to FAST, with the FAST Extended method, both indices S_i and S_i^T can be obtained with no additional cost.

At this point it is important to note that for both quantitative methods, FAST Extended and Sobol, the number of runs is proportional to k the number of inputs. It means that in the case of a model with a large number of inputs, a cheap preliminary screening sensitivity analysis, like Morris method, could be used to discard non-significant factors and save a large number of model evaluations when it comes to applying global sensitivity analysis.

A representation of sensitivity analysis classes, given in Figure 2.5, is proposed in Campolongo et al. (1999) to illustrate the trade-off the designer has to make between the information he wants to obtain and its computational cost. Note that in this

representation, the cost of each sensitivity analysis method is referred to the number of times the model must be executed.

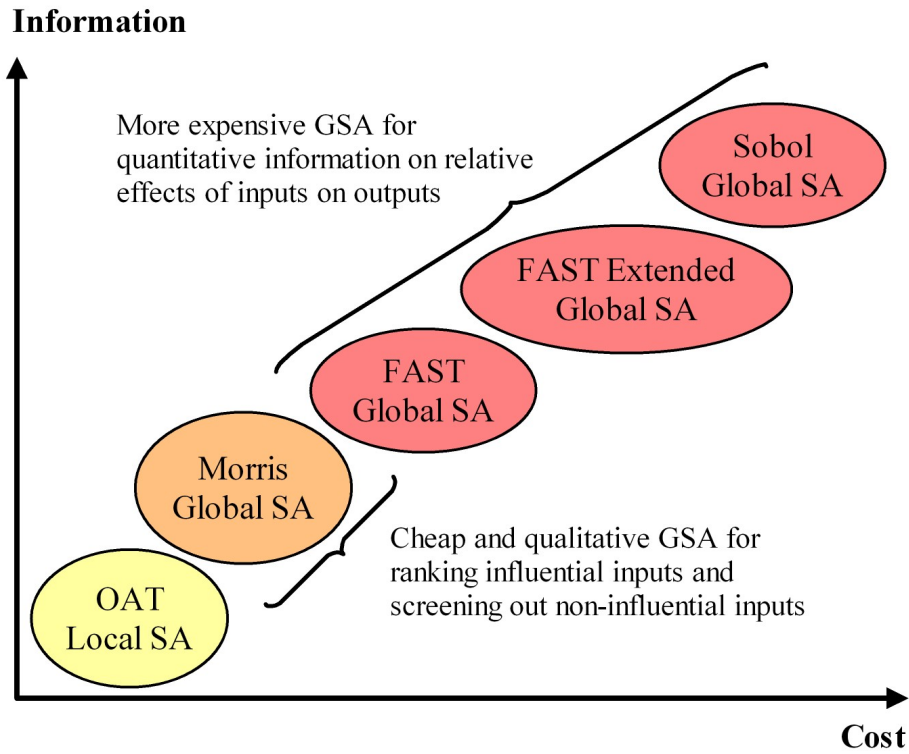


Figure 2.5: Representation of information vs cost for model-independent sensitivity analysis

2.2 Multi-objective optimisation problem

2.2.1 Formulation

The aim of a multi-objective optimisation problem is to find a vector of the design space that optimises a set of objectives and meets a set of constraints. The objective functions are the quantities that the designer wishes to maximise, minimise or match a certain value. The mathematical problem in standard form for minimisation can be formulated as follow:

$$\text{Minimise: } \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})\} \quad (2.40)$$

$$\text{Subject to: } g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, L \quad (2.41)$$

$$h_j(\mathbf{x}) = 0, \quad j = 1, \dots, K \quad (2.42)$$

$$x_l^l \leq x_l \leq x_l^u \quad l = 1, \dots, N \quad (2.43)$$

where \mathbf{x} is the design vector with N variables, $\mathbf{f}(\mathbf{x})$ is the objective vector with M objective functions, g and h are respectively the inequality and equality constraints on the design vector, and the constraints (2.43) are called boundary constraints. An input vector is said to be feasible if all the constraints are satisfied.

When $M = 1$, there is only one objective function to be minimised and the problem is referred to as single-objective optimisation. In this case, classic optimisation methods (Siddall, 1982) or evolutionary methods (Rao, 1996) such as *Genetic Algorithm* or *Simulated Annealing* can be used to solve the problem. When $M > 1$, the problem is known as multi-objective optimisation. In this case, minimising several objectives at the same time might not be possible and the concept of Pareto solution must be introduced.

2.2.2 Pareto concept

Solving a multi-objective optimisation problem with conflicting objectives usually does not have a unique solution, but a set of non-dominated solutions also known as Pareto solution. Pareto optimality is an important concept for multi-objective optimisation. A feasible design point is said to be Pareto optimal if no other feasible design can improve some of the objectives without simultaneously being detrimental to others. In the literature, the set of non-dominated solutions is referred to as Pareto surface or efficient frontier. In the case of the minimisation problem formulation presented above, a Pareto point is defined as:

$$\mathbf{x}^* \text{ is Pareto} \Leftrightarrow \text{no feasible } \mathbf{x} \text{ such that } \begin{cases} f_i(\mathbf{x}) \leq f_i(\mathbf{x}^*) \text{ for all } i = 1, \dots, M \\ f_j(\mathbf{x}) < f_j(\mathbf{x}^*) \text{ for } 1 \leq j \leq M \end{cases}$$

2.2.3 Obtaining Pareto solutions

The growing interest for multi-objective optimisation required the development of specific methods to obtain solutions of such a problem. A large range of methods is available. Reviews and comparative studies are also presented in the literature (Marler and Arora, 2004; De Weck, 2004). De Weck distinguishes between two main types of

methods: those based on a reformulation of the problem into a single objective optimisation and those based on the dominance of Pareto solutions.

In the first group of methods, the approach is to reformulate the original problem into a single objective optimisation to make use of classic single criterion optimisation methods. In its simplest formulation, the single objective takes the form of a *Weighted Sum* of objectives. However, it was demonstrated that such a method has limitations to capture non-convex parts of the Pareto set (Das and Dennis, 1997). More recently, a scalarization using utility functions, also referred as a *class function*, was proposed by Messac: *Physical Programming* method (Messac, 1996; Messac and Ismail-Yahaya, 2002, Messac and Mattson, 2002). Other popular methods making use of the reformulation into a single objective optimisation problem are the ε -*Constraint Method* (Palli et al. 1998), the *Goal Programming, Compromise Programming* (Tappeta and Renaud, 1999).

In the second group, multi-objective optimisation methods are based on the fact that the Pareto solutions are non-dominated. The simplest method explores the design space by performing an extensive grid search with possible use of surrogate models (Shan and Wang, 2005; Wilson et al. 2001) and filtering to select non-dominated points. Such method only enables to get an approximation of the Pareto front and may not be affordable in the case of a computationally expensive model. Two other methods have attracted an increasing interest: the Normal Boundary Intersection (Das and Dennis, 1998) and *Multi-Objective Genetic Algorithm* (MOGA) (Deb, 2001). The NBI allows to obtain evenly spread Pareto solutions of the efficient frontier. However, for complex problems it can converge to local optima which results in a reduced representation of the Pareto front. MOGA has the adverse feature of clustering the solutions in a particular region of the objective space, which can lead to a non-uniform representation of the Pareto front.

Parallel to the current study, a sophisticated method for obtaining an improved description of the Pareto surface was developed at Cranfield University in the same research group (Fantini, 2007; Fantini et al., 2007; Guenov et al. 2005; Utyuzhnikov et al., 2005). This multi-objective optimisation tool also referred as calculation engine allows to set up and find solutions to an arbitrary continuous multi-objective optimisation problem. The calculation engine is capable of determining well-distributed

Pareto points by reformulating the multi-objective optimisation problem for each of the Pareto points sought. Furthermore, it has the unique feature of determining both the global Pareto front as well as local Pareto fronts when present. This tool was used to obtain Pareto solutions of the multi-objective optimisation problems considered in the present research.

The focus of this research is not on Pareto solution generation but on sensitivity analysis. Therefore, these methods are not discussed in detail further.

2.3 Pareto frontier exploration and sensitivity analysis for decision making

The inherent difficulty in solving a multi-objective optimisation problem makes generating the entire Pareto set a complicated task. In their paper, Ruzika and Wiecek (2005) make a comprehensive survey of methods developed to approximate part of the/the entire Pareto set. They distinguish between techniques developed exclusively for bi-criteria optimisation problems and those developed in the more general case of multi-objective optimisation. Within each category, they propose a classification based on the type of approximation. Methods of 0th order approximate the Pareto front by generating a reduced number of Pareto optimal solutions. Methods of 1st and 2nd order approximate part of the/the entire Pareto front linking available Pareto solutions with line segments or quadratic curves. This section reviews some of these methods but focuses on those that provide sensitivity information of the Pareto set.

2.3.1 Obtaining the sensitivity of Pareto solutions

A simple method to obtain sensitivity analysis of Pareto solution is proposed by Andersson (2002). In this paper, Andersson studies how each design parameter affects the solution of the multi-objective optimisation problem at different locations on the Pareto front. This method combines DOE and response surface methodology to obtain the sensitivity. Typically, once a Pareto solution is available in the objective space, a D-optimal design DOE (Myers and Montgomery 1995) is centred on the corresponding point in the design space and a quadratic polynomial is used to build the response surface:

$$y = b_0 + \sum_{i=1}^N b_i x_i + \sum_{i=1}^N b_{ii} x_i^2 + \sum_{i=1}^{N-1} \sum_{j=i+1}^N b_{ij} x_i x_j \quad (2.44)$$

Sensitivity information is obtained by studying the variations of coefficients and the gradients of equation (2.44) along the efficient frontier. Sensitivity information is presented to the decision maker in the form of tables showing for each parameter the local effect at each Pareto point under consideration. Here no allowance is made to constraints that can be violated by moving away from the Pareto solution. This technique simply provides direct information on the relationship between objectives and parameters at different locations on the Pareto surface, and therefore at different locations in the design space. It gives an insight into how the influence of some parameters varies along the Pareto surface but does not allow to obtain new Pareto solutions. Such information allows the designer to learn more about the properties of the system under consideration.

Balbas et al. (2005) study the sensitivity of a Pareto solution related to a problem parameter. Based on the Karush-Kuhn-Tucker necessary condition for optimality, they derive first order derivatives with respect to the problem parameter. Balbas et al. studies how small perturbations of this parameter will affect the design variables and objectives of the current Pareto solution. Here, two types of sensitivity are investigated: the sensitivity in the design space and in the criterion space.

2.3.2 Articulation of preferences

In the case of a convex bi-criteria optimisation problem, Li et al. (2003) try to approximate the entire Pareto surface with a hyper-ellipse of the form:

$$\left(\frac{f_1 - f_1^{min}}{f_1^{max} - f_1^{min}} \right)^v + \left(\frac{f_2 - f_2^{min}}{f_2^{max} - f_2^{min}} \right)^v = 1 \quad (2.45)$$

where (f_1^{min}, f_2^{max}) and (f_1^{max}, f_2^{min}) are the two end points of the Pareto surface in the criterion space. It is assumed that the Pareto surface is continuous and convex. Three points lying on the Pareto surface are necessary to derive equation (2.45). The two

extreme points are simply obtained by minimising objective f_1 and f_2 individually. A third point is obtained by minimising a weighted sum of both objective, for a combination of weight coefficients that is to be decided by the designer based on some a priori engineering knowledge of the optimisation problem. Once the three Pareto solutions are obtained, equation (2.45) can be solved for v . Li *et al.* declare that a minimum of information (only three Pareto points) are needed to obtain a good approximation of the entire Pareto surface. The main advantages of the hyper-ellipse approximation are its simplicity and flexibility. An analytical description of the Pareto front can then be obtained. However, the approximation does not relate the point lying on the hyper-ellipse to the corresponding point in the feasible space. The hyper-ellipse approximation is used in Fadel and Li. (2002) to help decision-making of the designer when solving a bi-objective design problem. In this paper, they typically use the hyper-ellipse approximation to offer the designer an insight in to the trade-off on the Pareto surface. It is simply used as a guideline for deciding on the Pareto solution that will suit best preferences of the designer that where not taken into account in the initial formulation of the optimisation problem. After the preference information returned by the designer, an optimisation procedure is carried out to obtain the corresponding value of the design variable in the design space. It is proposed to solve two types of optimisation problem. If one of the objective values is specified by the decision maker, $f_1(\mathbf{x}) = \varepsilon_1$, it is used as a constraint and the ε -Constraint Method is applied to solve for the other objective and design parameters. The optimisation problem is as follows:

$$\min_{\mathbf{x}} f_2(\mathbf{x}) \quad (2.46)$$

subject to

$$g_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, L \quad (2.47)$$

$$f_1(\mathbf{x}) \leq \varepsilon_1 \quad (2.48)$$

When the decision maker specifies values for both objectives $f_1(\mathbf{x}) = b_1$ and $f_2(\mathbf{x}) = b_2$, it is proposed to use a *Goal Programming* to solve for the deviation variables and input variables:

$$\min_{\mathbf{x}} \left| \sum_{j=1}^2 (d_j^- + d_j^+)^p \right|^{1/p}, \quad p = 1 \text{ or } 2, \quad (2.49)$$

subject to

$$g_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, L \quad (2.50)$$

$$f_j(\mathbf{x}) + d_j^- - d_j^+ = b_j, \quad j = 1 \text{ or } 2 \quad (2.51)$$

$$d_j^- \geq 0 \text{ and } d_j^+ \geq 0, \quad j = 1 \text{ or } 2 \quad (2.52)$$

$$d_j^- \cdot d_j^+ = 0, \quad j = 1 \text{ or } 2 \quad (2.53)$$

In this method, the approximation is provided to the designer to help him/her make trade-off decisions and decide on a particular approximate compromise solution. The corresponding true Pareto solution is obtained after solving the optimisation problem. As a result, the error due to the hyper-ellipse approximation is eliminated. Note that in the case for the goal programming method, the result might exhibit an under or over achievement. This approach is applied to bi-criteria optimisation structural problem and performs well. Its use in the case of a non-continuous Pareto surface is also discussed. Compared to those obtained with the true Pareto front, the trade-offs given by the approximation might show a large discrepancy. However, this should not mislead the decision maker in his/her decision. As in the continuous case, the error due to the approximation is removed after the true Pareto solution closest to the decision maker preferences is obtained. It is shown that when using the *Weighted Sum* method to find a solution of the multi-objective optimisation problem, there is no relation between the preferences articulated by the designer before the optimisation and the results obtained. Indeed, the relative importance of the objectives of a Pareto solution obtained with a particular weighted sum does not correspond to the relation between the specified weight coefficients.

Similarly, in Kasprzak and Lewis (2001), the emphasis is put on the decision-making process rather than on the approximation itself. In this paper, the efficient frontier is obtained by making use of an extensive grid search. It ensures that both convex and non-convex parts of the Pareto surface are captured and do not require the designer to articulate a priori preferences. A high order polynomial is used to fit the set of non-dominated solutions and obtain an analytical description of the entire Pareto

front. After the approximation is obtained, the designer starts making preferences about the solution of the problem, which results in scaling the criterion space. An immediate geometric consequence of the scaling is that the best solution with respect to the designer preferences is changed. A collinearity theorem is introduced and provides an immediate relationship between the preferences of the designer, i.e., the scaling factor k , the coordinates of the utopia point and the corresponding optimum design. In the two objectives case where the scaling is on f_1 , the relationship is as follows:

$$k = \frac{(f_2 - f_{2u})f_2'}{f_1 - f_{1u}} \quad (2.54)$$

where f_{1u} , and f_{2u} correspond to the particular values obtained from the individual minimisation of objective f_1 and f_2 respectively. In the criterion space, the point defined by the coordinates (f_{1u}, f_{2u}) is called *utopia* point. f_2' is the derivative of f_2 expressed as a function of f_1 . Another method named contour space and developed by Kasprzak and Lewis (2000) relates the information on these results obtained in the criterion space to the design space.

A consensus among researchers working in multi-objective optimisation is that it can be difficult to come up with meaningful preferences between objectives before the optimisation process. It is argued in Fadel et al. (2002) that articulating preferences once a good description of the set of efficient solutions is obtained is preferable. Information about the Pareto surface could be obtained to help the designer achieve a better trade-off solution of the problem under study than using method of a priori preference assignment. In their paper, Fadel et al. (2002) propose to investigate the robustness of a two-objective Pareto set, not related to the design variables but to the relative importance of the criteria. The sensitivity is studied by perturbing the initial multi-objective optimisation problem and minimising each objective individually. Assume the initial optimisation problem is formulated as:

$$\min_{\mathbf{x}} (f_1(\mathbf{x}), f_2(\mathbf{x})) \quad (2.55)$$

subject to

$$\mathbf{x} \in \mathbf{X}^* \quad (2.56)$$

where \mathbf{X}^* is the set of all feasible design. The associated perturbed multi-objective optimisation problem is defined by:

$$\min_{\mathbf{x}} (f_1(\mathbf{x}), f_2(\mathbf{x})) \quad (2.57)$$

subject to

$$\begin{aligned} f_1(\mathbf{x}) - f_1^* &\geq \varepsilon \\ f_2(\mathbf{x}) - f_2^* &\geq \varepsilon \\ \mathbf{x} &\in \mathbf{X}^* \end{aligned} \quad (2.58)$$

where f_1^* and f_2^* are the minimum values obtained for each objective when minimised individually. The resulting Pareto curve for the perturbed problem (2.57) and (2.58) is the truncated efficient frontier obtained for initial problem (2.55) and (2.56). However, the entire set does not need to be obtained. Instead, both objectives are also minimised individually for problem (2.57) to obtain information on the end point of the Pareto curve. The cost of such study is reduced as only four optimisation runs are required to obtain the Pareto curve end points. This technique is particularly attractive when the designer is interested in trade-off solutions lying near the extremities of the efficient set but otherwise shows limited usefulness.

2.3.3 Understanding local trade-off with Pareto approximation

The literature shows that the number of methods attempting to approximate the entire Pareto surface is quite small. This comes from the fact that for real life cases, the efficient frontier might be non-smooth, discontinuous, and therefore these techniques can only apply to very simple optimisation problems. An alternative approach is to focus on a few Pareto solutions available and try to obtain more information about the Pareto frontier by studying locally trade-offs between objectives.

An attempt to provide the designer with a local analytical description of the Pareto set is given by Zhang J. *et al.* (2000) and Wiecek *et al.* (2001). This follows the work carried out by the same authors (Chen *et al.*, 1999) where an interactive procedure was suggested to solve a two-objective robust design problem with the assumption that

preferences of the decision-maker were not always taken into account in the utility function. In this paper, the authors perform robust optimisation and solve a bi-criteria problem where they attempt to minimise both mean and variance of the system under study. Here the *Compromise Programming* approach is used to generate the Pareto solutions. The efficient frontier is locally described with a utility function of the form:

$$q(\mathbf{f}) = \frac{1}{2} \alpha (w_1 f_1 - w_2 f_2)^2 + p_1 f_1 + p_2 f_2 + c \quad (2.59)$$

where coefficients p_1 , p_2 and c are completely determined by the value of both criteria at the utopia point and the combination of weights used to obtain the current Pareto point. The coordinates of the utopia points in the criterion space are the values obtained for objective f_1 and f_2 when minimised individually. This procedure is described in more detail in (Chen et al. 1999). Equation (2.59) defines a family of quadratic utility functions. In Zhang J. et al. (2000), the authors propose a method for determining the best value for α . More Pareto points are obtained near the candidate Pareto solution by solving ε -constrained problems. The best value for α is the one that minimises the difference between the predicted and true values of the utility function (2.59) at the Pareto points located in the neighbourhood. The discrepancy between the true Pareto front and the approximation might be sensitive to the range of variation. In this case, an iterative procedure where the range is reduced progressively should allow to find a range where the approximation given by the utility function is acceptable. This method was tested on different simple test cases and showed that it could provide the designer with valuable local information about the Pareto surface. However, as previously, once new preferences are articulated by the decision maker, there is no easy way to obtain the corresponding value for the design parameters and like in Fadel and Li (2002), an optimisation problem must be solved. Although they illustrate it in the case of robust optimisation, the method could be easily adapted for bi-criteria optimisation problems.

More sophisticated approaches based on the gradient-projection method and feasible descent directions have been developed recently (Tappeta and Renaud, 1999, 2001, Zhang, W.H. 2003a, 2003b). They enable to obtain the local trade-offs between

objectives on the Pareto surface and the effects of a perturbation of the Pareto solution on the design parameters.

Tappeta and Renaud (1999, 2001) propose an interactive procedure to study solutions of the multi-objective optimisation problem under consideration, taking into account the designer preferences. Assuming the objective and constraint functions are twice differentiable and that the set of active constraints does not change in the vicinity of the current Pareto point, they derive local linear and quadratic approximations of the Pareto surface of the form:

$$f_p = f_p^* + \sum_{\substack{i=1 \\ i \neq p}}^M \frac{df_p}{df_i} \Delta f_i \quad (2.60)$$

$$f_p = f_p^* + \sum_{\substack{i=1 \\ i \neq p}}^M \frac{df_p}{df_i} \Delta f_i + \frac{1}{2} \sum_{\substack{j,k=1 \\ i \neq p}}^M H_{jk}^{(p)} \Delta f_j \Delta f_k \quad (2.61)$$

where $H_{jk}^{(p)}$ are the components of the Hessian calculated on the Pareto surface. In their paper, they follow the *Compromise Programming* method to obtain Pareto solution, which leads to the following optimisation problem:

$$\min_{\mathbf{x}} z + \alpha \sum_{i=1}^M w_i f_i \quad (2.62)$$

subject to

$$\begin{aligned} z &\geq w_i (f_i(\mathbf{x}) - f_i^u) \quad i = 1, \dots, M \\ \mathbf{g}_i(\mathbf{x}) &\leq 0 \quad i = 1, \dots, L \\ \mathbf{x}^l &\leq \mathbf{x} \leq \mathbf{x}^u \end{aligned} \quad (2.63)$$

where

$$w_i = \frac{1}{f_i - f_i^u} \quad (2.64)$$

α is a sufficiently small positive number, and the components of \bar{f} are *aspiration levels* and f^u is the utopia point. When a Pareto solution is available, it is presented to the designer. If not satisfied, s/he can decide to explore the neighbourhood of this point according to her/his preferences. First order directional derivatives on the Pareto surface are obtained to derive a linear approximation. To obtain the second order approximation, the Hessian is evaluated based on Pareto points generated in the neighbourhood of the point under study using least-squares minimisation. The designer uses the linear approximation of the Pareto surface to provide new *aspiration levels*. A two-step optimisation procedure performed with *Compromise Programming* method give new Pareto solutions. The prediction step uses linear approximation of objectives and constraints with respect to the design variables to predict an approximate Pareto point. The second step is a correction step where the approximate solution is projected onto the true Pareto surface by solving the initial *Compromise Programming* problem. A local quadratic approximation of the Pareto surface is obtained and allows the decision maker to explore the trade-offs in the neighbourhood of the candidate point. In Tappeta et al. (2000), the same approach is used, but *Physical Programming* is introduced to capture the decision maker preferences. In Tappeta et al. (2002), the interactive procedure for Pareto solution decision-making is applied to multidisciplinary engineering problems, namely hover-craft design and aircraft concept sizing and shows very promising results for MDO.

Zhang W. H. (2003a, 2003b) uses the same approach based on a gradient-projection method to derive directional derivatives on the Pareto surface and study how improving one objective will impact the others. In the case of a two-objective problem, the directional derivative corresponds to the minimum amount one must sacrifice on one objective to improve the other one. However, in his paper, Zhang only considers first order directional derivatives and no general method for obtaining a local approximation of the Pareto surface is proposed. As part of the local exploration of the Pareto surface, Zhang W. H. also proposes to use linear approximation of the non-active constraints to ensure they remain satisfied when moving away from the Pareto point under study. In addition to the local linear approximation, Zhang W. H. also proposes an approach to detect non-differentiability of the Pareto surface related to changes in the set of active constraints. The approach is based on the Karush-Kuhn-Tucker condition

and the gradient-projection method. Zhang W. H. shows that perturbations from the current Pareto point in the criterion space can lead to changes in the coefficients of a weighted sum of the objective functions. It indicates a modification of the normal to the Pareto surface, which means that the Pareto surface is not smooth. Zhang developed the approach to identify non-differentiability along a direction but not in the general case.

2.4 Summary

In this chapter, the context of the present research is introduced. First, the concept of sensitivity analysis is presented and various methods used in the field are described. Their applications, advantages and drawbacks are outlined. It is shown that variance-based methods offer several desirable properties for use in complex system design. Then, the multi-objective optimisation problem and the Pareto optimality concept are presented. The most popular methods for obtain solutions of such problem are briefly described. Finally, different techniques developed for obtaining sensitivity information of the Pareto front are reviewed.

Chapter 3

A sensitivity analysis approach for an efficient multi-objective optimisation

In this chapter, requirements are specified and an approach is proposed for sensitivity analysis aiming to increase the efficiency of the multi-objective optimisation. First, the method used for reducing the number of variables of the optimisation problem is described. Second, the requirements for an improved method for approximating the Pareto surface are outlined.

3.1 Reducing the dimensionality of the optimisation problem

In Kim *et al.* (2004), variance-based methods (VBM) are used to identify and fix non-significant random variables through an adaptive procedure for reliability-based optimisation. It enables them to reduce the number of simulations required to construct the stochastic response surface required for uncertainty propagation. In this adaptive procedure, the reduction of the number of variables is carried out locally when a high order stochastic response surface is constructed. To the author's knowledge, the possibility of using VBM to reduce the number of variables before solving a deterministic optimisation problem has not been explored yet.

3.1.1 Identifying non-significant inputs

In the context of MDO, we assume that the model, which evaluates the objectives and constraints of the optimisation problem, can be represented as a black-box. This stands for the fact that the designer is unaware of the complex internal mechanisms of the model, which relate objectives and constraints to the input variables. The methodology developed in this general case can then be readily implemented for a more specific purpose. Let us consider the following multi-optimisation problem:

$$\text{Minimise: } \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_M(\mathbf{x})\} \quad (3.1)$$

$$\text{Subject to: } g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, L \quad (3.2)$$

$$h_j(\mathbf{x}) = 0, \quad j = 1, \dots, K \quad (3.3)$$

$$x_l^l \leq x_l \leq x_l^u \quad l = 1, \dots, N \quad (3.4)$$

where $\mathbf{x} = (x_1, \dots, x_N)$ is the vector of independent variables. In the context of deterministic optimisation, it is also assumed that all variables have uniform distribution defined by a minimum and maximum bound.

In this approach, it is proposed to use VBM to evaluate the global sensitivity indices of each input with respect to all outputs. This allows to quantify the effect of the variations of the inputs on the outputs variance. Both main and total sensitivity indices described in chapter 2 are calculated. The discrepancy between these two indices represents the contribution to the output variance due to interaction effects among input variables. The main and total effects represent the minimum and maximum expected reduction of the output variance if the input is fixed to a specific value. Therefore, freezing an input variable with a negligible total sensitivity index will not affect the output variance.

It is proposed to remove from the original optimisation formulation a variable with a negligible effect on all objectives and constraints. When more than one variable is non-significant, the designer should consider these as a group of variables and see whether the sensitivity of the group is still below the significance threshold. Typically, this would imply performing another sensitivity analysis and resample the design space. Such procedure could be very expensive. Instead, it is recommended to make sure that

the sum of the group's total indices remains below a particular threshold value. Without loss of generality, let us assume that the first R variables are non-significant to all objectives and constraints. The problem can be reformulated as follows:

$$\text{Minimise: } \mathbf{f}(\mathbf{x}_{\text{red}}) = \{f_1(\mathbf{x}_{\text{red}}), f_2(\mathbf{x}_{\text{red}}), \dots, f_M(\mathbf{x}_{\text{red}})\} \quad (3.5)$$

$$\text{Subject to: } g_i(\mathbf{x}_{\text{red}}) \leq 0, \quad i = 1, \dots, L \quad (3.6)$$

$$h_j(\mathbf{x}_{\text{red}}) = 0, \quad j = 1, \dots, K \quad (3.7)$$

$$x_l^l \leq x_l \leq x_l^u \quad l = R+1, \dots, N \quad (3.8)$$

where $\mathbf{x}_{\text{red}} = (x_{R+1}, \dots, x_N)$. All variables with negligible effect are fixed to a value $x_m = x_m^*$ for $m = 1, \dots, R$.

The approach is presented in the case of a multi-criteria optimisation, but the same considerations can be made when a single objective is optimised.

3.1.2 Obtaining the sensitivity analysis indices

Two main methods exist to compute the sensitivity indices: Sobol and FAST Extended. Both require large samples to estimate accurately the indices. The numerical calculation techniques together with the sampling scheme are implemented in Simlab (Saltelli and Tarantola, 2003).

There is no clear answer on the size of the sample required. As mentioned in the literature review, FAST Extended can provide both main and total effect indices at a lower cost than Sobol method and therefore appear more efficient. However, its inherent complexity makes it very hard to implement if one does not use Simlab. Instead, Sobol is based on a more practical method for estimating the sensitivity indices. In the current implementation of Simlab, the sampling used to calculate the Sobol indices has a particular property that large samples re-use points from smaller samples. Therefore, an iterative procedure can be implemented. The user progressively increases the size of the sample until the estimate for the Sobol indices has achieved convergence. Such procedure cannot be carried out with the FAST Extended sampling scheme where it would require to regenerate the whole sample.

When the model is too expensive, a possibility is to obtain a surrogate model on a small data sample. This enables to compute estimates of the sensitivity indices at a reduce cost. If the formulation of the surrogate model is explicitly known, the indices can be derived analytically. Chen et al. (2005) and Sudret (2006) have derived these formulas for Kriging models and Polynomial chaos expansions respectively. In this procedure, the designer is only interested in finding non-significant input rather than obtaining accurate quantification of the effect of significant inputs. Therefore, when inputs with negligible effect are present, they can be identified with relatively small samples.

Finally, one must keep in mind that the objective of the procedure is to save time during the optimisation process. To make the approach efficient, it is crucial that the total computational effort required performing the SA and the optimisation of the reduced problem remain significantly cheaper than the optimisation on the full problem.

3.1.3 Freezing non-significant variables to a specific value

The question here is how to fix the non-significant variables. In design under uncertainty and robust design, a common practice is to fix non-significant variables to their mean value. However, this might not always be the most appropriate choice in the case of an optimisation problem. Their final impact on the solution of the optimisation problem is difficult to predict. In this research, the author came across some problems where fixing non-significant inputs variables to an arbitrary value could lead to a small degradation of the solution of the optimisation. A simple example is considered to investigate the effect of fixing a non-significant variable to an arbitrary value.

Test optimisation problem:

$$\text{Minimise: } \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x})\} \quad (3.9)$$

$$\text{Subject to: } g(\mathbf{x}) = 1 - x_1^2 - x_2^2 - 0.2x_3^2 \leq 0 \quad (3.10)$$

$$0 \leq x_l \leq 1 \quad l = 1, \dots, 3 \quad (3.11)$$

where

$$f_1(\mathbf{x}) = \exp(4x_1) \quad (3.12)$$

$$f_2(\mathbf{x}) = \exp(4x_2) \quad (3.13)$$

Objectives f_1 and f_2 depend only on a single variable, x_1 and x_2 respectively. The third variable appears only in constraint g and therefore has no effect on the objectives. It is clear that f_1 and f_2 take small values when x_1 and x_2 are close to 0. However, because of constraint (3.10), x_1 and x_2 cannot be minimised together. Moreover, minimum values for x_1 and x_2 are obtained when $x_3 = 1$. The sensitivity indices can be derived analytically as given in Table 3.1. It is worth noting that in this case, there is no interaction effect between the various inputs and therefore main and total effect sensitivity indices are equal.

	Main effect			Total effect		
	x_1	x_2	x_3	x_1	x_2	x_3
f_1	1.0	0.0	0.0	1.0	0.0	0.0
f_2	0.0	1.0	0.0	0.0	1.0	0.0
g	0.49	0.49	0.02	0.49	0.49	0.02

Table 3.1: Main and total sensitivity indices of objectives f_1 and f_2 and constraint g for test example

Table 3.1 shows that input x_3 has a small effect compared to input x_1 and x_2 and therefore could be removed from the optimisation formulation. Variable x_3 is fixed to its mean value, i.e. $x_3 = 0.5$. The solutions for the full and reduced problem optimisation are obtained using the in-house developed gradient-based optimisation tool for multi-objective optimisation mentioned in chapter 2 (Fantini *et al.*, 2007; Guenov *et al.*, 2005; Utyuzhnikov *et al.*, 2005). The Pareto fronts are plotted in Figure 3.1. The full problem required 1744 calls to the function to obtain 20 solutions while this number was reduced to 1324 for the reduced problem. Fixing one variable results in a significant reduction of the number of function calls. However, it causes degradation in the quality of the Pareto front. This optimisation problem proves that fixing arbitrarily the non-significant variable to its mean is not always an appropriate choice. It is worth

noting that for all solutions of the full optimisation problem, the constraint on the minimum bound for x_3 is activated: $x_3 = 1$.

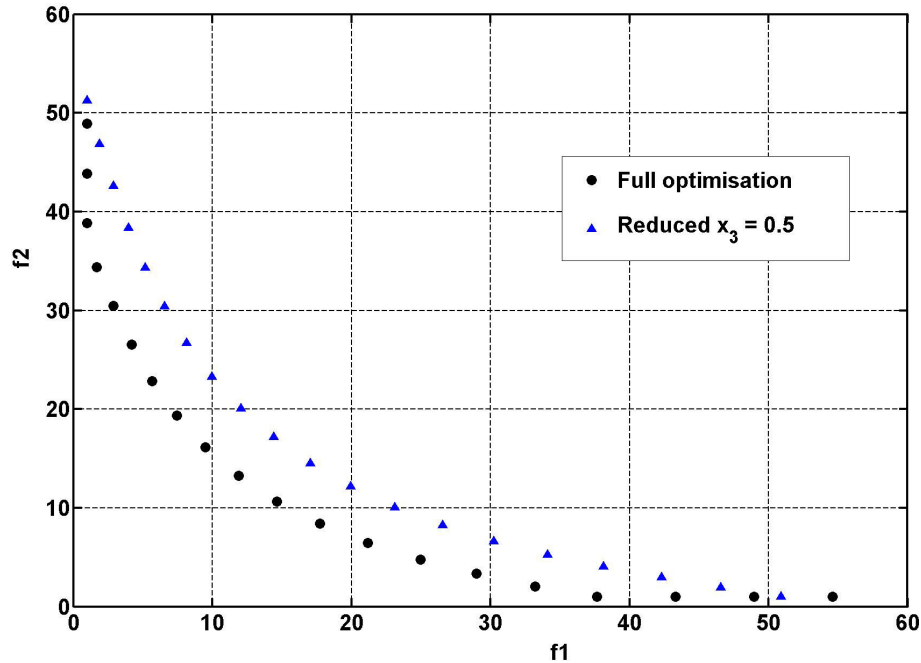


Figure 3.1: Pareto fronts obtained with full optimisation and reduced problem for $x_3 = 0.5$

In this approach, it is proposed to build a self-organizing map (SOM) of the problem to check whether there is a preferred value to fix the non-significant variables. The SOM is built with the Matlab toolbox developed at Helsinki University (Vesanto et al., 2000; Vesanto, 1999) and is shown in Figure 3.2. Feasible regions are marked with white dots while black dots are used for non-feasible ones. The examination of the SOM shows that objectives f_1 and f_2 are conflicting and cannot be minimised together since their component maps exhibit opposite pattern. It also shows that small values of f_1 or f_2 can be reached for small and large values for x_3 . This can be seen on Figure 3.1 where the discrepancy between both Pareto fronts is smaller at the ends of the curves. However, for the region where both objectives take intermediate values (marked with a circle in Figure 3.2), x_3 takes large values. Such considerations show that a large value for x_3 should be preferred in order to improve the capture of the Pareto surface. The reduced optimisation problem is solved for $x_3 = 0.75$. The corresponding Pareto front is plotted in Figure 3.3 and a summary of the number of evaluations for each optimisation

run is given in Table 3.2. The SOM indicates that a high value for x_3 would help capture parts of the Pareto front. Unfortunately, it appears difficult to derive accurately a preferred value for the variable that is to be frozen.

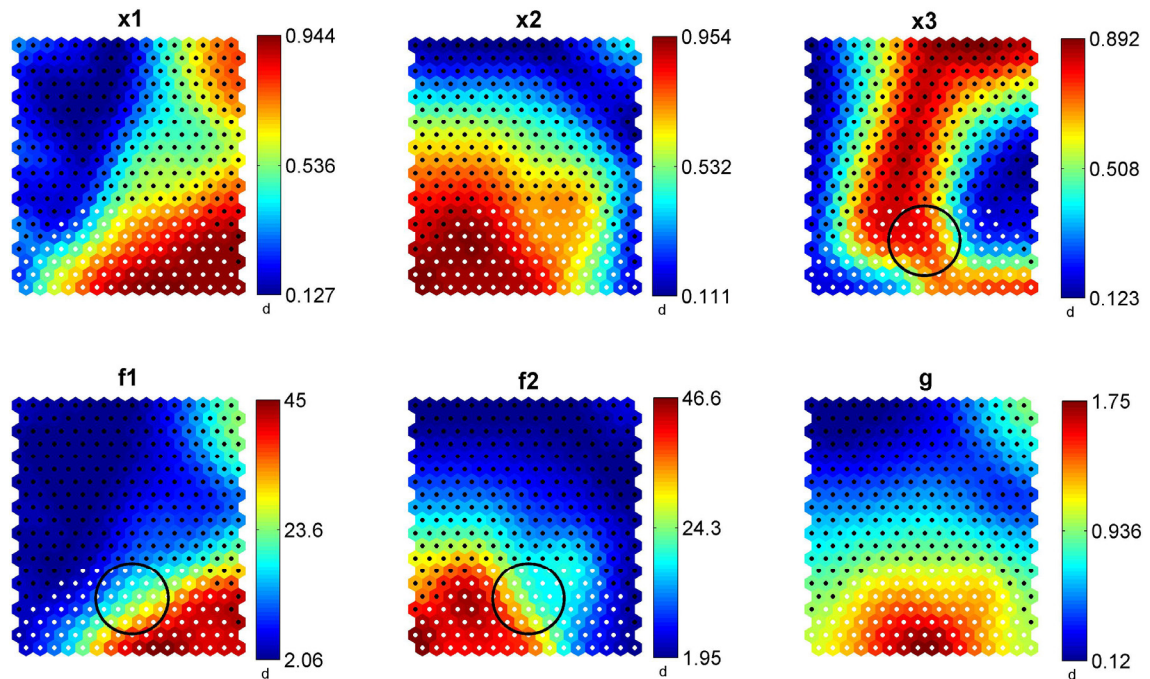


Figure 3.2: Self-organizing map for test optimisation problem

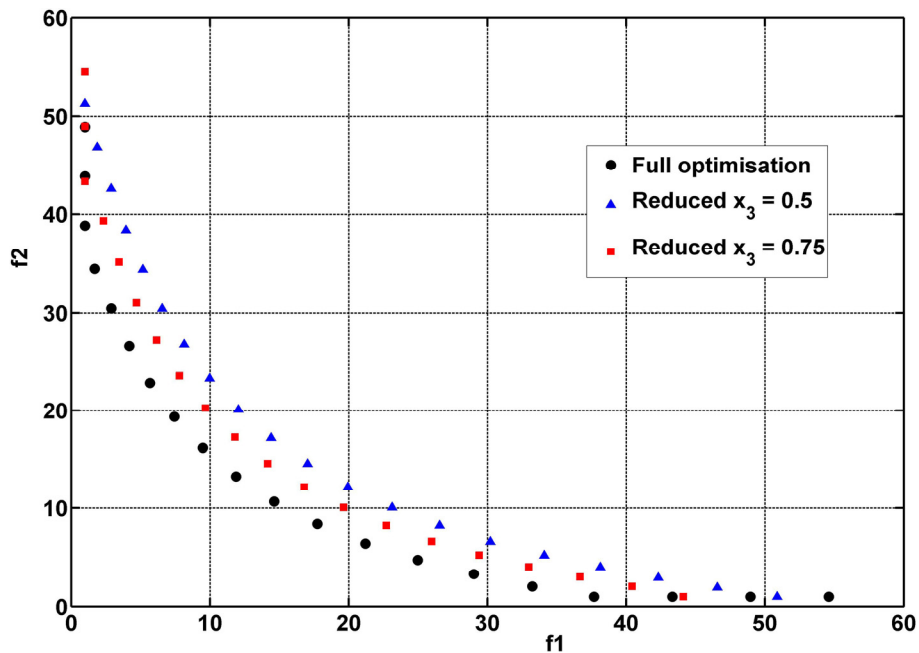


Figure 3.3: Pareto fronts obtained with full optimisation, reduced problem for $x_3 = 0.5$ and $x_3 = 0.75$

	All inputs	$x_3 = 0.5$	$x_3 = 0.75$
Function evaluations	1744	1324	1290
Pareto points obtained	20	20	20

Table 3.2: Computational cost comparison between full optimisation and reduced problem for $x_3 = 0.5$ and $x_3 = 0.75$

However, the SOM should be used to check whether there is a preferred value by studying the structure of the data (inputs, objectives and constraints). The SOM can be built on the same sample used for obtaining the sensitivity indices so that no extra computational effort is required. A summary of the approach for reducing the dimensionality of the optimisation problem is given in Figure 3.4.

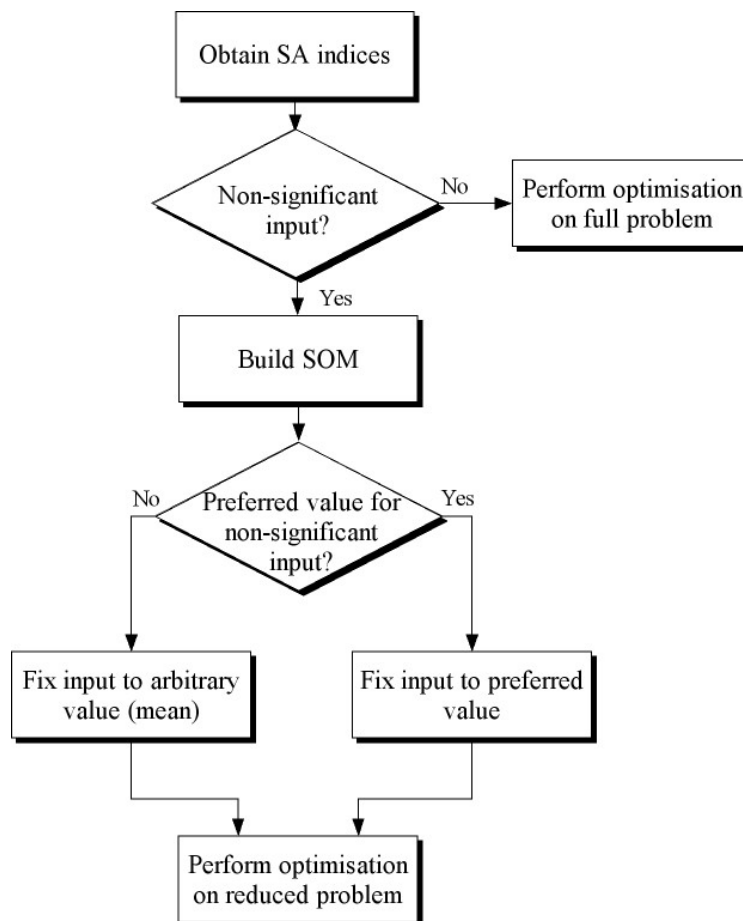


Figure 3.4: Global sensitivity analysis approach flowchart

3.2 Pareto sensitivity analysis

The Pareto sensitivity can play a predominant role in the decision making process because it can provide essential information about optimum solutions.

3.2.1 Understanding the trade-offs on the Pareto surface

As discussed in chapter 1, in the *a posteriori articulation of preferences*, the decision-making process occurs after the optimisation process. An initial set of solutions is obtained and presented to the designer. All these designs are solutions of the multi-objective optimisation problem. Therefore, the selection of the final design must be based on preferences that are not included in the mathematical formulation of the problem. The decision-maker can make considerations with regards to other disciplines, previous design experience and so forth. From the analysis of the initial set of designs, the designer can get a better understanding of the solutions that can be obtained. At this point, he/she can even decide to attribute a higher priority to one or more objectives. If the decision-maker cannot select a design from the initial set of solution, he/she might be interested in exploring the Pareto front further.

As part of the decision process, the designer would benefit from having a better insight into the local trade-offs between objectives on the Pareto surface. It would give him/her essential information about other solutions on the Pareto front because relationships between Pareto solutions in the objective space and their corresponding point in the design space are not easy to comprehend.

3.2.2 Requirements for sensitivity analysis on the Pareto surface

As outlined in chapter 2, most of the methods attempting to obtain the SA of the Pareto surface have been developed for simple and specific cases, which are not representative of an MDO optimisation problem. Approximations techniques are interesting since they not only provide the sensitivity of the Pareto surface, but they also enable to explore locally other possible solutions. Here are outlined some desirable properties for a Pareto approximation which are not present in the methods found in literature.

- The approximation should be developed in a general case and should not be limited by the shape of the Pareto front, the number of variables, constraints or objectives.
- The derivation of the approximation should aim to limit the number of model evaluations and come at a little cost.
- The approximation should provide a mean to relate the information derived on the Pareto surface in the objective space to the design space. This should avoid running optimisation that could be costly.
- Recommendations on the extent of validity of the approximation should be available.

3.3 Summary

In this chapter, both components of the SA approach proposed for multi-objective optimisation are described. First, it is explained how global SA (VBM) is used to identify inputs with negligible effect on the output variance and fixed them to a specific value. It is proposed to use the data sample generated for the SA to build a SOM and ensure whether there is preferred value for the non-significant variables. Then, it is emphasised how understanding the local trade-offs on the Pareto surface can provide the designer with essential information to make the decision on the final solution of the problem. The lack of desirable features of the available methods is outlined.

In chapter 4, a new improved Pareto approximation is derived to address the requirements specified in this chapter.

Chapter 4

Obtaining local sensitivity on the Pareto surface

The design of complex systems is often the solution an optimisation problem, which involves different disciplines and where all design criteria have to be optimised simultaneously. Mathematically this problem can be reduced to a vector optimisation problem. The solution of this problem is not unique and is represented by a Pareto surface in the objective functions space. Once a candidate Pareto solution is obtained, it may be very useful for the decision-maker to be able to perform a quick local approximation in its vicinity for sensitivity analysis. The literature review presented in chapter two shows that there exist no general equations for approximating locally the Pareto surface. Most attempts are only valid in a particular case and necessitate more discipline analysis.

In this chapter, a new method for obtaining a local approximation the Pareto surface is presented. In the first section, a formulation of the multi-objective optimisation problem and a definition of the Pareto concept are given. In section 2, a promising technique is reviewed in details and clarifications of its formulation in the linear case are made. The formulation is extended to obtain directly a quadratic approximation of the Pareto surface. The limitations of the existing method are pointed

out in an analytical example. Then, a more theoretically sound method is derived to obtain linear and quadratic approximations in the general case and an approach to assess the extent of their validity is discussed in section 3. Section 4 concerns the detection of non-differentiable Pareto points and section 5 summarises the entire Pareto analysis procedure. Finally, the potential of the approach is illustrated through a few analytical examples and conclusions are drawn.

4.1 Multi-objective optimisation problem formulation and assumptions

A more general and complete formulation of the multi-objective optimisation problem is introduced. Then, concepts and assumptions that are necessary to understand and discuss existing approximation techniques of interest are presented.

It is assumed that an optimisation problem is described in terms of a design variable vector $\mathbf{x} = (x_1, x_2, \dots, x_N)^T$ in the design space $\mathbf{X} \subset \mathbb{R}^N$. A function $\mathbf{f} \in \mathbb{R}^M$ evaluates the quality of a solution by assigning it to an objective vector $\mathbf{y} = (y_1, y_2, \dots, y_M)^T$, where $y_i = f_i(\mathbf{x})$, $f_i: \mathbb{R}^N \rightarrow \mathbb{R}^1$, $i = 1, \dots, M$ in the objective space $\mathbf{Y} \subset \mathbb{R}^M$. Thus, \mathbf{X} is mapped onto \mathbf{Y} by $\mathbf{f}: \mathbf{X} \rightarrow \mathbf{Y}$. A multi-objective optimisation problem may be formulated in the following form:

$$\min_{\mathbf{x}} [\mathbf{y}(\mathbf{x})] \quad (4.1)$$

Subject to L inequality constraints:

$$g_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, L \quad (4.2)$$

which may also include equality constraints.

The feasible design space \mathbf{X}^* is defined as the set $\{\mathbf{x} \mid g_i(\mathbf{x}) \leq 0, j = 1, 2, \dots, L\}$. A feasible design point is a point that does not violate any constraint. The feasible criterion (objective) space \mathbf{Y}^* is defined as the set $\{\mathbf{Y}(\mathbf{x}) \mid \mathbf{x} \in \mathbf{X}^*\}$. A design vector \mathbf{a} ($\mathbf{a} \in \mathbf{X}^*$) is called a Pareto optimum if, and only if, there exists no vector $\mathbf{b} \in \mathbf{X}^*$ such that

$y_i(\mathbf{b}) \leq y_i(\mathbf{a})$, for all $i = 1, \dots, M$ and $y_j(\mathbf{b}) < y_j(\mathbf{a})$ for $1 \leq j \leq M$. Henceforth it is assumed that all vectors are considered in the appropriate Euclidean spaces.

A constraint g_i is said to be active at a Pareto point \mathbf{x}^* of the design space \mathbf{X} if a strict equality holds at this point, i.e. $g_i(\mathbf{x}) = 0$ (Tappeta *et al.*, 1999, 2000). Let us note the set of active constraints (equation (4.2)) as \mathbf{G} . At the given point \mathbf{x}^* of the design feasible space \mathbf{X}^* it means that

$$\mathbf{G}(\mathbf{x}^*) = \mathbf{0}. \quad (4.3)$$

Two main assumptions are made in this chapter.

First assumption: $\mathbf{f} \in C^2(\mathbb{R}^M)$ and $\mathbf{G} \in C^2(\mathbb{R}^I)$.

It follows that locally the constraints can be written in the linear form

$$\mathbf{J}(\mathbf{x} - \mathbf{x}^*) = \mathbf{0}, \quad (4.4)$$

where \mathbf{J} is the Jacobian of active constraints at \mathbf{x}^* : $\mathbf{J} = (\nabla \mathbf{G})^T$.

Second assumption: The gradients of the active constraints are linearly independent at the Pareto point \mathbf{x}^* .

Therefore, point \mathbf{x}^* is a regular point of the feasible space (Vincent and Grantham, 1981). Another important result is that matrix $(\mathbf{J}\mathbf{J}^T)$ is invertible (see Appendix A) and the projection matrix onto the hyperplane normal to all gradients of active constraints can then be defined by

$$\mathbf{P} = \mathbf{I} - \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}. \quad (4.5)$$

4.2 Discussion on an existing technique

4.2.1 Obtaining the sensitivity coefficients on the Pareto surface

In this section, it is intended to present an existing technique that was essential to the development of the general Pareto surface approximation, presented in the next section. This method is described in Tappeta and Renaud (1999, 2001) and Tappeta *et al.* (2000). Based on the calculation of sensitivity coefficient on the Pareto surface, a linear approximation is derived. This information on the linear approximation is used together with Pareto data generated in the neighbourhood of the Pareto point to derive a quadratic approximation. This approach implies running locally more optimisation loops to obtain the quadratic approximation.

In this approach, it is assumed that the set of active constraints does not change in the vicinity of the Pareto point under study. The feasible direction with the greatest improvement of objective f_i is obtained by projecting its gradient onto the hyperplane normal to the gradients of all constraints:

$$d_{f_i} = \mathbf{P}(-\nabla f_i). \quad (4.6)$$

This method is known as gradient-projection method and was first introduced by Rosen (1960). It is illustrated in Figure 4.1.

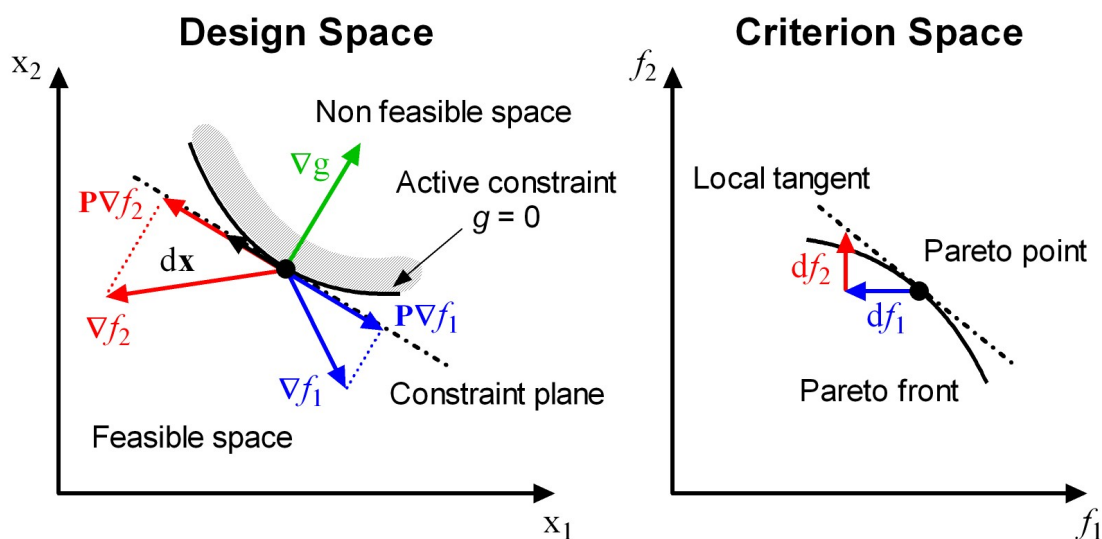


Figure 4.1: Gradient projection method

To obtain the Pareto sensitivity df_i/df_j along the feasible direction with the greatest improvement of objective f_j , objectives f_i and f_j are differentiated with respect to \mathbf{x} :

$$df_i = \mathbf{d}_{f_i}^T \mathbf{d}\mathbf{x} \quad (4.7)$$

$$df_j = \mathbf{d}_{f_j}^T \mathbf{d}\mathbf{x} \quad (4.8)$$

and equation (4.8) is solved for \mathbf{x} as follows:

$$\mathbf{d}\mathbf{x}|_j = \left[\mathbf{d}_{f_j}^T \mathbf{d}_{f_j} \right]^{-1} \mathbf{d}_{f_j} df_j \quad (4.9)$$

In equation (4.9), $\mathbf{d}\mathbf{x}|_j$ represents the direction in the design space along the feasible direction with the greatest improvement of objective f_j to accommodate for a change df_j of objective f_j . It is also important to note that $\mathbf{d}_{f_j}^T \mathbf{d}_{f_j}$ is a scalar and therefore the inverse is always defined. Substituting equation (4.9) in equation (4.8), the trade-off between objective f_i and f_j along the direction $\mathbf{d}\mathbf{x}|_j$ can be obtained as

$$\frac{df_i}{df_j} = \mathbf{d}_{f_i}^T \left[\mathbf{d}_{f_j}^T \mathbf{d}_{f_j} \right]^{-1} \mathbf{d}_{f_j} \quad (4.10)$$

and are called sensitivity coefficients. A similar analysis can be conducted for all $i = 1, \dots, M$ to obtain the sensitivity information for each objective along the same direction $\mathbf{d}\mathbf{x}|_j$. Tappeta *et al.* (1999,2000) propose to represent the sensitivity information in the form of a matrix, where the j^{th} row corresponds to the trade-offs required on all objectives for a unit improvement in the j^{th} objectives. This matrix is called the trade-off matrix and is given by:

$$\mathbf{T} = \begin{bmatrix} 1 & \frac{df_2}{df_1} & \dots & \frac{df_M}{df_1} \\ \frac{df_1}{df_2} & 1 & \dots & \frac{df_M}{df_2} \\ \dots & \dots & \dots & \dots \\ \frac{df_1}{df_M} & \frac{df_2}{df_M} & \dots & 1 \end{bmatrix} \quad (4.11)$$

There is a trade-off between two objectives along a particular direction if the corresponding off-diagonal element is negative. If the matrix element is positive, it means that both objectives can be improved together. Note that on the same row, at least one element is negative otherwise, all objectives could be locally improved together which would contradict the fact that the point under study is a Pareto solution of the problem defined by equations (4.1) and (4.2).

A local linear approximation of the Pareto surface of the form

$$f_p = f_p^* + \sum_{\substack{i=1 \\ i \neq p}}^M \frac{df_p}{df_i} \Delta f_i \quad (4.12)$$

is proposed, where df_p/df_i are based on the elements of \mathbf{T} . The formulation and notation used in Tappeta *et al.* (2000) can be a little confusing at first. Indeed, the df_p/df_i in equation (4.12) do not correspond to the elements of the trade-off matrix \mathbf{T} . To understand this, let us consider the following simple analytical multi-objective optimisation problem:

Unit-sphere optimisation problem

$$\text{Minimise:} \quad \mathbf{f} = (x, y, z)^T \quad (4.13)$$

$$\begin{aligned} \text{Subject to:} \quad & g(\mathbf{x}) = 1 - x^2 - y^2 - z^2 \leq 0, \\ & x > 0, \\ & y > 0, \\ & z > 0. \end{aligned} \quad (4.14)$$

The design space and objective space coincide in this example. It is easy to see that the Pareto surface corresponds to the part of the unit sphere in the first quadrant and is represented by the following formula:

$$z = \sqrt{1 - x^2 - y^2}. \quad (4.15)$$

The exact first order derivatives on the Pareto surface can be derived analytically as follows:

$$\begin{aligned} \frac{\partial f_3}{\partial f_1} &= \frac{\partial z}{\partial x} = \frac{-x}{\sqrt{1 - x^2 - y^2}} \\ \frac{\partial f_3}{\partial f_2} &= \frac{\partial z}{\partial y} = \frac{-y}{\sqrt{1 - x^2 - y^2}}. \end{aligned} \quad (4.16)$$

According to Tappeta *et al.* (2000), the elements of the last column of the trade-off matrix \mathbf{T} on the Pareto surface can be derived for any points as follows:

$$\mathbf{J} = [-2x \quad -2y \quad -2z], \quad (4.17)$$

$$\mathbf{P} = \begin{bmatrix} y^2 + z^2 & -xy & -xz \\ -xy & x^2 + z^2 & -yz \\ -xz & -yz & x^2 + y^2 \end{bmatrix}. \quad (4.18)$$

Equation (4.10) gives:

$$\begin{aligned} T_{13} &= \frac{df_3}{df_1} = \frac{-xz}{y^2 + z^2} = \frac{(1 - x^2 - y^2)(-x)}{(1 - x^2)\sqrt{1 - x^2 - y^2}}, \\ T_{23} &= \frac{df_3}{df_2} = \frac{-yz}{x^2 + z^2} = \frac{(1 - x^2 - y^2)(-y)}{(1 - y^2)\sqrt{1 - x^2 - y^2}}. \end{aligned} \quad (4.19)$$

It is obvious that the elements of the trade-off matrix do not correspond to the derivatives on the Pareto surface. The linear approximation obtained with the derivative taken as the element of the trade-off matrix is given in Figure 4.2. This problem can be explained by the fact that in the trade-off matrix \mathbf{T} only elements in the same row relate to each other. Each row corresponds to a particular direction in the design space. Therefore, sensitivity coefficients df_i/df_j can only be compared if they are associated with the same direction. The next section describes in more details how the linear approximation can be obtained from the elements of \mathbf{T} . It also shows how to obtain a direct quadratic approximation of the Pareto surface.

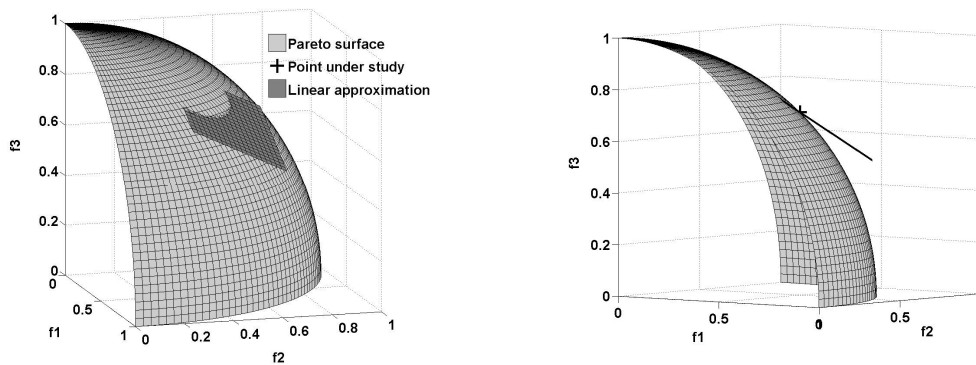


Figure 4.2: Unit-sphere optimisation problem – Erroneous linear approximation

Note that using directly equation (4.10) to obtain the derivative on the Pareto surface would be correct in the case of a two-objective optimisation problem.

4.2.2 Deriving a linear approximation from the sensitivity coefficients

Without loss of generality let us consider the first row of the trade-off matrix \mathbf{T} . One element indicates the appropriate change in the corresponding objective when objective f_1 is improved of one unit along its greatest feasible descent direction. As a result, for an infinitesimal improvement df_1 in f_1 along the direction defined by equation (4.9) in the design space, the new approximate Pareto point is given by:

$$\begin{bmatrix} f_1^* \\ f_2^* \\ \dots \\ f_m^* \end{bmatrix} \rightarrow \begin{bmatrix} f_1^* + d f_1 \\ f_2^* + \frac{d f_2}{d f_1} d f_1 \\ \dots \\ f_m^* + \frac{d f_m}{d f_1} d f_1 \end{bmatrix} \quad (4.20)$$

where $[f_1^* \ f_2^* \ \dots \ f_m^*]^T$ corresponds to the coordinates of the Pareto point under study in the objective space. Each row vector of the trade-off matrix defines a tangent vector to the Pareto surface. Therefore, the tangent plane to the Pareto surface can be obtained by finding the vector normal to all row vectors of the trade-off matrix \mathbf{T} . The normal vector \mathbf{N} can be found by solving the linear system of equations:

$$\mathbf{T} \cdot \mathbf{N} = \mathbf{0} \quad (4.21)$$

and the linear approximation(4.12) can be deduced from the equation of a surface given a normal vector:

$$[\Delta f_1 \ \Delta f_2 \ \dots \ \Delta f_M] \cdot \mathbf{N} = 0 \quad (4.22)$$

The correct linear approximation and the tangent direction corresponding to the row vectors of the trade-off matrix are shown in Figure 4.3.

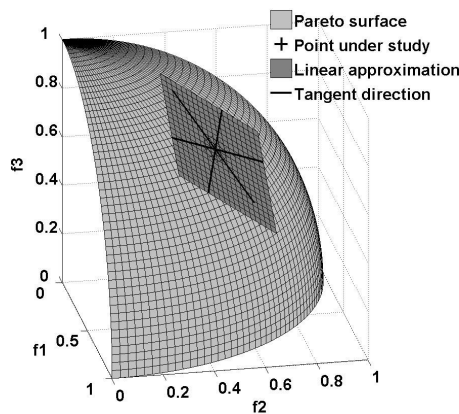


Figure 4.3: Unit-sphere optimisation problem – Linear approximation derived from tangent directions

4.2.3 Deriving a quadratic approximation from the sensitivity coefficients

Unlike the method proposed by Tappeta *et al.* (2000), in this section it is shown how a quadratic approximation of the Pareto surface can be derived without running further local optimisation. The derivation of the quadratic approximation is based on the same approach. In this case, when changing objective f_1 of an increment $d f_1$ along its greatest feasible descent direction in the design space, to the second order, all other objectives are modified as follows:

$$\begin{bmatrix} f_1^* \\ f_2^* \\ \dots \\ f_M^* \end{bmatrix} \rightarrow \begin{bmatrix} f_1^* + d f_1 \\ f_2^* + \frac{d f_2}{d f_1} d f_1 + \frac{1}{2} \frac{d^2 f_2}{d f_1^2} d f_1^2 \\ \dots \\ f_M^* + \frac{d f_M}{d f_1} d f_1 + \frac{1}{2} \frac{d^2 f_M}{d f_1^2} d f_1^2 \end{bmatrix} \quad (4.23)$$

In this case, all other objectives are locally approximated by a Taylor expansion up to the second order. The same method used for calculating the first order term of the trade-off matrix is used to calculate the second order term $d f_i/d f_j$ along the greatest feasible descent direction of objective f_j :

$$\frac{d^2 f_i}{d f_j^2} = \left[\mathbf{P} \nabla \left(\frac{d f_i}{d f_j} \right) \right]^T \cdot d \mathbf{x} \Big|_j \quad (4.24)$$

where $d \mathbf{x} \Big|_j$ is defined by equation (4.10). Note that now the directions are not tangent vectors but parametric quadratic curves locally tangent to the Pareto surface as shown in Figure 4.4. The idea is now to fit these parametric curves to a quadratic hypersurface and to derive its equation. For convenience, let us use the following notation:

$$(f_i)_j = \frac{d f_i}{d f_j} \quad \text{and} \quad (f_i)_{jj} = \frac{d^2 f_i}{d f_j^2}, \quad (4.25)$$

for the first and second order sensitivity coefficient defined by (4.10) and (4.24) respectively.

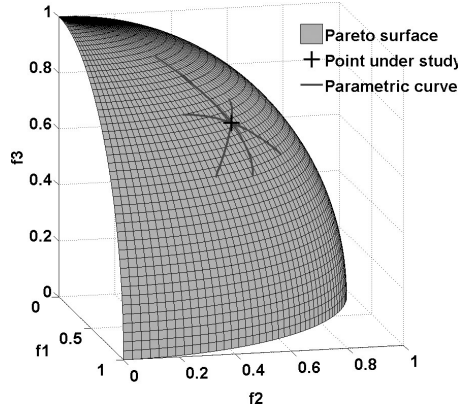


Figure 4.4: Unit-sphere optimisation problem – Parametric curves

In the case of the unit-sphere optimisation problem with three objectives, it is proposed to obtain a quadratic approximation for f_3 of the form:

$$\alpha_1(f_1 - f_1^*)^2 + \beta_1(f_1 - f_1^*) + \alpha_2(f_2 - f_2^*)^2 + \beta_2(f_2 - f_2^*) + (f_3 - f_3^*)^2 + \beta_3(f_3 - f_3^*) = 0. \quad (4.26)$$

Note that in equation (4.26) the coefficient for the second order term of objective f_3 is set to one, $\alpha_3 = 1$, to avoid the trivial solution where all coefficients are zero when it is fitted to the parametric curves defined previously. We then impose that each parametric curve must lie locally on the quadratic surface. Therefore, substituting (4.23) into (4.26) one finds:

$$\begin{aligned} & (\beta_1 + \beta_2(f_2)_1 + \beta_3(f_3)_1)df_1 + \\ & \left(\alpha_1 + \alpha_2((f_2)_1)^2 + \frac{1}{2}\beta_2(f_2)_{11} + \beta_3\frac{1}{2}(f_3)_{11} + ((f_3)_1)^2 \right)df_1^2 = 0. \end{aligned} \quad (4.27)$$

To satisfy equation (4.27) for any small improvements for objective f_1 , the coefficient of both first and second order terms must be equal to zero. Therefore, performing the

analysis for the two remaining objectives, a linear system of equations can be obtained to calculate the coefficient α_i and β_i :

$$\begin{bmatrix} 0 & 1 & 0 & (f_2)_1 & (f_3)_1 \\ 1 & 0 & ((f_2)_1)^2 & \frac{1}{2}(f_2)_{11} & \frac{1}{2}(f_3)_{11} \\ 0 & (f_1)_2 & 0 & 1 & (f_3)_2 \\ ((f_1)_2)^2 & \frac{1}{2}(f_1)_{22} & 1 & 0 & \frac{1}{2}(f_3)_{22} \\ 0 & (f_1)_3 & 0 & (f_2)_3 & 1 \\ ((f_1)_3)^2 & \frac{1}{2}(f_1)_{33} & ((f_2)_3)^2 & \frac{1}{2}(f_2)_{33} & 0 \end{bmatrix} \times \begin{bmatrix} \alpha_1 \\ \beta_1 \\ \alpha_2 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} 0 \\ -((f_3)_1)^2 \\ 0 \\ -((f_3)_2)^2 \\ 0 \\ -1 \end{bmatrix} \quad (4.28)$$

Note that there are six equations and five unknowns. Row 1, 2 and 3 correspond to the local linear approximation of the Pareto surface. Only two equations are necessary to ensure the local tangency in the linear approximation and therefore, one equation among row 1,2 and 3 can be removed.

After solving the reduced set of linear equations, for α_i and β_i , the equation for the quadratic approximation is obtained and values for objective f_3 can be derived for each new value of f_1 and f_2 by solving the following quadratic equation:

$$\Delta f_3^2 + \beta_3 \Delta f_3 + \underbrace{\alpha_1(f_1 - f_1^*)^2 + \beta_1(f_1 - f_1^*) + \alpha_2(f_2 - f_2^*)^2 + \beta_2(f_2 - f_2^*)}_{r} = 0. \quad (4.29)$$

This equation always has a solution because the three quadratic parametric curves satisfied (4.26). The solution is the obvious solution of a second order polynomial:

$$\Delta f_3 = \frac{-\beta_3 \pm \sqrt{\beta_3^2 - 4r}}{2}. \quad (4.30)$$

When $\sqrt{\beta_3^2 - 4r} \neq 0$, a couple of solutions, depending on the sign in equation (4.30) are obtained, but only one of them approximates the Pareto surface. A simple consideration helps decide on the right solution for Δf_3 . In particular, equation (4.30) must be valid

for the Pareto solution under study (f_1^*, f_2^*, f_3^*) . For this reason, when $(f_1, f_2) = (f_1^*, f_2^*)$ we must find $\Delta f_3 = 0$. For $(f_1, f_2) = (f_1^*, f_2^*)$, $r = 0$ and the expression for the solution is reduced to:

$$\Delta f_3 = \frac{-\beta_3 \pm |\beta_3|}{2} = 0 \quad (4.31)$$

- If $\beta_3 > 0$, to satisfy $\Delta f_3 = 0$ for $(f_1, f_2) = (f_1^*, f_2^*)$, we must have a positive sign so that $\Delta f_3 = \frac{-\beta_3 + \beta_3}{2} = 0$.
- If $\beta_3 < 0$, to satisfy $\Delta f_3 = 0$ for $(f_1, f_2) = (f_1^*, f_2^*)$, we must have a negative sign so that $\Delta f_3 = \frac{-\beta_3 - |\beta_3|}{2} = 0$.

Therefore,

$$\text{If } \beta_3 > 0 \quad f_3 = f_3^* + \frac{-\beta_3 + \sqrt{\beta_3^2 - 4r}}{2},$$

$$\text{If } \beta_3 < 0 \quad f_3 = f_3^* + \frac{-\beta_3 - \sqrt{\beta_3^2 - 4r}}{2}.$$

The resulting quadratic approximation obtained for the unit-sphere optimisation problem is given in Figure 4.5.

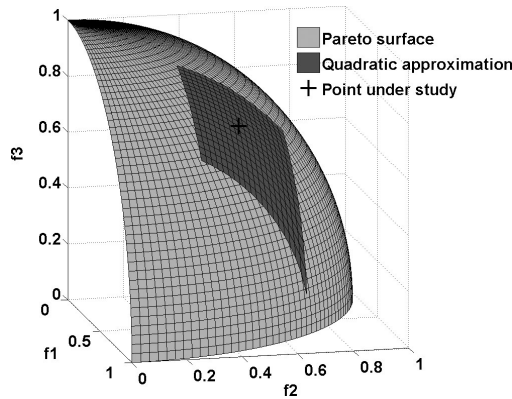


Figure 4.5: Unit-sphere optimisation problem – Quadratic approximation derived from parametric curves

Note that in this case, there is an ideal fit of the Pareto surface. It can be explained by the fact that when solved for α_i and β_i , equation (4.26) describes perfectly the surface of a sphere. This is a special case as the Pareto surface corresponds to the surface of a sphere. The unit-sphere optimisation example has been chosen for its analytical simple formulation with more than two objectives. Unlike the method proposed by Tappeta, this approach shows that it is possible to derive a linear and quadratic approximation from the sensitivity coefficients (4.10) without running locally more optimisation loops. Only gradients and Hessian of objectives and active constraints at the Pareto point under study are needed.

The implementation of a more general case with M objectives is obvious. Indeed, the quadratic approximation (4.26) requires solutions for $2M-1$ coefficients ($M-1$ alpha and M beta). From fitting the M quadratic parametric curves (4.23) to the quadratic approximation (4.26), M equations of the same type as (4.27) can be obtained. $M-1$ equations corresponding to the first order term df_1 and M to the second order term $d^2 f_1$ from equation (4.27) to constitute a system of linear equations similar to (4.28).

4.2.4 Limitations

In this section, limitations of the previous approach are presented in the case of a numerical test case. Let us consider the unit-sphere optimisation problem and add an extra constraint. The modified optimisation problem is as follows:

$$\text{Minimise: } \quad \mathbf{f} = (x, y, z)^T \quad (4.32)$$

$$\begin{aligned} \text{Subject to: } \quad & g_1(\mathbf{x}) = 1 - x^2 - y^2 - z^2 \leq 0, \\ & g_2(\mathbf{x}) = x - y = 0, \\ & x > 0, \\ & y > 0, \\ & z > 0. \end{aligned} \quad (4.33)$$

As in the unit-sphere optimisation problem, design and objective space coincide. The addition of an equality constraint imposes that all Pareto surface correspond to the

intersection of the surface of the sphere located in the first quadrant and the plane defined by g_2 . It results that the set of Pareto solutions for this three objectives optimisation problem is not a surface anymore (two degrees of freedom), but a curve (one degree of freedom) as shown in Figure 4.6.

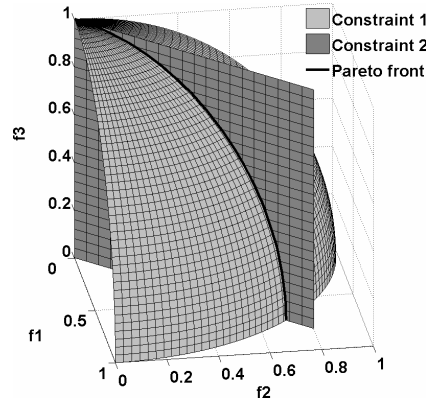


Figure 4.6: Modified unit-sphere optimisation problem – Limitations of the method

The analytical expression of the Pareto front can be obtained easily by substituting the equation corresponding to the added constraint into (4.15). It results that the Pareto front can be represented analytically by the parametric curve:

$$\begin{cases} x \\ y = x \\ z = \sqrt{1 - 2x^2} \end{cases} \quad (4.34)$$

If $\nabla \mathbf{f}$ is the matrix whose column vectors are the gradients of the M objective functions, Figure 4.6 corresponds to the situation where $\text{rank}(\mathbf{P}\nabla \mathbf{f}) < M - 1$. This situation is discussed in more detail in the next section. In this case, the Pareto surface cannot be locally approximated with a hyperplane of dimension $M - 1$. For this reason, the linear and therefore the quadratic approximations developed on the sensitivity coefficient show a lack of generalization. In next section, this issue is addressed by deriving linear and quadratic approximations in the general case.

4.3 New approximations of the Pareto surface

4.3.1 Derivation of linear and quadratic approximations in the general case

In this section, sound theoretical linear and quadratic approximations of the Pareto surface are derived in the general case. As in the previous section, it is assumed that constraints that are active at a particular Pareto point remain active in its vicinity. Thus, the sensitivity predicted at the given Pareto point is valid until the set of active constraints remains unchanged (Hernandez, 1995; Tappeta et al., 2000). Without loss of generality, let us assume that the first I constraints are active and the first Q of those correspond to inequalities ($Q \leq I \leq L$).

Let us assume that in the objective space \mathbf{Y} the Pareto surface is given by

$$S(\mathbf{y}) = 0 \quad (4.35)$$

and at the Pareto point $\mathbf{y}^* = \mathbf{f}(\mathbf{x}^*)$ function $S \in C^2(\mathbb{R}^1)$. In the general case, let us define the values of the gradient of any differentiable function F at point \mathbf{x}^* under constraints by the reduced gradient formula (Fletcher 1989):

$$\nabla F|_{S_l} = \mathbf{P} \nabla F, \quad (4.36)$$

where S_l is the hyperplane tangent to \mathbf{X}^* in the design space given by

$$S_l = \{\mathbf{x} \mid \mathbf{J}(\mathbf{x} - \mathbf{x}^*) = 0\} \quad (4.37)$$

and \mathbf{P} is projection matrix. Directional derivatives corresponding to equation (4.36) in the objective space are represented by

$$\frac{dF}{df_i} = \frac{dF}{d\mathbf{x}} \Big|_{S_l} \frac{d\mathbf{x}}{df_i}. \quad (4.38)$$

The first element of the product corresponds to the reduced gradient and in the second element $d\mathbf{x}$ represents the infinitesimal change in the design vector \mathbf{x} to accommodate the infinitesimal shift $d\mathbf{f}$ of the objective vector tangent to the Pareto surface. The last derivative in equation (4.38) can be represented via the gradients obtained in the design space \mathbf{X} . Assume that matrix $\mathbf{P}\nabla\mathbf{f}$, ($\mathbf{f} = (f_1, f_2, \dots, f_M)^T$) has $n_f < M$ linearly independent columns.

Here, it is to be noted that $n_f \neq M$. Indeed, we have:

$$d\mathbf{f} = (\mathbf{P}\nabla\mathbf{f})^T d\mathbf{x}. \quad (4.39)$$

$n_f = M$ would mean that for any $d\mathbf{f}$, in particular one where all objectives are improved together, there would exist a direction $d\mathbf{x}$ so that the set of active constraints remains unchanged. This contradicts the fact that the point under study is a Pareto solution.

Without loss of generality, we assume further that the first n_f components of $\mathbf{P}\nabla\mathbf{f}$ are linearly independent and represented by $\mathbf{P}\nabla\tilde{\mathbf{f}}$. Thus,

$$d\tilde{\mathbf{f}} = (\mathbf{P}\nabla\tilde{\mathbf{f}})^T d\mathbf{x}, \quad (4.40)$$

where all columns of matrix $\mathbf{P}\nabla\tilde{\mathbf{f}} \equiv (\mathbf{P}\nabla f_1, \dots, \mathbf{P}\nabla f_{n_f})$ are linearly independent. Now, let us write $d\mathbf{x}$ in the following form:

$$d\mathbf{x} = \mathbf{A}d\tilde{\mathbf{f}}. \quad (4.41)$$

Then, having multiplied both sides of equation (4.41) by $(\mathbf{P}\nabla\tilde{\mathbf{f}})^T$ and taking into account equation (4.40) we obtain

$$(\mathbf{P}\nabla\tilde{\mathbf{f}})^T \mathbf{A}d\tilde{\mathbf{f}} = d\tilde{\mathbf{f}}. \quad (4.42)$$

Hence,

$$\mathbf{A} = \mathbf{P}\nabla\tilde{\mathbf{f}}[(\mathbf{P}\nabla\tilde{\mathbf{f}})^T \mathbf{P}\nabla\tilde{\mathbf{f}}]^{-1}. \quad (4.43)$$

Thus, matrix \mathbf{A} is the right-hand generalized inverse matrix to $(\mathbf{P}\nabla\tilde{\mathbf{f}})^T$. It is possible to prove that the inverse matrix $[(\mathbf{P}\nabla\tilde{\mathbf{f}})^T \mathbf{P}\nabla\tilde{\mathbf{f}}]^{-1}$ is always non-singular because all the

vectors $\mathbf{P}\nabla f_i$, ($i = 1, \dots, n_f$) are linearly independent (see appendix). From the definition of matrix \mathbf{A} it follows that $(\mathbf{P}\nabla\tilde{\mathbf{f}})^T\mathbf{A} = \mathbf{I}$ and $\mathbf{A}_i^T\mathbf{P}\nabla\tilde{f}_j = \delta_{ij}$, where \mathbf{I} is the unit matrix and δ_{ij} is the Kronecker symbol. Hence, the system of vectors $\{\mathbf{A}_i\}$ ($i = 1, \dots, n_f$) creates the basis reciprocal to the basis of vectors $\{\mathbf{P}\nabla\tilde{f}_j\}$ ($j = 1, \dots, n_f$). From equation (4.43), it follows that $\mathbf{P}\mathbf{A} = \mathbf{A}$ and $d\mathbf{x}$ in equation (4.41) belongs to the tangent plane S_l at the Pareto point. Thus:

$$\frac{d\mathbf{x}}{d\tilde{\mathbf{f}}} = \mathbf{A} \equiv \mathbf{P}\nabla\tilde{\mathbf{f}}[(\mathbf{P}\nabla\tilde{\mathbf{f}})^T\mathbf{P}\nabla\tilde{\mathbf{f}}]^{-1} \quad (4.44)$$

and for any $i \leq n_f$

$$\frac{d\mathbf{x}}{df_i} = \mathbf{A}_i, \quad (4.45)$$

where $\mathbf{A} = (\mathbf{A}_1, \mathbf{A}_2, \dots, \mathbf{A}_{n_f})$. Then, from equations (4.36), (4.38) and (4.44), it follows that for any $i \leq n_f$

$$\frac{dF}{df_i} = (\mathbf{P}\nabla F)^T \mathbf{A}_i = \mathbf{A}_i^T \mathbf{P}\nabla F = \mathbf{A}_i^T \nabla F. \quad (4.46)$$

If $F = f_j$ ($n_f < j \leq M$), then we obtain the first order derivative of objective f_j with respect to objective f_i on the Pareto surface. Thus:

$$\frac{df_j}{df_i} = \mathbf{A}_i^T \nabla f_j \quad (0 \leq i \leq n_f, n_f < j \leq M). \quad (4.47)$$

It is important to note that this formula coincides with the formula of sensitivity of objective f_j with respect to objective f_i along the greatest feasible descent direction for objective f_i :

$$\frac{df_j}{df_i} = \frac{((\mathbf{P}\nabla f_j)^T \mathbf{P}\nabla f_i)}{((\mathbf{P}\nabla f_i)^T \mathbf{P}\nabla f_i)} \equiv \frac{((\nabla f_j)^T \mathbf{P}\nabla f_i)}{((\nabla f_i)^T \mathbf{P}\nabla f_i)}, \quad (4.48)$$

(Tappeta *et al.*, 1999, 2000, 2001) if and only if either $n_f = 1$ or the vectors $\mathbf{P}\nabla\tilde{\mathbf{f}}$ create an orthogonal basis. In this case, the matrix $[(\mathbf{P}\nabla\tilde{\mathbf{f}})^T\mathbf{P}\nabla\tilde{\mathbf{f}}]$ is diagonal. In particular, in the case of a two-objective optimisation these formulas always coincide since $n_f = 1$.

On the Pareto surface in the objective space the operator of the first derivative can be defined by:

$$\frac{d}{df_i} = \mathbf{A}_i^T \nabla. \quad (4.49)$$

By applying this operator to the first order derivative found previously, one can obtain the reduced Hessian as follows:

$$\frac{d^2 F}{df_i df_j} = \mathbf{A}_i^T \nabla (\mathbf{A}_j^T \nabla F) \quad (0 \leq i, j \leq n_f). \quad (4.50)$$

Thus, the Pareto surface can be locally represented as a linear hyperplane:

$$\sum_{i=1}^{n_f} \frac{dS}{df_i} \Delta f_i = 0 \quad (4.51)$$

or a quadratic surface:

$$\sum_{i=1}^{n_f} \frac{dS}{df_i} \Delta f_i + \frac{1}{2} \sum_{j,k=1}^{n_f} \frac{d^2 S}{df_j df_k} \Delta f_j \Delta f_k = 0, \quad (4.52)$$

where $\Delta \mathbf{f} = \mathbf{f} - \mathbf{f}(\mathbf{x}^*)$. Approximations (4.51) and (4.52) can be rewritten with respect to the trade-off relations between the objective functions as follows

$$f_p = f_p^* + \sum_{i=1}^{n_f} \frac{df_p}{df_i} \Delta f_i \quad (p = n_f + 1, \dots, M) \quad (4.53)$$

and

$$f_p = f_p^* + \sum_{i=1}^{n_f} \frac{df_p}{df_i} \Delta f_i + \frac{1}{2} \sum_{j,k=1}^{n_f} H_{jk}^{(p)} \Delta f_j \Delta f_k \quad (p = n_f + 1, \dots, M), \quad (4.54)$$

where $H_{jk}^{(p)} = \frac{d^2 f_p}{df_j df_k}$.

A quadratic approximation where the reduced Hessian matrix H_{ij} is evaluated with a least-squared minimisation using the Pareto set generated around the original Pareto point has been derived previously (Tappeta *et al.*, 1999, 2000). However, in preliminary design such evaluation can be unsuitable because it would require generating more Pareto points in the vicinity of the point under study. Instead, the local determination of the reduced Hessian using equation (4.50) is more accurate and is entirely based on the value of the objective and constraint gradients with respect to the independent design variables. It is important to note that in contrast to the approximations developed earlier (Tappeta *et al.*, 1999, 2000, 2001), the ones derived in this paper precisely correspond to the first two terms of the Taylor expansion in the general case.

Let us consider again the unit-sphere optimisation problem as defined by (4.13) and (4.14). According to the method developed in this section,

$$\mathbf{A} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -x/z & -y/z \end{bmatrix}. \quad (4.55)$$

One can easily ensure that using equation (4.47) we obtain the exact first order derivatives (equation (4.16)). In the same way, it is also possible to verify that the exact second order derivatives can be obtained with equation (4.50). The linear and quadratic approximation are shown in Figure 4.7 and Figure 4.8 respectively. The relative error in predicting f_3 is defined by

$$\text{Relative error} = \frac{|f_3 - f_3^{\text{approx}}|}{f_3} \quad (4.56)$$

and is given in Figure 4.9. The analysis of the relative error gives an indication on how quick the approximations depart from the Pareto surface. As it can be expected, the quadratic approximation performs much better than the linear approximation.

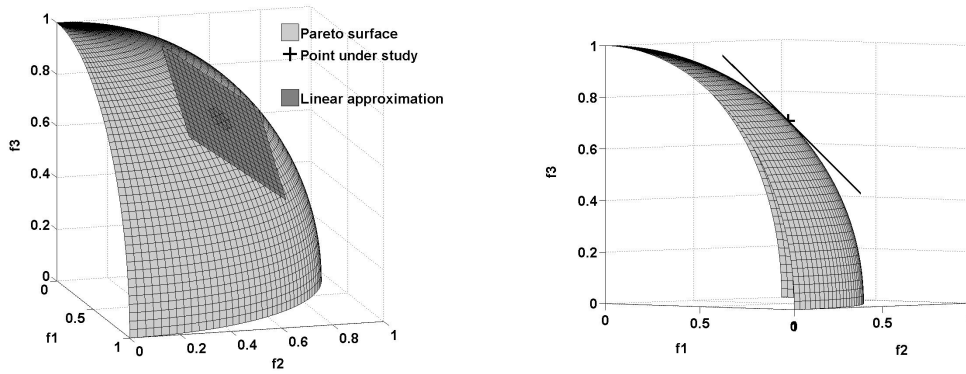


Figure 4.7: Unit-sphere optimisation problem – New linear approximation

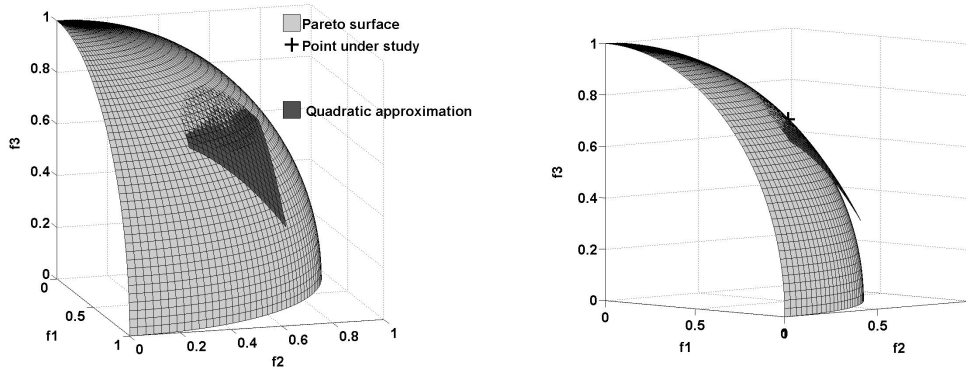


Figure 4.8: Unit-sphere optimisation problem – New quadratic approximation

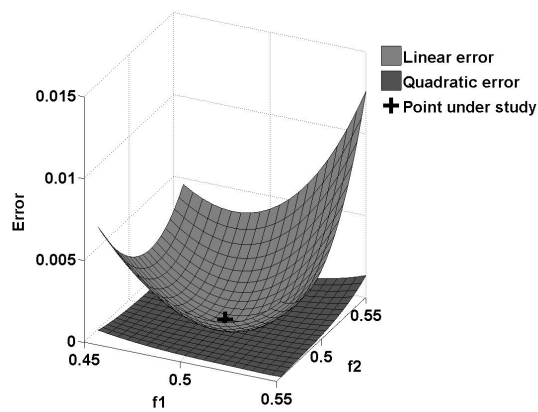


Figure 4.9: Unit-sphere optimisation problem – Relative error in predicting f_3

Let us considered here the modified unit-sphere optimisation problem defined by equations (4.32) and (4.33), which was used to show the limitations of the method presented in Tappeta *et al* (1999, 2000, 2001). In this example, the new linear and quadratic approximations are derived as follows:

$$\mathbf{J} = \begin{bmatrix} -2x & -2y & -2z \\ 1 & -1 & 0 \end{bmatrix} \quad (4.57)$$

$$\mathbf{P} = \frac{1}{2} \times \begin{bmatrix} z^2 & z^2 & -2zx \\ z^2 & z^2 & -2zx \\ -2zx & -2zx & 4x^2 \end{bmatrix} \quad (4.58)$$

It is clear that

$$\text{rank}(\mathbf{P}\nabla\mathbf{f}) = 1. \quad (4.59)$$

Therefore, in equations (4.53) and (4.54), $n_f = 1$ and $p = 2, 3$. As a result, matrix \mathbf{A} is reduced to a column vector:

$$\mathbf{A} = \begin{bmatrix} 1 \\ 1 \\ \frac{-2x}{\sqrt{1-2x^2}} \end{bmatrix} \quad (4.60)$$

and the first order derivatives on the Pareto front are given by:

$$\frac{df_2}{df_1} = \mathbf{A}^T \nabla f_2 = 1, \quad (4.61)$$

$$\frac{df_3}{df_1} = \mathbf{A}^T \nabla f_3 = \frac{-2x}{\sqrt{1-2x^2}}. \quad (4.62)$$

The resulting linear approximations are:

$$f_2 = y = y^* + \Delta x, \quad (4.63)$$

$$f_3 = z = z^* + \frac{-2x}{\sqrt{1-2x^2}} \Delta x \quad (4.64)$$

and clearly coincide with the derivatives of the analytical description of the Pareto front given with equation (4.34). The exact second order derivative can also be obtained using (4.50).

Even if very simple, this example shows that unlike the method proposed in Tappeta *et al* (2000), the new formulation is not limited in the case where $\text{rank}(\mathbf{P}\nabla\mathbf{f}) < M-1$ and does not require any further local approximation. Moreover, the derivative obtained, corresponds exactly to the terms of the Taylor expansion on the Pareto surface.

4.3.2 Local trade-off study

The first order derivatives df_p/df_i provide us with first order sensitivity of an objective f_p along the feasible descent direction of an objective f_i when all other objectives are kept constant. It is to be noted here that all the derivatives df_p/df_i are non-positive for all $i = 1, \dots, n_f$. Otherwise, two objectives could be locally improved together, which would contradict the fact that the point under study is a Pareto solution. The local approximations of the Pareto surface can be used to study the local sensitivity of a Pareto solution. As in a real-life problem it can be time and computationally expensive to obtain a single Pareto solution, local approximate solutions around a Pareto point can be obtained using either equation (4.53) or (4.54) without additional full-run computations. As discussed above, it can be very beneficial to the decision maker if s/he is able to perform a quick sensitivity analysis of the solution obtained to understand the trade-offs between different objectives. It has to be emphasized that in the multi-objective case, the change of one objective does not fully determine the changes of the others. If the designer freezes all objectives except two or three, it is then possible to obtain information, which is useful for understanding the trade-off between the selected objectives. The analysis of solutions around a Pareto point allows the

decision maker to correct locally the solution with respect to additional preferences, which can be articulated after a first solution is obtained. In the design practice, the knowledge for further improvement of some objectives at the expense of local degradation of some other objectives is important as it give the designer room for finding the most appropriate solution to the problem.

Another important feature of local Pareto analysis is that when investigating the vicinity of a Pareto solution in criterion space, the decision maker can immediately relate the new approximate solution to the corresponding approximate point in the design space using equation (4.41). The designer can then ensure that the changes on the design vector required to accommodate for the local changes in the Pareto solution satisfy his/her preferences.

4.3.3 Extent of validity of the local Pareto approximations

As part of the trade-off analysis, the decision maker is able to analyze possible violations of the constraints. Representations given by equations (4.53) and (4.54) are only local approximations and there is a question on the range of $\Delta\mathbf{x}$ where the approximations are valid. In the framework of a local analysis, giving a strict answer to this question is not possible. However, the reliable range of the variation of $\Delta\mathbf{x}$ can be evaluated qualitatively by comparing the solutions obtained by the linear and quadratic approximations. It is reasonable to expect that the approximations are suitable as long as the difference between them remains small.

In the sensitivity analysis procedure, due to a perturbation $\delta\mathbf{f}$ and the appropriate displacement $\delta\mathbf{x}$, some constraints, which are inactive at point \mathbf{x}^* , can become either violated or active. The exact verification of the constraints validation may require running more discipline analysis and therefore may be time consuming. To investigate this problem, it was suggested to obtain a local linear approximation of the inactive constraints to study the degree of constraint violation (Zhang W.H., 2003a, 2003b). The approach developed in the previous section can be used to obtain the appropriate linear and quadratic approximations for non-active inequality constraints in any direction:

$$\mathbf{g}_k(\mathbf{x}) = \mathbf{g}_k(\mathbf{x}^*) + \sum_{i=1}^{n_f} \frac{d\mathbf{g}_k}{df_i} \Delta f_i, \quad (I < k \leq L), \quad (4.65)$$

$$\mathbf{g}_k(\mathbf{x}) = \mathbf{g}_k(\mathbf{x}^*) + \sum_{i=1}^{n_f} \frac{d\mathbf{g}_k}{df_i} \Delta f_i + \frac{1}{2} \sum_{i,j=1}^{n_f} \frac{d^2\mathbf{g}_k}{df_i df_j} \Delta f_i \Delta f_j. \quad (4.66)$$

These approximations can be used to verify that inactive constraints remain inactive at a new approximate Pareto point. Similarly, equation (4.41) can be used to ensure that inactive constraints on the bounds of variables remain inactive within the approximation. Therefore, when performing the local Pareto analysis, the decision maker must ensure that the following constraints are satisfied:

$$\begin{aligned} & \bullet \quad \mathbf{g}_k(\mathbf{x}^*) + \sum_{i=1}^{n_f} \frac{d\mathbf{g}_k}{df_i} \delta f_i + \frac{1}{2} \sum_{i,j=1}^{n_f} \frac{d^2\mathbf{g}_k}{df_i df_j} \delta f_i \delta f_j \leq 0 \quad (I < k \leq L), \\ & \bullet \quad \mathbf{x}_{\min} \leq \mathbf{x} + \delta \mathbf{x} \leq \mathbf{x}_{\max}, \\ & \bullet \quad \frac{\frac{1}{2} \sum_{j,k=1}^{n_f} H_{jk}^{(p)} \delta f_j \delta f_k}{\sum_{i=1}^{n_f} \frac{df_p}{df_i} \delta f_i} < \alpha. \end{aligned} \quad (4.67)$$

The last point represents the constraint on the discrepancy between the linear and quadratic approximation of the Pareto surface. In the first case, the contribution of the second order term is compared to the value of objective f_p while in the second case, it must be smaller than a fraction of the first order contribution. The choice of the constraint to handle the discrepancy between linear and quadratic approximation is left to the decision maker.

4.4 Identification of non-differentiable Pareto points

In general, there is no guarantee that the Pareto surface is smooth. If the designed Pareto solution appears at a point of lack of smoothness, the approximations

derived above are not formally valid. In such a case, a substantial discrepancy can appear between the first and second order approximations in the nearest vicinity of the point under study.

4.4.1 Non-differentiability concept on the Pareto surface

The concept of non-differentiable Pareto point concerns the lack of smoothness of the Pareto surface in the criterion space. In the case of two objectives, it corresponds to a sudden change of slope as illustrated in Figure 4.10. For higher dimensional objective space, a more general way to characterize non-differentiability is related to local changes of the normal to Pareto surface (Zhang, W.H., 2003a, 2003b).

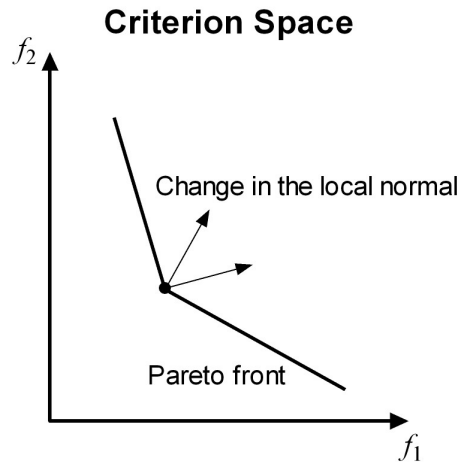


Figure 4.10: Non-differentiable Pareto point concept

The idea to identify non-differentiability of a point on the Pareto front is to detect changes in the local normal based on information available in the design space. A method to reveal such non-smooth points is based on the analysis of Lagrange multipliers and is described in this section.

4.4.2 Derivation of a sufficient condition

Assume that an aggregate function Φ such that

$$\Phi = \Phi(\mathbf{f}), \quad \frac{\partial \Phi}{\partial f_i}(\mathbf{x}_p) \geq 0, \quad i = 1, 2, \dots, M \quad (4.68)$$

reaches the minimum at a Pareto point \mathbf{x}_p . In particular, such aggregate function satisfying conditions (4.68) is used in the physical programming method (Messac *et al.*, 1996; Utyuzhnikov *et al.*, 2005). Then, from the Karush-Kuhn-Tucker (KKT) optimality condition, (Fletcher, 1989) it follows that:

$$\mathbf{S}_\Phi \equiv -\nabla\Phi - \mathbf{J}^T\boldsymbol{\lambda} = \mathbf{P}(-\nabla\Phi) = 0, \quad (4.69)$$

where the vector $\boldsymbol{\lambda}$ corresponds to the Lagrange multipliers:

$$\begin{aligned} \boldsymbol{\lambda} &= -(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}\nabla\Phi, \\ \lambda_j &\geq 0 \quad (j = 1, \dots, I). \end{aligned} \quad (4.70)$$

It is to be noted here that all analysis in this section is limited by the assumption of linear local constraints. In the objective space \mathbf{Y} , the normal to the Pareto surface is determined by the vector:

$$\mathbf{n}_f = \frac{\partial\Phi}{\partial\mathbf{f}} \equiv \nabla_f\Phi. \quad (4.71)$$

Let us introduce a test function Ψ satisfying conditions:

$$\frac{\partial\Psi}{\partial f_i}(\mathbf{x}_p) \geq 0 \quad (i = 1, 2, \dots, M) \quad (4.72)$$

in the vicinity of the Pareto point \mathbf{x}_p . This condition means that $\nabla\Psi \in C$, where C is a polyhedral cone defined by:

$$C = \{\mathbf{y} \in \mathbb{R}^N \mid \mathbf{y} = \sum_{i=1}^M a_i \nabla f_i, \quad a_i \geq 0\} \quad (4.73)$$

From equation (4.36) it follows that the descent direction \mathbf{S}_Ψ of the function Ψ is determined by:

$$\mathbf{S}_\Psi = -\nabla\Psi - \mathbf{J}^T\boldsymbol{\mu} = \mathbf{P}(-\nabla\Psi), \quad (4.74)$$

where $\boldsymbol{\mu} \in \mathbb{R}^l$ is defined by:

$$\boldsymbol{\mu} = -(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}\nabla\Psi. \quad (4.75)$$

The vector \mathbf{S}_Ψ is the orthogonal projection of $-\nabla\Psi$ onto the tangent plane to the Pareto surface in the design space. The vector \mathbf{S}_Ψ is a null-vector if and only if $\nabla\Psi$ belongs to the linear manifold V_I defined by

$$V_I = \left\{ \mathbf{y} \in \mathbb{R}^N \mid \mathbf{y} = \sum_{i=1}^l \nu_i \nabla \mathbf{g}_i \right\}. \quad (4.76)$$

If \mathbf{S}_Ψ is not a null-vector, then the mapping of \mathbf{S}_Ψ in the objective space gives a tangent direction to the Pareto surface. Indeed, multiplying \mathbf{S}_Φ by \mathbf{S}_Ψ and taking into account the orthogonality condition $\mathbf{J}\mathbf{S}_\Psi = 0$, we obtain

$$\sum_{i=1}^M \frac{\partial \Phi}{\partial f_i} \delta f_i = 0 \quad (4.77)$$

where $\delta f_i = \nabla f_i^T \mathbf{S}_\Psi \alpha$ is an infinitesimal variations of f_i along the direction \mathbf{S}_Ψ and $\alpha \rightarrow 0$ is a scaling parameter. Therefore, if \mathbf{S}_Ψ is not a null-vector, then it can be used to determine a tangent direction in the objective space and provide the sensitivity along \mathbf{S}_Ψ

$$\left. \frac{df_i}{df_j} \right|_{\mathbf{S}_\Psi} = \frac{df_i}{d\mathbf{S}_\Psi} \frac{d\mathbf{S}_\Psi}{df_j} = \frac{((\nabla f_i)^T \mathbf{S}_\Psi)}{((\nabla f_j)^T \mathbf{S}_\Psi)} \quad (i, j = 1, \dots, M; i \neq j). \quad (4.78)$$

Assume now that \mathbf{S}_Ψ is a null-vector. Then, $\nabla\Psi \in V_I$ and $\nabla\Psi$ is represented by a linear combination of the vectors normal to the hyperplanes of the active constraints

$$\nabla\Psi = -\sum_{i=1}^l \mu_i \nabla \mathbf{g}_i. \quad (4.79)$$

If in equation (4.79) $\mu_i \geq 0$ for all $1 \leq i \leq Q$, then there is no a descent direction for the test function Ψ . It immediately follows from Farkas' lemma (Fletcher, 1989; see also Appendix B). Indeed, feasible directions are defined by the set of vectors \mathbf{e} that satisfy:

$$(\mathbf{e}, \nabla \mathbf{g}_i) \leq 0 \text{ for } 1 \leq i \leq Q \quad \text{and} \quad (\mathbf{e}, \nabla \mathbf{g}_i) = 0 \text{ for } Q+1 \leq i \leq I, \quad (4.80)$$

while a descent direction of the function Ψ , must satisfy:

$$(\mathbf{e}, \nabla \Psi) < 0. \quad (4.81)$$

From Farkas' lemma (see appendix), the set of vectors \mathbf{e} satisfying conditions (4.80) and (4.81) is empty if and only if in equation (4.79) $\mu_i \geq 0$ for all $1 \leq i \leq Q$.

Let us now determine the set of Lagrange multipliers $\Omega_{I,Q}$ in such a way that the first Q Lagrange multipliers μ_i corresponding to the active constraints are positive

$$\Omega_{I,Q} = \{\boldsymbol{\mu} \in \mathbb{R}^I \mid \mu_i \geq 0 \text{ for } 1 \leq i \leq Q\}. \quad (4.82)$$

If some of the first Q multipliers are negative ($\boldsymbol{\mu} \notin \Omega_{I,Q}$), then the appropriate inequality constraints become inactive in the direction \mathbf{S}_Ψ and the Pareto solution might correspond to a non-differentiable point of the Pareto frontier. In order to identify if the Pareto point is non-differentiable, it is proposed to remove the constraints corresponding to negative components of $\boldsymbol{\mu}$ and repeat the analysis reconsidering the active constraints (Zhang W.H., 2003a, 2003b).

Let us now consider the requirements for the test function Ψ . This analysis is based on the consideration of the constraint cone and polar cone at a Pareto point. The constraint cone K is defined as follows (Vincent and Grantham, 1981)

$$K = \{\mathbf{y} \in \mathbb{R}^N \mid \mathbf{y} = -\sum_{i=1}^I \mu_i \nabla \mathbf{g}_i, \boldsymbol{\mu} \in \Omega_{I,Q}\}. \quad (4.83)$$

Along with the constraint cone K we define the positive polar cone K^*

$$K^* = \{\mathbf{z} \in \mathbb{R}^N \mid \mathbf{z}^T \mathbf{y} \geq 0 \quad \forall \mathbf{y} \in K\}. \quad (4.84)$$

For any regular point of the feasible space, the cone T tangent to the feasible design space coincides with the cone K^* : $K^* = T$ (Vincent and Grantham, 1981). It is to be noted that if the function Ψ reaches a minimum at point \mathbf{x}_p then for any vector $\mathbf{e} \in T$ the following condition must be valid

$$\mathbf{e}^T \nabla \Psi \geq 0, \quad (4.85)$$

otherwise there is a feasible direction of minimisation for Ψ . Inequality (4.85) also means that $\nabla \Psi \in T^* = K$, where T^* is the positive polar cone to the cone T . The identity $T^* = K$ follows from the Polar theorem: $(K^*)^* = K$ (Vincent and Grantham, 1981). One can note that since the function Φ reaches a minimum at point \mathbf{x}_p , then $\nabla \Phi \in K$.

If $\nabla \Psi \notin K$ then $\boldsymbol{\mu} \notin \Omega_{i,Q}$ and there is a direction along which the function Ψ can be further diminished. Assume, that there is only one Lagrange multiplier μ_q in equation (4.75): $\mu_q < 0$, ($q \leq Q$). By removing the q -th active constraint from \mathbf{J} , we obtain the reduced matrix \mathbf{J}_r . The new descent vector $\tilde{\mathbf{S}}_\Psi$ is defined by the set of remaining active constraints as follows:

$$\tilde{\mathbf{S}}_\Psi = -\nabla \Psi - \mathbf{J}_r^T \boldsymbol{\mu}_r, \quad (4.86)$$

where

$$\boldsymbol{\mu}_r = -(\mathbf{J}_r \mathbf{J}_r^T)^{-1} \mathbf{J}_r \nabla \Psi. \quad (4.87)$$

Vector $\tilde{\mathbf{S}}_\Psi$ defines a descent direction along which the q -th constraint becomes inactive. Before going any further, let us make sure the new direction is feasible. Subtracting equation (4.74) from (4.86) we obtain:

$$\tilde{\mathbf{S}}_\Psi = \mathbf{J}^T \boldsymbol{\mu} - \mathbf{J}_r^T \boldsymbol{\mu}_r = \mu_q \nabla g_q + \mathbf{J}_r^T (\tilde{\boldsymbol{\mu}}_r - \boldsymbol{\mu}_r), \quad (4.88)$$

where vector $\tilde{\boldsymbol{\mu}}_r \in \mathbb{R}^{l-1}$ is obtained from $\boldsymbol{\mu}$ by dropping μ_q . Note that $\tilde{\mathbf{S}}_\Psi$ is not a null vector, otherwise from equation (4.88), there would exist a non-null vector $\boldsymbol{\beta}$ such that $\mathbf{J}^T \boldsymbol{\beta} = \mathbf{0}$ which would contradict the assumption of local linear independence of the active constraints. Multiplying equation (4.88) by itself and taking into account the orthogonality condition $\mathbf{J}_r \tilde{\mathbf{S}}_\Psi = \mathbf{0}$, we find:

$$|\tilde{\mathbf{S}}_\Psi|^2 = \mu_q \tilde{\mathbf{S}}_\Psi^T \nabla g_q > 0. \quad (4.89)$$

Hence, $\tilde{\mathbf{S}}_\Psi^T \nabla g_q < 0$ and the direction $\tilde{\mathbf{S}}_\Psi$ is feasible.

If several Lagrange multipliers in the vector $\boldsymbol{\mu}$ are negative, we define the set \bar{Q} such that $\mu_q < 0$ if $q \in \bar{Q}$. Only one ∇g_q with $q \in \bar{Q}$ needs to be removed from \mathbf{J} so that the above analysis remains valid. A strategy is proposed (Zhang, 2003b), to remove the gradient of the constraint that satisfies:

$$\frac{\lambda_k}{\mu_k} = \max_{q \in \bar{Q}} \frac{\lambda_q}{\mu_q} < 0 \quad (4.90)$$

to obtain the reduced matrix \mathbf{J}_r for the descent vector $\tilde{\mathbf{S}}_\Psi$. Along the descent direction $\tilde{\mathbf{S}}_\Psi$, we have:

$$\delta \Psi = \sum_{i=1}^M \frac{\partial \Psi}{\partial f_i} \delta f_i < 0 \quad (4.91)$$

and another Pareto solution is achievable in the framework of the linear approximation because all the coefficients $\partial \Psi / \partial f_i$ in equation (4.91) are non-negative.

To find the new normal to the Pareto surface in the \mathbf{Y} space, we multiply the null-vector \mathbf{S}_Ψ from equation (4.74) by the multiplier α_Ψ defined as:

$$\alpha_\Psi = -\max_{q \in \bar{Q}} \frac{\lambda_q}{\mu_q} = -\frac{\lambda_k}{\mu_k} > 0 \quad (4.92)$$

and add it to the null-vector \mathbf{S}_Φ from equation (4.69). Then, we obtain:

$$\nabla(\Phi + \alpha_\Psi \Psi) = -\mathbf{J}^T(\boldsymbol{\lambda} + \alpha_\Psi \boldsymbol{\mu}). \quad (4.93)$$

This equality can be seen as the KKT optimality condition for the new aggregate function $\Phi^* = \Phi + \alpha_\Psi \Psi$ as we can show that:

$$\nabla \Phi^* + \mathbf{J}^T \boldsymbol{\lambda}^* = 0 \quad (4.94)$$

with the Lagrange multipliers: $\boldsymbol{\lambda}^* = \boldsymbol{\lambda} + \alpha_\Psi \boldsymbol{\mu}$. Additionally, one can verify the following inequalities:

$$\frac{\partial \Phi^*}{\partial f_j} \geq 0, \quad (j = 1, \dots, M), \quad (4.95)$$

$$\lambda_i^* \geq 0, \quad (i = 1, \dots, I). \quad (4.96)$$

Note that in the KKT condition (4.94) there exists $k \in \bar{Q}$ such that $\lambda_k^* = 0$. It corresponds to the active constraint that becomes inactive. As a result, the normal to the Pareto surface might shift to its new value:

$$\mathbf{n}_f = \nabla_f(\Phi + \alpha_\Psi \Psi). \quad (4.97)$$

It is to be noted here that the change of the number of active constraints, Q , does not necessarily mean that the Pareto point is not differentiable. More precisely the non-differentiability corresponds to a shift of the normal to the Pareto surface (Zhang, 2003b) not a change in the set of active constraints.

It is possible to show that the vector $\tilde{\mathbf{S}}_\Psi$ defines a direction tangent to the Pareto surface. Indeed, multiplying the null-vector \mathbf{S}_Φ from equation (4.69) by $\tilde{\mathbf{S}}_\Psi$, we obtain:

$$\nabla \Phi^T \tilde{\mathbf{S}}_\Psi = -\lambda_k \nabla g_k^T \tilde{\mathbf{S}}_\Psi. \quad (4.98)$$

Then, after multiplication of \mathbf{S}_Ψ from equation (4.74) by $\tilde{\mathbf{S}}_\Psi$ we have:

$$\nabla g_k^T \tilde{\mathbf{S}}_\Psi = -\frac{1}{\mu_k} \nabla \Psi^T \tilde{\mathbf{S}}_\Psi, \quad (4.99)$$

$$\nabla \Phi^T \tilde{\mathbf{S}}_\Psi = \frac{\lambda_k}{\mu_k} \nabla \Psi^T \tilde{\mathbf{S}}_\Psi = -\alpha_\Psi \nabla \Psi^T \tilde{\mathbf{S}}_\Psi, \quad (4.100)$$

hence,

$$\nabla(\Phi + \alpha_\Psi \Psi)^T \tilde{\mathbf{S}}_\Psi = \nabla(\Phi^*)^T \tilde{\mathbf{S}}_\Psi = \mathbf{0}. \quad (4.101)$$

In turn, if $\nabla \Psi \in K$, then $\nabla \Psi$ can be expressed as a linear combination of the gradients of active constraints (cf. equation (4.79)) with $\boldsymbol{\mu} \in \Omega_{l,q}$. It follows that for such a test function, the feasible descent direction \mathbf{S}_Ψ is a null vector. From the previous analysis, it also follows that the test function Ψ does not allow to obtain any additional information on the differentiability of the Pareto surface.

Thus, to verify the differentiability of a Pareto point the function Ψ must be chosen in such a way that (cf. equations (4.73), (4.76), (4.83))

$$\nabla \Psi \in C \cap (V_I \setminus K). \quad (4.102)$$

It is worth noting that the equality constraints have an effect on the matrix \mathbf{J} but do not influence the analysis with regard to the detection of non-differentiable Pareto points. This follows from the fact that equality constraints are to be strictly satisfied, i.e. they cannot be inactive, and the sign of the appropriate Lagrange multipliers is undetermined. It is to be noted that in contrast to (Zhang W.H. 2003a, 2003b) where only a weighted-sum of objective functions is considered as the test (preference) function Ψ , we consider here a general test function. Therefore, the approach becomes applicable to many other algorithms and is not limited to the weighted-sum method, which can only be used for convex problems. In particular, the analysis developed to detect non-differentiable Pareto solutions can be implemented into the *Physical-Programming* based method (Guenov et al., 2005; Utyuzhnikov et al., 2005), which guarantees a well-distributed representation of the entire Pareto frontier.

4.4.3 Use of the sufficient condition for non-differentiability

In a practical situation, using conditions (4.102) to see whether a candidate Pareto solution is non-differentiable might not always be very convenient. If only two or three variables are involved in the optimisation problem, the designer can simply draw in the design space the relevant set and cones V_I, C and K to see whether there exists a vector satisfying conditions (4.102). However, in the case where more than three variables are involved, this option is clearly not suitable.

Let us reconsider conditions (4.102) and try to reformulate it in a more convenient way. To show that a Pareto point is not differentiable, one must try to find a vector $\nabla\Psi$ that belongs to C and $V_I \setminus K$. Therefore, one must find a set of scalars μ_i and a_j such that:

$$\nabla\Psi = -\sum_{i=1}^I \mu_i \nabla g_i = \sum_{j=1}^M a_j \nabla f_j \quad (4.103)$$

where at least one $\mu_i < 0$, $1 \leq i \leq Q$ and $a_j \geq 0$ for all $1 \leq j \leq M$. It can be reformulated as follows:

$$\boldsymbol{\mu} = -(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}[\nabla\mathbf{f}]\mathbf{a} \quad (4.104)$$

$$\mathbf{P}[\nabla\mathbf{f}]\mathbf{a} = \mathbf{0} \quad (4.105)$$

where at least one $\mu_i < 0$, $1 \leq i \leq Q$ and $a_j \geq 0$ for all $1 \leq j \leq M$.

To see whether there exist a vector satisfying (4.102), it is proposed to solve the following linear optimisation problem:

$$\begin{aligned} & \min_{\mathbf{a}} \mu_i \\ & \text{Subject to } \begin{cases} \mathbf{P}[\nabla\mathbf{f}]\mathbf{a} = \mathbf{0} \\ \mathbf{a} \geq \mathbf{0} \\ \mu_i \geq -1 \end{cases} \end{aligned} \quad (4.106)$$

where μ_i is obtained from equation (4.104) and the constraint $\mu_i \geq -1$ is only here to stop the optimisation as soon as μ_i is negative. The linear optimisation problem is

solved for each $i = 1, \dots, Q$ individually until one feasible μ_i is negative. If the linear optimisation problem has such a solution, the corresponding vector \mathbf{a} is such that:

$$\nabla\Psi = \sum_{j=1}^M a_j \nabla f_j \in C \cap (V_I \setminus K) \quad (4.107)$$

and the Pareto point under study is non-differentiable.

4.5 Procedure for local Pareto sensitivity analysis

In this section, an overview of the Pareto analysis concept developed in the two previous sections is presented. The problem of identifying non-differentiable Pareto point is of main importance. Indeed, if the candidate point is not differentiable, the general linear and quadratic approximations will perform poorly. Condition (4.102) on differentiability is only sufficient, i.e. if there exists a vector $\nabla\Psi$ satisfying condition (4.102), it follows that the Pareto point is not differentiable. However, if such vector cannot be found, it does not ensure that the point is differentiable and that the Pareto approximations can be used safely. In the case where condition (4.102) is not satisfied, the designer can decide to investigate further the differentiability of the Pareto point under study by finding directional derivatives along principal descent directions of positive linear combinations of gradients of objective functions using equation (4.78). As for the trade-off matrix, these derivatives define tangent vectors to the Pareto surface. If all these vectors are in the same hyperplane, then the designer can fairly assume that the Pareto surface is differentiable. The whole approach is summarised in Figure 4.11.

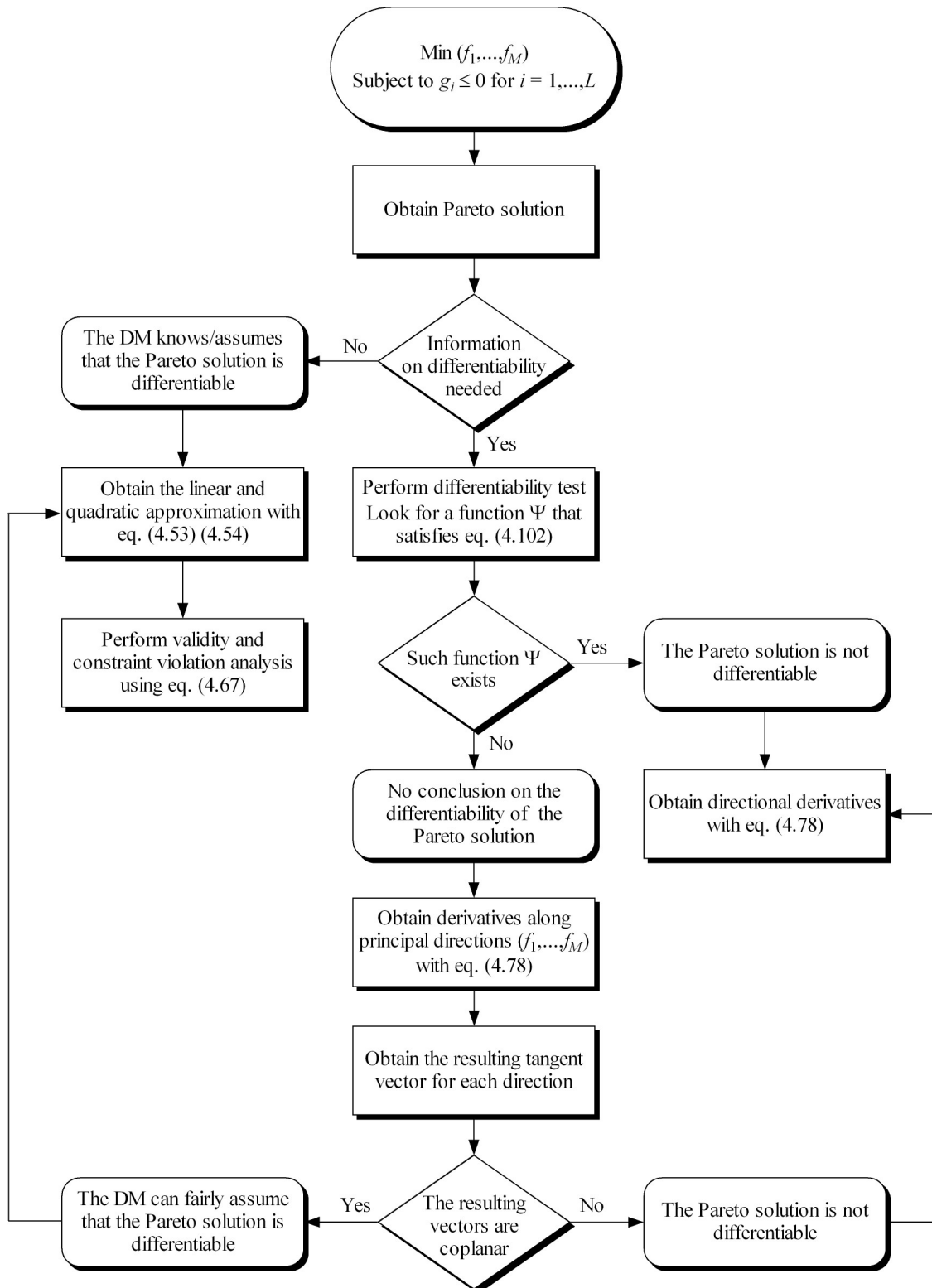


Figure 4.11: Pareto analysis flowchart

4.6 Limitations of the Pareto sensitivity analysis

At this point, it is important to introduce the concept of *global* and *local Pareto front*. Points of a *local Pareto front* correspond to Pareto optimal points, but with respect to a localised region of the design space. Instead, points of the *global Pareto front* are Pareto optimal with respect to the entire design space. The *global Pareto front* might be composed of solutions from several *local Pareto fronts*, which correspond to different regions of the design space. Such situation is described in Figure 4.12.

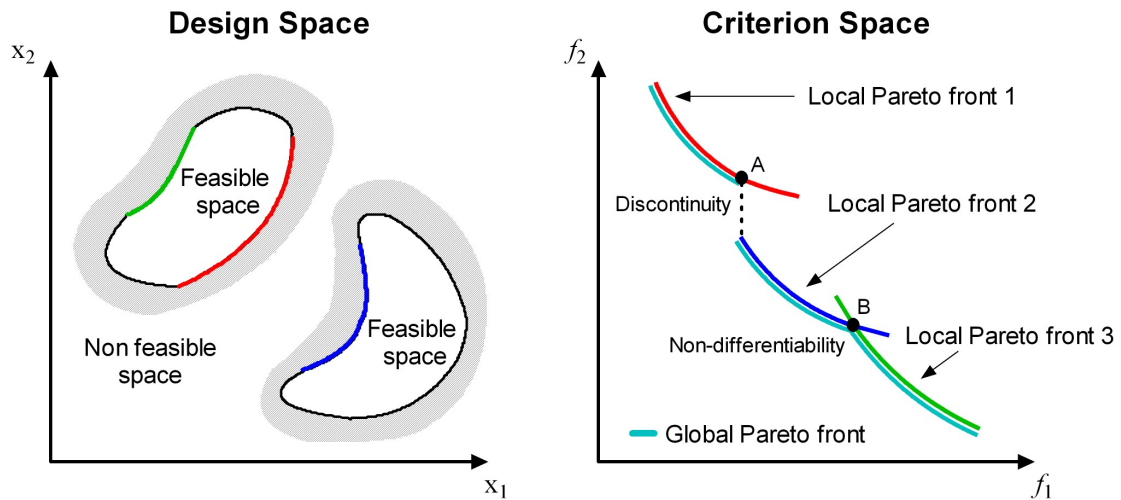


Figure 4.12: Global and local Pareto fronts

Figure 4.12 also shows that the presence of several *local Pareto fronts* can result in non-smooth regions on the *global Pareto front* such as discontinuity (point A) and non-differentiability (point B).

The local Pareto approximations derived in this chapter are based on local information and are only valid if the Pareto point under study is differentiable. If local Pareto approximations (4.53) and (4.54) are used at point A, one will actually approximate *local Pareto front 1* for which point A is differentiable. For point B, one will approximate either *local Pareto front 2* or *local Pareto front 3* depending on the corresponding point in the design space.

The method developed in this chapter for detecting non-differentiable Pareto points can only detect non-differentiability due to local changes in the set of active constraints. It will fail to detect lack of smoothness corresponding to points A and B.

The Pareto analysis developed in this chapter is entirely based on local information on the solution set. It cannot deal with non-smooth regions of the *global Pareto front* described in Figure 4.12 because they result from the structure of the solution set over the entire design space. It is important to observe that currently no method to the best of the author's knowledge can address this type of non-differentiability.

4.7 Numerical examples

In this section, the new linear and quadratic equations are used to obtain the local Pareto approximations

4.7.1 Example 1: Non linear three-objective optimisation

Let us now consider the following multi-objective problem studied in (Tappeta *et al.*, 2000):

$$\text{Minimise: } \mathbf{f}(\mathbf{x}) = \{f_1(\mathbf{x}), f_2(\mathbf{x}), f_3(\mathbf{x})\} \quad (4.108)$$

$$\text{Subject to: } \begin{aligned} g(\mathbf{x}) &= 12 - x_1^2 - x_2^2 - x_3^2 \geq 0, \\ \mathbf{x} &\geq 0 \end{aligned} \quad (4.109)$$

where the objective functions are given by:

$$\begin{aligned} f_1 &= 25 - (x_1^3 + x_1^2(1 + x_2 + x_3) + x_2^3 + x_3^3)/10, \\ f_2 &= 35 - (x_1^3 + 2x_2^3 + x_2^2(1 + x_1 + x_3) + x_3^3)/10, \\ f_3 &= 50 - (x_1^3 + x_2^3 + 3x_3^3 + x_3^2(1 + x_1 + x_2))/10. \end{aligned} \quad (4.110)$$

The linear and quadratic Pareto approximations obtained with (4.53) and (4.54) are given in Figure 4.13 and Figure 4.14 respectively. The comparison between both approximations is given in Figure 4.15. As expected, a much better approximation of the Pareto surface is provided by the quadratic approximation.

In the next two examples, non-smooth Pareto surfaces are considered. The Pareto analysis is used to detect non-differentiable Pareto points.

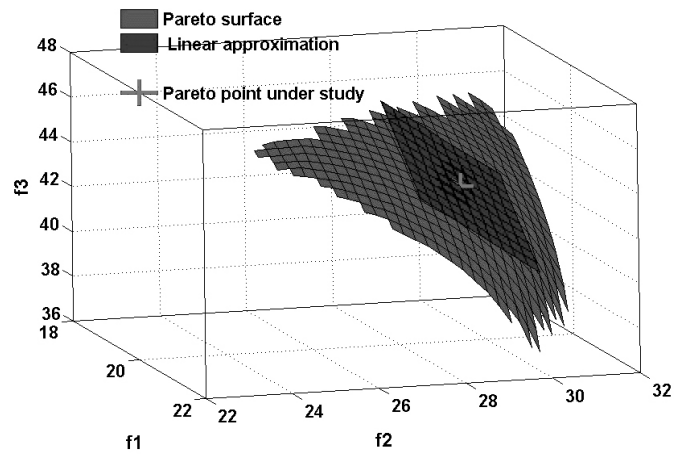


Figure 4.13: Example 1 – Non linear three objectives problem – linear approximation

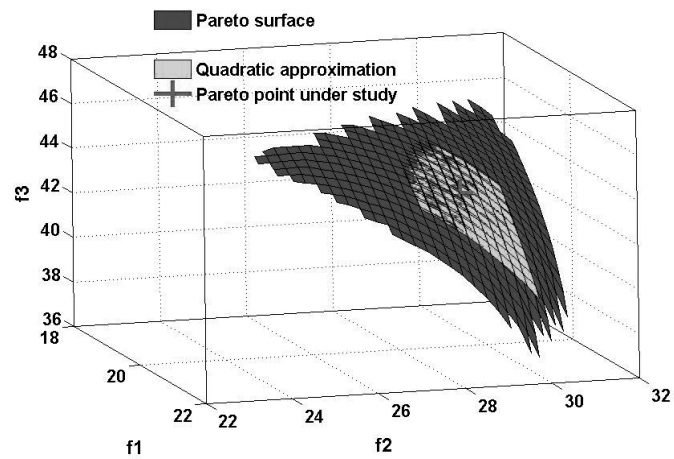


Figure 4.14: Example 1 – Non linear three objectives problem – Quadratic approximation

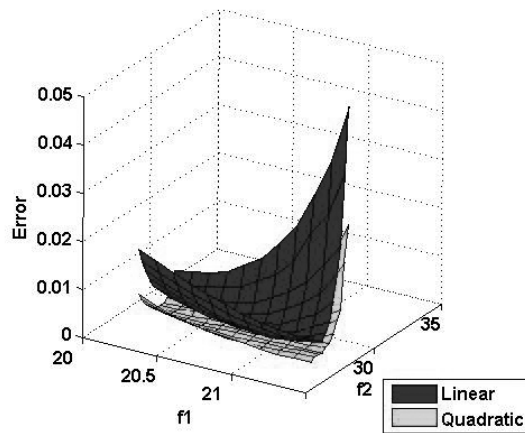


Figure 4.15: Example 1 – Non linear three objectives problem – Relative error in predicting f_3

4.7.2 Example 2: Non-differentiable Pareto point when design and criterion spaces coincide

The linear bi-criteria test case taken from Zhang W.H. 2003b is considered:

$$\text{Minimise: } \mathbf{x} = (x, y) \quad (4.111)$$

$$g_1(\mathbf{x}) = -\frac{x}{4} - \frac{y}{4} + 1 \leq 0,$$

$$\text{Subject to: } g_2(\mathbf{x}) = -\frac{x}{3} - \frac{y}{6} + 1 \leq 0, \quad (4.112)$$

$$g_3(\mathbf{x}) = -x \leq 0,$$

$$g_4(\mathbf{x}) = -y \leq 0.$$

In this test case, the design space coincides with the objective space. Hence, each point of the Pareto set is a point of the design space for which at least one constraint is active. The Pareto set is represented by the solid portions of the lines $g_1 = 0$ and $g_2 = 0$. The kink on the Pareto surface corresponds to the only non-differentiable Pareto solution.

At the non-differentiable Pareto solution, the tangent cone $T = K^*$ and its polar cone K are constructed and shown in Figure 4.16. In this example $K \subset K^*$.

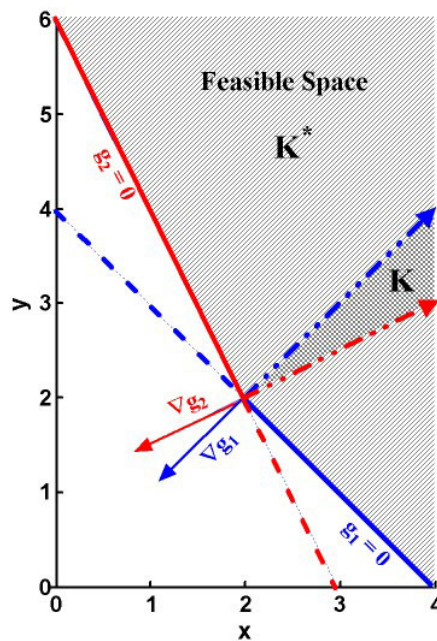


Figure 4.16: Example 2 – Relative position of the tangent cone and its polar at a non-differentiable Pareto point

A vector $\nabla\Psi$ satisfying condition (4.102) can be obtained and is displayed in Figure 4.17. Therefore, according to the analysis developed in section 4, the Pareto point under study is non-differentiable.

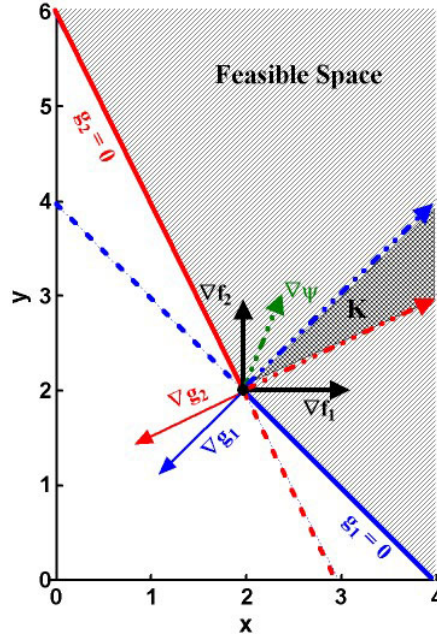


Figure 4.17: Example 2 – Existence of vector $\nabla\Psi$

As the point is non-differentiable, the linear and quadratic approximations (4.53) and (4.54) cannot be used. Instead, tangent derivatives along the Pareto surface can be obtained using equation (4.78).

At the non-differentiable point, both constraints are active and the corresponding matrix \mathbf{J} is defined by

$$\mathbf{J} = \begin{bmatrix} -\frac{1}{4} & -\frac{1}{4} \\ -\frac{1}{3} & -\frac{1}{6} \end{bmatrix}. \quad (4.113)$$

Hence,

$$\mathbf{P} = \mathbf{0}. \quad (4.114)$$

Let us study the sensitivity for the reduction of f_1 choosing $\Psi_1 = f_1$. The projection of $-\nabla\Psi_1$ gives $\mathbf{S}_1 = \mathbf{P}(-\nabla\Psi_1) = \mathbf{0}$ with $\boldsymbol{\mu}_1 = -(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}\nabla\Psi_1 = [-4 \ 6]^T$. The negative value for

the first component of $\boldsymbol{\mu}_1$ indicates that in the direction of reduction of f_1 , constraint g_1 becomes inactive. Therefore, its gradient should be removed from \mathbf{J} to obtain the reduced projection matrix

$$\mathbf{P}_{r1} = \begin{bmatrix} \frac{1}{5} & -\frac{2}{5} \\ -\frac{2}{5} & \frac{4}{5} \end{bmatrix}. \quad (4.115)$$

Thus, $\tilde{\mathbf{S}}_{\Psi_1}$ can be calculated as $\tilde{\mathbf{S}}_{\Psi_1} = -\mathbf{P}_{r1}(\nabla\Psi_1)$ and we obtain $\tilde{\mathbf{S}}_{\Psi_1} = [-1/5 \ 2/5]^T$. The sensitivity along $\tilde{\mathbf{S}}_{\Psi_1}$ for f_1 is then

$$\left. \frac{df_2}{df_1} \right|_{\tilde{\mathbf{S}}_{\Psi_1}} = \frac{(\nabla f_2, \tilde{\mathbf{S}}_{\Psi_1})}{(\nabla f_1, \tilde{\mathbf{S}}_{\Psi_1})} = -2 \quad (4.116)$$

which corresponds to the slope of the line $g_2 = 0$.

The same analysis can be performed along the greatest feasible direction for objective f_2 by choosing $\Psi_2 = f_2$. It follows that $\mathbf{S}_2 = \mathbf{P}(-\nabla\Psi_2) = \mathbf{0}$ with the vector of Lagrange multipliers $\boldsymbol{\mu}_2 = -(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}\nabla\Psi_2 = [4 \ -6]^T$. In this case, constraint g_2 becomes inactive and the new reduced projection matrix is

$$\mathbf{P}_{r2} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \quad (4.117)$$

and $\tilde{\mathbf{S}}_{\Psi_2} = -\mathbf{P}_{r2}(\nabla\Psi_2) = [1/2 \ -1/2]^T$. The sensitivity along $\tilde{\mathbf{S}}_{\Psi_2}$ for f_2 is then

$$\left. \frac{df_2}{df_1} \right|_{\tilde{\mathbf{S}}_{\Psi_2}} = \frac{(\nabla f_2, \tilde{\mathbf{S}}_{\Psi_2})}{(\nabla f_1, \tilde{\mathbf{S}}_{\Psi_2})} = -1 \quad (4.118)$$

which is the slope of the line $g_1 = 0$.

4.7.3 Example 3: Non-differentiable Pareto point when design and criterion space are different

In this example, the quadratic bi-criteria test case under linear constraints is considered:

$$\text{Minimise: } (f_1(\mathbf{x}), f_2(\mathbf{x})) \quad (4.119)$$

$$\begin{aligned} \text{Subject to: } & g_1(\mathbf{x}) = y - 2x + 1 \leq 0, \\ & g_2(\mathbf{x}) = -y + \frac{x}{2} + \frac{1}{2} \leq 0 \end{aligned} \quad (4.120)$$

with

$$\mathbf{x} = (x, y) \quad (4.121)$$

$$\begin{aligned} f_1(\mathbf{x}) &= \left(y - \frac{3}{2}x - \frac{1}{2} \right)^2 \\ f_2(\mathbf{x}) &= \left(y - \frac{2}{3}x + \frac{1}{2} \right)^2 \end{aligned} \quad (4.122)$$

The feasible space is given in Figure 4.18.

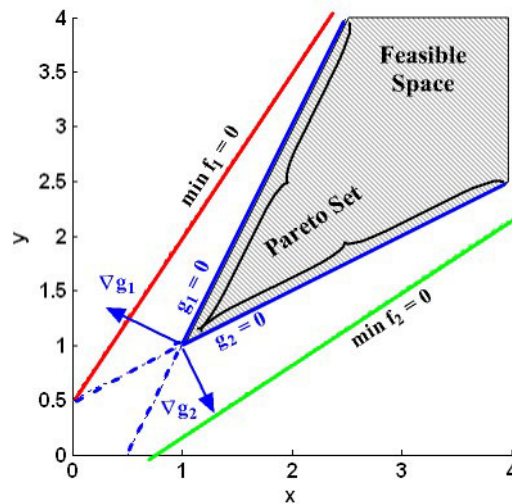


Figure 4.18: Example 3 – Design space / Pareto set

Only the two contour lines corresponding to the minimum value of the objectives are represented in the design space. Isovalue contour lines for each objective are straight lines parallel to the one corresponding to their respective minimum value. The set of Pareto solutions for the problem is given in Figure 4.19. In this example, design space

and objective spaces are different. This example puts emphasis on the fact that only information in the design space is used to identify a non-differentiable point in the objective space.

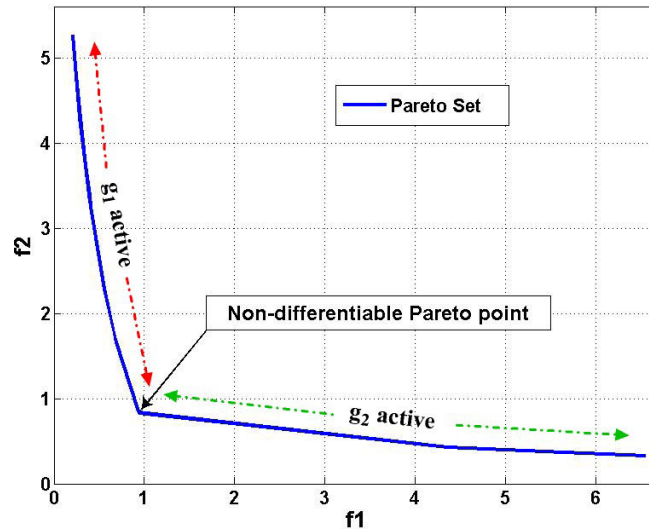


Figure 4.19: Example 3 – Objective space / Pareto set

The Pareto point, corresponding to the design point where both constraints are active, is clearly non-differentiable point in the objective space. Moving away from this point, the set of active constraints is changed and implies a shift in the direction tangent to the Pareto surface. Figure 4.20 shows the tangent cone $T = K^*$ and its polar cone K in the design space.

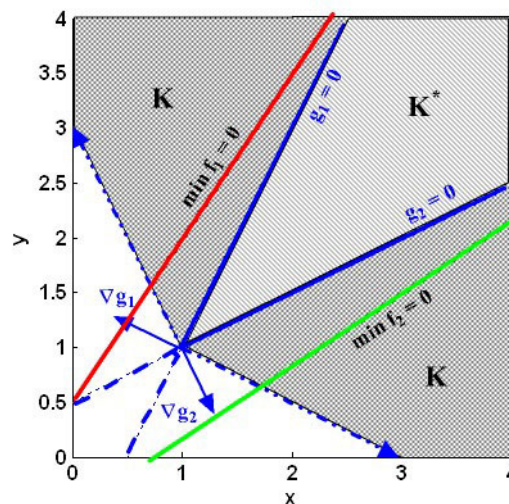


Figure 4.20: Example 3 – Relative position of the tangent cone and its polar at a non-differentiable Pareto point

As shown in Figure 4.21, there exists a vector $\nabla\Psi$ satisfying condition (4.102), therefore, the Pareto point is not differentiable.

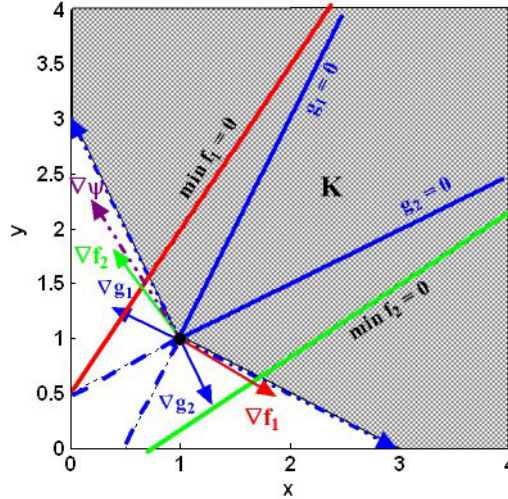


Figure 4.21: Example 3 – Existence of vector $\nabla\Psi$

Only derivatives along descent direction for each objective can be obtained using equation (4.78).

At the Pareto point under study, both constraints are active and the corresponding matrix \mathbf{J} is defined by

$$\mathbf{J} = \begin{bmatrix} -2 & 1 \\ \frac{1}{2} & -1 \end{bmatrix}. \quad (4.123)$$

Hence,

$$\mathbf{P} = \mathbf{0}. \quad (4.124)$$

Let us evaluate the sensitivity for the reduction of objective f_1 and choose $\Psi_1 = f_1$. Proceeding as in the previous example, we obtain $\mathbf{S}_1 = \mathbf{P}(-\nabla\Psi_1) = \mathbf{0}$ and the corresponding vector of Lagrange multipliers $\boldsymbol{\mu}_1 = -(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}\nabla\Psi_1 = [4/3 \ -2/3]^T$. The negative value for the second component of $\boldsymbol{\mu}_1$ indicates that in the direction of reduction of f_1 , constraint g_2 becomes inactive. Therefore, its gradient should be removed from \mathbf{J} to obtain the reduced projection matrix:

$$\mathbf{P}_{r1} = \begin{bmatrix} \frac{1}{5} & \frac{2}{5} \\ \frac{2}{5} & \frac{4}{5} \end{bmatrix}. \quad (4.125)$$

Thus, $\tilde{\mathbf{S}}_{\Psi_1}$ can be calculated as $\tilde{\mathbf{S}}_{\Psi_1} = -\mathbf{P}_{r1}(\nabla\Psi_1)$ and we obtain $\tilde{\mathbf{S}}_{\Psi_1} = [1/5 \ 2/5]^T$. The sensitivity along the greatest feasible direction for f_1 is then given by

$$\left. \frac{df_2}{df_1} \right|_{\tilde{\mathbf{S}}_{\Psi_1}} = \frac{(\nabla f_2, \tilde{\mathbf{S}}_{\Psi_1})}{(\nabla f_1, \tilde{\mathbf{S}}_{\Psi_1})} = -\frac{2}{9}. \quad (4.126)$$

The same analysis can be performed along the greatest feasible direction for objective f_2 by choosing $\Psi_2 = f_2$. It follows that $\mathbf{S}_2 = \mathbf{P}(-\nabla\Psi_2) = \mathbf{0}$ with the corresponding vector of Lagrange multipliers $\boldsymbol{\mu}_2 = -(\mathbf{J}\mathbf{J}^T)^{-1}\mathbf{J}\nabla\Psi_2 = [-0.1852 \ 1.4815]^T$. In this case, constraint g_1 becomes inactive and the new reduced projection matrix is

$$\mathbf{P}_{r2} = \begin{bmatrix} \frac{4}{5} & \frac{2}{5} \\ \frac{2}{5} & \frac{1}{5} \end{bmatrix} \quad (4.127)$$

and $\tilde{\mathbf{S}}_{\Psi_2} = -\mathbf{P}_{r2}(\nabla\Psi_2) = [2/9 \ 1/9]^T$. The sensitivity along the greatest feasible direction for f_2 is then

$$\left. \frac{df_2}{df_1} \right|_{\tilde{\mathbf{S}}_{\Psi_2}} = \frac{(\nabla f_2, \tilde{\mathbf{S}}_{\Psi_2})}{(\nabla f_1, \tilde{\mathbf{S}}_{\Psi_2})} = -0.1389. \quad (4.128)$$

4.8 Conclusions

In this chapter, a promising existing technique (Tappeta *et al.*, 1999, 2000, 2001) is reviewed in detail. Some clarifications about the formulation of the derivation of the linear approximation are made. Based on the same concept, the method is extended to a quadratic approximation where no local optimisation is needed. However, a simple

analytical example proved that this approach could not always provide a local Pareto approximation.

Taking into account the limitations of the first method, new theoretically sound linear and quadratic approximations are derived in the general case. The assumptions are clearly stated and an approach to use them as part of a Pareto local sensitivity analysis is given. It includes an approach to assess the extent of validity of the approximation by defining the range of local feasible moves on the Pareto surface.

A sufficient condition to identify non-differentiable Pareto points is derived. It is shown how it can be conveniently reformulated as a linear optimisation.

The limitations of the sensitivity analysis procedure in the context of a global Pareto surface composed of solution from several local Pareto front are outlined.

Finally, a summary of the entire Pareto analysis procedure is given and numerical tests are performed to evaluate the advantages of the approach.

Chapter 5

Beam optimisation problem

In this chapter, a typical engineering design optimisation problem is considered. The optimisation of an I-beam, namely minimisation of its cross section area and the resulting displacement, subject to a particular load is carried out. The test case can be entirely described with analytical equations, which makes the interpretation of the results easier as the relations between inputs, and outputs are straightforward. This example serves as the first of two representative test cases to demonstrate the advantages of using local Pareto approximation once a Pareto solution is available. Emphasis is put on the analysis that can be carried out during the search of a satisfactory solution of the multi-objective optimisation problem and the information about trade-offs that can be obtained.

5.1 Description

The I-beam design test case is taken from Yang *et al.* (2002). The problem is to find the dimensions of the cross section of a beam of fixed length, which satisfies geometric and strength constraints, and minimise its weight and static deflection under

the loads P and Q applied along the z and y axis, respectively. A description of the I-beam problem is given in Figure 3.1.

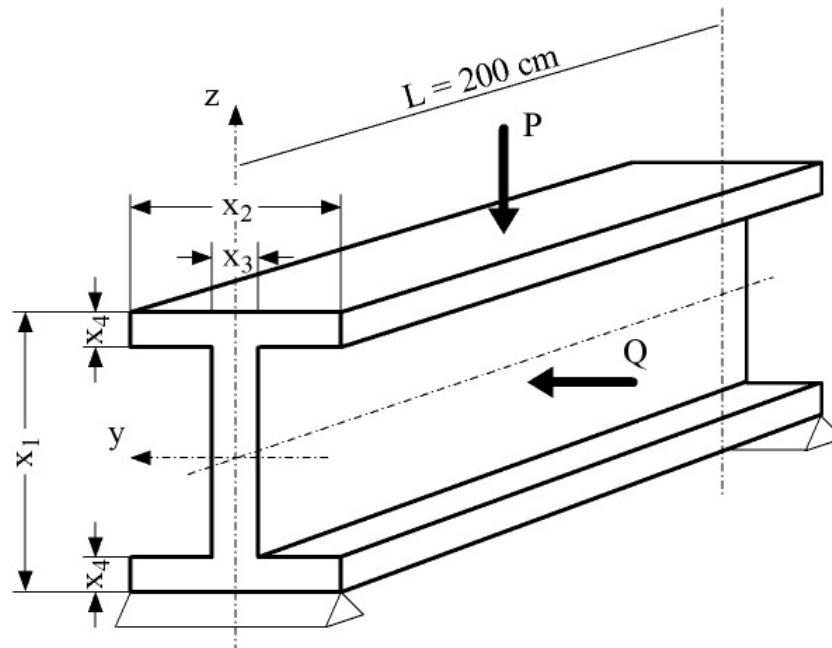


Figure 5.1: The I-beam design problem

Since the length of the beam is fixed, minimising its weight is equivalent to minimising its cross-sectional area. Therefore, the problem can be formulated as follows:

$$\min_{\mathbf{x}} \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} = \begin{pmatrix} \text{area} = 2x_2x_4 + x_3(x_1 - 2x_4) \\ \text{displacement} = \frac{PL^3}{48EI} \end{pmatrix} \quad (5.1)$$

Subject to a strength constraint:

$$g(\mathbf{x}) = \frac{M_y}{Z_y} + \frac{M_z}{Z_z} - \sigma_a \leq 0 \quad (\text{kN.cm}^{-2}) \quad (5.2)$$

and geometric constraints on the input range of variation:

$$\begin{aligned} 10 &\leq x_1 \leq 80 && (\text{cm}) \\ 10 &\leq x_2 \leq 50 && (\text{cm}) \\ 0.9 &\leq x_3 \leq 5 && (\text{cm}) \\ 0.9 &\leq x_4 \leq 5 && (\text{cm}) \end{aligned} \quad (5.3)$$

where the moments are given by:

$$M_y = \frac{P}{2} \times \frac{L}{2}, \quad (5.4)$$

$$M_z = \frac{Q}{2} \times \frac{L}{2} \quad (5.5)$$

and the moments of inertia are calculated as follows:

$$Z_y = \frac{1}{6x_1} \{x_3(x_1 - x_4)^3 + 2x_2x_4[4x_4^2 + 3x_1(x_1 - 2x_4)]\}, \quad (5.6)$$

$$Z_z = \frac{1}{6x_2} \{(x_1 - x_4)x_3^3 + 2x_4x_2^3\}, \quad (5.7)$$

with $E = 2 \times 10^4 \text{ kNcm}^{-2}$, $\sigma_a = 16 \text{ kNcm}^{-2}$, $P = 600 \text{ kN}$, $Q = 50 \text{ kN}$ and $L = 200 \text{ cm}$.

5.2 Pareto surface

The set of solutions for this problem is obtained using an in-house developed gradient-based optimisation tool for multi-objective optimisation (Fantini, 2007; Fantini *et al.*, 2007; Guenov *et al.*, 2005; Utyuzhnikov *et al.*, 2005). This tool searches for solutions that are uniformly distributed over the Pareto front. Since all equations necessary to describe the present problem are analytical, obtaining a large number of Pareto solutions is not a problem and 260 points can be obtained in a matter of a couple of minutes. The large number of solutions allows to compare easily results of the approximation with the true Pareto front.

The set of Pareto solutions obtained for this test case is plotted in Figure 5.2. The high density of points gives a good representation of the Pareto front, particularly in the region where the curvature is important. It appears quite clearly that the Pareto front is differentiable everywhere. Therefore, in this chapter, the analysis about the Pareto differentiability is omitted. Instead, the analysis will focus on the use of local Pareto approximation for understanding the local trade-offs at different candidate Pareto solutions.

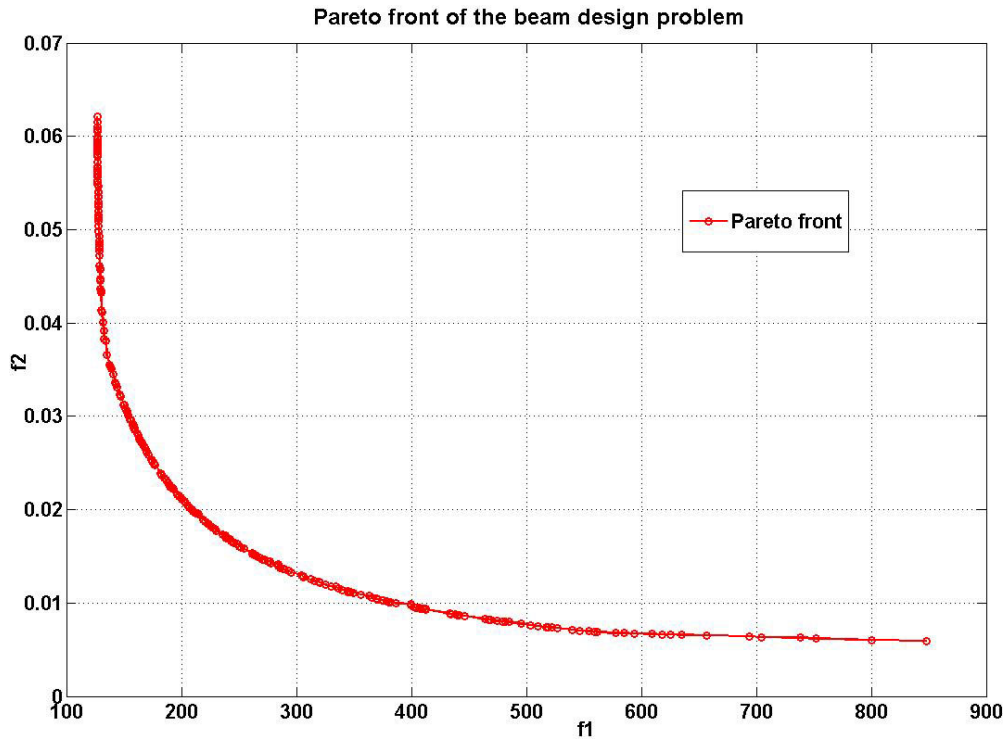


Figure 5.2: Pareto front for the I-beam design problem

5.3 Local Pareto sensitivity analysis

In this section, the approximation analysis is carried out on two candidate Pareto solutions and approximations are derived to understand their local sensitivity.

The first candidate solution considered is given in Table 5.1. Its location on the Pareto surface is shown in Figure 5.3.

Pareto solution 1	
x_1 (cm)	80
x_2 (cm)	50
x_3 (cm)	0.9
x_4 (cm)	3.48
<hr/>	
f_1 (cm ²)	413.83
f_2 (cm)	0.0093
g (kN. cm ⁻²)	-12.930

Table 5.1: Candidate Pareto solution ($f_1 = 413.83$ cm², $f_2 = 0.0093$ cm)

For this candidate Pareto solution, three constraints on the input range are active: two constraints on the maximum bound for x_1 and x_2 , and one constraint on the minimum bound for x_3 .

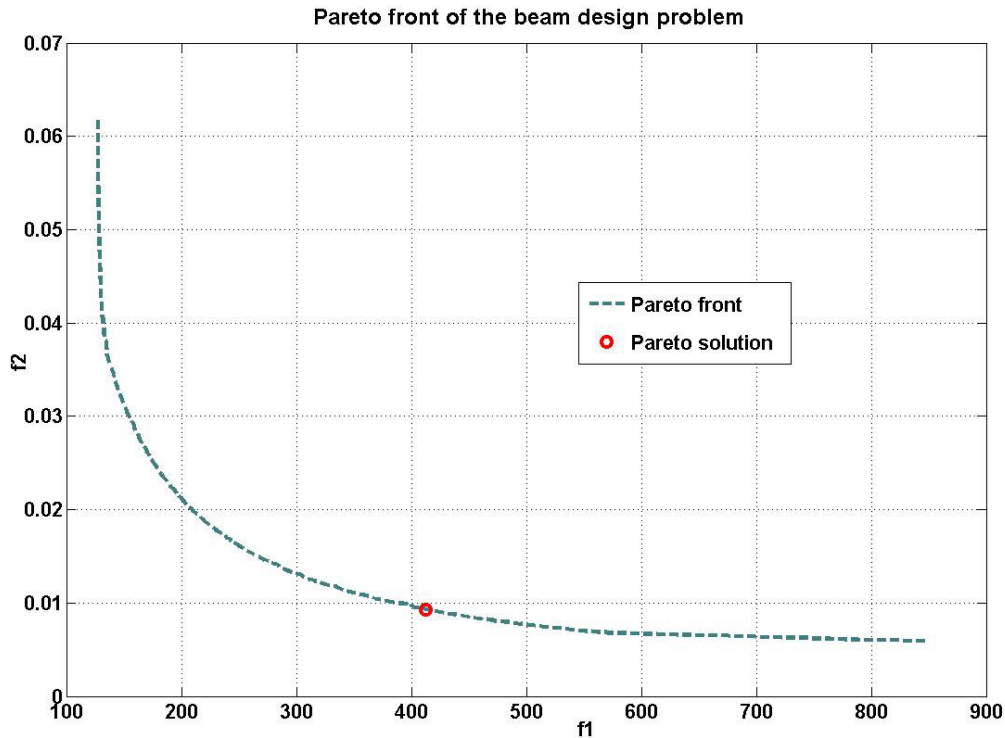


Figure 5.3: Candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2$, $f_2 = 0.0093 \text{ cm}$) in criterion space

Equations derived in chapter 4 are used to find the first and second order derivatives on the Pareto surface at this point and are given in Table 5.2.

Derivatives	
df_2/df_1	-2.29×10^{-5}
d^2f_2/df_1^2	1.26×10^{-7}

Table 5.2: First and second order derivative on the Pareto surface at candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2$, $f_2 = 0.0093 \text{ cm}$)

Simple considerations on the values for the first and second order derivatives provide an insight into the local trade-offs. The small value of the first order derivative suggests that to improve (decrease) slightly objective f_2 , a larger compromise (increase) in objective f_1 , must be conceded. Indeed, an improvement for f_2 , of 0.001 cm implies that

f_1 , must be increased of around 43 cm². The small value for the second derivative indicates that the local curvature is small and therefore the candidate solution is located on a flat portion of the Pareto front.

The linear and quadratic approximations are derived as follows:

$$f_2 = f_2^* + \frac{df_2}{df_1} \Delta f_1 \quad (5.8)$$

$$f_2 = f_2^* + \frac{df_2}{df_1} \Delta f_1 + \frac{1}{2} \frac{d^2 f_2}{df_1^2} \Delta f_1^2 \quad (5.9)$$

and plotted in Figure 5.4. Equations (5.8) and (5.9) provide us with the local description of the Pareto curve. However, more information can be obtained than just local approximate sections of the Pareto front.

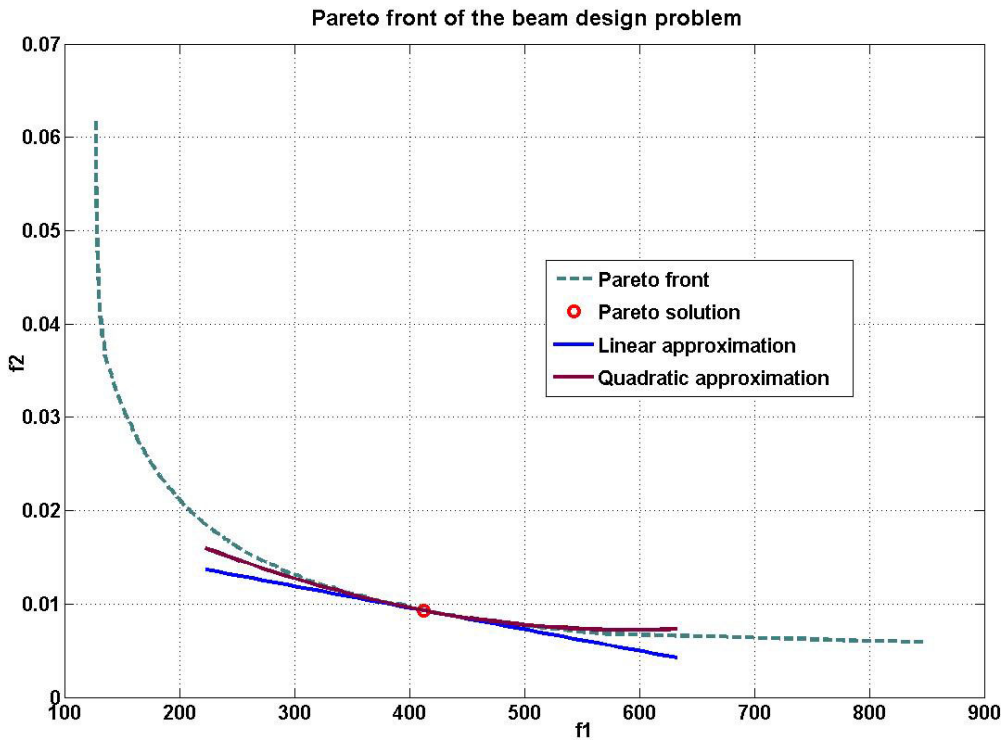


Figure 5.4: Linear and quadratic approximation at candidate Pareto solution
($f_1 = 413.83$ cm², $f_2 = 0.0093$ cm)

Indeed, in chapter 4, it was described how the extent of validity of the local approximation can be estimated and how solutions lying on the approximate portion on

the Pareto curve can be mapped back into the design space. With respect to the aforementioned analysis, the study of the extent of validity of the solution can be formulated as a single objective optimisation problem, which reduces to a linear problem in the case of two objectives:

Minimise and maximise: δf_1

Subject to:

$$-1 \leq \mathbf{x} + \delta \mathbf{x} \leq 1 \quad (5.10)$$

$$g + \frac{dg}{df_1} \delta f_1 + \frac{1}{2} \frac{d^2g}{df_1^2} (\delta f_1)^2 \leq 0 \quad (5.11)$$

$$\left| \frac{1}{2} \frac{d^2f_2}{df_1^2} \delta f_1 \right| / \left| \frac{df_2}{df_1} \right| \leq \alpha \quad (5.12)$$

where alpha is fixed to 0.2 for this test case. The values for inputs variables, objectives and constraint of the approximate solutions defining the bounds of validity of the Pareto approximation are given in Table 5.3. Here, the values for objective f_1 are the results of the linear optimisation problem and approximate values for f_2 and g are obtained with the quadratic approximation. Figure 5.5 plots the approximate solutions in the criterion space and Figure 5.6 shows the geometry of the corresponding beams.

	Approx solution 1		Pareto solution 1	Approx solution 2	
x_1 (cm)	80		80	80	
x_2 (cm)	50		50	50	
x_3 (cm)	0.9		0.9	0.9	
x_4 (cm)	2.74		3.48	4.22	
	Approx	Model	Model	Approx	Model
f_1 (cm ²)	341.175	341.175	413.83	486.493	486.493
f_2 (cm)	0.0113	0.0113	0.0093	0.0079	0.0078
g (kN. cm ²)	-12.232	-12.202	-12.930	-13.392	-13.412

Table 5.3: Inputs, objectives and constraint values of candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm}$) and approximate Pareto solutions associated with the limits of validity of the approximation

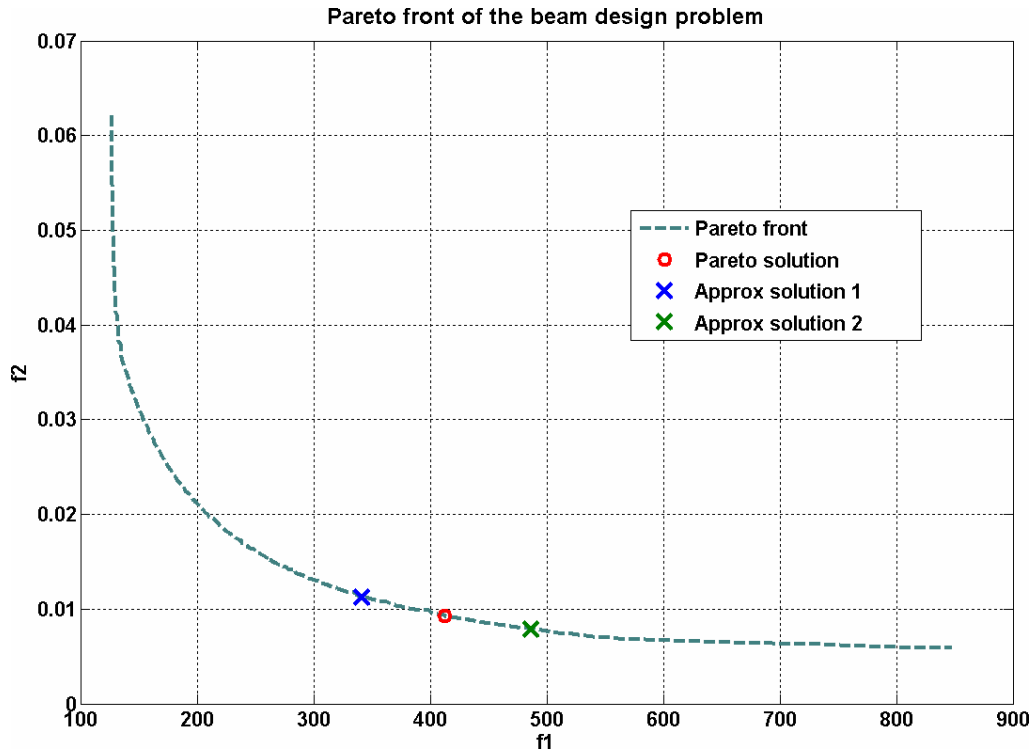


Figure 5.5: Extent of validity of approximation at candidate Pareto solution
 $(f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm})$

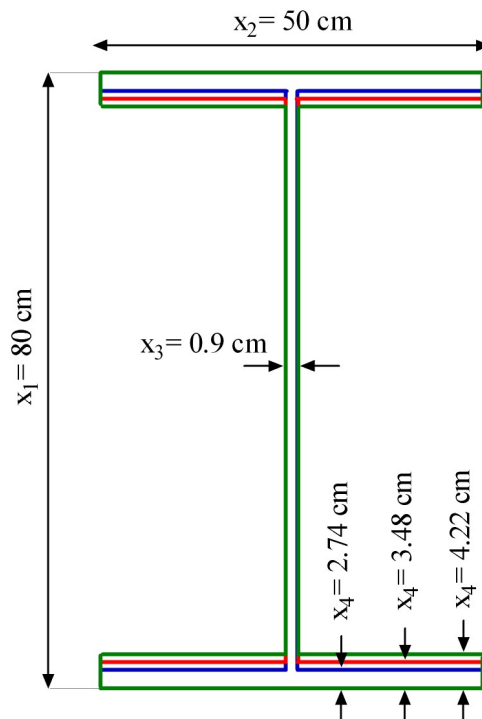


Figure 5.6: Beams associated with the limits of validity of the approximation at candidate Pareto solution $(f_1 = 413.83 \text{ cm}^2, f_2 = 0.0093 \text{ cm})$

The small value of the first order derivative of objective f_2 with respect to f_1 and the fact that no inactive constraints are close to activation, results in a large range of variation for f_1 associated with the extent of validity of the approximation.

The predictions given by the Pareto approximation are in accordance with the estimates obtained with the analytical equations. The small discrepancy on the strength constraint values obtained with the approximation and the model does not really matter as the constraint at the candidate Pareto solution is not close to being activated. Indeed, at the candidate Pareto solution, the first and second order derivatives for the strength constraint given in Table 5.4 are relatively small when compared to the actual value of the constraint and would require a much larger change in objective f_1 to activate it.

Derivatives	
dg_2/df_1	-7.98×10^{-3}
d^2g_2/df_1^2	4.48×10^{-5}

Table 5.4: First and second order derivative on the Pareto surface for strength constraint at candidate Pareto solution ($f_1 = 413.83 \text{ cm}^2$, $f_2 = 0.0093 \text{ cm}$)

The back-mapping analysis indicates what variables are changed as a result of the Pareto approximation. The analysis derived in chapter 4 was based on the assumption that active constraints remain active in the vicinity of the Pareto solution under study. For this candidate point, three constraints on the bounds of the inputs variables are active (x_1 , x_2 and x_3) and remain active for the Pareto approximation. Therefore, the Pareto approximation analysis indicates that locally, this part of the Pareto front corresponds to changes with respect to x_4 only. This can be clearly seen in Figure 5.6 where only the thickness of the foot and head of the beam, i.e., x_4 , is changed. Such a consideration provides the designer with valuable information about the Pareto solution in the vicinity of the point under study. It also indicates what parameters can be adjusted locally if the compromise solution does not entirely satisfy some preferences on the type of solutions that were not articulated in the formulation of the optimisation problem.

As mentioned in chapter 4, the Pareto approximation comes at a reduced cost compared to Tappeta *et al.* (2000). Gradients and Hessians of the objectives and constraints are required to obtain the first and second order derivatives. In the case of a gradient-based optimiser, obtaining the gradients at the point under study has no cost

since these are obtained during the optimisation procedure. If Hessians are not calculated during the optimisation, they must be computed if the designer wants to make use of the approach described to assess the extent of validity of the Pareto approximation. Dennis and Schnabel (1983) show that in the case of forward differences, when the gradients are already available, the Hessian is obtained at an additional cost of $(n^2+n)/2$ where n is the number of variables. In the case of central differences, Abramowitz and Stegun (1972) show that the additional cost of obtaining the Hessian with finite differences is $2n^2$. Finally, two extra runs of the model: one for each bound of the extent of validity, allow to verify that the approximation is consistent with the model and can be used safely.

The second candidate solution considered is given in Table 5.5. Two constraints on the minimum bounds for x_3 and x_4 are active at this point. The constraint on the strength is also activated here. This means that on this portion on the Pareto surface, input variables x_3 and x_4 are fixed to their minimum value. Only x_1 and x_2 are left free to move along the Pareto curve. As mentioned before, the apparent smoothness of the Pareto surface given by the high density of Pareto points obtained led to the conclusion that the Pareto surface was differentiable. However, the set of active constraints at this Pareto point is different from the one at the first candidate point. As explained in chapter 4, it shows that a change in the set of active constraints on the Pareto surface does not necessarily indicate non-differentiability. Only the study of the local normal vector to the Pareto surface can reveal whether the surface is smooth or not.

Pareto solution 2	
x_1 (cm)	70.72
x_2 (cm)	37.05
x_3 (cm)	0.90
x_4 (cm)	0.90
f_1 (cm ²)	128.718
f_2 (cm)	0.0472
g (kN. cm ⁻²)	0.000

Table 5.5: Candidate Pareto solution ($f_1 = 128.718$ cm², $f_2 = 0.0472$ cm)

Compared to the first candidate Pareto point, this one corresponds to a very different solution. Indeed, its cross-section area is much smaller than the previous one: 128.7 cm^2 compared to 413.8 cm^2 . A 70% reduction in weight has been traded for a 0.0379 cm increase in displacement. However, this came at a price: the constraint on the maximum stress allowable is now activated. It means that the current point corresponds to a critical solution where an essential structural constraint is strictly satisfied, with no margin.

The first and second order derivatives on the Pareto surface in the criterion space are given in Table 5.6.

Derivatives	
df_2/df_1	-4.01×10^{-3}
d^2f_2/df_1^2	1.42×10^{-3}

Table 5.6: First and second order derivative on the Pareto surface at candidate Pareto solution ($f_1 = 128.718 \text{ cm}^2$, $f_2 = 0.0472 \text{ cm}$)

The first order derivative is clearly larger than the first candidate solution, which indicates that the point corresponds to a part of the Pareto curve with a steeper slope. It also shows that at this point, a smaller cross-section area, i.e., less weight, must be sacrificed to significantly decrease the displacement. However, compared to the first candidate solution, the relatively large value for the second order derivative suggests that the quadratic approximation will depart early from the linear approximation. A direct consequence is that the analysis of the extent of validity of the approximation will give a limited range for legitimate variations of objective f_1 .

The extent of validity of the approximation is estimated with the same approach, i.e., by trying to minimise and maximise the change in objective f_1 , subject to constraints defined by equations (5.10), (5.11) and (5.12). The values for these input variables, objectives and constraint of the approximate solutions defining the bounds of validity of the Pareto approximation are given in Table 5.7, while Figure 5.7 plots the approximate solutions in the criterion space. As previously discussed, values for f_1 correspond to the solution of the optimisation problem that is solved to assess the extent of validity of the approximation. Values for objective f_2 and constraint g are given by

their respective quadratic approximation. The design points associated to the bounds of validity of the Pareto approximation are found using back-mapping to the design space. The resulting beam designs are given in Figure 5.8.

	Approx solution 1		Pareto solution 1	Approx solution 2	
x_1 (cm)	66.91		70.72	74.53	
x_2 (cm)	38.33		37.05	35.77	
x_3 (cm)	0.90		0.90	0.90	
x_4 (cm)	0.90		0.90	0.90	
	Approx	Model	Model	Approx	Model
f_1 (cm ²)	127.588	127.588	128.718	129.845	129.848
f_2 (cm)	0.0527	0.0521	0.0472	0.0436	0.0430
g (kN·cm ⁻²)	0.000	0.046	0.000	0.000	0.045

Table 5.7: Inputs, objectives and constraints values of candidate Pareto solution ($f_1 = 128.718$ cm², $f_2 = 0.0472$ cm) and approximate Pareto solutions associated with the limits of validity of the approximation

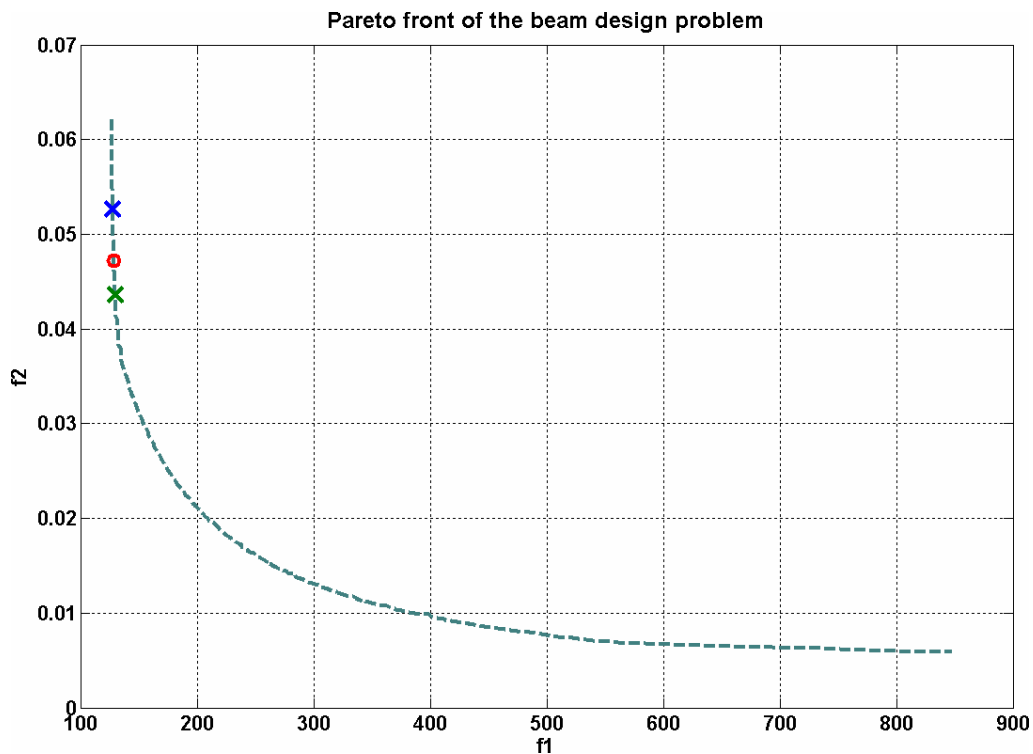


Figure 5.7: Extent of validity of approximation at candidate Pareto solution ($f_1 = 128.718$ cm², $f_2 = 0.0472$ cm)

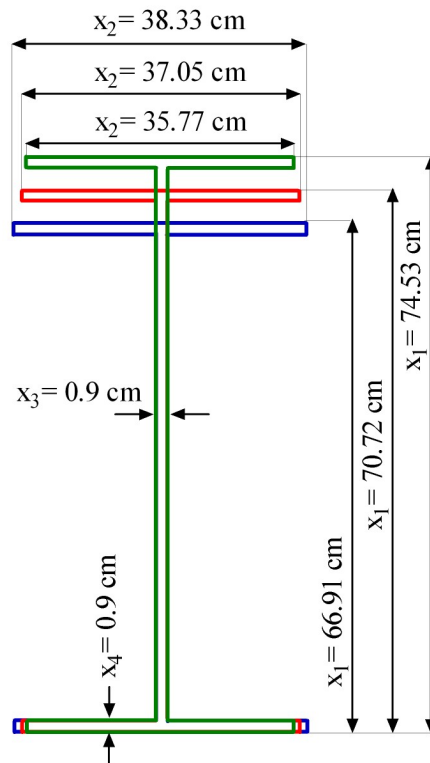


Figure 5.8: Beams associated with the limits of validity of the approximation at candidate Pareto solution ($f_1 = 128.718 \text{ cm}^2$, $f_2 = 0.0472 \text{ cm}$)

It is worth noting that when a constraint is active at the point under study, the approximation guarantees that it will remain active and therefore will not be violated within the extent of validity of the approximation. However, when moving away from the point under study, the model will most of the time depart from the approximation. Therefore, it is difficult to ensure that the active constraints will strictly remain active. Indeed, Table 5.7 shows that at the bounds of the approximation, the strength constraint does not remain active exactly. The assumption behind the approximation theory is that locally the move in the design space to accommodate for the change in the objective functions on the Pareto surface is locally normal to the set of active constraints. For active bound constraints, it ensures that the constraint will remain active in this direction, however, for any other active constraint, this is only valid locally.

5.4 Conclusions

In this chapter, the Pareto approximation methodology proposed in chapter 4 is used in the context of a typical engineering multi-objective optimisation: beam design problem.

It is described in detail how the approximation can be used at a discrete number of Pareto points to gain valuable information about the local sensitivity of the Pareto surface. It is shown how the approximation derived in the criterion space enables to obtain new approximate Pareto solutions. The back-mapping allows the corresponding design in the design space to be obtained. It indicates what design variables can be modified to remain locally on the Pareto surface, and to what extent these inputs must be changed. Therefore, input variables can be adjusted according to the designer preferences on the objectives that are not included in the initial formulation of the optimisation problem.

Approximations are derived for the constraint to ensure that the constraint is not violated within the extent of validity of the approximation.

The Pareto approximation is entirely based on first and second order derivatives of the objectives and constraints and therefore comes at a reduced cost. It is also suggested to perform a small number of runs on the true model to ensure the quality of the approximation at the bounds of the range of validity.

The Pareto approximation analysis provides valuable information about the Pareto surface to facilitate the trade-off study between available solutions of the multi-objective optimisation problem.

Chapter 6

Airfoil design optimisation

In this chapter, a multi-objective airfoil optimisation is considered. The implementation of the problem together with the surrogate model strategy adopted for this test case are described. Variance-based indices are obtained for each criterion and constraint to identify non-significant design variables. The reduced problem is formulated and the set of Pareto solutions is obtained. The results are compared with those obtained without reduction of dimensionality and followed by a discussion on the performance of the method. Then, local Pareto approximations described in section 4, are obtained for several candidate Pareto solutions to illustrate the advantages of the method in a complex design optimisation situation.

6.1 Description of the problem and model implementation

This section concerns the implementation of the model used for this test case. More precisely, it describes the chosen method for the airfoil parameterisation and the aerodynamic code used to assess the aerodynamic performance.

6.1.1 PARSEC-11 parameterization

In this test case, the PARSEC-11 parameterisation proposed by Sobieczky (1998), was used to represent the airfoil with 11 geometric variables as shown in Figure 6.1. A description of all input parameters is given in Table 6.1.

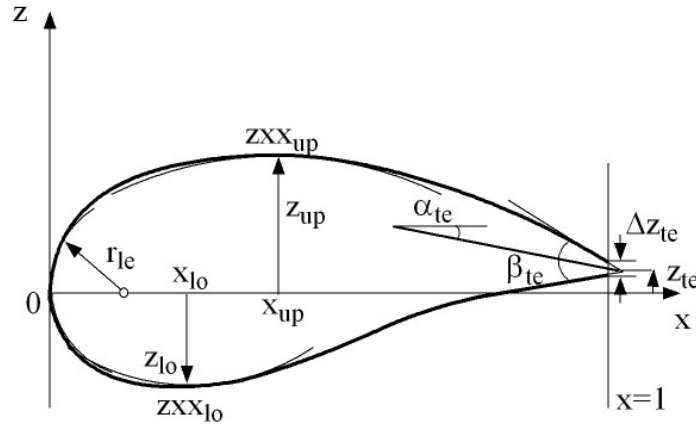


Figure 6.1. PARSEC-11 airfoil parameterization

Inputs	Description
r_{le}	Leading edge radius
x_{up}	Upper surface max thickness location
x_{lo}	Lower surface max thickness location
z_{up}	Upper surface max thickness
z_{lo}	Lower surface max thickness
ZXX_{up}	Upper surface curvature at x_{up}
ZXX_{lo}	Lower surface curvature at x_{lo}
z_{te}	Trailing edge position
Δz_{te}	Trailing edge thickness
β_{te}	Trailing edge aperture angle
α_{te}	Angle of trailing edge bisector

Table 6.1: Description of model inputs

A polynomial is used to describe the upper and lower surface:

$$z = \sum_{n=1}^6 a_n(\mathbf{p}) x^{n-1/2}, \quad (6.1)$$

where x varies between 0 and 1. The coefficients a_n of the polynomial are real and depend on the design vector \mathbf{p} made of the 11 geometric parameters

$$\mathbf{p} = \left[r_{le} \quad x_{up} \quad x_{lo} \quad z_{up} \quad z_{lo} \quad ZXX_{up} \quad ZXX_{lo} \quad z_{te} \quad \Delta z_{te} \quad \beta_{te} \quad \alpha_{te} \right]. \quad (6.2)$$

Here, the coefficients a_n of the polynomial are not taken as design parameters. Instead, the geometric parameters are used to define the polynomial for the upper and lower surface, which entirely determine the shape of the airfoil. The 11 geometric parameters are used to derive 12 equations on the airfoil shape based on the radius of leading edge, position of maximum thickness, curvature and so on (see Appendix C). It results that the geometric parameters and polynomial coefficient are related through a linear algebraic system of 12 equations. Once the geometric parameters are given, the linear system is solved for the coefficients a_n to obtain the polynomials describing the upper and lower surface. The high order polynomial representation of PARSEC-11 might result in the upper or/and lower surface to have more than one relative maximum for particular combinations of these parameters. This implies that humps might appear locally on the airfoil surface. This feature can be interesting in the context of airfoil optimisation as these humps might have a significant impact on the flow characteristics but it is not clear how they can be controlled with the geometric parameters. This problem is discussed later in this chapter.

The PARSEC parameterization is a popular and efficient approach for transonic wing profile parameterization (Jeong, 2005; Oyama, 2000; Oyama et al., 2000a, 2000b; Vavalle, 2005). Its most attractive feature is the use of a reduced number of intuitive and explicit geometric design parameters that the designer can directly relate to changes in the flow characteristics. On one hand, Oyama (1999) shows that for the optimisation of a transonic airfoil for lift to drag ratio with evolutionary algorithm, PARSEC-11 leads to a better design than a B-spline parameterisation with a similar number of design parameters (11 for PARSEC and 12 for B-spline). On the other hand, Vavalle (2005) and Kumano (2006) report that PARSEC-11 sometime fails to describe accurately well-known existing profiles. Another parametric geometry description based on polynomials has been developed by Kulfan (2007). In parameterisation, the *Class function/Shape function Transformation* (CST) extends to different types of geometry and allows for an intuitive description using a reduced number of variables. The geometry is described with a *Class function*, which defines classes of geometries (airfoils, axi-symmetric bodies, nacelles...), and a *Shape function*, which defines specific geometric shapes within each class. However, for this test case, the PARSEC-11 describes a design space, which is complex enough to apply the proposed

methodology and demonstrate its potential to assist the study of multi-objective optimisation problems. PARSEC-11 representation was implemented in a Matlab script to generate the airfoil geometry.

6.1.2 VGK

The aerodynamic features (C_d and C_m) of the airfoil were obtained using an ESDU code, VGK (ESDU 96028, ESDU 96029), which was developed for subsonic free streams and includes viscous and weak transonic effects. VGK (Viscous Garabedian-Korn) is a FORTRAN code developed by DERA to predict two-dimensional transonic attached flow around an airfoil placed in a subsonic freestream, with allowance for viscous effect. Solutions of inviscid flow about an airfoil and displacement effects of boundary layer and wake are coupled to compute the flow around the profile. Laminar and turbulent boundary layer elements are calculated with integral methods. The computation of displacement and momentum thickness takes into account the effect of pressure obtained from the solution of the inviscid flow. The coupling between potential equations and boundary layer effect is obtained by iterative procedure and allows to reach a converged and consistent solution of the flow around the airfoil.

The equations governing the region of inviscid flow are the full potential equations for steady compressible gas. These equations are exact in the case of shock-free flow and only approximate for flows with relatively strong shocks. Nevertheless, VGK can still predict a solution whose pressure distribution over the surface is close to the true physical flow provided the shock is not too strong. The viscous solution is obtained by means of integral turbulent boundary layer method where high-order effects disregarded in the first-order boundary theory (zero normal pressure gradient) are taken into account. VGK has good accuracy for viscous flows with attached boundary layer and weak shock waves, for $M_{L_{shock}} < 1.3$. When the boundary layer is close to separation, valuable information about the flow can still be obtained even if the prediction is not regarded as accurate (ESDU 96028).

The inviscid flow potential equations are solved on a mesh by means of iterative finite-difference scheme with a relaxation method. A conformal mapping function generates the mesh in the external region of the airfoil. Airfoil surfaces with concave

regions and/or rapid changes of slope may cause the mesh generation to diverge/crash or lead to anomalies in the mesh. This must be avoided as a poor mesh will result in poor estimates of the aerodynamic characteristic of the airfoil. VGK uses a two-level mesh strategy to obtain the solution of the inviscid flow. A number of iterations is first performed on a coarse mesh to speed up the procedure, followed by a larger number of iterations on a fine mesh to achieve a converged and accurate solution. Boundary layer conditions updates are made at regular intervals during the iterative process for solving the inviscid flow on either the coarse or fine mesh. When separation appears, i.e. the friction coefficient goes to zero, VGK has a strategy to compute the flow. It fixes the friction coefficient to a value close to zero, $c_f = 2 \times 10^{-6}$, which allows to complete the boundary layer calculation. This is how VGK handles separation bubbles. However, when VGK fixes the friction coefficient to this value at more than two locations on either upper or lower surface, it is recommended to consider the results of the flow with care as they might not be reliable (ESDU 96028, 96029).

In VGK, two different approaches are used to determine the drag coefficient: *near-field* and *far-field* approach. The former is based on the evaluation of the total drag by integrating *pressure* and *skin friction* over the airfoil surface: $C_d = C_{dp} + C_{df}$. In the latter, the overall drag is calculated as the sum of a viscous term and a wave drag component due to the presence of a shock: $C_d = C_{dv} + C_{dw}$. The *viscous* drag component is derived from the momentum thickness of the wake far downstream of the airfoil. The *wave* drag component is computed by means of two methods incorporated in the solver and referred as *first-order* and *improved* method. Both are based on the assumption that the shock is normal to the airfoil surface but the *improved* approach makes allowance for effects of the profile upstream the shock location and often leads to better prediction. The *far-field* approach is recommended in the case of shock-free flows, but there is no clear indication in the case of flows with significant wave drag (ESDU 96028). Also in the case of the *near-field* approach, the inherent difficulty in the evaluation of the pressure term by means of numerical integration can lead to phenomena of numerical cancellation and the consequent numerical instabilities. Therefore, the *improved far-field* approach was preferred to estimate the overall airfoil drag. The FORTRAN code VGK was relatively fast to run (2-3 seconds) and could be

easily implemented in a Matlab environment for immediate use with an existing in-house developed optimisation tool.

6.1.3 Implementation for optimisation

VGK was originally conceived to work under the direct control of the user, who had to provide the inputs and analyze the outputs to read the required results and check for their soundness and accuracy. Both inputs and outputs are written to text files, which work as interfaces for the program core. In order to automate this process in a Matlab environment, two appropriate preprocessing and postprocessing modules are created. The overall configuration of the CFD code used for this test case is shown in Figure 6.2.

The preprocessing module is a Matlab *mex* function, which links the original preprocessing code for VGK: *vgkcon.for*, to the Matlab main script. The *mex* function allows for a readily modification of *vgkcon.for* input parameters from the main Matlab script. These inputs include the conditions of the simulation, (fixed incidence angle/fixed C_l), flow conditions (Mach and Reynolds number), parameters regarding the solver (inviscid/viscous number of iterations/relaxation factors), the transition locations for boundary layer (upper and lower surface), and so forth.

In this test case, the optimisation is performed for a fixed lift coefficient and the incidence angle is adjusted by the solver during the flow calculation to accommodate for the specified lift coefficient. There is no transition prediction implemented in VGK, instead, the user must specify the transition location. If a laminar separation occurs upstream of the specified separation point, transition to turbulent boundary layer is assumed where the laminar flow separates. In practice, many different factors can affect the location of the transition point and it becomes difficult to provide the VGK with a valid transition location before the flow around the airfoil is calculated, especially when VGK is used as a black box. In this case, it is proposed to fix it to the usual transition location of a small passenger transport aircraft. Typically, $x_{TU}/x_{TL} = 1\%$ of the chord where x_{TU} and x_{TL} are the position of the boundary layer transition point on the upper and lower surface respectively. The VGK settings adopted for this test case are given in Table 6.2.

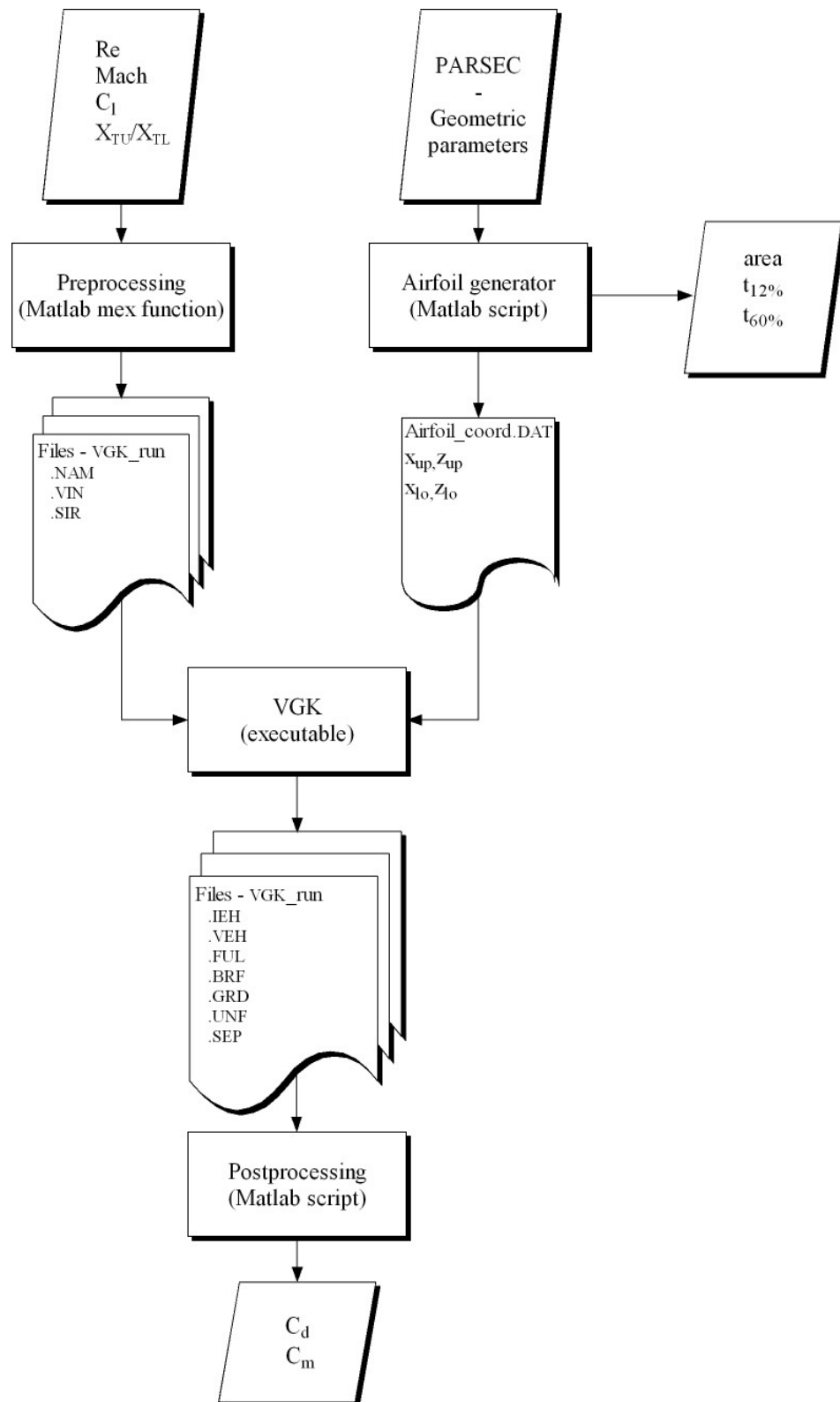


Figure 6.2: Overall configuration of the model

Airfoil flow conditions	
C_l	0.71
Mach	0.72
Re	21×10^6
x_{TU}	0.01
x_{TL}	0.01

Table 6.2: VGK settings for flow calculation about an airfoil

In ESDU 96028 and ESDU 96029, it is suggested to check the output files written by VGK to verify that the results obtained are reliable. These checks must be performed by the user on the various output files produced by VGK. In this test case, the model is included in an optimisation loop where calls to VGK are automated. Therefore, Matlab routines are written to perform the appropriate checks on the relevant output files.

- *check_crash* ensures the VGK code is executed without failure from the main Matlab script.
- *check_diverged* reads VGK output file .FUL to ensure that the run has not diverged.
- *check_converged* reads the residuals of inviscid flow and boundary layer calculations from VGK output files .IEH and .VEH respectively, to ensure that the run is sufficiently converged.
- *check_cf* reads VGK output file .FUL to ensure that the friction coefficient c_f has not been fixed to 2×10^{-6} more than twice.
- *check_hbar* reads VGK output file .FUL to ensure that the shape parameter has not reach its limit value on either upper or lower surface.

The post processing module sequentially scans the various output text files produced by the VGK. If a check fails, it automatically adjusts some parameters or uses a continuation run procedure to obtain reliable results corresponding to this particular run.

First, if *check_crash* or *check_diverged* fails, the VGK simulation is run again for smaller values for the relaxation factors for supersonic and subsonic region of the inviscid flow, XM and XS respectively. For particular profiles and transonic flow conditions, regions of the flow around the airfoil might be difficult to compute. If diminishing the relaxation factor does not help the convergence, the Mach number is reduced to a value for which the flow can be computed and increased iteratively. In this continuation procedure, the flow of the previous step is used as a starting point for the computation of the new flow for an increased Mach number. When VGK has not crashed or diverged, *check_converged* is performed to make sure that the residuals are small enough and the run is sufficiently converged. Then, *check_cf* and *check_hbar* are carried out to verify that there is no boundary layer separation. If *check_converged* or *check_cf* or *check_hbar* fails, a continuation run starting with the result of the current run is performed. In this continuation procedure, the viscous relaxation factors are decreased and the number of inviscid iterations between two boundary layer updates is doubled. If all checks are fine, the results are considered reliable and a routine reads the respective values for C_d and C_m . In the case when a check fails and the parameter adjustments do not help, suitable values for C_d and C_m are passed to the optimiser to penalize that particular combination of independent variables. The postprocessing flowchart is given in Figure 6.3.

It is also recommended to check the quality of mesh generated around each airfoil (ESDU, 96028; ESDU, 96029). However, when VGK is wrapped in a Matlab script to be called as a black box into an optimisation loop, this is not possible. It can happen that when VGK is fed with an odd profile, the mesh is not properly generated. In the case where the mesh generation process crashes, the flow around the airfoil cannot be computed and the model simply returns penalisation values for the drag and pitching moment. However, when a poor mesh is generated a converged flow might be obtained, but the estimate should not be considered reliable. Such situation is a problem as it is difficult to distinguish between good and poor quality meshes when VGK is executed automatically. In the next section, it is shown that some of the poor quality meshes or failed runs are caused by an erroneous parameterization of the airfoil.

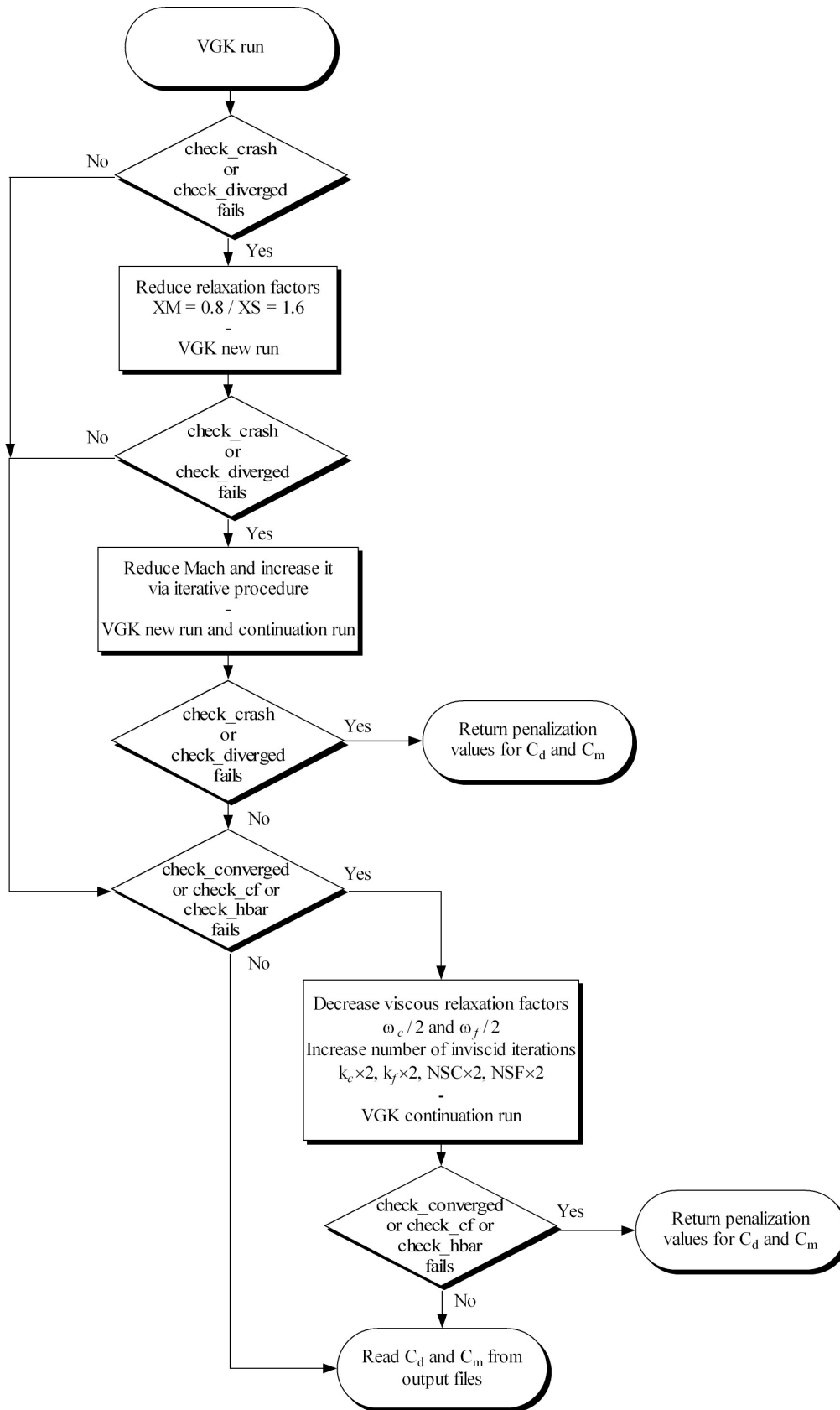


Figure 6.3: Postprocessing checks flowchart

6.2 Optimisation strategy

6.2.1 Initial range of variation and PARSEC-11 issues

An initial range of variation for PARSEC-11 input variables was defined and is given in Table 6.3.

Inputs	Min value	Max value
r_{le}	0.005	0.00938
x_{up}	0.3	0.45
x_{lo}	0.3	0.56
z_{up}	0.045	0.057
z_{lo}	-0.058	-0.04
zxx_{up}	-0.555	-0.1
zxx_{lo}	0.28	1.1
z_{te}	-0.02	0.0
Δz_{te}	0.005	0.0082
β_{te} (rad)	0.003491	0.41958
α_{te} (rad)	-0.20944	0.0

Table 6.3: Initial input range of variation

Preliminary results obtained on this range of variation showed that many VGK runs crashed or failed to converge, even after the checks and continuation run procedure. Giving penalisation values to C_d and C_m corresponding to these failed runs was not a suitable strategy. Indeed, the large number of failed runs indicated that the optimiser was stuck in a particular region of the design space corresponding to profiles that could not be calculated by VGK.

These odd profiles are essentially due to the formulation of PARSEC-11 parameterization. Both upper and lower surface are described by equation (6.1), which can be re-written as:

$$z = x^{1/2} \left(a_1(\mathbf{p}) + \sum_{n=2}^6 a_n(\mathbf{p}) x^{n-1} \right) \quad (6.3)$$

where $x^{1/2}$ will provide a round nose for the airfoil and the function in brackets is a 5th order polynomial. Using the 11 parameters, geometric conditions regarding airfoil shape are derived to fit the polynomial between $x = 0$ and $x = 1$ corresponding to the leading and trailing edge positions respectively. An adverse feature of high order polynomials when their coefficients are derived to fit particular conditions, is their tendency to

produce uncontrolled oscillations. In the case of transonic airfoil design, small bumps on either the upper or the lower surface can lead to interesting designs. However, when the size of these oscillations is too large, PARSEC-11 can fail to give meaningful profiles, which will result in the mesh generation procedure to crash or produce poor quality mesh.

In PARSEC-11 formulation, the position of the maximum thickness is given as an input and is used to derive a condition on the airfoil surface to determine the coefficients of the polynomial. A first order condition is used to impose the position of maximum thickness on either upper or lower surface. It is simply required that the first order derivative of the polynomial given in equation (6.1) must be zero for x_{up} and x_{lo} (see Appendix C). Such condition is necessary but not sufficient, which means it ensures that x_{up} and x_{lo} are a local maximum or minimum respectively, but does not guarantee that it is the only one between 0 and 1. In this situation, the airfoil given by PARSEC does not correspond to the one we intend to describe. The resulting profile is therefore not meaningful. Such situation could occur within the bounds of the initial range of variation given in Table 6.3. An example is shown in Figure 6.4, where both airfoils correspond to the same parameterization, but the z axis for the upper airfoil has been stretched to make the variation of thickness more obvious. The corresponding parameter values are given in Table 6.4. Another type of meaningless profile corresponds to the situation when upper and lower surface intersect. In this case, the mesh generation algorithm will fail to build a mesh around the airfoil and crash. In PARSEC-11 formulations, no condition is derived to impose that upper and lower surface do not intersect. An example of airfoil where upper and lower surfaces intersect for parameter values lying within the initial bounds is plotted in Figure 6.5. The corresponding geometric inputs are given in Table 6.5.

It is shown that for particular combinations of values for the input parameters, PARSEC-11 fails to generate the corresponding profile. This issue is entirely due to its representation using high order polynomial which exhibits some adverse features such as uncontrolled oscillations on the resulting surface.

In the next section, it is attempted to identify the regions of the design space that correspond to erroneous profiles in order to redefine a valid input range of variation for PARSEC-11.

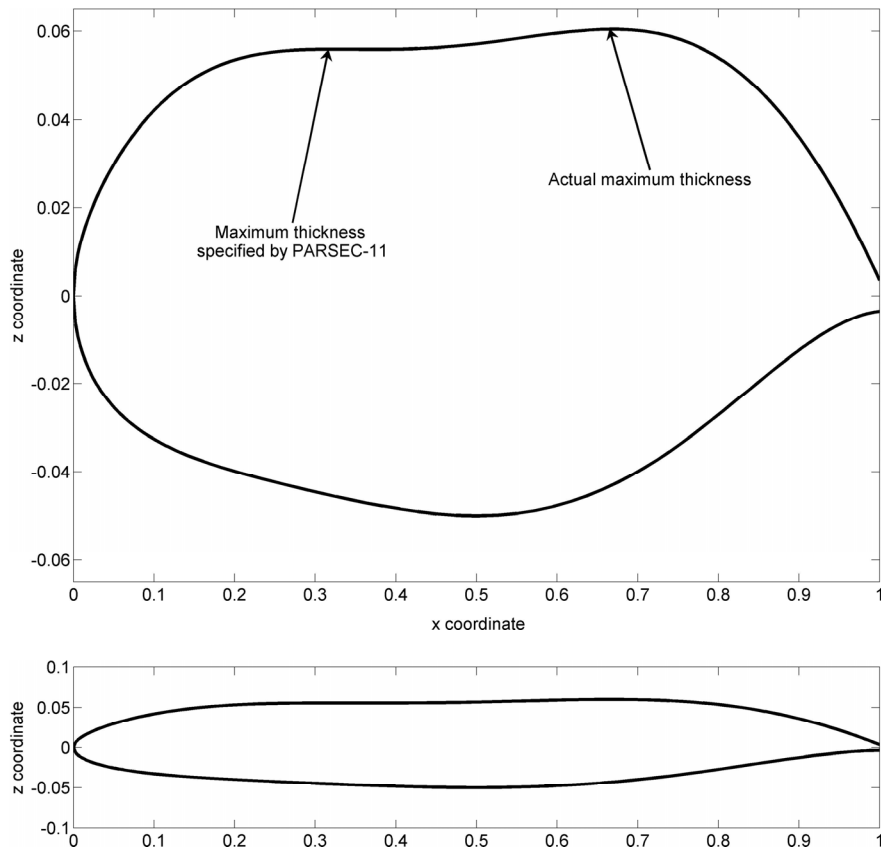


Figure 6.4: PARSEC-11 erroneous parameterization – Inconsistent maximum thickness location

Inputs	Value
Γ_{le}	0.008
X_{up}	0.32
X_{lo}	0.5
Z_{up}	0.056
Z_{lo}	-0.05
ZXX_{up}	-0.11
ZXX_{lo}	0.4
Z_{te}	0.0
ΔZ_{te}	0.007
β_{te} (rad)	0.4
α_{te} (rad)	-0.18

Table 6.4: Example of PARSEC-11 input values giving an airfoil with inconsistent maximum thickness location

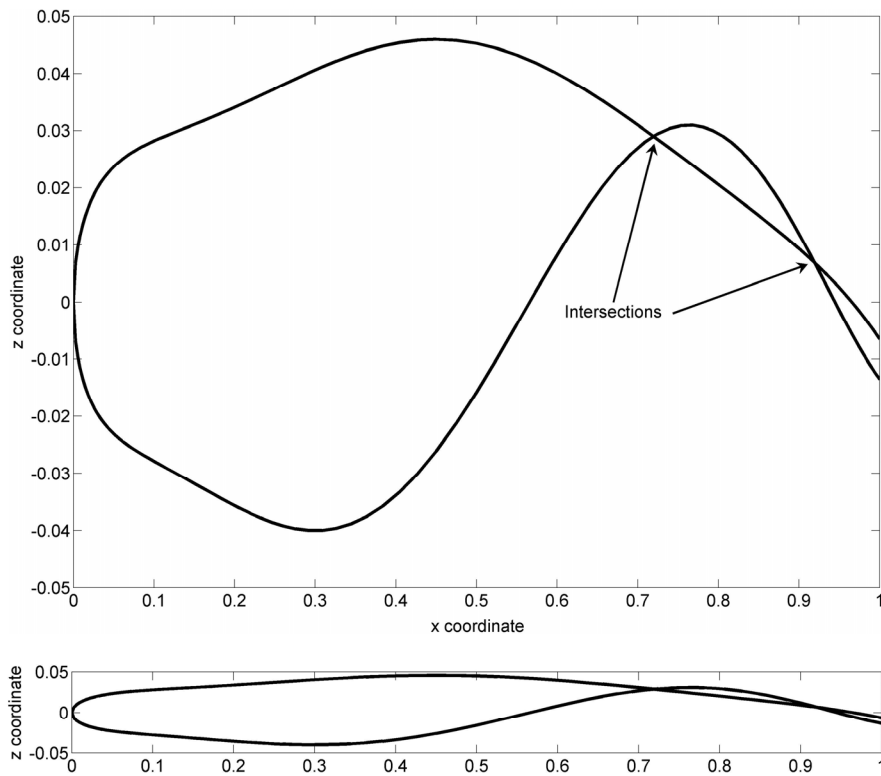


Figure 6.5: PARSEC-11 erroneous parameterization – Intersection between upper and lower surfaces

Inputs	Value
r_{1e}	0.009
x_{up}	0.45
x_{lo}	0.3
z_{up}	0.046
z_{lo}	-0.04
zxx_{up}	-0.55
zxx_{lo}	1.1
z_{te}	-0.01
Δz_{te}	0.007
β_{te} (rad)	0.02
α_{te} (rad)	-0.2

Table 6.5: Example of PARSEC-11 input values giving an airfoil where upper and lower surface intersect

6.2.2 Definition of a suitable range of variation for input variables

The strategy adopted to define a valid range of variation was to use SOM to identify the regions of the design space that correspond to erroneous PARSEC-11 parameterizations. SOM have the ability to classify large amount of data and identify clusters. The idea is to understand how these erroneous profiles are clustered in the design space and reduce the input ranges to discard these regions.

The geometry generator is run on a large LP-Tau design of experiments (Statnikov and Matusov, 1995) to screen the design space. Checks for the position of maximum thickness and detection of intersection between upper and lower surfaces are performed to identify erroneous profiles. A large number of erroneous profiles is required to build a meaningful SOM. Here LP-Tau sampling is very convenient as it ensures the sample is evenly spread and points from smaller sets are re-used for larger sets. Therefore, the sample can be iteratively increased until a sufficiently large number of erroneous profiles is obtained. For this test case, an LP-Tau sampling of 131072 points was used and 2052 erroneous profiles were identified. The large sample size was not a problem as the geometry generator was extremely fast to run and the whole analysis only takes a few minutes.

The SOM of erroneous profile is built and given in Figure 6.6. The component maps corresponding to the U-matrix and the 11 parameters of PARSEC parameterization are plotted. The U-matrix (top left corner) gives an indication of how the data points are clustered. Low values correspond to clusters, while high values correspond to borders between clusters. Here, the U-matrix clearly shows that most of the erroneous profiles are clustered together (blue region) in the same region of the design space. This region corresponds to low values of x_{up} , high values of $z_{xx_{up}}$ and z_{te} , and extreme values (low and high with respect to the input range) of β_{te} and α_{te} . Therefore, to discard the region corresponding to meaningless profiles, the range of variation for these inputs is truncated. The new reduced range of variation is given in Table 6.6. The same screening analysis with same size LP-Tau sampling was carried out on the reduced design space. Out of the 131072 sample profiles, only 2 failed. This method allows for a significant reduction of the number of erroneous profiles generated by the PARSEC-11 parameterization.

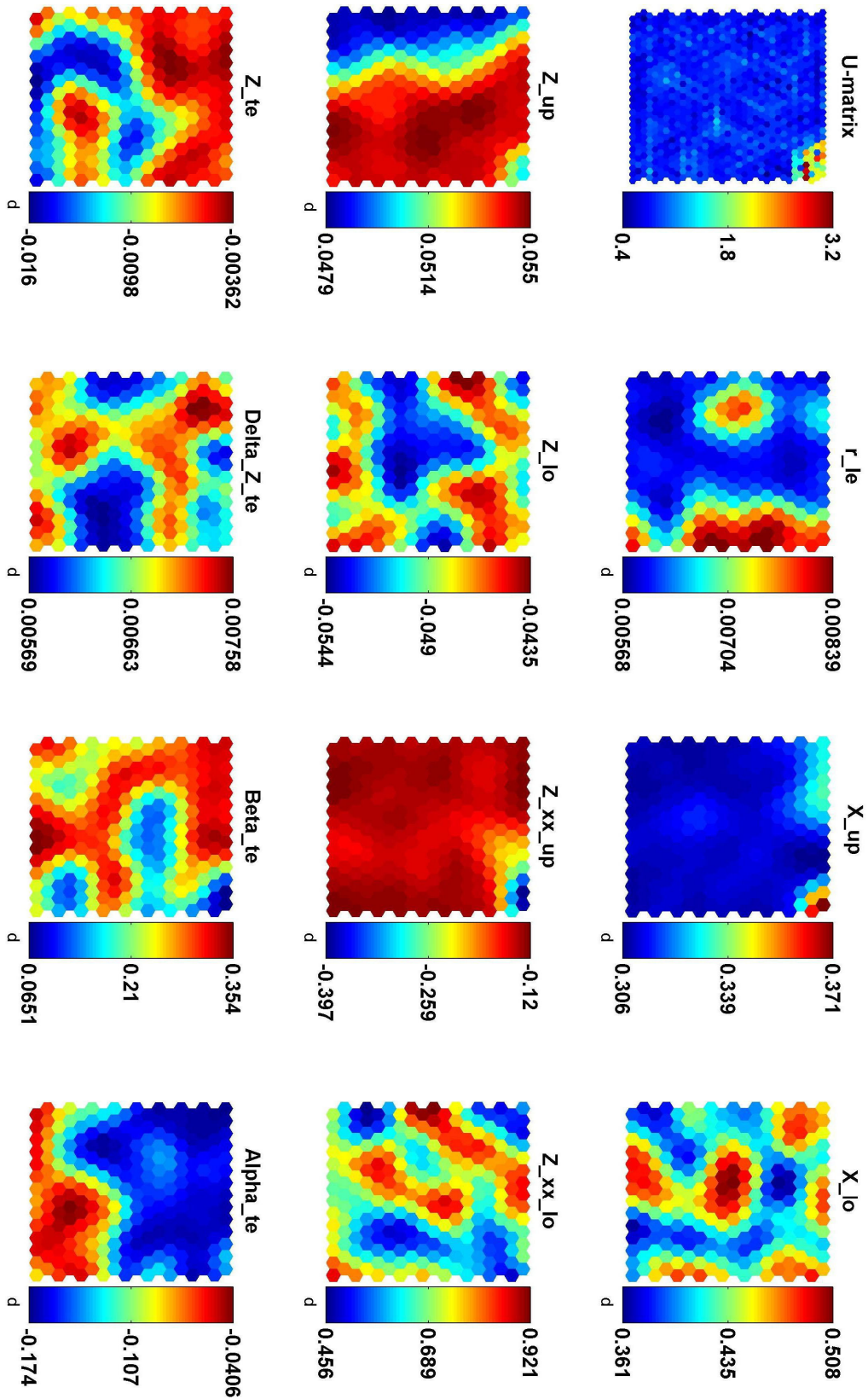


Figure 6.6: SOM of erroneous PARSEC-11 profiles obtained on initial input range

This also allows to concentrate the optimisation on regions of the design space where the PARSEC-11 parameterization is meaningful.

Inputs	Min value	Max value
r_{le}	0.005	0.00938
x_{up}	0.36	0.45
x_{lo}	0.3	0.56
z_{up}	0.045	0.057
z_{lo}	-0.058	-0.04
$z_{xx_{up}}$	-0.555	-0.26
$z_{xx_{lo}}$	0.28	1.1
z_{te}	-0.02	-0.009
Δz_{te}	0.005	0.0082
β_{te} (rad)	0.1	0.29
α_{te} (rad)	-0.13	-0.08

Table 6.6: Reduced input range of variation

6.2.3 Kriging adaptive sequential training

The noise inherent to VGK in the estimates of C_d and C_m implies that any gradient-based method (optimisation or sensitivity analysis) will perform poorly as the output response is not smooth. Therefore, a surrogate model was built for both C_d and C_m . Another consequence is that it will also speed up the optimisation procedure. The choice went for a Kriging model as it presents some attractive features for this test case. First, Kriging models are capable of accurately approximating complex functions and usually perform better than low order polynomial response surfaces (Jones *et al.*, 1998; Simpson *et al.*, 2001, 2004; Van Beers and Kleijnen, 2003). Secondly, their formulation allows to calculate analytically the sensitivity indices (Chen, 2005) as described in Appendix D. Finally, as a surrogate model, the resulting output response will be noiseless, continuous and differentiable.

An appropriate design of experiments (DOE) strategy is necessary to obtain a suitable Kriging. Sequential designs (Martin and Simpson, 2002; Van Beers and Kleijnen, 2003) usually perform much better than standard space filling designs. They are based on adding iteratively to the training sample, points of a design of experiment with large prediction error variance. Different techniques such as cross-validation and parametric bootstrapping can be used to select the candidate points (Martin and Simpson, 2002; Van Beers and Kleijnen, 2003) but they become burdensome when the size of the sample increases. For this test case, a large number of points was required to

obtain a sufficient description of C_d and C_m . Therefore, these strategies were not appropriate. Instead, an alternative sequential strategy was adopted to ensure that training points were selected in regions that required a better description.

In the proposed sequential strategy for fitting the Kriging model, an LP-Tau (Statnikov and Matusov, 1995) DOE of 4096 design points is first generated. The resulting design vectors are supplied to VGK to obtain the corresponding estimates for C_d and C_m . LP-Tau sequences are deterministic space-filling DOE that ensure uniform distribution of design points over the design space. As mentioned above, an important feature of LP-Tau DOE is that larger LP-Tau sequences re-use points from smaller ones. From the total sample (4096 LP-Tau sequence), an initial training sample (256 LP-Tau sequence) is used to train the first Kriging model. The remaining points compose the test sample. Since the Kriging models are to be used to predict outputs for the optimisation, the Max error (6.4) over the test sample is preferred to the Mean Square Error (MSE) (6.5) as a measure of accuracy. Indeed, considering only an average value of the error could lead to erroneous solution of the optimisation problem. In the case of a Kriging model, the max error on the total sample and on the test sample are the same as the model predictions for the points used for training are exact. Next, a set of new points with large max error is taken from the test sample and is added to the training sample. A minimum Euclidian distance is defined to ensure that the selected points are not clustered in the same region.

$$\text{Max error} = \max_{\text{test sample}} \left(\frac{|\hat{y}_i - y_i|}{|y_i|} \right) \quad (6.4)$$

$$\text{MSE} = \frac{1}{n_{\text{test}}} \sum_{\text{test sample}} (\hat{y}_i - y_i)^2 \quad (6.5)$$

Then, the new Kriging is fitted and its max error is compared with the previous one. If the max error has increased significantly, the new points are rejected and another set of points with large max error is taken from the test sample. The aim is to reduce the max error by adding selected points. Typically, we want:

$$\text{new max error} \leq \gamma \times \text{max error} , \quad (6.6)$$

with $\gamma < 1$. We observe occasionally that adding such points might instead slightly increase the max error. Also, in this case, the training strategy experiences difficulties and takes longer to reach lower max error. Our experience shows that better results are achieved if γ is relaxed. For this problem, we used $\gamma = 1.1$. The sequential strategy adopted for fitting the Kriging model is described in Figure 6.7.

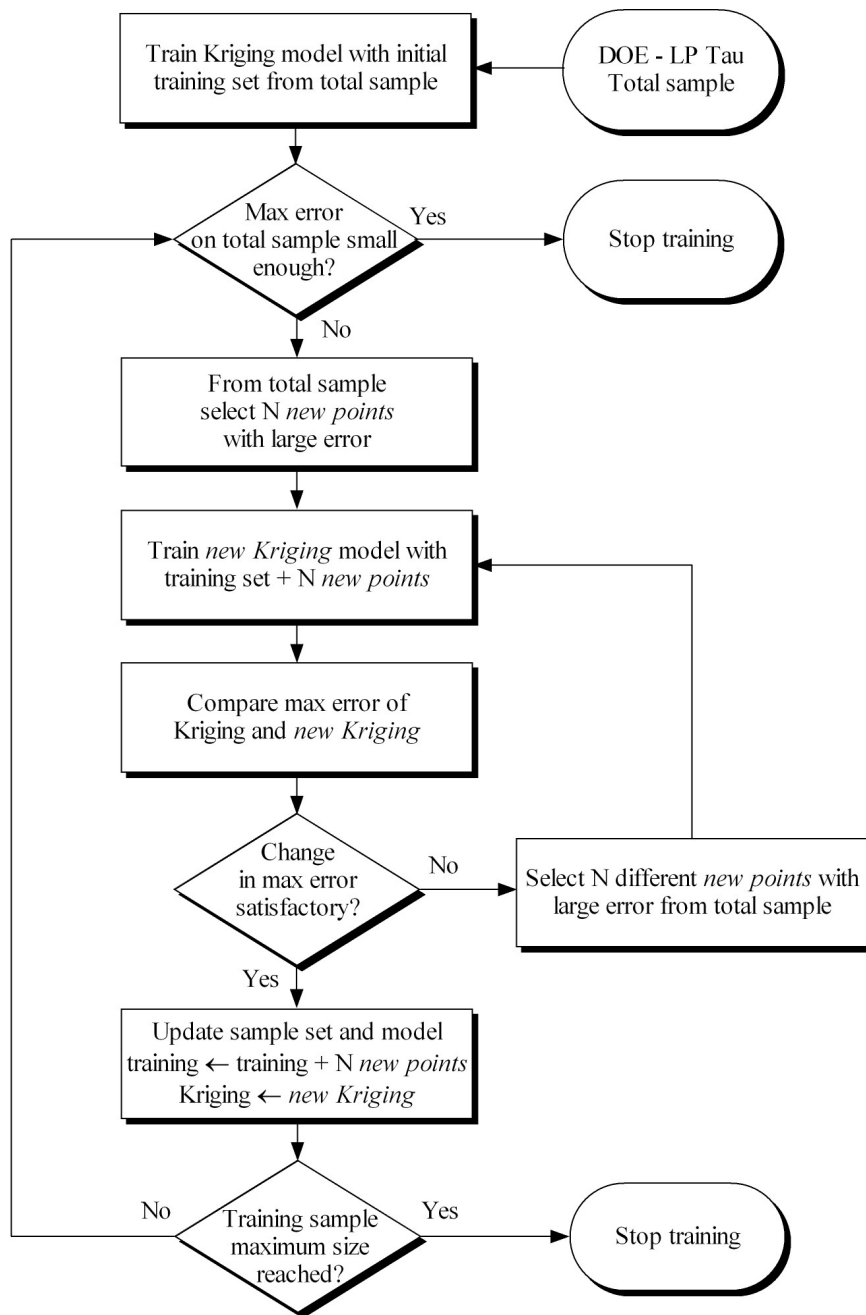


Figure 6.7: Adaptive sequential strategy for Kriging model training

In this test case, DACE toolbox (Lophaven *et al.*, 2002a, 2002b) for Matlab is used to fit the Kriging model. The Gaussian correlation model is chosen as it is the most popular. However, the choice of the regression model was not clear. Therefore, the adaptive sequential Kriging fitting described above was carried out for three regression models available in DACE toolbox, i.e constant, linear and quadratic. The history plot of max error and MSE during the sequential fitting procedure for C_d and C_m are given in Figure 6.8 and Figure 6.9, respectively. For this test case, the maximum training sample size was fixed to 1024 and the stopping criterion for max error was set to 0.02.

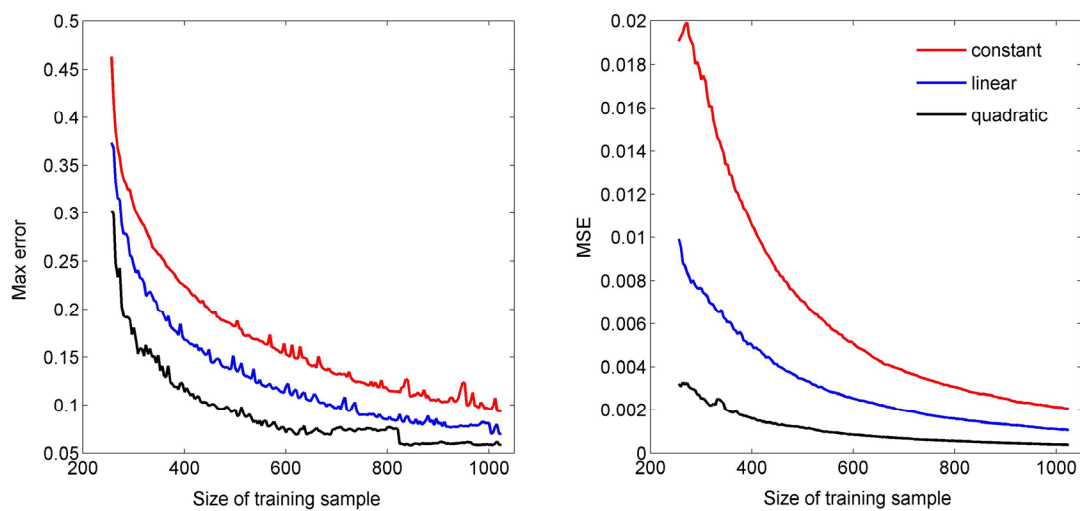


Figure 6.8. Error history of Kriging model for C_d during adaptive training for different regression models

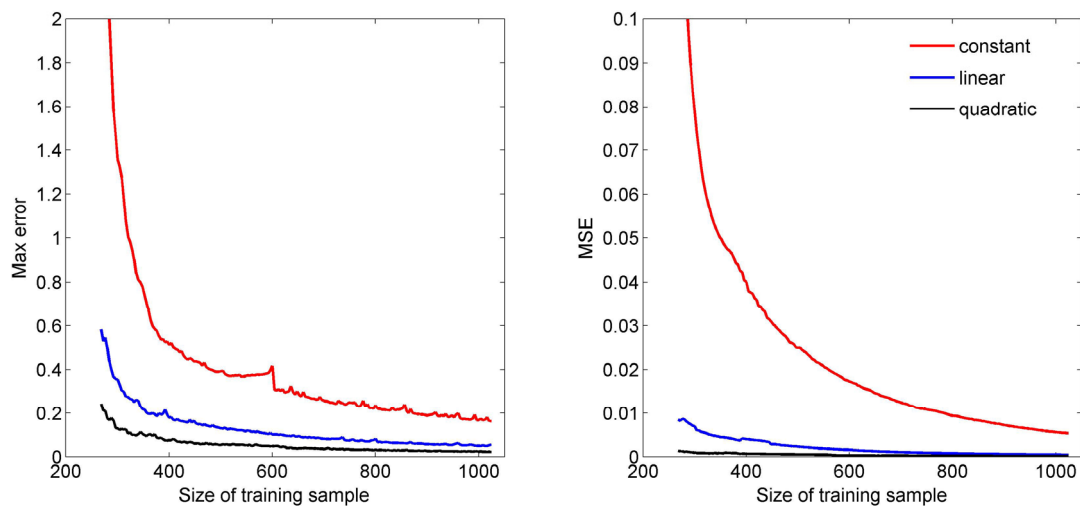


Figure 6.9. Error history of Kriging model for C_m during adaptive training for different regression models

The general tendency is that the max error and MSE decrease as the number of training points increases. Note that the curves for the MSE appear much smoother than the ones corresponding to the max error. This is a clear effect of the averaging of the error. For this test case, the quadratic regression model clearly outperforms both constant and linear regression models. Therefore, the resulting Kriging models with quadratic regression and Gaussian correlation model are used to predict the output values for C_d and C_m during the multi-objective optimisation procedure. The max error and MSE for the final Kriging models when adaptive training reached the maximum training sample size of 1024 points are provided in Table 6.7.

Regression model	Kriging - C_d		Kriging - C_m	
	Max error	MSE	Max error	MSE
Constant	9.38×10^{-2}	2.05×10^{-3}	1.63×10^{-1}	5.37×10^{-3}
Linear	6.98×10^{-2}	1.06×10^{-3}	5.46×10^{-2}	4.36×10^{-4}
Quadratic	5.82×10^{-2}	3.73×10^{-4}	2.24×10^{-2}	1.11×10^{-4}

Table 6.7: Max error and MSE of Kriging models with different regression functions after adaptive training

6.3 Formulation of optimisation problem

The design problem is to optimise the airfoil shape for 11 PARSEC input variables, namely to minimise the total drag and maximise the wing box volume. In the case of a 2-d airfoil, the area of the trapezoid defined by the position of the front and rear spars as shown in Figure 6.10 is to be maximised.

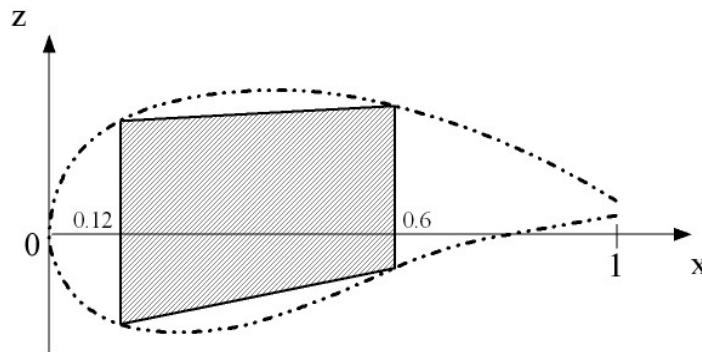


Figure 6.10. Area of inner trapezoid box defined by front and rear spar

The optimisation problem can be formulated as follows:

$$\min_{\mathbf{x}} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} C_d \\ -\text{area} = -0.24 \times (t_{12\%} + t_{60\%}) \end{pmatrix} \quad (6.7)$$

Subject to

$$\begin{pmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{pmatrix} = \begin{pmatrix} -C_m - 0.07 \\ C_m + 0.04 \\ -t_{12\%} + 0.076 \\ -t_{60\%} + 0.072 \end{pmatrix} \leq \mathbf{0} \quad (6.8)$$

and the input range defined in Table 6.6.

A description of the outputs considered in the optimisation problem is given in Table 6.8. Here g_1 and g_2 are constraints on the achieved pitching moment while g_3 and g_4 are constraints on the thickness of the airfoil where the front and rear spar are positioned, respectively.

Outputs	Description
C_d	Total drag
C_m	Achieved pitching moment
area	Area of trapezoid defined by front and rear spar
$t_{12\%}$	Airfoil thickness at front spar position: 12% chord
$t_{60\%}$	Airfoil thickness at rear spar position: 60% chord

Table 6.8: Description of model outputs

During the optimisation procedure, the drag and pitching moment coefficient are predicted with the Kriging model and the geometric outputs (area, $t_{12\%}$, and $t_{60\%}$) are evaluated with the geometry generator module which does not require running VGK. For practical reasons, the input variables are scaled between -1 and 1. If not specified otherwise, all design input values given in the following sections are scaled.

6.4 Reducing the dimensionality of the problem

6.4.1 Identifying non-significant inputs using VBM indices

For the aerodynamic outputs, C_d and C_m , the exact first order and total sensitivity analysis indices are analytically derived from the Kriging models (Chen,

2005; see also Appendix D). For the geometric outputs, i.e. the area, $t_{12\%}$ and $t_{60\%}$, the first order and total sensitivity analysis indices are obtained numerically using the Sobol method implemented in Simlab (Saltelli, 2003; Saltelli, 2004). It implies running extensively the PARSEC-11 geometry generator on a particular sample of points (49152). However, this only require a few minutes on a normal computer. The first order and total sensitivity indices for all outputs are given in Table 6.9 and **Table 6.10**, respectively.

First order indices		C_d	C_m	area	$t_{12\%}$	$t_{60\%}$
1	r_{le}	0.0912	0.0039	0.0144	0.0592	0.0015
2	x_{up}	0.0320	0.0332	0.0018	0.0154	0.0232
3	x_{lo}	0.0314	0.3737	0.0262	0.3360	0.4270
4	z_{up}	0.1927	0.0954	0.1661	0.0628	0.1042
5	z_{lo}	0.1240	0.1392	0.4117	0.1597	0.2542
6	z_{XXup}	0.1030	0.0182	0.0370	0.0220	0.0165
7	z_{XXlo}	0.0980	0.1824	0.3425	0.2806	0.1106
8	z_{te}	0.0400	0.0566	0.0002	0.0003	0.0001
9	Δz_{te}	0.0004	0.0016	0.0000	0.0001	0.0000
10	β_{te}	0.0025	0.0033	0.0000	0.0020	0.0012
11	α_{te}	0.0077	0.0461	0.0000	0.0000	0.0000

Table 6.9: First order sensitivity indices for C_d , C_m , area, $t_{12\%}$ and $t_{60\%}$

Total order indices		C_d	C_m	area	$t_{12\%}$	$t_{60\%}$
1	r_{le}	0.1896	0.0055	0.0162	0.0602	0.0025
2	x_{up}	0.1730	0.0365	0.0022	0.0169	0.0240
3	x_{lo}	0.0462	0.4160	0.0279	0.3985	0.4878
4	z_{up}	0.3740	0.0959	0.1660	0.0634	0.1036
5	z_{lo}	0.1524	0.1400	0.4098	0.1574	0.2550
6	z_{XXup}	0.1662	0.0208	0.0371	0.0227	0.0176
7	z_{XXlo}	0.1205	0.2224	0.3424	0.3415	0.1694
8	z_{te}	0.0655	0.0576	0.0011	0.0017	0.0002
9	Δz_{te}	0.0010	0.0017	0.0001	0.0001	0.0001
10	β_{te}	0.0045	0.0038	0.0009	0.0033	0.0013
11	α_{te}	0.0103	0.0464	0.0003	0.0004	0.0000

Table 6.10: Total order sensitivity indices for C_d , C_m , area, $t_{12\%}$ and $t_{60\%}$

A more intuitive visualization of the results is given in Figure 6.11 and Figure 6.12 where each bar represents the total contribution of each design variable to the output variance and is broken down into main effect and interaction effect. This bar chart representation gives a clear idea of the nature of the contributions and interactions between inputs.

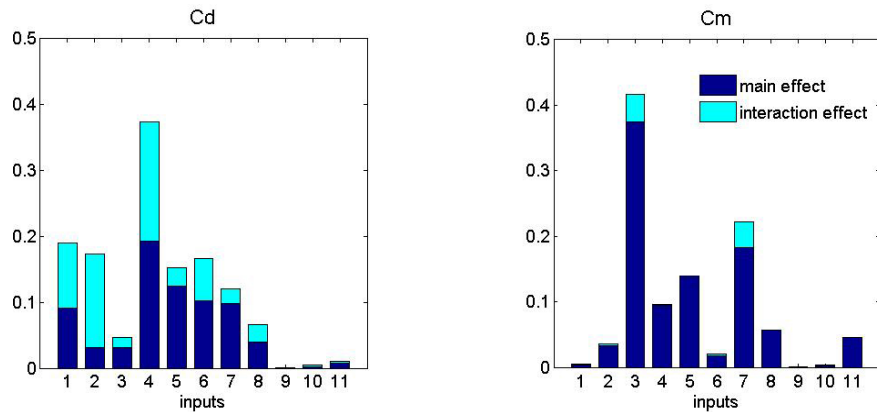


Figure 6.11: Variance-based sensitivity analysis – Main and interaction effect for C_d and C_m

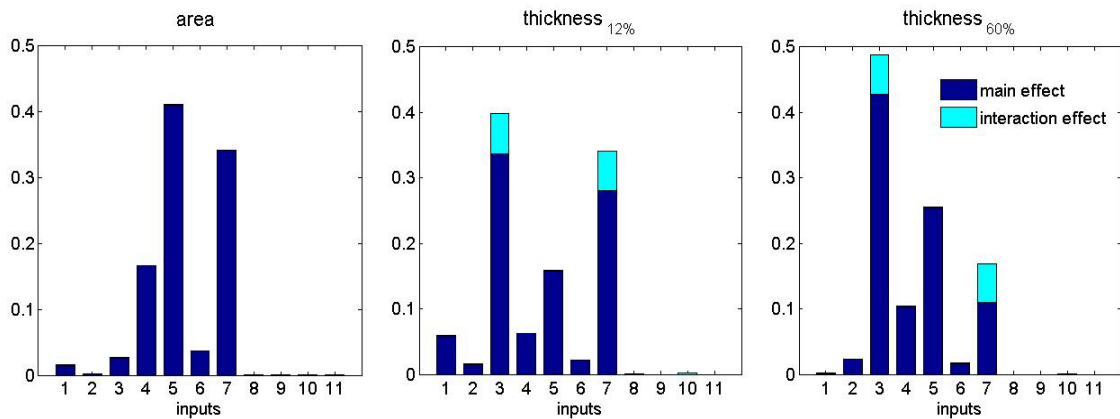


Figure 6.12: Variance-based sensitivity analysis – Main and interaction effect for area, $t_{12\%}$ and $t_{60\%}$

Here we consider as non-significant, inputs with a total order index smaller than 0.02. It is clear that variables 9 and 10, i.e. Δz_{te} and β_{te} are non-significant and according to the methodology described in section 2 can be discarded from the optimisation problem by fixing them to a specific value.

6.4.2 Formulation of the reduced optimisation problem

The reduced optimisation problem is formulated as follows:

$$\min_{\mathbf{x}_{\text{red}}} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} C_d \\ -\text{area} = -0.24 \times (t_{12\%} + t_{60\%}) \end{pmatrix} \quad (1.9)$$

Subject to

$$\begin{pmatrix} g_1(\mathbf{x}_{\text{red}}) \\ g_2(\mathbf{x}_{\text{red}}) \\ g_3(\mathbf{x}_{\text{red}}) \\ g_4(\mathbf{x}_{\text{red}}) \end{pmatrix} = \begin{pmatrix} -C_m - 0.07 \\ C_m + 0.04 \\ -t_{12\%} + 0.076 \\ -t_{60\%} + 0.072 \end{pmatrix} \leq \mathbf{0} \quad (1.10)$$

where

$$\mathbf{x}_{\text{red}} = \left(I_{\text{te}} \ x_{\text{up}} \ x_{\text{lo}} \ z_{\text{up}} \ z_{\text{lo}} \ z_{\text{XXup}} \ z_{\text{XXlo}} \ z_{\text{te}} \ \Delta z_{\text{te}}^* \ \beta_{\text{te}}^* \ \alpha_{\text{te}} \right)^T \quad (1.11)$$

with Δz_{te}^* and β_{te}^* constant.

6.4.3 Choosing values for non-significant variable using SOM

As described in section 3, a SOM is used to investigate the most appropriate value to freeze Δz_{te} and β_{te} . All inputs and outputs are taken into account to build the SOM on the same data set used for training the Kriging model (LP-Tau 4096 set). The same set is re-used to try to obtain more information about the design problem at no extra computational cost. Figure 6.13 shows only the component maps for Δz_{te} , β_{te} and all outputs. The map units corresponding to feasible design are marked with white dots while non-feasible units are displayed with a black dot. It can be noted that in the representation given by the SOM, the region of feasible design represents a small portion of the entire design space. This region clearly corresponds to large value of area and relatively small value of drag. A look at the values taken by Δz_{te} and β_{te} for these map units shows that they are centred on their mean value. Therefore, there is no preferred value for Δz_{te} and β_{te} . For that reason, it is decided to fix both values for Δz_{te} and β_{te} to zero.

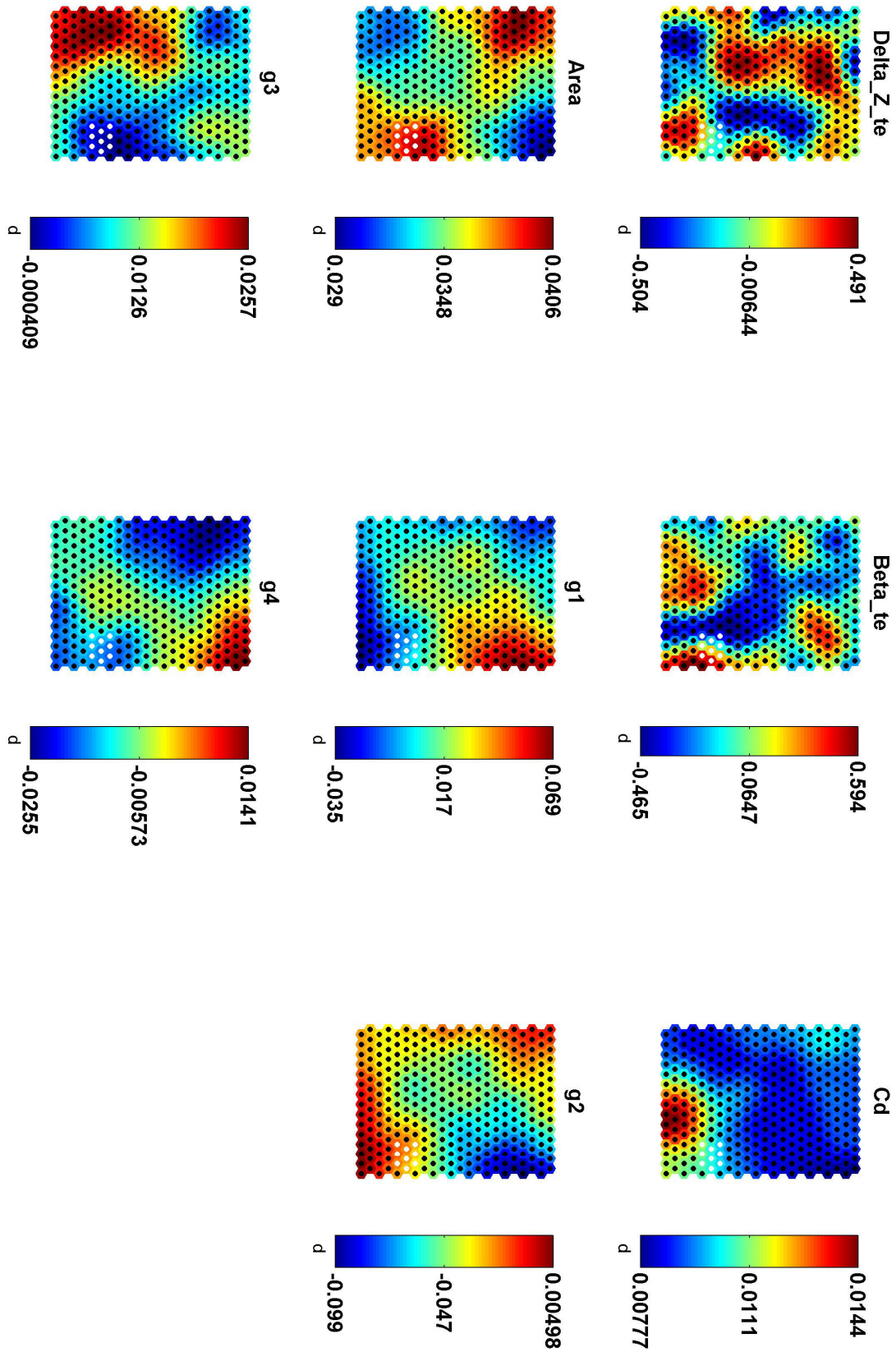


Figure 6.13: SOM with feasible regions for Δz_{te} , β_{te} and all outputs

6.4.4 Obtaining the different Pareto fronts

The set of Pareto solutions of the optimisation problem is obtained using an in-house developed gradient-based optimisation tool (Fantini, 2007; Fantini *et al.*, 2007; Guenov *et al.*, 2005; Utyuzhnikov *et al.*, 2005). Three runs are performed. In the first run, all inputs are considered. In the two other runs, the optimisation is performed on the reduced problem. The non-significant variables Δz_{te} and β_{te} are fixed to zero in the second run. In the last one, Δz_{te} and β_{te} are fixed to their lower bound value as a significant number of Pareto solutions of the full optimisation problem correspond to these values. Table 6.11 provides information on the optimisation procedure for the different runs.

	All inputs	$\Delta z_{te} = 0 / \beta_{te} = 0$	$\Delta z_{te} = -1 / \beta_{te} = -1$
Time	3h53min	2h11min	2h36min
Function evaluations	314266	181997	219989
Pareto points obtained	201	167	221

Table 6.11: Performance comparison between different three optimisation runs

It is clear that removing two inputs from the optimisation procedure by fixing them to a specific value considerably reduces the number of function evaluation and therefore the total computational time. It is a direct consequence of the reduction of the problem dimensionality. In the case of gradient optimisation, fewer input variables implies that less effort is spent to obtain numerical estimates of gradient using finite differences. The Pareto fronts corresponding to these three optimisation runs are plotted in Figure 6.14. In this figure, the extreme values for C_d and area obtained on the 4096 LP-Tau DOE are taken as the bounds of the x and y axes. Figure 6.14 clearly shows that all Pareto fronts are located in the same region. It is important to notice that the extent of the range of variation for C_d and for the area remains the same. The Pareto front, where Δz_{te} and β_{te} are fixed to zero, shows good agreement with the solutions of the full problem for values of C_d up to 0.0098, but fails to do so for higher values. It can be explained by the fact that for solutions of the full problem that correspond to values of C_d higher than 0.0098, the constraints on the minimum values for Δz_{te} and β_{te} are active. As it could be

expected, the solutions obtained, when Δz_{te} and β_{te} are fixed to -1, show a better match with this part of the Pareto front obtained for the full problem.

Reducing the dimensionality of the optimisation problem by freezing non-significant variables to a specific value might come at the expense of the quality of the Pareto front. Indeed, some parts of the Pareto surface are not captured accurately by the reduced problem. Therefore, when the designer decides to reduce the number of variables using this approach he/she must be aware that the quality of the resulting Pareto surface might be compromised.

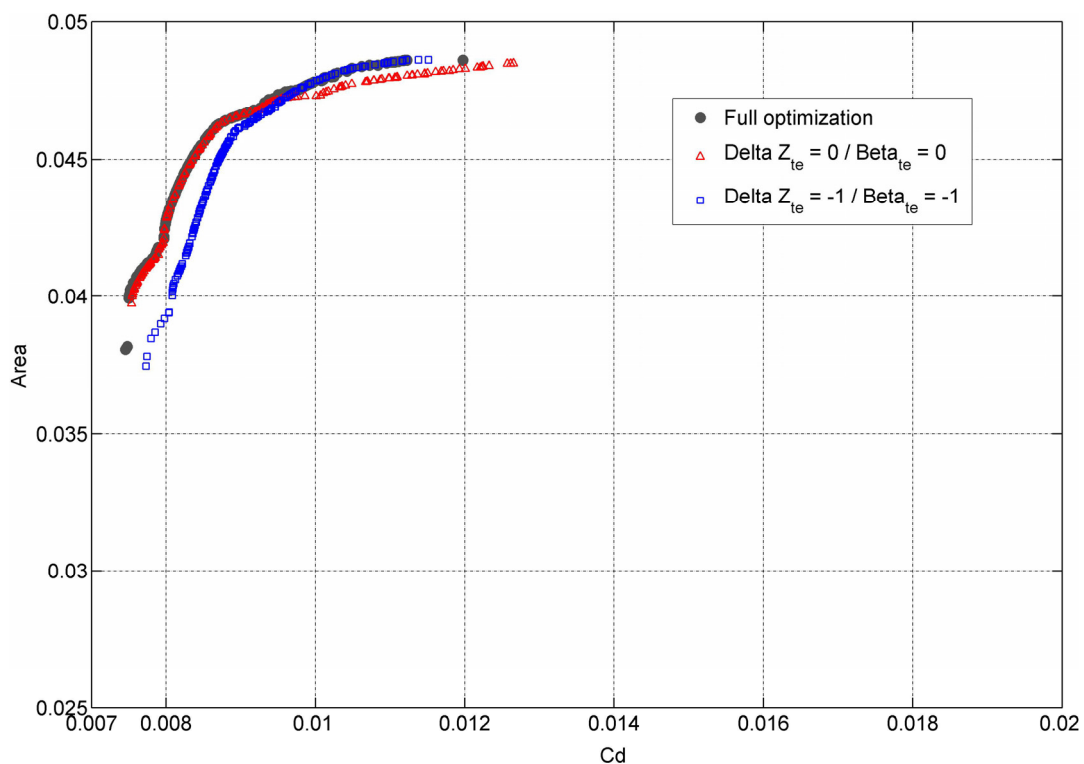


Figure 6.14: Pareto surface obtained for the two-objective airfoil optimisation problem

6.5 Local Pareto sensitivity analysis

In this section, we focus on the use of local approximations of the Pareto surface to understand the trade-offs between objectives and the adaptability of a Pareto solution. Figure 6.14, shows that all Pareto solutions are located in a specific region of the criterion space but the discontinuities in the Pareto surface seems to indicate that they correspond to different type of airfoils. Figure 6.15 shows the location in the criterion space of a few representative Pareto solutions. The corresponding airfoils are plotted in

Figure 6.16. It gives an idea of the diversity of the Pareto solutions lying on the Pareto surface.

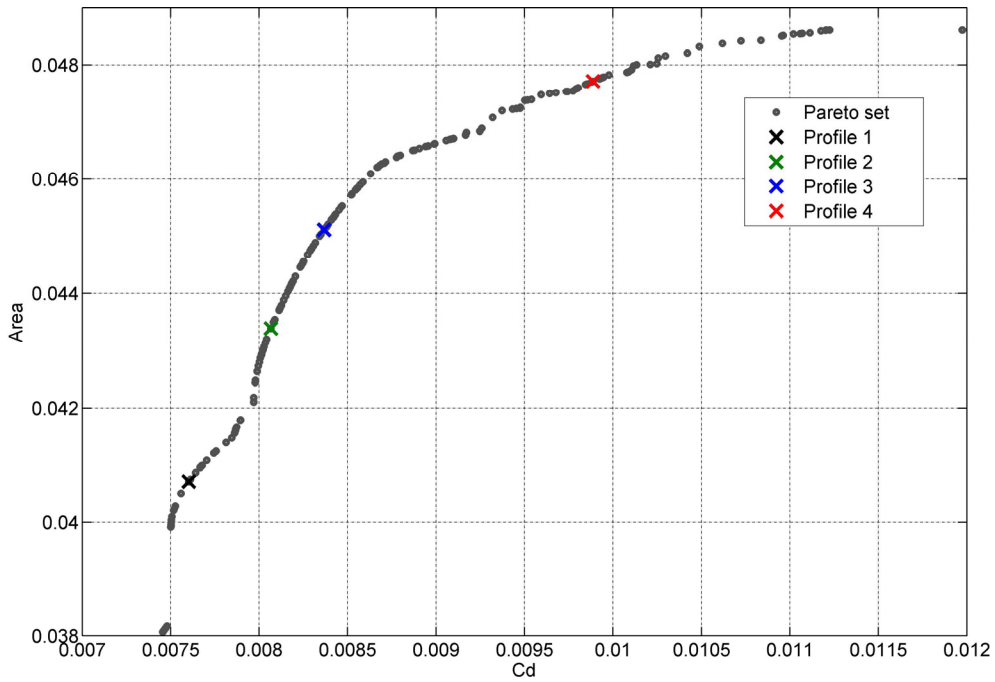


Figure 6.15: A few characteristic solutions in the criterion space

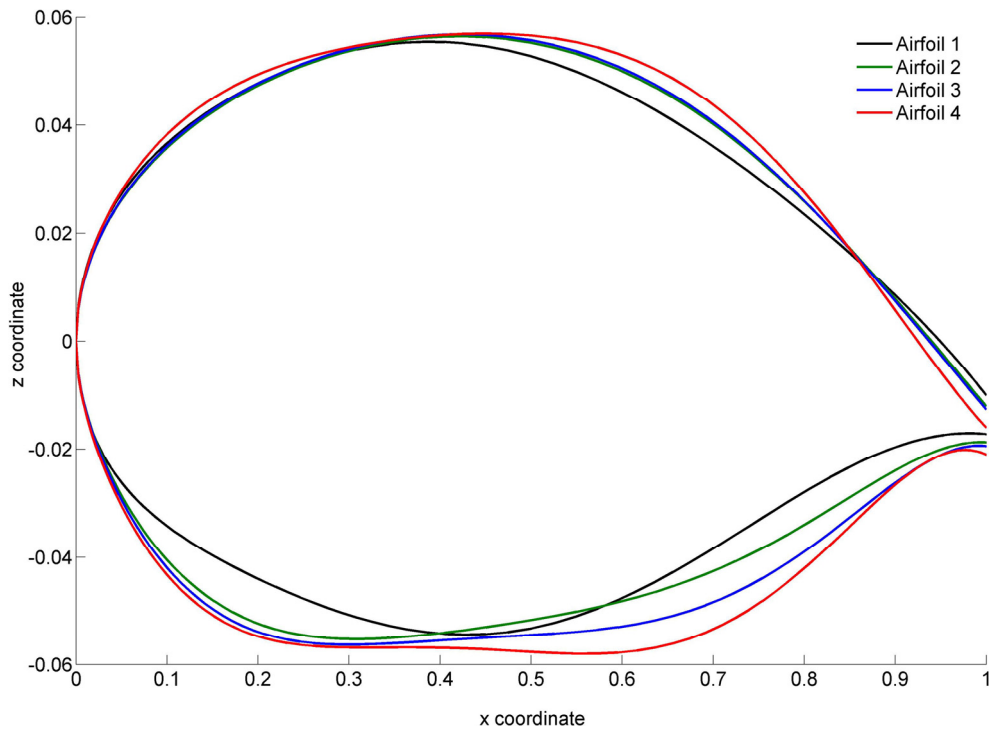


Figure 6.16: A few characteristic Pareto solution airfoils

In this test case, a good description of the Pareto surface is given as a significant number of Pareto points can be obtained at an affordable computational cost. However, when high fidelity codes are used in place of surrogate models, obtaining a single Pareto solution could be very demanding and the designer will benefit from obtaining information about the local sensitivity analysis of the candidate solution.

When a candidate Pareto solution is obtained, the linear and quadratic approximations developed in chapter 4 can be derived. For a small change in one objective, the local trade-off in the other objective and the corresponding change in the design space can be computed. In this context, the range of validity of the local approximation is not clear. It is proposed to use the difference between quadratic and linear prediction to measure the extent of the validity of the approximation. When the contribution of the quadratic term becomes too important with respect to the linear term, the linear approximation is not considered reliable. Meanwhile, the designer must ensure that all constraints from the optimisation problem are still satisfied. In other words, in the case of a two objective optimisation problem, to assess the extent of the validity of the linear approximation, the designer must solve a double linear optimisation problem:

Minimise and maximise: δf_1

Subject to:

$$-1 \leq \mathbf{x} + \delta \mathbf{x} \leq 1, \quad (6.12)$$

$$\mathbf{g} + \frac{d\mathbf{g}}{df_1} \delta f_1 + \frac{1}{2} \frac{d^2\mathbf{g}}{df_1^2} (\delta f_1)^2 \leq \mathbf{0}, \quad (6.13)$$

$$\left| \frac{1}{2} \frac{d^2 f_2}{df_1^2} \delta f_1 \right| / \left| \frac{df_2}{df_1} \right| \leq \alpha. \quad (6.14)$$

The first two equations impose that constraints on the design variables range and that the other constraints (6.13) are still satisfied. Equation (6.14) requires that the contribution of the quadratic term remains smaller than a fraction of the linear term. It allows to detect when the curvature becomes important and where the Pareto surface

departs from the linear approximation. The scalar α is defined by the designer and represents the relative contribution of the quadratic term with respect to the linear one.

Let us consider the candidate solution ($C_d = 0.00807$, area = 0.04339) and study its local adaptability. The first and second order derivatives are given in Table 5.2 and the linear optimisation problem is solved for $\alpha = 0.1$ to derive the extent of the validity of the approximation. Note that f_2 is defined as the negative value of the area, see (6.7). The limits of the range of validity of the linear approximation are plotted in Figure 6.17 and the associated airfoils are shown in Figure 6.18.

Derivatives	
df_2/df_1	-7.51
d^2f_2/df_1^2	3.28×10^4

Table 6.12: First and second order derivative on the Pareto surface at candidate Pareto solution ($C_d = 0.00807$, area = 0.04339)

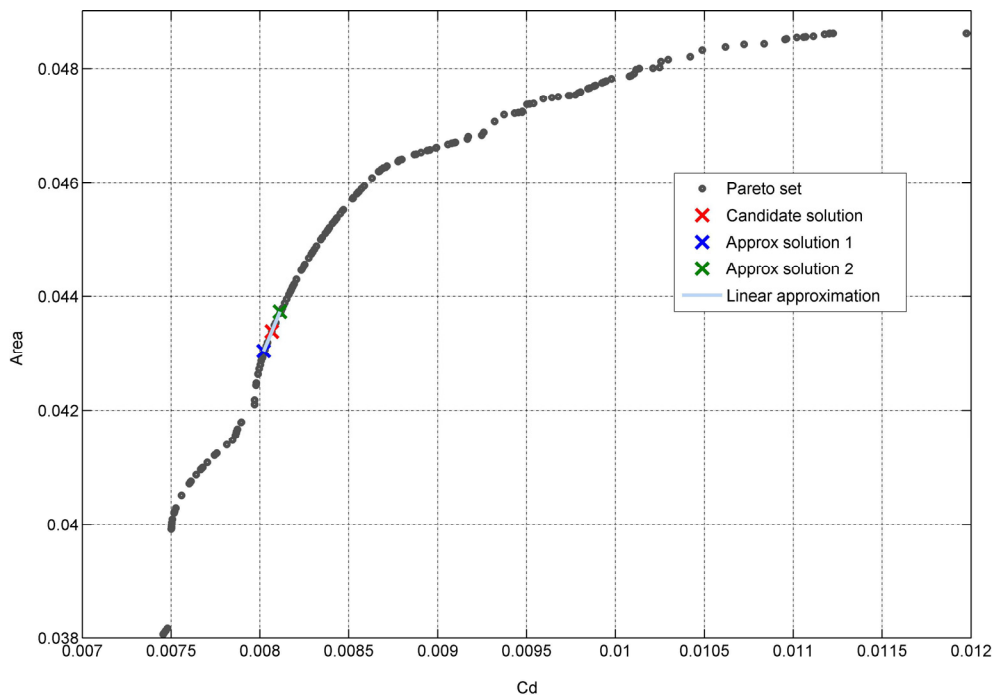


Figure 6.17: Linear approximation at candidate Pareto solution ($C_d = 0.00807$, area = 0.04339)

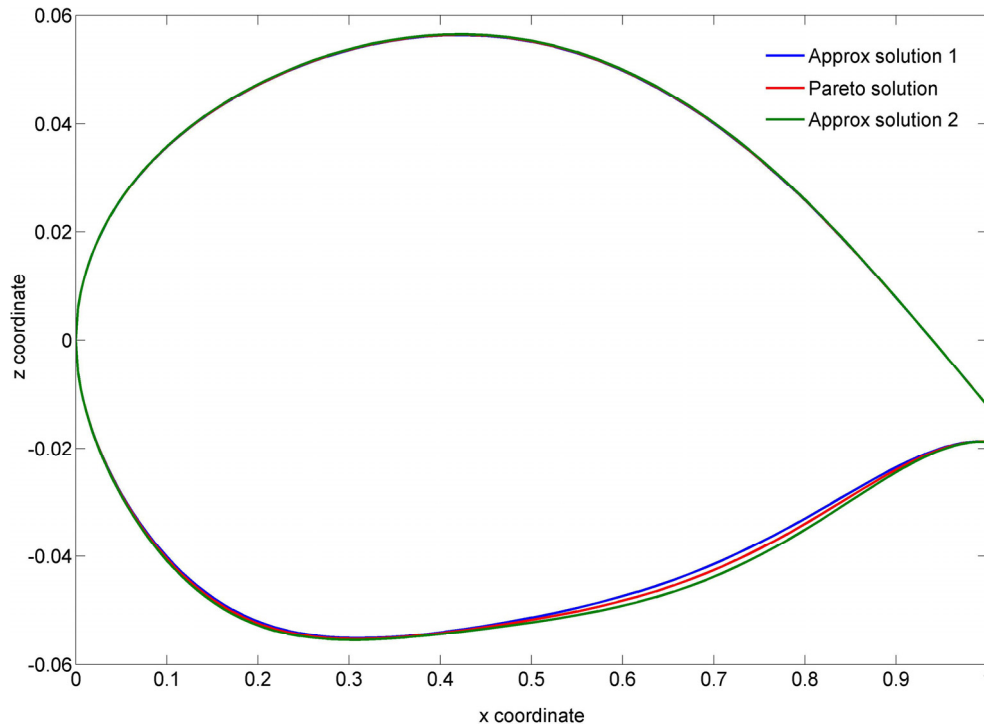


Figure 6.18: Airfoils associated with the limits of validity of the approximation at candidate Pareto solution ($C_d = 0.00807$, area = 0.04339)

It is clear that along the direction given by the linear approximation, approximated Pareto airfoils differ mainly in their lower surface. The corresponding values for their inputs, objectives and constraints are given in Table 6.13. The output values predicted by the approximation show good agreement with those obtained with the model. The linear approximation provides the designer with an insight in the Pareto solution that can be found in this region of the Pareto surface.

For candidate Pareto solution ($C_d = 0.00807$, area = 0.04339), the range of validity of the linear approximation is limited by constraint (6.14) related to the departure of the Pareto surface from its local tangent. Table 6.13 shows that no other constraint is activated. It follows that the range of validity of the linear approximation is centred on the candidate Pareto solution.

	Approx solution 1		Pareto solution 1	Approx solution 2	
r_{ie}	0.06400		0.06521	0.06641	
x_{up}	0.37032		0.37263	0.37493	
x_{lo}	-0.91566		-0.94435	-0.97303	
z_{up}	0.90145		0.91516	0.92887	
z_{lo}	-0.67471		-0.69336	-0.71202	
zxx_{up}	0.16030		0.16645	0.17260	
zxx_{lo}	-0.81905		-0.83604	-0.85303	
z_{te}	-0.16488		-0.16715	-0.16942	
Δz_{te}	0.04200		0.04235	0.04270	
β_{te}	-0.09983		-0.09997	-0.10011	
α_{te}	0.01455		0.01341	0.01227	
	Approx	Model	Model	Approx	Model
C_d	0.00802	0.00802	0.00807	0.00811	0.00812
area	0.04305	0.04306	0.04339	0.04374	0.04375
g_1	0.00000	0.00032	0.00000	0.00000	0.00029
g_2	-0.03000	-0.03032	-0.03000	-0.03000	-0.03029
g_3	-0.00617	-0.00617	-0.00663	-0.00708	-0.00708
g_4	-0.02520	-0.02524	-0.02617	-0.02715	-0.02719

Table 6.13: Inputs, objectives and constraints values of candidate Pareto solution ($C_d = 0.00807$, area = 0.04339) and approximate Pareto solutions associated with the limits of validity of the approximation

Let us consider a different candidate solution ($C_d = 0.00837$, area = 0.04512) and carry out a similar analysis. First, the first and second order derivatives are obtained and given in Table 6.14. For this candidate solution, the trade-off between objectives is clearly less important than for the first candidate solution.

Derivatives	
df_2/df_1	-4.35
d^2f_2/df_1^2	9.01×10^3

Table 6.14: First and second order derivative on the Pareto surface at candidate Pareto solution ($C_d = 0.00837$, area = 0.04512)

As for the first Pareto candidate, the linear optimisation problem is solved for $\alpha = 0.1$ to derive the extent of the validity of the approximation. For this candidate Pareto solution,

the minimum value for df_1 is limited by the constraint on the quadratic term (6.14) while the maximum value for df_1 is limited by a constraint on the range of variation of \mathbf{x} (see equation (6.12)), which becomes active first. It results that the range of the validity of the linear approximation is not centred on the candidate Pareto solution as shown in Figure 6.19. It illustrates that all constraints must be taken into account when estimating the extent of validity of the local approximation since for a candidate Pareto solution, inactive constraints might be close to become active. The associated airfoils are given in Figure 6.20 and show that the changes in the airfoil geometry for the approximated Pareto solutions are very small.

The corresponding values for their inputs, objectives and constraints are given in Table 6.15. Similar to the first candidate Pareto solution, the values of the outputs obtained with the linear approximation for the new approximated Pareto solution match the values obtained with the model. As mentioned before, for the second approximated solution, variable z_{up} reaches its maximum value ($z_{up} = 1$).

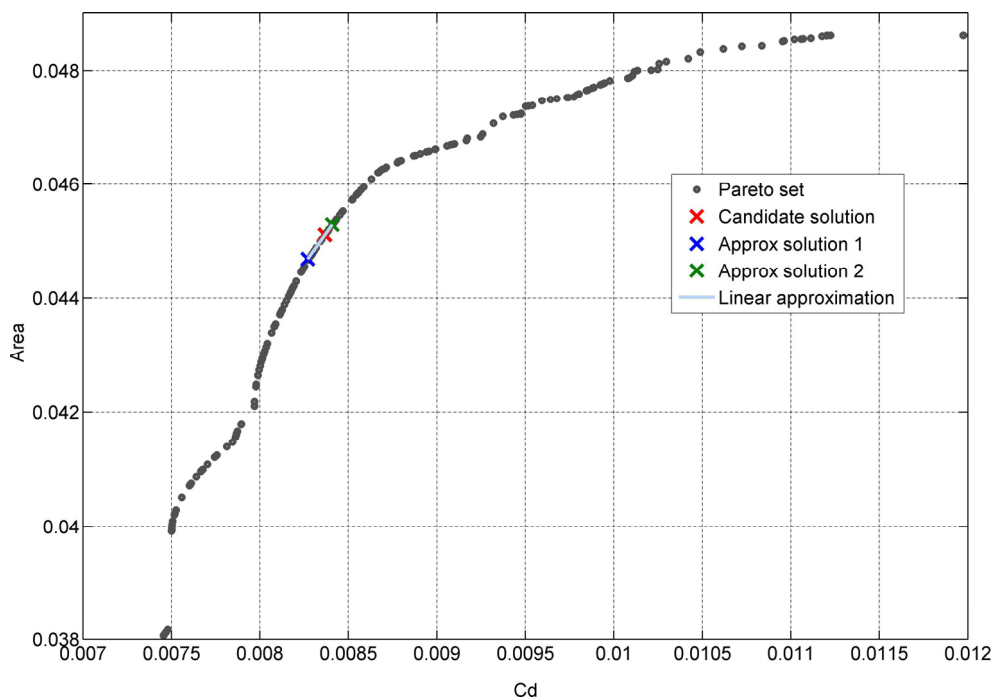


Figure 6.19: Linear approximation at candidate Pareto solution ($C_d = 0.00837$, area = 0.04512)

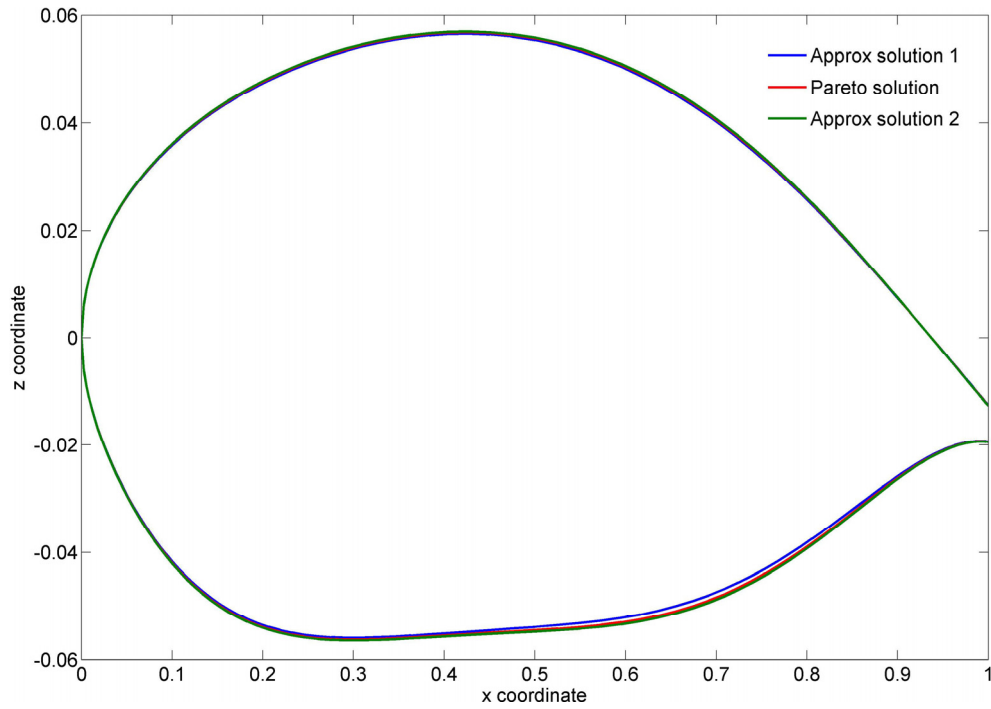


Figure 6.20: Airfoils associated with the limits of validity of the approximation at candidate Pareto solution ($C_d = 0.00837$, area = 0.04512)

	Approx solution 1		Pareto solution 2	Approx solution 2	
r_{te}	0.07227		0.07590	0.07744	
x_{up}	0.39724		0.41098	0.41678	
x_{lo}	-1.00000		-1.00000	-1.00000	
z_{up}	0.92966		0.97911	1.00000	
z_{lo}	-0.77073		-0.81022	-0.82691	
zxx_{up}	0.19328		0.21416	0.22298	
zxx_{lo}	-0.95874		-0.98073	-0.99002	
z_{te}	-0.26966		-0.28366	-0.28958	
Δz_{te}	0.10616		0.10855	0.10955	
β_{te}	-0.21414		-0.21912	-0.22122	
α_{te}	-0.10551		-0.11622	-0.12075	
	Approx	Model	Model	Approx	Model
C_d	0.00827	0.00828	0.00837	0.00841	0.00841
area	0.04469	0.04469	0.04512	0.04529	0.04529
g_1	0.00000	0.00005	0.00000	0.00000	0.00001
g_2	-0.03000	-0.03005	-0.03000	-0.03000	-0.03001
g_3	-0.00803	-0.00803	-0.00856	-0.00878	-0.00879
g_4	-0.03020	-0.03020	-0.03142	-0.03194	-0.03194

Table 6.15: Inputs, objectives and constraints values of candidate Pareto solution ($C_d = 0.00837$, area = 0.04512) and approximate Pareto solutions associated with the limits of validity of the approximation

6.6 Conclusions

In this chapter, the optimisation of an airfoil shape to minimise drag and maximise the inner trapezoid area is carried out.

The popular PARSEC-11 parameterisation used in this test case seems very attractive because it makes use of a reduce number of intuitive geometric parameters to control the airfoil shape. However, in some cases, this parameterisation can produce erroneous profiles causing the VGK code to crash. An approach using SOM was proposed to tackle this problem. The use of SOM enabled to identify regions of the design space where PARSEC-11 failed to generate meaningful profiles. These regions could be removed from the design space, to allow the optimisation to concentrate on well parameterised regions only. The use of a Kriging model allowed to eliminate the inherent noise problem in estimates for C_d and C_m obtained with VGK. Moreover, it saved time for the estimation of VBM indices for C_d and C_m and accelerated the optimisation procedure. An adaptive procedure was used to train the Kriging model as some regions of the design space required a more detailed description.

The identification of non-significant variables (Δz_{te} and β_{te}) enabled to reduce the problem from 11 to 9 variables. A SOM was built to investigate the most appropriate values to fix the non-significant variables. It showed that for feasible regions, Δz_{te} and β_{te} were taking values centred on their mean, making it a preferred choice. The Pareto set obtained with the reduced problem indicated that a reduction of the dimensionality might come at the expense of the quality of the Pareto solutions. Indeed, it showed a good match with the Pareto front obtained with the full optimisation problem for small values of the drag coefficient. However, a small discrepancy occurred for higher values because this part of the Pareto front corresponds to solutions where constraints on the minimum bound for Δz_{te} and β_{te} are active.

The use of local approximations derived in section 4 enabled to investigate the type of airfoils that corresponded to the part of the Pareto front where the candidate solutions were located. It indicated what parameters could be varied to remain on the Pareto set, and to what extent. It also provided essential information on the local trade-offs of the candidate Pareto point and gave a better insight into the solution set at a reduced cost.

Chapter 7

Summary and conclusions

This thesis focuses on the optimisation aspect of the MDO process. A novel strategy for sensitivity analysis (SA) is proposed to improve the process efficiency and effectiveness. It includes two components. The first one is to provide the designer with a methodology for identifying non-significant variables that can be removed from the optimisation formulation. The second component is linked to the decision-making process in the case of a multi-objective optimisation where it is essential for the designer to obtain an insight in the trade-offs between objectives. The main elements of novelty of this thesis include:

- A new strategy combining global sensitivity analysis and self-organising maps to reduce the dimensionality of the optimisation problem.
- A new method for studying the sensitivity of Pareto solutions in which:
 - New general equations for obtaining local linear and quadratic approximations of the Pareto surface are derived.
 - A method for using the local approximations, including limits of validity and back-mapping is proposed.

- A new general sufficient condition for identifying non-differentiable Pareto points is derived.

7.1 Reducing the dimensionality of the design problem

The aim of the first component of the proposed approach is to reduce the dimensionality of the problem prior to the optimisation to reduce the computational effort. Sensitivity analysis is used to understand the relationship between variables and outputs of the models involved in the design procedure. A review of available sensitivity analysis methods shows that Variance-based methods (VBM) are the most appealing candidate. They present many desirable properties to develop a general SA approach in the context of MDO:

- VBM are model independent, no assumption on the model additivity or linearity is required.
- VBM can capture contributions to the output, which are due to interaction effects between input variables.
- VBM can examine the effect of each input over the whole range of variation.
- VBM can treat groups of factors. This property allows to perform studies by grouping factors by discipline or component.

In this thesis, a new approach for reducing the dimensionality of the optimisation problem is proposed. Global SA is used to understand and quantify over the entire design space the effect of input variables on the various outputs. It provides a means to identify which inputs are the most important for the design. In this approach, it is recommended to retain only significant input variables to reduce the computational effort and make the optimisation procedure more efficient. Typically, one must fix the non-significant variables to a specific value. In this research, it is shown that in some cases, fixing these to an arbitrary value might lead to a small degradation of the solutions of the optimisation problem. It is suggested to make use of the data generated for the SA to build a self-organising map (SOM) and determine whether there is a preferred value for the non-significant variables.

7.2 Pareto sensitivity analysis

The second component of this approach is a novel method for SA of the Pareto solutions and includes local approximation and differentiability study of the Pareto surface. At this stage of the design process, when Pareto solutions have been obtained, it is essential for the designer to understand the trade-offs between objectives in order to select the final solution that will suit his/her preferences best. The local sensitivity of the solutions of the multi-objective optimisation problem is obtained to understand the trade-offs between objectives.

7.2.1 Investigation into available methods

The author's investigation into available methods for SA of the Pareto front shows that most of these have been developed for specific and simple cases. Therefore, they present little interest in an MDO situation. However, a technique based on gradient projection was found to offer interesting features for Pareto approximation. It is thoroughly investigated in the thesis and its advantages and limitations are outlined. This technique relies on two assumptions:

- The objective and constraint functions are twice continuously differentiable
- The set of active constraints remains the same in the vicinity of the Pareto point under consideration

This method enables to obtain directional derivatives to derive a linear approximation of the Pareto surface. It also proposes to generate additional Pareto information in the neighbourhood of the solution under study to obtain a quadratic approximation.

7.2.2 Derivation of new local Pareto approximations

In this work, new general equations based on the gradient projection method under the same assumptions are derived for local linear and quadratic approximations. Results on several algebraic examples show that they give the exact analytical approximations. This new method for obtaining local approximations presents several advantages with respect to those found in literature:

- The new approximations are not limited by the number of design variables, constraints or objectives.
- Both exact linear and quadratic approximations of the Pareto surface are derived in the general case.
- No additional Pareto information is needed in the vicinity of the current Pareto solution and the quadratic approximation is obtained without running further optimisation.
- The computational cost for obtaining the linear and quadratic approximations can be clearly measured since it is directly linked to the effort required for evaluating the gradients and Hessians of the constraint and objective functions.
- A unique feature of these approximations is that local moves on the approximated Pareto surfaces are directly related to moves in the design space and is referred as back-mapping. The designer not only has the possibility to obtain new approximate solutions in the objective space, he/she can also understand what are the degrees of freedom on the Pareto surface and how variables must be varied with respect to each other.
- The linear and quadratic approximations of the non-active constraints are obtained and are used as part of the back-mapping to ensure that moves on the Pareto surface remains feasible.

It is also proposed in this thesis to use the discrepancy between the linear and quadratic approximation to assess the extent of validity of the moves on the Pareto surface.

7.2.3 Development of a new method for identifying non-differentiable Pareto points

Previous work found in the literature is used as a starting point to develop a more general method for detecting non-smooth area of the Pareto surface. In this thesis, a sufficient condition entirely based on consideration of gradient obtained in the design space is derived to identify non-differentiable Pareto points. It is also shown how it can be formulated into a linear optimisation problem to make it more convenient in the high dimensional case.

Non-differentiability detection and Pareto surface approximation are the two components of the Pareto SA approach. A detailed scheme describing how to implement it is also provided.

7.2.4 Limitations of the proposed approach

The Pareto approximations are only valid at differentiable regions of the efficient frontier. However, the *global Pareto front* might be composed of solutions from several *local Pareto fronts* and as a result might present non-smooth regions. In such situation, the new linear and quadratic equations can only approximate the *local Pareto front* and do not guarantee a good approximation of the *global Pareto front*. In the same way, the method for identifying non-differentiable Pareto points developed in this thesis can only detect non-differentiability due to changes in the set of active constraints and will fail to identify such situations. It is important to observe that currently no method to the best of the author's knowledge can address this type of non-differentiability.

7.3 Test results

The approach was conducted on two representative design examples: beam optimisation problem and airfoil design. In the beam problem, the approximation enabled to identify typical families of Pareto solutions on the Pareto surface. This gives a better insight in the solutions of the problem. In the airfoil design problem, the entire SA approach was tested. Two non-significant variables were identified which allowed to reduce the problem from eleven to nine design parameters. It resulted in a very significant reduction of the computational effort (up to 40%), which came only at a small degradation of Pareto front. The strategy developed for local sensitivity of the Pareto front proved to be very effective. Part of the Pareto set could be approximated locally with a high degree of accuracy, providing new approximate Pareto solutions at a reduced cost. This additional information could be very useful for the designer during the decision-making process as it could help articulate *a posteriori* preferences to select the final solution.

7.4 Further work

In this thesis, VBM and SOM are used together to identify non-significant variables and freeze them to a specific value. It is also shown that for some parts of the Pareto front, there might be a small discrepancy between the results obtained with full optimisation and the reduced one. It will be interesting to investigate this problem further.

A significant feature of the Pareto approximation approach is its ability to find approximate solutions on the efficient frontier. It is thought that it could contribute to improve methods striving to generate evenly distributed solutions on the Pareto front. If the distribution is not homogeneous, approximate points could be obtained to help fill in gaps.

A further direction for research would be to investigate further the non-differentiability problem due to the presence of several *local Pareto fronts*.

Appendix A

Theorem

If \mathbf{M} is a matrix of shape $m \times n$ and all column vectors are linearly independent then matrix $\mathbf{M}^T \mathbf{M}$ is invertible.

Proof

Let \mathbf{M} be a $m \times n$ matrix where all column vectors \mathbf{M}_j , $j = 1, \dots, n$ are linearly independent:

$$\mathbf{M} = (\mathbf{M}_1 \mid \dots \mid \mathbf{M}_n). \quad (\text{A.1})$$

Let us consider equation:

$$(\mathbf{M}^T \mathbf{M}) \mathbf{x} = \mathbf{0}. \quad (\text{A.2})$$

Assume that matrix $\mathbf{M}^T \mathbf{M}$ is not invertible. Therefore, there exists a solution \mathbf{x} such that $\mathbf{x} \neq \mathbf{0}$. Multiplying equation (A.2) by \mathbf{x}^T we obtain:

$$\mathbf{x}^T (\mathbf{M}^T \mathbf{M}) \mathbf{x} = 0 \quad (\text{A.3})$$

which can be rewritten as

$$(\mathbf{M}\mathbf{x})^T (\mathbf{M}\mathbf{x}) = 0. \quad (\text{A.4})$$

Therefore

$$\|\mathbf{M}\mathbf{x}\|_2 = 0 \quad (\text{A.5})$$

and

$$\mathbf{M}\mathbf{x} = \mathbf{0} \quad (\text{A.6})$$

which contradicts the fact that the columns of \mathbf{M} are linearly independent. Therefore, $\mathbf{M}^T \mathbf{M}$ is invertible.

Appendix B

As given in Fletcher (1989)

Farkas' lemma

Given any vectors $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_m$ and \mathbf{g} then the set

$$S = \{ \mathbf{s} \mid \mathbf{s}^T \mathbf{g} < 0, \tag{B.1}$$

$$\mathbf{s}^T \mathbf{a}_i \geq 0, \quad i=1,2,\dots,m \} \tag{B.2}$$

is empty if and only if there exist multipliers $\lambda_i \geq 0$ such that

$$\mathbf{g} = \sum_{i=1}^m \mathbf{a}_i \lambda_i$$

Appendix C

PARSEC-11 parameterisation

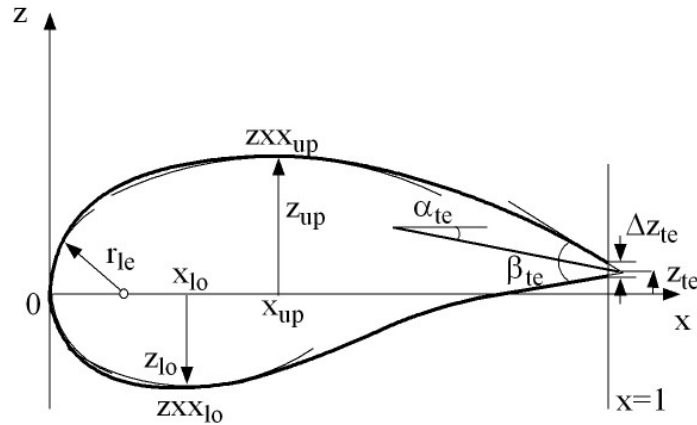


Figure C.1. PARSEC-11 airfoil parameterization

In PARSEC-11 (see Figure C.1), the upper and lower surface of an airfoil are described with a polynomial of the form:

$$z = \sum_{n=1}^6 a_n(\mathbf{p})x^{n-1/2} \quad (\text{C.1})$$

where x varies between 0 and 1. The coefficients a_n of the polynomial are real and depend on the design vector \mathbf{p} made of the 11 geometric parameters:

$$\mathbf{p} = \left[r_{le} \quad x_{up} \quad x_{lo} \quad z_{up} \quad z_{lo} \quad z_{xx_{up}} \quad z_{xx_{lo}} \quad z_{te} \quad \Delta z_{te} \quad \beta_{te} \quad \alpha_{te} \right] \quad (\text{C.3})$$

The 11 geometric parameters are used to derive 12 equations on the airfoil shape characteristics:

Equations

Characteristics

$$a_1^+ = \sqrt{2r_{le}} \quad \text{leading edge radius} \quad (C.4)$$

$$a_1^- = -\sqrt{2r_{le}} \quad \text{leading edge radius} \quad (C.5)$$

$$\sum_{n=1}^6 (n-1/2) a_n^+ \cdot x_{up}^{n-3/2} = 0 \quad \text{max thickness location} \quad (C.6)$$

$$\sum_{n=1}^6 (n-1/2) a_n^- \cdot x_{lo}^{n-3/2} = 0 \quad \text{max thickness location} \quad (C.7)$$

$$z_{up} = \sum_{n=1}^6 a_n^+ \cdot x_{up}^{n-1/2} \quad \text{max thickness} \quad (C.8)$$

$$z_{lo} = \sum_{n=1}^6 a_n^- \cdot x_{lo}^{n-1/2} \quad \text{max thickness} \quad (C.9)$$

$$z_{XX_{up}} = \sum_{n=1}^6 (n-1/2)(n-3/2) a_n^+ \cdot x_{up}^{n-5/2} \quad \text{curvature at max thickness} \quad (C.10)$$

$$z_{XX_{lo}} = \sum_{n=1}^6 (n-1/2)(n-3/2) a_n^- \cdot x_{lo}^{n-5/2} \quad \text{curvature at max thickness} \quad (C.11)$$

$$\Delta z_{te} = \sum_{n=1}^6 (a_n^+ - a_n^-) \quad \text{trailing edge thickness} \quad (C.12)$$

$$z_{te} = \sum_{n=1}^6 \frac{1}{2} (a_n^+ + a_n^-) \quad \text{trailing edge position} \quad (C.13)$$

$$\tan\left(\frac{\beta_{te}}{2} - \alpha_{te}\right) = -\sum_{n=1}^6 (n-1/2) a_n^+ \quad \text{trailing edge angles} \quad (C.14)$$

$$\tan\left(\frac{\beta_{te}}{2} + \alpha_{te}\right) = \sum_{n=1}^6 (n-1/2) a_n^- \quad \text{trailing edge angles} \quad (C.15)$$

where a_n^+ and a_n^- correspond to the coefficients of the polynomial for the upper and lower surface respectively.

Appendix D

In Chen et al. (2005), it is shown how the VBM main and total indices of a Kriging model with constant regression term can be derived analytically. To obtain the sensitivity indices in Chapter 6, these formulas have been extended in the case of a Kriging model with quadratic regression term.

Formulas for variance-based sensitivity analysis indices

In their paper, Chen et al. (2005) derive convenient formulas to obtain the VBM indices for tensor-product functions:

$$f(\mathbf{x}) = a_0 + \sum_{i=1}^{M_b} a_i B_i(\mathbf{x}) \quad (\text{D.1})$$

where

$$B_i(\mathbf{x}) = \prod_{l=1}^N h_{il}(x_l), \quad i = 1, \dots, M_b \quad (\text{D.2})$$

are called multivariate tensor-product basis functions and $h_{il}(x_l)$ are univariate basis functions.

It is shown that for a tensor-product function:

$$V_u = \sum_{i_1=1}^{M_b} \sum_{i_2=1}^{M_b} \left\{ a_{i_1} a_{i_2} \prod_{l=1}^N (C_{1,i_1 l} C_{1,i_2 l}) \left\{ C_{2,i_1,i_2,u} / (C_{1,i_1 u} C_{1,i_2 u}) - 1 \right\} \right\} \quad (\text{D.3})$$

$$V_{\sim u} = \sum_{i_1=1}^{M_b} \sum_{i_2=1}^{M_b} \left\{ a_{i_1} a_{i_2} \prod_{l=1}^N (C_{1,i_1 l} C_{1,i_2 l}) \left\{ \prod_{l=1, l \neq u}^N C_{2,i_1,i_2,l} / (C_{1,i_1 l} C_{1,i_2 l}) - 1 \right\} \right\} \quad (\text{D.4})$$

$$V = \sum_{i_1=1}^{M_b} \sum_{i_2=1}^{M_b} \left\{ a_{i_1} a_{i_2} \prod_{l=1}^N (C_{1,i_1 l} C_{1,i_2 l}) \left\{ \prod_{l=1}^N C_{2,i_1,i_2,l} / (C_{1,i_1 l} C_{1,i_2 l}) - 1 \right\} \right\} \quad (\text{D.5})$$

where

$$C_{1il} = \int h_{il}(x_l) p_l(x_l) dx_l \quad (D.6)$$

$$C_{2i_1i_2l} = \int h_{i_1l}(x_l) h_{i_2l}(x_l) p_l(x_l) dx_l \quad (D.7)$$

The main and total VBM sensitivity indices are given by:

$$S_i^1 = \frac{V_u}{V} \quad (D.8)$$

$$S_i^T = \frac{V - V_{\sim u}}{V} \quad (D.9)$$

In their paper, Chen et al. (2005) also show that some commonly used surrogate models can be written in the form of a tensor-product function. Here are given the results obtained for the second order regression model and Kriging model.

Second order regression model

In its more general form, it can be expressed as:

$$f(\mathbf{x}) = \beta_0 + \sum_{i=1}^N \beta_i x_i + \sum_{i=1}^N \beta_{ii} x_i^2 + \sum_{i=1}^{N-1} \sum_{j=i+1}^N \beta_{ij} x_i x_j \quad (D.10)$$

Defining:

$$\beta_{(i,j)} = \begin{cases} x_j & i = 0 \\ x_j^2 & i = j \\ x_i x_j & i < j \end{cases} \quad (D.11)$$

$$h_{(i,j)l} = \begin{cases} 1 & \text{none of } (i, j) = l \\ x_l & \text{only one of } (i, j) = l \\ x_l^2 & \text{both of } (i, j) = l \end{cases} \quad (D.12)$$

the second order regression model can be written in a tensor-product function:

$$f(\mathbf{x}) = \beta_0 + \sum_{0 \leq i, j \leq N, j \neq 0}^N \beta_{ij} B_{(i,j)} = \beta_0 + \sum_{0 \leq i, j \leq N, j \neq 0}^N \beta_{ij} \prod_{i=1}^N h_{(i,j)l} \quad (\text{D.13})$$

with $\beta_{0j} = \beta_j$.

The corresponding C_1 and C_2 are given by:

$$C_{1,(i,j)l} = \begin{cases} 1 & \text{none of } (i, j) = l \\ \int x_l p_l(x_l) dx_l = \mu_l & \text{only one of } (i, j) = l \\ \int x_l^2 p_l(x_l) dx_l = \mu_l^2 + \sigma_l^2 & \text{both of } (i, j) = l \end{cases} \quad (\text{D.14})$$

$$C_{2,(i,j)l} = \begin{cases} 1 & \text{none of } (i_1, j_1, i_2, j_2) = l \\ \int x_l p_l(x_l) dx_l = \mu_l & \text{only one of } (i_1, j_1, i_2, j_2) = l \\ \int x_l^2 p_l(x_l) dx_l = \mu_l^2 + \sigma_l^2 & \text{only two of } (i_1, j_1, i_2, j_2) = l \\ \int x_l^3 p_l(x_l) dx_l = 3\mu_l \sigma_l^2 + \mu_l^3 & \text{only three of } (i_1, j_1, i_2, j_2) = l \\ \int x_l^4 p_l(x_l) dx_l = \frac{9}{5} \sigma_l^2 + 6\mu_l^2 \sigma_l^2 + \mu_l^2 & \text{all of } (i_1, j_1, i_2, j_2) = l \end{cases} \quad (\text{D.15})$$

Kriging model with constant regression term

In the paper, the formulas are derived for a Kriging in the case of a gaussian correlation model and with the regression model reduced to a constant term:

$$f(\mathbf{x}) = \hat{\beta} + \sum_{i=1}^{M_b} \kappa_i \prod_{l=1}^N h_{il}(x_l) \quad (\text{D.16})$$

with

$$h_{il}(x_l) = \exp\left(-\theta_l (x_l - x_{il})^2\right) \quad (\text{D.17})$$

The corresponding C_1 and C_2 are defined by:

$$C_{1,il} = \frac{1}{2\delta_l} \sqrt{\frac{\pi}{\theta_l}} \left\{ \phi(a_{il}\sqrt{2\theta_l}) - \phi(b_{il}\sqrt{2\theta_l}) \right\} \quad (D.18)$$

$$C_{2,i_1,i_2,l} = \frac{1}{2\delta_l} \sqrt{\frac{\pi}{2\theta_l}} \exp\left(-\frac{\theta_l}{2}(x_{i_1l} - x_{i_2l})^2\right) \times \left\{ \phi\left(\sqrt{\theta_l}(a_{i_1l} + a_{i_2l})\right) - \phi\left(\sqrt{\theta_l}(b_{i_1l} + b_{i_2l})\right) \right\} \quad (D.19)$$

where

$$a_{il} = \mu_l + \delta_l - x_{il} \quad (D.20)$$

$$b_{il} = \mu_l - \delta_l - x_{il} \quad (D.21)$$

Kriging model with quadratic regression term

In this case, the Kriging model can be written as a sum of (D.10) and (D.16):

$$f(\mathbf{x}) = \beta_0 + \sum_{i=1}^N \beta_i x_i + \sum_{i=1}^N \beta_{ii} x_i^2 + \sum_{i=1}^{N-1} \sum_{j=i+1}^N \beta_{ij} x_i x_j + \sum_{i=1}^{M_b} \kappa_i \prod_{l=1}^N h_{il}(x_l) \quad (D.22)$$

which is still of the form of a tensor-product function. Therefore, equations (D.3)-(D.9) can be used to obtain analytically the VBM sensitivity indices.

All C_1 's have already been derived in (D.14) and (D.18). C_2 's where both univariate basis functions h_{il} come from the quadratic part or the Gaussian term have already been calculated and are given by (D.15) and (D.19) respectively. Only the C_2 cross-terms between the regression part and the Gaussian part must be derived:

$$C_{2(i,j)il} = \int h_{(i,j)l}(x_l) h_{il}(x_l) p_l(x_l) dx_l \quad (D.23)$$

Using the same notation as in Chen et al. (2005), it can be shown that:

$$C_{2(i,j)_{i,l}} = \begin{cases} C_{1,i,l} & \text{none of } (i, j) = l \\ I_1 + x_{i,l} C_{1,i,l} & \text{only one of } (i, j) = l \\ I_2 + \left(\frac{1}{2\theta_l} + x_{i,l}^2 \right) C_{1,i,l} & \text{both of } (i, j) = l \end{cases} \quad (\text{D.24})$$

where

$$I_1 = \frac{1}{4\delta_l\theta_l} \left(\exp(-\theta_l b_{i,l}^2) - \exp(-\theta_l a_{i,l}^2) \right) \quad (\text{D.25})$$

and

$$I_2 = \frac{1}{4\delta_l\theta_l} \left((2\mu_l - a_{i,l}) \exp(-\theta_l b_{i,l}^2) - (2\mu_l - b_{i,l}) \exp(-\theta_l a_{i,l}^2) \right) \quad (\text{D.26})$$

References

- Abramovitz, M., and Stegun, I. A. (1972). *Handbook of Mathematical Functions*. Dover Publication, New York.
- Andersson, J. (2002). Sensitivity analysis in Pareto optimal design. *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning (SEAL'02)*, Singapore, Nov. 18-22, 2002, pp. 156-161.
- Antony, J. (2003). *Design of Experiments for Engineers and Scientists*. Butterworth & Heinemann, Oxford.
- Balbas, A., Galperin, E., and Jimenez Guerra, P. (2005). Sensitivity of Pareto solutions in multiobjective optimization. *Journal of Optimization Theory and Applications*, Vol. 126, No. 2, pp. 247-264.
- Balling, R. J., and J. Richard. (2000). Pareto sets in decision-based design. *Journal of Engineering Valuation and Cost Analysis*, Vol. 3, No. 2, pp. 189–198.
- Balling, R. J., and Sobieszczanski-Sobieski, J. (1996). Optimization of couple systems: a critical overview of approaches. *AIAA Journal*, Vol. 34, No. 1, pp. 6-17.
- Bartholomew, P. (1998). The role of MDO within aerospace design and progress towards an MDO capability. *7th AIAA/USAF/NASA/ ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St.Louis, Missouri, USA, Sept. 2-4 1998, AIAA 98-4705.
- Bowcutt, K. G. (2003). A perspective on the future of aerospace vehicle design. *12th AIAA International Space Planes and Hypersonic Systems and Technologies*, Norfolk, Virginia, 15-19 Dec. 2003, AIAA 2003-6957.
- Campolongo, F., Tarantola, S., and Saltelli, A. (1999). Tackling quantitatively large dimensionality problems. *Computer Physics Communications*, Vol. 117, pp. 75-85.
- Chan, K., Saltelli, A., and Tarantola, S. (1997). Sensitivity analysis of model output: variance-based methods make the difference. *Proceeding of 1997 Winter Simulation Conference (WSC'97)*, Atlanta, GA, USA, Dec. 7-10 1997, pp. 261-268.
- Chen, W., Jin, R., and Sudjianto, A. (2005). Analytical variance-based global sensitivity analysis in simulation-based design under uncertainty. *ASME Journal of Mechanical Design*, Vol. 127, No. 5, pp. 875-886.

- Chen, W., Wiecek, M., and Zhang, J. (1999). Quality utility: a compromise programming approach to robust design. *ASME Journal of Mechanical Design*, Vol. 121, No. 2, pp. 179-187.
- Cukier, R. I., Levine, H. B., and Shuler, K., E. (1978). Non-Linear Sensitivity Analysis of Multi-Parameter Model Systems. *Journal of Computational Physics*, Vol. 26, No. 1, pp. 1-42.
- Das, I. and Dennis, J. (1997). A closer look at drawbacks of minimizing weighted sums of objectives for Pareto set generation in multicriteria optimization problems. *Structural Optimization*, Vol. 14, No. 1, pp. 63–69.
- Das, I. and Dennis, J. (1998). Normal-boundary intersection: a new method for generating Pareto optimal points in multicriteria optimization problems. *SIAM Journal on Optimization*, Vol. 8, No. 3, pp. 631-657.
- Deb K. (2001). *Multi-Objective Optimization using Evolutionary Algorithms*. Wiley and Sons, New York.
- Dennis, J. E. and Schnabel, R. B. (1983). *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Englewood Cliffs.
- Deremaux, Y., Willcox, K., and Haines, R. (2003). Physically-based, real-time visualization and constraint analysis in multidisciplinary design optimization. *33rd AIAA Fluid Dynamics Conference and Exhibit*, Orlando, Florida, USA, June 23-26 2003. AIAA 2003-3876.
- De Weck, O. L. (2004). Multiobjective optimization: history and promise. *3rd China-Japan-Korea Joint Symposium on Optimization of Structural and Mechanical Systems*, Kanazawa, Japan, Oct. 30-Nov. 2, 2004.
- ESDU 96028. VGK method for two-dimensional aerofoil sections – Part 1: Principles and results.
- ESDU 96029. VGK method for two-dimensional aerofoil sections – Part 2: User manual for operation with MS-DOS and UNIX systems.
- Fadel, G., Konda, S., Blouin, V., and Wiecek M. (2002). Epsilon optimality in bi-criteria optimization. *43rd AIAA/ASME Structures, Structural Dynamics, and Materials Conference*, Denver, Colorado, USA, April 22-25 2002. AIAA 2002-1233.
- Fadel, G., and Li, Y. (2002). Approximating the Pareto curve to help solve biobjective design problems. *Structural and Multidisciplinary Optimization*, Vol. 23, No. 4, pp. 280-296.

- Fantini, P. (2007). *Improving the Effectiveness of Multi-Disciplinary Design Optimisation at the Aircraft Conceptual Design Phase*. PhD thesis. Cranfield University.
- Fantini, P., Balachandran, L. K., and Guenov, M. D. (2007). Computational intelligence in multi-disciplinary optimization at feasibility design stage. *First International Conference on Multidisciplinary Design Optimization and Applications*, ASMDO, Besançon, France, April 17-20 2007.
- Fletcher, R. (1989). *Practical Methods of Optimization*. John Wiley & Sons, New York.
- Frey, H. C., and Patil, S. R. (2002). Identification and review of sensitivity analysis methods. *Risk Analysis*, Vol. 22, No. 3, pp. 553-578.
- Fulton, R. E., Sobieszczanski, J., Storaasli, O., Landrum, E. J., and Loendorf, D. (1974). Application of computer-aided aircraft design in a multidisciplinary environment. *Journal of Aircraft*, Vol. 11, No. 7, pp. 369-370.
- Giunta, A. A., Golivodov, O., Knill, D. L., Grossman, B., Haftka, R. T., Mason, W. H., and Watson, L. T. (1996). Multidisciplinary design optimization of advanced aircraft configurations. *MAD Center Report*, Virginia Tech, MAD 96-06-01.
- Guenov, M. D., Utyuzhnikov, S. V., and Fantini, P. (2005). Application of the modified physical programming method to generating the entire Pareto frontier in multi-objective optimization. *Proceedings of the 6th Eurogen Conference*, Munich, Sept. 12-14 2005.
- Gumbert, C. R., Hou, W., and Newman, P. A. (2001). Simultaneous aerodynamic analysis and design optimization for a 3-d flexible wing. AIAA 2001-1107
- Hernandez, S. (1995). A general sensitivity analysis for unconstrained and constrained Pareto optima in multi-objective optimization. *Proceedings of the 36th AIAA/ASME Structures Dynamics and Materials Conference*, 1995, AIAA-1288-CP.
- Hines, W. W., and Montgomery, D. C. (1980). *Probability and Statistics in Engineering and Management Science*, 2nd Ed. John Wiley & Sons, New York.
- Holden, C. M. E. (2004). *Visualization Methodologies in Aircraft Design Optimization*. PhD thesis. University of Southampton.
- Holden, C. M. E., and Keane, A. J. (2004). Visualization methodologies in aircraft design. *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, Aug. 30-Sept. 1 2004, AIAA 2004-4449.

- Holden, C. M. E., Davies, R., and Keane, A. J. (2002). Optimization methodologies in conceptual design. *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, Georgia, Sept. 4-6 2002, AIAA 2002-5524.
- Homma, T., and Saltelli, A. (1996). Importance measures in sensitivity analysis of non-linear models. *Reliability Engineering and System Safety*, Vol. 52, No. 1, pp. 1-17.
- Jeong, S., Chiba, K., and Obayashi, S. Data mining for aerodynamic design space. *23rd AIAA Applied Aerodynamics Conference*, Toronto, Canada, June 6-8 2005, AIAA 2005-5079.
- Jones, D., Schonlau, M., and Welch, W. (1998). Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, Vol. 13, No. 4, pp. 455-492.
- Kasprzak, E. M., and Lewis, K. E. (2001). Pareto analysis in multiobjective optimization using the collinearity theorem and scaling method. *Structural and Multidisciplinary Optimization*, Vol. 22, No. 3, pp. 208-218.
- Kasprzak, E. M., and Lewis, K. E. (2000). An approach to facilitate decision trade-offs in Pareto solution sets. *Journal of Engineering Valuation and Cost Analysis*, Vol. 3, No. 1, pp. 173-187.
- Kennedy, J. B., and Neville, A. M. (1986). *Basic Statistical Methods for Engineers and Scientists*, 3rd Ed. Harper & Row Publishers, New York.
- Kelton, W. D., and Barton, R. R. (2003). Experimental design for simulation. *Proceeding of 2003 Winter Simulation Conference (WSC'03)*, New Orleans, Louisiana, USA, Dec. 7-10 2003, pp. 59-65.
- Kim, Y., Kim, J., Jeon, Y., Bang, J., Lee, D. H., Kim, Y., and Park, C. (2002). Multidisciplinary aerodynamic-structural design optimization of supersonic fighter wing using response surface methodology. *40th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, Jan. 14-17 2002, AIAA 2002-0322.
- Kim, N. H., Wang, H., and Queipo, N. V. (2004). Adaptive reduction of design variables using global sensitivity analysis in reliability-based optimization. *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, Aug. 30-Sept. 1 2004, AIAA 2004-4515.
- Kleijnen, J. P. C. (2004a). Design and analysis of experiments with simulation models: classic and novel methods for sensitivity analysis. *4th International Conference on Sensitivity Analysis of Model Output, SAMO*, Santa Fe, New Mexico, March 8-11 2004.

- Kleijnen, J. P. C. (2004b). An overview of the design and analysis of simulation experiments for sensitivity analysis. *European Journal of Operational Research*, Vol. 164, No. 2, pp. 287-300.
- Kohonen, T. (2001). *Self-Organizing Maps*. Springer-Verlag, 3rd Edition.
- Kroo, I., Altus, S., Braun, R., Cage, P., and Sobieski, I. (1994). Multidisciplinary optimization methods for aircraft preliminary design. 5th *AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Panama City, Florida, Sept. 7-9 1994, AIAA 94-4325.
- Kulfan, B. (2007). Recent extensions and applications of the “CST” universal parametric geometry representation method. 7th *AIAA/ATIO Conference*, Belfast, Northern Ireland, Sept. 18-20 2007, AIAA 2007-7709.
- Kumano, T., Jeong, S., and Obayashi, S. (2006). Multidisciplinary optimization of wing shape for a small jet aircraft using Kriging Model. 44th *AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, Jan. 9-12 2006, AIAA 2006-932.
- La Rocca, G., Krakkers, L., and van Tooren, M. (2002). Development of an ICAD generative model for blended wing body aircraft. 9th *AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, Georgia, Sept. 4-6 2002, AIAA-2002-5447.
- Li, Y., Fadel, G. M., Wiecek, M., and Blouin, V. Y. (2003). Minimum effort approximation of the Pareto space of convex bi-criteria problems. *Optimization and Engineering*, Vol. 4, No. 3, pp. 231-261.
- Lophaven, S. N., Nielsen, H. B., and Sondergaard, J. (2002). DACE: A Matlab Kriging toolbox – Version 2. Informatics and Mathematical Modelling, Technical University of Denmark. Technical Report IMM-TR-2002-12.
- Lophaven, S. N., Nielsen, H. B., and Sondergaard, J. (2002). Aspects of the Matlab toolbox DACE. Informatics and Mathematical Modelling, Technical University of Denmark. Technical Report IMM-REP-2002-13.
- Marler, R. T., and Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, Vol. 26, No. 6, pp. 369-395.
- Martin, J. D., and Simpson, T. W. (2002). Use of adaptative metamodeling for design optimization. 9th *AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, Georgia, Sept. 4-6 2002, AIAA 2002-5631.
- Mason, R. L., Gunst, R. F., and Hess, J., L. (2003). *Statistical Design and Analysis of Experiments with Application to Engineering and Sciences*, 2nd Ed. John Wiley & Sons, Hoboken, New Jersey.

- Mc Kay, M. D., Conover, W. J., and Beckman, R. J. (1979). A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, Vol. 21, No. 2, pp. 239-245.
- Messac, A. (1996). Physical programming effective optimization for computational design. *AIAA Journal*, Vol. 34, No. 1, pp. 149-158.
- Messac, A., and Ismail Yahaya, A. (2002). Multiobjective robust design using physical programming. *Structural and Multidisciplinary Optimization*, Vol. 23, No. 5, pp. 357-371.
- Messac, A., and Mattson, C. A. (2002). Generating well-distributed sets of Pareto points for engineering design using physical programming. *Optimization and Engineering*, Vol. 3, No. 4, pp. 431-450.
- Miettinen, K. M. (1999). *Nonlinear Multi-Objective Optimization*. Kluwer Academic, Boston.
- Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics*, Vol. 33, No. 2, pp.161-174.
- Myers, R. H., and Montgomery, D. C. (1995). Response Surface Methodology. Wiley Series in Probability and Statistics, John Wiley & Sons, New York.
- Oyama, A., Obayashi, S., and Nakahashi, K., and Hirose, N. (1999). Fractional factorial design of genetic coding for aerodynamic optimization. *14th Computational Fluid Dynamics Conference*, Norfolk, VA, June 28-July 1 1999, Vol. 1 pp. 430-441, AIAA 99-3298
- Oyama, A. (2000). Multidisciplinary optimization of transonic wing design based on evolutionary algorithms coupled with CFD solver. *European Congress on Computational Methods in Sciences and Engineering (ECCOMAS 2000)*, Barcelona, Sept. 11-14 2000,
- Oyama, A., Obayashi, S., Nakahashi, K., Nakamura, T. (2000a). Aerodynamic optimization of transonic wing design based on evolutionary algorithm. *Third International Conference on Non-linear Problems in Aviation and Aerospace Methods and Software*, Daytona Beach, FL, USA.
- Oyama, A., Obayashi, S., Nakahashi, K., and Hirose, N. (2000b). Aerodynamic wing optimization via evolutionary algorithms based on structured coding. *Computational Fluid Dynamics Journal*, Vol. 8, No. 1. pp. 570-577.
- Palli, N., Azarm, S., Mc Cluskey, P., and Sundararajan, R. (1998). An interactive multistage ϵ -inequality constraint method for multiple objectives decision making. *ASME Journal of Mechanical Design*, Vol. 120, pp. 678-686.

- Rao, S. S. (1996). *Engineering Optimization*. 3rd Ed. John Wiley & Sons, New York.
- Rosen, J. B. (1960). The gradient projection method for nonlinear programming. Part I. Linear constraints. *Journal of the Society for Industrial and Applied Mathematics*, Vol.8, No. 1, pp.181-217.
- Ruzika, S., and Wiecek, M. M. (2005). Approximation methods in multiobjective programming (Survey paper). *Journal of Optimization Theory and Applications*, Vol. 126, No. 3, pp. 473-501.
- Sacks, J., Schiller, S. B., and Welch, W. J. (1989a). Design of computer experiments. *Technometrics*, Vol. 31, No. 1, pp. 41-47.
- Sacks, J., Welch, W. J., Mitchell, T. J., and Wynn, H. P. (1989b). Design and analysis of computer experiments. *Statistical Science*, Vol. 4, No. 4, pp. 409-435.
- Saisana, M., Saltelli, A., and Tarantola, S. (2005). Uncertainty and sensitivity analysis techniques as tool for the quality assessment of composite indicators. *Journal of the Royal Statistical Society - A*, Vol. 168, No. 2, pp. 307-323.
- Saltelli, A. (2002). Sensitivity analysis for importance assessment. *Risk Analysis*, Vol. 22, No. 3, pp. 579-590.
- Saltelli, A., Andres, T. H., and Homma, T. (1993). Sensitivity analysis of model output: an investigation of new techniques. *Computational Statistics & Data Analysis*, Vol. 15, No. 2, pp. 211-238.
- Saltelli, A., and Bolado, R. (1998). An alternative way to compute Fourier Amplitude Sensitivity Test (FAST). *Computational Statistics & Data Analysis*, Vol. 26, No. 4, pp. 445-460.
- Saltelli, A., Chan, K., and Scott, M. (2000a). *Sensitivity Analysis*. John Wiley & Sons, New York.
- Saltelli, A., Tarantola, S., and Campolongo, F. (2000b). Sensitivity analysis as an ingredient of modelling. *Statistical Science*, Vol. 15, No 4, pp. 377-395.
- Saltelli, A., and Tarantola, S. (2003). *SimLab 2.2, Reference Manual*. European Commission - Institute for the Protection and Security of the Citizen (IPSC).
- Saltelli, A., Tarantola, S., and Chan, K. P. S. (1999). A quantitative model-independent method for global sensitivity analysis of model output. *Technometrics*, Vol. 41, No. 1, pp. 39-56.
- Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. (2004). *Sensitivity Analysis in Practice: a guide to assessing scientific models*. John Wiley & Sons, Chichester.

- Scott, A. T. (2001). An evaluation of three commercially available integrated design framework packages for use in space systems design lab. *MS Special Problem Report*, Georgia Institute of Technology Atlanta, GA, USA, April 2001.
- Shan, S., and Wang, G. G. (2005). An efficient Pareto set identification approach for multiobjective optimization on black-box functions. *ASME Journal of Mechanical Design*, Vol. 127, No. 5, pp. 866-874.
- Siddall, J. N. (1982). *Optimal Engineering Design – Principles and applications*, Marcel Dekker Inc., New York.
- Simpson, T. W., Lin, D. K. L., and Chen, W. (2001). Sampling Strategies for Computer Experiments: Design and Analysis, *International Journal of Reliability and Applications*, Vol. 2, No. 3, pp. 209-240.
- Simpson, T. W., Booker, A. J., Ghosh, D., Giunta, A. A., Koch, P. N., and Yang, R. J. (2004). Approximation methods in multidisciplinary analysis and optimization: a panel discussion. *Structural and Multidisciplinary Optimization*, Vol. 27, No. 5, pp. 302-313.
- Sobieczky, H. (1998). Parametric airfoils and wings. *Notes on Numerical Fluid Mechanics*, Vol. 68, pp. 71-88.
- Sobieszczanski-Sobieski, J. (1990). *Sensitivity of Complex, Internally Coupled Systems*, AIAA Journal, Vol. 28, No. 1, pp. 153-160.
- Sobieszczanski-Sobieski, J., and Loendorf, D. (1972). A mixed optimization method for automated design of fuselage structures. *Journal of Aircraft*, Vol. 9, No. 12, pp. 805-811.
- Sobieszczanski-Sobieski, J., Barthelemy, J. F. M., and Giles, G. L. (1984). Aerospace engineering design by systematic decomposition and multilevel optimization. NASA , Langley Research Center, Hampton, VA, June 1984, NASA TM 85823.
- Sobol', I. M. (1993). Sensitivity Estimates for non-linear mathematical models. *Mathematical Modelling & Computational Experiments*, Vol. 1, No. 4, pp. 407-414.
- Sobol', I. M. (2001). Global sensitivity indices for non-linear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, Vol. 55, pp. 271-280.
- Statnikov, R. B., and Matusov, J. B. (1995). *Multicriteria Optimization and Engineering*. Chapman & Hall, New York.
- Sudret, B. (2006). Global sensitivity analysis using polynomial chaos expansion. *5th Computational Stochastic Mechanics Conference (CSM5)*, Rhodes, Greece, June 19-23, 2006.

- Tappeta, R. V., and Renaud, J. E. (1999). Interactive multiobjective optimization procedure. *AIAA Journal*, Vol. 37, No. 7, pp. 881-889, AIAA 99-1207.
- Tappeta, R. V., and Renaud, J. E. (2001). Interactive multiobjective optimization design strategy for decision based design. *ASME Journal of Mechanical Design*, Vol. 123, No. 6, pp. 205-215.
- Tappeta, R. V., Renaud, J. E., Messac, A., and Sundararaj, G. J. (2000). Interactive physical programming: trade-off analysis and decision making in multidisciplinary optimization. *AIAA Journal*, Vol. 38, No. 5, pp. 917-926.
- Tappeta, R. V., Renaud, J. E., and Rodriguez, F. (2002). An interactive multiobjective optimization design strategy for decision based multidisciplinary design. *Engineering Optimization*, Vol. 34, No. 5, pp. 523-544.
- Utyuzhnikov, S. V., Guenov, M. D., and Fantini, P. (2005). Numerical method for generating the entire Pareto frontier in multi-objective optimization. *Proceedings of the 6th Eurogen Conference*, Munich, Sept. 12-14 2005.
- Valvalle, A. (2005). Response Surface Aerodynamic optimisation for Blended Wing Body Aircraft. PhD thesis. Cranfield University.
- Van Beers, W. C., M. and Kleijnen, J. P. C. (2003). Kriging for interpolation in random simulation. *Journal of Operational Research Society*, Vol. (54), pp. 255-262.
- Vesanto, J. (1999). SOM-based data visualization methods. *Intelligent Data Analysis*, Vol. 3, pp. 111-126.
- Vesanto, J., Himberg, J., Alhoniemi, E., and Parhankangas, J. (2000). SOM Toolbox for Matlab. Helsinki University of Technology. Report A57.
- Vincent, T. L., and Grantham, W. J. (1981). *Optimality in Parametric Systems*. John Wiley & Sons, New York.
- Walsh, J. L., Townsend, J. C., Salas, A. O., Samareh, J. A., Mukhopadhyay, V., and Barthelemy, J. F. (2000). Multidisciplinary high-fidelity analysis and optimization of aerospace vehicle, part 1: formulation. *38th AIAA Aerospace Sciences Meeting and Exhibit*, Reno, Nevada, Jan. 10-13, 2000, AIAA 2000-0418.
- Wiecek, M., Chen, W., and Zhang, J. (2001). Piecewise quadratic approximation of the non-dominated set for bi-criteria programs. *Journal of Multi-Criteria Decision Analysis*, Vol. 10, No. 1, pp 35-47.
- Wilson, B., Cappelleri, T., Simpson, T. W., and Frecker, M. (2001). Efficient Pareto frontier exploration using surrogate approximations. *Optimization and Engineering*, Vol. 2, No. 1, pp. 31-50.

- Yang, B. S., Yeun, Y. S., and Ruy, W. S. (2002). Managing approximation models in multiobjective optimization. *Structural and Multidisciplinary Optimization*, Vol. 24, No. 2, pp. 141-156.
- Zhang, J., Wiecek, M., and Chen, W. (2000). Local approximation of the efficient frontier in robust design. *ASME Journal of Mechanical Design*, Vol. 122, No. 2, pp. 232-236.
- Zhang, W. H. (2003a). Pareto optimum sensitivity analysis in multi-criteria optimization. *Finite Elements in Analysis and Design*, Vol. 39, No. 5, pp. 505-520.
- Zhang, W. H. (2003b). On the Pareto optimum sensitivity analysis in multi-criteria optimization. *International Journal for Numerical Methods in Engineering*, Vol. 58, No. 6, pp. 955-977.

Bibliography

- Abbot, I. H., and Von Doenhoff, A. E. (1959). *Theory of Wing Sections including a Summary of Airfoil Data*. Dover Publications, New York.
- Baccari, A. (2004). On the classical necessary second-order optimality conditions. *Journal of Optimization Theory and Applications*, Vol. 123, No. 1, pp. 213-221.
- Chiba, K., Jeong, S., Obayashi, S., and Morino, H. (2005). Data mining for multidisciplinary design space of regional-jet wing. *Proceeding of the 2005 IEEE congress on Evolutionary Computation*, Sept. 2-5 2005, Vol. 3, pp. 2333- 2340.
- Koch, P. N., Wujek, B., Golovidov, O., and Simpson, T. W. (2002). Facilitating probabilistic multidisciplinary design optimization using Kriging approximation models. *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, Georgia, Sept. 4-6 2002, AIAA 2002-5415.
- Liu, H., Chen, W., and Sudjianto, A. (2004). Probabilistic sensitivity analysis method for design under uncertainty. *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, New York, August 30-Sept. 1 2004, AIAA 2004-4589
- Mavris, D., Roth, B., and Macsotai, N. (1999). A method for probabilistic sensitivity analysis of commercial aircraft engines. *14th International Symposium on Air-Breathing Engines*, Florence, Italy, Sept. 5-10 1999, ISABE 99-7195.
- Meeker, W. Q., and Escobar, L. A. (2001). Use of sensitivity analysis to assess the effect of model uncertainty in analyzing accelerated life test data. *Proceedings of the 2001 Joint Statistical Meetings*, Atlanta, Georgia, August 2001, 300971
- Papalambros, P. Y., and Wilde, D. J. (2000). *Principles of Optimal Design – Modeling and Computation*. 2nd Ed. Cambridge University Press, New York.
- Papila, M., and Haftka, R. T. (2001). Uncertainty and response surface approximations. *42nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit*, Seattle, WA, April 16-19 2001, AIAA 2001-1680.
- Pinto, V. D., Pottenger, W. M., and Thompkins, W. (2000). A survey of optimization techniques being used in the field. *Proceedings of the 3rd International Meeting on Research in Logistics (IMRL 2000)*, Quebec, Canada, May 9-11 2000.
- Raymer, D. P. (1999). *Aircraft Design: A Conceptual Approach*, 3rd Ed. AIAA Education Series, Reston, Virginia.
- Shahpar, S. (2004). Design of experiment, screening and response surface modelling to minimise the design cycle time. *Optimization Methods & Tools for*

multicriteria/Multidisciplinary Design Applications to Aeronautics and Turbomachinery, VKI Lecture Series. Von Karman Institute, Rhode Saint Genèse, Belgium. VKI 2004-07.

Simpson, T. W., Peplinski, J., Koch, P. N., and Allen, J. K. (2001). Metamodels for computer-based engineering design: survey and recommendations. *Engineering with Computers*, Vol. 17, No. 2, pp. 129-150.

Tamura, K. (1979). A method for constructing the polar cone of a polyhedral cone, with applications to linear multicriteria decision problems. *Journal of Optimization Theory and Applications*, Vol. 19, No. 4, pp. 547-564.

Wah, B. W., and Wang, T. (1999). Efficient and adaptative Lagrange-multiplier methods for nonlinear continuous global optimization. *Journal of Global Optimization*, Vol. 14, pp. 1-25.

Wehrens, R., and Buydens., L. M. C. (2000). Classical and non-classical optimization methods. *Encyclopedia of Analytical Chemistry*. John Wiley & Sons, Chichester, pp. 9678-9689.

List of publications

Conference papers

Maginot, J., Guenov, M. D., Fantini, F., and Padulo, M. (2007). A method for assisting the study of Pareto solutions in multi-objective optimization. 7th *AIAA/ATIO Conference*, Belfast, Northern Ireland, Sept. 18-20 2007.

Utyuzhnikov, S. V., Maginot, J., and Guenov, M. D. (2007). Local approximation of Pareto surface. *World Congress on Engineering (WCE)*, London, UK, July 2-4 2007.

Utyuzhnikov, S. V., Maginot, J., and Guenov, M. D. (2006). Local Pareto analyser for preliminary design. 25th *International Congress of Aeronautical Sciences (ICAS)*, Hamburg, Germany, Sept. 3-8 2006.

Padulo, M., Campobasso, S., Guenov, M. D., and Maginot, J. (2007). Derivative-free uncertainty propagation methods for airfoil robust design. *EUROMECH Colloquium 482*, London, Sept. 10-12 2007.

Journal papers

Utyuzhnikov, S. V., Maginot, J., and Guenov, M. D. (2007). Local Pareto approximation for multi-objective optimization. *Engineering Optimization*. Accepted for publication.

Utyuzhnikov, S. V., Maginot, J., and Guenov, M. D. (2007). Local Pareto approximation. *AIP*. Accepted for publication.