

**UNIVERSITÀ DEGLI STUDI DI ROMA  
TOR VERGATA**



Facoltà di Ingegneria

Tesi di Laurea Magistrale in Ingegneria Meccanica

**ADVANCED OPTIMIZATION OF THE WT MODEL  
OF THE RIBES PROJECT**

Supervisor:

Prof. Marco Evangelos Biancolini

Candidate:

Domiziano Beltramme

Assistant supervisors:

Dt. Corrado Groth

Dt. Ubaldo Cella

Anno accademico 2013/2014

*Le suprême bonheur de la vie,  
c'est la conviction qu'on est aimé*  
*Victor Hugo*

# Acknowledgements

I would like to express the deepest gratitude to my professor Marco Evangelos Biancolini, which guided me through this important experience with his enthusiasm and knowledge. Without his help this thesis would have not be possible.

I would like to thank eng. Corrado Groth for his daily availability and his persistent support, always ready to discuss in order to figure out every thesis' problem.

In addition, a thank you to eng. Ubaldo Cella for his constructive contribution to the build of this dissertation.

A tender and hearty thanks to my dears, eternal pillar of life, and to my friend, irreplaceable traveling companions.

# Contents

<b>1</b>	<b>Introduction</b>	<b>7</b>
<b>2</b>	<b>RIBES project</b>	<b>9</b>
2.1	Overview . . . . .	9
2.2	The Wind Tunnel model . . . . .	11
2.3	Load transfer challenge . . . . .	15
<b>3</b>	<b>Optimization</b>	<b>17</b>
3.1	State of art . . . . .	17
3.2	Response Surface Methodology . . . . .	22
3.3	Optimization strategy . . . . .	24
<b>4</b>	<b>Tools</b>	<b>26</b>
4.1	Radial Basis Function . . . . .	26
4.2	Latin hypercube design . . . . .	31
4.3	Software . . . . .	32
4.3.1	Matlab . . . . .	32
4.3.2	Nastran . . . . .	32
4.3.3	Femap . . . . .	35

<b>5</b>	<b>Workflow development</b>	<b>36</b>
5.1	Structure . . . . .	37
5.2	Program components . . . . .	43
5.2.1	Input file . . . . .	43
5.2.2	Matlab scripts . . . . .	46
5.3	Model parametrization . . . . .	52
5.3.1	Thickness change . . . . .	52
5.3.2	Shape change . . . . .	53
<b>6</b>	<b>Results</b>	<b>57</b>
6.1	Connecting rod case . . . . .	57
6.2	WT model case . . . . .	61
6.2.1	Description and previous results . . . . .	61
6.2.2	WT model optimization . . . . .	65
<b>7</b>	<b>Conclusion</b>	<b>74</b>

# Abstract

The present thesis deals with the realization of a program that implements a DOE (design of experiment) structural study, a consequent interpolation of the results and allows their analysis, with the aim to pursuit a graphical optimization. The entire process is automatic, but some input files has to be customized and provided by the user depending on the model and on the study. The main application of the workflow will be the wind tunnel model of the RIBES project.

Chapter 1 exhibits a brief introduction, in order to make aware the reader about the context of the thesis.

Chapter 2 deals with the RIBES project: its little description, its goals and technological challenges, and the WT model to be analyzed, with the parameters of the real wing and the finite element model. An own subsection has been created for the load transfer challenge, in order to underline the importance and the difficulty of this transfer process.

Chapter 3 describes the optimization, including the state of art of the structural one, paying attention especially on the CAD and mesh based optimizations, and the entire strategy adopted in the workflow. Furthermore the response surface methodology is deeply described because is the basis of

the optimization process implemented.

Chapter 4 summarizes all tools used to realize the workflow: the radial basis functions, that are the interpolating tools used to obtain the metamodel, the Latin hypercube, the preprocessing method adopted to select the design variables for the DOE campaign, and the employed programs, with a brief description of their exploited features.

Chapter 5 exhibits the development of the workflow itself, with all the information, which are essential for its proper use. Each part of the program is described and explained with accurate illustrations.

Chapter 6 analyzes and comments two performed cases: the connecting rod and the WT model. In both cases, the model is first described, then customized and finally the results are showed and discussed.

Chapter 7 summarizes the conclusions and future works are briefly exhibited.

# Chapter 1

## Introduction

One of the most important topic, discussed all over the world, is the environment pollution. Since the industrial revolution, tons of polluters have been released and today this problem has gained a primary role in the engineering optimization of the automotive and aeronautical structures.

The reduction of pollutant emissions can be achieved by:

1. Abatement systems, which generally lead an improvement of the efficiency;
2. Employ of finer, but more expensive, fuels;
3. Optimization of the whole system.

Different project have been created to tackle the pollution such as Clean Sky 2 program; its key pillar is to demonstrate that substantial reduction in CO<sub>2</sub>, NO<sub>X</sub> emissions and fuel consumption can be achieved. To assess the Clean Sky's progress towards these environmental gaols, data concerning effi-



ciency, fuel consumption, weight, etc. are taken and analyzed by Technology Evaluator (TE). The goals are:

- 50% reduction in CO<sub>2</sub> emissions per ton-km;
- 80% reduction in NO<sub>X</sub> emissions per ton-km;
- 50% reduction in perceived noise;
- a substantial improvement in the aircraft life cycle impact on the environment.

It is thus clear that one of the main problems is to reduce weight with a consequential reduction of fuel consumption and pollutant emissions, taking advantage of the maximum limit of the strength of materials. Thereby the design optimization must be performed and tools with higher performances are essential to get solution to technical and scientific problems. One of these problem is to transfer loads from fluid dynamic mesh to a structural one, due to the different number of nodes. The program oriented to this particular aspect is the RIBES project.

# Chapter 2

## RIBES project

### 2.1 Overview

The goal of RIBES (Radial basis function at fluid Interface Boundaries to Envelope flow results for advanced Structural analysis) is to mitigate pollutant emissions and fuel consumption by means of a generalized optimization process, not only changing the shape and the materials, as in the past, but also through efficient design tools. These tools would allow to reduce weight and therefore increase the performances of the aircraft.

To reach this perspective, it's mandatory to simulate properly the aerodynamic interaction by means of numerical methodologies and software tools. The right behavior of the aircraft structure is obtained by the accurate transfer of loads between the CFD and FEM numerical models and the transfer itself represents the technological challenge. The complexity lies in the non matching meshes when flow data exchange has to be performed on huge calculation models.

This project proposes an innovative approach for load mapping exploiting the Radial Basis Function (see section 4.1) that was previously implemented in the field of mesh morphing (RBF Morph<sup>TM</sup>). This technology has to be demonstrated through an aeronautic wing structure, comparing the numerical and the experimental results.

## 2.2 The Wind Tunnel model

The accuracy of the developed load transfer numerical tools, will be validated against a case of aeronautical interest. Such a test case should accomplish the task of being significant of a realistic design problem and being suitable to be experimentally verified.

A typical wing structure will be developed referring to a realistic aircraft. In order to maximize the interaction between aerodynamic loads and wing deformation, a swept back wing will be adopted. Furthermore, to maximize the load similitude with a scaled wing structure model to be experimentally verified in a low speed wind tunnel, a relatively low wing load case should be selected. With this vision, the wing is supposed to refer to a geometry suitable for an ultra-light jet class aircraft (fig.2.1).

Table 2.1: Real aircraft parameters

<b>Wing span</b>	9.5 <i>m</i>
<b>Reference wing surface</b>	12 <i>m</i> <sup>2</sup>
<b>Total mass</b>	21660 <i>Kg</i>
<b>Service ceiling</b>	12500 <i>m</i>
<b>Cruise Mach</b>	0.75

The main data are summarized in tab.2.1.

The objective of the wind tunnel test is to provide a database of experimental measurements, in terms of loads, stresses and displacements, to be compared with the analysis performed, in the same conditions, by a numerical aero-elastic analysis environment. Such tool integrates the FEM model of the wing structure and the CFD computation coupled by the load mapping and transferring tool developed within the project. To reach this goal, a

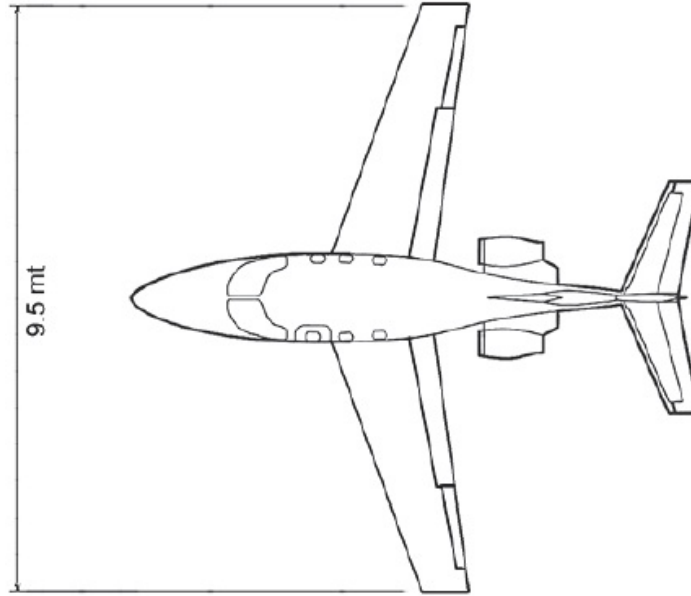


Figure 2.1: Ultra-light jet

scaled wind tunnel model of half wing in clean configuration and reproducing a typical wing box structure will be built.

The scale of the model will be 1:2.5 to which correspond a span of 1.6 meters (see fig.2.2 and tab.2.2).

Table 2.2: model wing parameters

<b>Model scale</b>	1 : 2.5
<b>Span</b>	1600 <i>mm</i>
<b>Reference surface</b>	0.754 <i>m</i> <sup>2</sup>
<b>MAC</b>	495 <i>mm</i>
<b>Root chord</b>	653 <i>mm</i>
<b>Tip chord</b>	288 <i>mm</i>
<b>Root thickness</b>	85 <i>mm</i>
<b>Tip thickness</b>	29 <i>mm</i>
<b>LE sweep angle</b>	20°

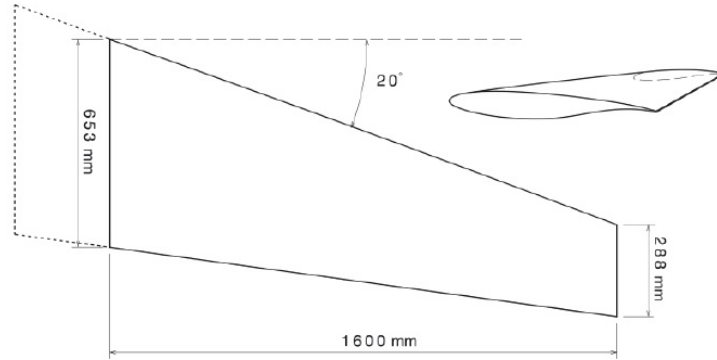


Figure 2.2: Model wing

A complete structural and load similitude at testing conditions would require a relatively high speed which is not compatible with a typical not pressurized low speed wind tunnel with a sufficient large test section (the speed required to generate a similar scaled load on a model with a span smaller than 2 meter, assuming the same full scale aircraft wing load and design cruising lift coefficient, would be higher than Mach 0.3). Such task, in any case, is not the objective of the experimental tests. It is then assumed to orient the setup of the tests campaign to the conservation of the similitude in terms of amount of model deformation and shape of wing load distribution. This configuration will provide an efficient base of validation representative of a typical design case.

In order to reproduce in subsonic conditions the pressure distribution that the aircraft would exhibit in cruise, a proper aerodynamic design of the WT model will be performed. The model design is oriented to reproduce as much as possible the span-wise load and the shape of pressure distribution will be elliptical.

The wing will be installed on the side wall of the test section as a cantilever and equipped with strain gauges, to measure stresses, located in several points of the wing structure according to the solution of a dimensioning FEM analysis.

A number of 80 pressure taps will be installed on both pressure and suction side of the model, along the span and along two sections at two stations, in order to provide a reasonable crowded cloud of point to be compared to the CFD pressure distribution estimation.

A marker matrix will be placed on the surface model and a photogrammetry technique will be adopted to evaluate the structural deformation. The geometric displacement will be reconstructed in 3D and compared to the FEM computed deformation.

The test matrix will cover a speed range from 35 to 45 m/s. Several wing polars will be measured and the wing deformation at the most significant test conditions will be reconstructed from the recorded images. The fluid dynamic computational domain will reproduce the wind tunnel configuration.

## 2.3 Load transfer challenge

As mentioned before, the accurate transfer of loads between the CFD and FEM numerical models represents a technological challenge. It is a very important topic especially for multiphysics applications where quantities obtained on a numerical grids need to be transferred across different models.

In aeronautical field a common application is the transfer of pressure distribution computed by CFD simulation on the FEM grid of the structure. The structural solution can be updated and directly used for stress assessment or, in the case of deformable structure in which a strong coupling is observed, to get the deformation field at wet boundaries to update the CFD mesh in the deformed position using mesh morphing. Information exchanging has to be iterated until a stable solution is observed on the deformed model.

Mapping methods needs to fulfill accuracy, flexibility , handling of dissimilar meshes including the cases fine to coarse and vice versa, and performance requirements, i.e. capability to manage very large models in a reasonably short time. It is worthwhile to notice that last requirement can be the most difficult to satisfy. The great fidelity of CFD and CSM simulation is achieved using so large meshes (hundreds millions in industrial applications, some billions in research studies) that very large problems can arise at mesh boundaries.

A good review about load transfer schemes can be found in the studies by Jiao & Heat [8] and Jaiman et al. [6].

Among them, the high quality interface needs particular attention. This method leads to a detailed surface exploiting the RBF interpolation and it



combines several features of existing methods (point wise, area weighted, etc...). This mapping methodology is based on a point wise representation of the source that allows to accurately interpolate as a point function both the source value (for instance the pressure) and the source geometry itself. So the original flow solution becomes meshless and is defined onto a meshless representation of the surface. It's similar to an artificial thin shell structure that covers the fluid-structure interface as a vehicle for transferring the load and the displacement data between the fluid and the structural domains. Such interpolator is then tuned with respect to the original source solution and mesh so that local and global equilibrium is satisfied and the high quality interface becomes available. It is then used for mapping onto the target enforcing the equilibrium on an element by element basis between target element and its projection onto the high quality interface. Shape functions of target elements are used to represent the map applied on the target and the coefficients are obtained enforcing the equilibrium between target and its projection onto the high quality interface.

It is worth to notice that the high quality interface can be defined regardless the level of the source refinement.

The most sophisticated component of this method is the interpolator: the Radial Basis Function; for more detail see section 4.1.

# Chapter 3

## Optimization

### 3.1 State of art

The concept of optimization is basic to much of what we do in our daily lives and the dictionary help to understand why: it means "making the best of something" i.e. a the best result we can perform in a specific field.

This point of view is began obviously a pillar in the engineering field: we wish to produce the best possible result with the available resources. In a highly competitive modern world it is no longer sufficient to design a system whose performance of the required task is just satisfactory. It is essential to design the best system. Thus in "designing" new products in any field: aerospace, automotive, chemical, electrical, biomedical, agricultural, etc, we must use design tools which provide the desired results in a timely and economical fashion. Numerical optimization is one of the tools at our disposal. It is an automated design technique and its purpose is to aid users in searching among alternative designs solution choosing the best one.

The alternative designs depend on some parameters called design variables; changing these values, the optimization process look for the maximum or minimum of some characteristics called objective function. Most of the engineering problem need that certain requirements must be satisfied; these requirements are called design constraints.

The best combination between the design variables, that optimizes the objectives function, is the optimal solution.

As regard the structural optimization, it is a very important topic especially for transport applications where weight reduction is directly connected to an improvement of performances and to a reduction of energy request. Finite Element Analysis (FEA) allows to accurately reproduce the response of a structural component and optimization tools are commonly used in several commercial software capable to address both shape and topological optimization.

The first type allows to remove material from the structure without the constraint of preserving the original topology; a typical example is the creation of holes. It's useful especially in the first stage of design to define a rough starting shape of the candidate component. It has several drawbacks: it's difficult to control how the optimizer will remove material and furthermore it's tricky to digitize obtained shape in a new geometrical model. Nevertheless it's commonly used and it can be combined with mesh morphing tool to refine obtained shape so that the definition of CAD representation can be facilitated.

Once that a good topological layout is available there is still the room for further improvements acting on the shape of the component. Shape

optimization tools available for structural analysis can be divided mainly in two family: CAD based and mesh based.

CAD based shape optimization relies in the CAD parameterization of the geometry. Several shape features can be introduced as CAD parameters (and in this case topology could not be preserved) each one controlled by a specific scalar parameter. The CAD is capable to generate a new solid (brep representation) for a given set of parameter. A new mesh is then defined and submitted to the solver.

Mesh based parameterization usually is accomplished by mesh morphing. New shapes are generated by deforming the mesh of the baseline model, i.e. just updating nodal positions, which requires a negligible computational time compared to any re-meshing procedure. Importantly, preserving the same mesh structure eliminates the re-meshing noise that can be confused with the effect of the design parameters.

Several algorithms have been explored for this task. In 2007 an innovative approach was created using Radial Basis Functions (RBF) interpolation that combines the benefits of a meshless method with great precision [see 7, 4]. The first industrial implementation of RBF mesh morphing was introduced in 2009 with the software RBF Morph [see 2].

Mesh morphing for shape optimization is available as an out of the box option in the Nastran solver using the SOL 200, a practical application of this solution strategy is given by [see 3] as shown in fig.3.1. Sensitivity information are used by the optimizer that exploits the gradient method in an optimization sequence in which several parameters of the FEM model (material, thickness, beam cross section, mesh shape) can be controlled to

optimize desired objective functions (weight, max deflection, max stress,..).

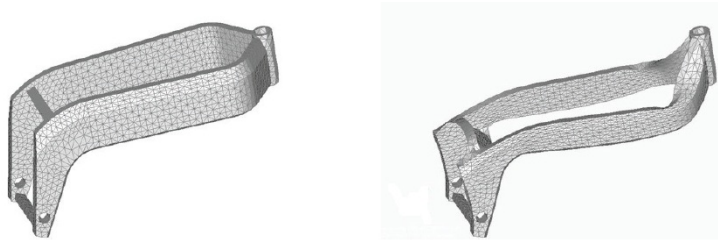


Figure 3.1: Motorbike frame, example of shape optimization

Multi-objective multi-parameters structural optimization using large FEA model could be costly for large meshes usually adopted in industrial design. This makes attractive the option of Response Surface (RS) methods 3.2. Several design are evaluated on a predefined optimal filled design matrix using the Design Of Experiment (DOE) approach. Then the matrix is completed with responses by means of an automated evaluation of FEM solutions.

Optimization algorithms are then used on the RS model that is an interpolation of responses in the space parameters. Optimal candidates need to be verified by direct calculation so the original DOE map is usually refined adding new points in most promising area that deserve a better resolution. RS evaluation, i.e. the operation of interpolation in the design space, can be effectively executed using several methods. Among them the better performing ones are the Kriging method and the RBF interpolation [see 9].

Several families of RBF can be used to interpolate responses in the space parameters. Very good results are achieved using method for DOE post processing (Lapack solver included in Mathcad software), with the RBF spline  $\varphi(r) = r^3$ . An example of shape optimization of an engine airbox is given by

Biancolini & Abbasciano [see 1].

## 3.2 Response Surface Methodology

Response surface methodology (RSM) is a set of statistical and mathematical techniques which allow to developing, improving and optimizing processes [11].

It is widely used in the industrial world, especially in problems where several input variables potentially influence the performances of the problem itself and it is often an important concurrent engineering tool. The performance measure or quality characteristic is called the **response** and the input variables **design variables** (DV).

The purpose of the RSM is to explore the space of the DVs, create an appropriate approximating relationship between the process variables and the objective function, and exploit optimization methods for finding the best solution.

It is worth to point out that this last objective is not performed by mathematical optimization methodology, as the method of steepest ascent, but through a screening method. It means that we want to represent by diagrams the interaction between DVs and objective function and find graphically the best design point. Obviously this entails that the number of variables must be maximum two, otherwise it is impossible to represent the entire response surface.

Moreover, in most response surface problems there are several responses that must in some sense be simultaneously considered. For example in a wing structural engineering analysis both the mass of the structure and the maximum stress must be taken into account. One way that we could solve this problem is to obtain response surfaces for all two objective functions and

then superimpose their contours, as shown in fig.3.2.

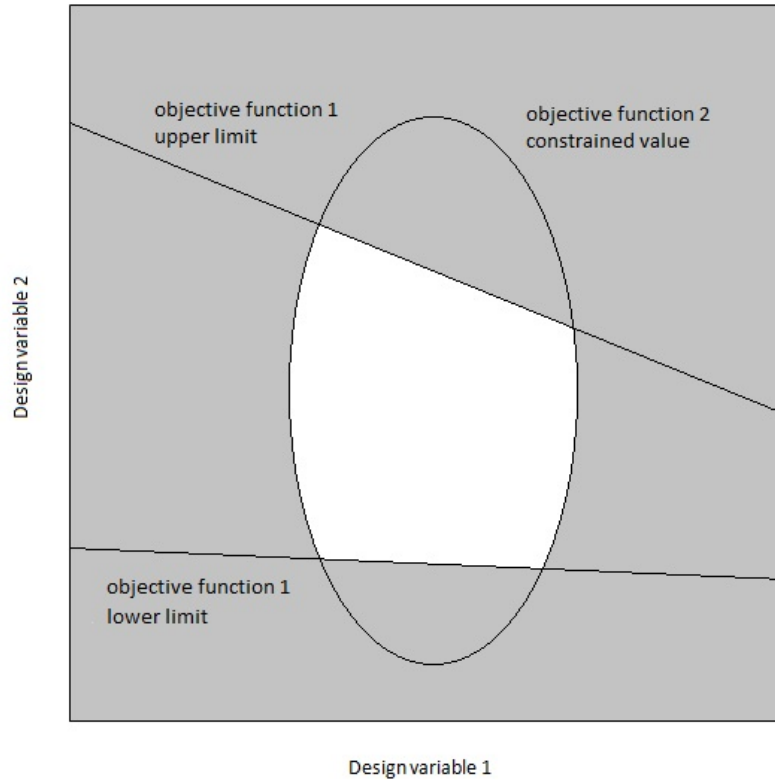


Figure 3.2: Multiresponse contour plot

The unshaded region in the figure represents all the possibilities that simultaneously satisfy all requirements on the process.

In practice, complex process optimization problems can often be solved by superimposing appropriate response surface contours. However, it is not unusual to encounter problems with more than two process variables and more complex response requirements to satisfy. In such problems, other optimization methods that are more effective than overlaying contour plots will be necessary.



### 3.3 Optimization strategy

The optimization strategy is based on direct parametrization of the FEM model, using Nastran solver. It is conducted on the RS metamodel (sec.3.2) based on cubic RBF (sec.4.1) that has proven to well perform in this field.

It is possible to divide the method in the following step:

1. Preliminary selection of the design variables; it is extremely important for properly customize the problem. The same effect can be obtained acting on several input parameters.
2. Planning of the DOE campaign; the aim is to find a set o design variables able to better explore the input parameters domain. The method implemented to pursue this objective is the Latin hypercube method because of quality achievable (see section 3.2).
3. Evaluating an approximate model of the system by means of RSM. Among several possibility available, the metamodel is obtained using the cubic function of the RBF.
4. Automatic multi objective optimization using approximate model that explore the response surface by means of screening methods. The responses can be represented as a scatter plot (two objective functions) or as a space cloud (three objective functions). The solution will be the optimal choice included in the frontier of the two type of diagrams, called the Pareto curve (2D) or Pareto surface (3D), depending on the necessary constraints.

5. Direct evaluations of the optimum, that has to be verified because this value is obtained from an interpolated surface.

# Chapter 4

## Tools

In this chapter we'll focus on the tools used in the present thesis to let the reader understand aspects otherwise unexplained.

### 4.1 Radial Basis Function

A **Radial Basis Function (RBF)** is a function that depends only on the distance, usually euclidean, and satisfies the property  $\varphi(\mathbf{x}) = \varphi(\|\mathbf{x}\|)$ . The sum of these function (eq. 4.1) is a very powerful interpolation tool for scattered data, capable to interpolate everywhere in the space a function defined at discrete points, giving the exact value at the original ones. Its behavior is deeply influenced by the type of basis adopted.

$$s(\mathbf{x}) = \sum_{i=1}^N \gamma_i \varphi(\|\mathbf{x} - \mathbf{x}_i\|) + h(\mathbf{x}) \quad (4.1)$$

The radial function can be fully or compactly supported and in any case a polynomial corrector is added to guarantee compatibility for rigid modes;

it is worth to point out that the degree of the polynomial has to be chosen depending on the kind of function adopted.

Typical radial functions are reported in table 4.1.

Table 4.1: RBF with global support

<b>Spline type</b>	$r^n, n \text{ odd}$
<b>Thin plate spline</b>	$r^n \log(r), n \text{ even}$
<b>Multiquadratic</b>	$\sqrt{1 + r^2}$
<b>Inverse quadratic</b>	$\frac{1}{1 + r^2}$
<b>Inverse multiquadratic</b>	$\frac{1}{\sqrt{1 + r^2}}$
<b>Gaussian</b>	$e^{-r^2}$

To obtain the right RBF equation, a linear system has to be solved, in order to evaluate the  $\gamma_i$  and the polynomial coefficients. The constraints that has to be imposed are:

$$s(\mathbf{x}_{k_i}) = g(\mathbf{x}_{k_i}), 1 \leq i \leq N \quad (4.2)$$

to impose that the desired function values  $g$  are obtained at source points.

$$0 = \sum_{i=1}^N \gamma_i q(\mathbf{x}_{k_i}) \quad (4.3)$$

too obtain that the polynomial terms give 0 contribution at source points.

The minimal degree of polynomial  $h$  depends on the choice of the basis function. A unique interpolant exists if the basis function is a conditionally positive definite function. If the basis functions are conditionally positive definite of order  $m \leq 2$ , a linear polynomial can be used:

$$h(\mathbf{x}) = \beta_1 + \beta_2 x + \beta_3 y + \beta_4 z$$

Note that in this case  $\mathbf{x} = \begin{pmatrix} x \\ y \\ z \end{pmatrix}$ .

Using a linear polynomial, the rigid body translations are exactly recovered.

Imposing the equation 4.2 and 4.3, it is possible to solve the following system:

$$\begin{pmatrix} \mathbf{M} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{pmatrix} \begin{pmatrix} \gamma \\ \beta \end{pmatrix} = \begin{pmatrix} \mathbf{g} \\ \mathbf{0} \end{pmatrix} \quad (4.4)$$

where  $\mathbf{M}$  is the interpolation matrix obtained by calculating all the radial

interactions between source points:

$$M_{ij} = \varphi (\| \mathbf{x}_{k_i} - \mathbf{x}_{k_j} \|) \quad i \geq 1 \quad j \leq N$$

$\mathbf{P}$  is a constrained matrix that arises balancing the polynomial contribution and contains a column of 1 and the  $x$   $y$   $z$  positions of source points in the other three columns:

$$\mathbf{P} = \begin{pmatrix} 1 & x_{k_1}^0 & y_{k_1}^0 & z_{k_1}^0 \\ 1 & x_{k_2}^0 & y_{k_2}^0 & z_{k_2}^0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x_{k_N}^0 & y_{k_N}^0 & z_{k_N}^0 \end{pmatrix}$$

Obviously the radial basis interpolation can be extended to vector fields interpolating three components:

$$\begin{cases} u = s_u(\mathbf{x}) \\ v = s_v(\mathbf{x}) \\ w = s_w(\mathbf{x}) \end{cases}$$

Radial basis method has several advantages that make it very attractive in the area of load mapping. It allows to transform the original quantity (i.e. pressure, temperature, load) from a mesh wise defined in a continuum one. Acting on the kernel parameters (and in the solver strategy) it can be tuned

so that it satisfy constraint as local and global equilibrium conditions. Extra points can be added to refine and improve the interpolation behavior. For instance, for element with an high aspect ratio, extra points with solution data coming from cell internal gradient are required in the long edge direction.

Furthermore RBF are proven to be an excellent tool for surface reconstruction and so they can be used to transform mesh information in an implicit surface so that source and target share a unique and meshless representation of the actual interface.

An example of the result provided by RBF interpolation is showed in picture 4.1. The first picture 4.1a is is composed by 9 points while the second one 4.1b with 1681.

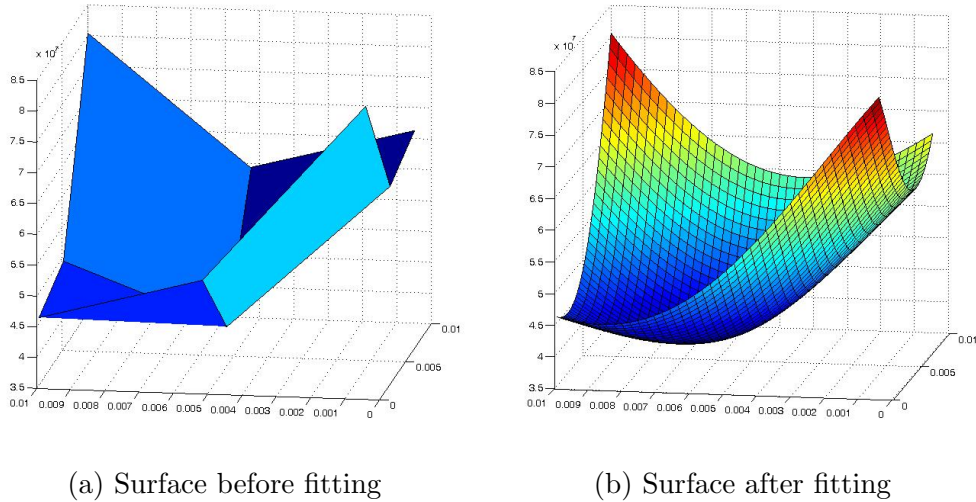


Figure 4.1: Comparison between a surface to be fitted (4.1a) and the result after RBF interpolation (4.1b)

## 4.2 Latin hypercube design

To better explore the space of variables, reducing the computational time of the study of a particular model, it's necessary the assistance of a statistical method involving the selection of the design variables from an ordered sampling frame. **Latin hypercube sampling (LHS)** is one of those method often used to construct computer experiments.

In the field of statistical sampling, the basis concept is represented by the Latin square: a square grid divided in equally probable intervals is a Latin square only if there is one sample in each row and each column. A Latin hypercube is th generalization of it to an N-dimensional hypercube. The condition will be that each sample must be the only one in each axis-aligned hyperplane.

LHS operates in the following manner to generate a sample size  $N$  from the  $n$  variables  $\xi_1, \xi_2, \dots, \xi_n$  [10]. The range of each variable is partitioned into  $N$  non overlapping intervals on the basis of equal probability size  $\frac{1}{N}$ . One value from each interval is selected at random with respect to the probability density in the interval. The  $N$  values thus obtained for  $\xi_1$  are paired in a random manner with the  $N$  values of  $\xi_2$ . These  $N$  pairs are combined in a random manner with the  $N$  values of  $\xi_3$  to form  $N$  triplets, and so on, until a set of  $N$  n-tuples is formed. This set is the Latin hypercube sample. An example is shown in tab.4.2.

The range of the three parameters is between  $[0, 1]$ , thus the equal probability spaced values are  $0, 0.1, \dots, 0.9, 1$ .



Table 4.2: LHS example for 3 parameters with 10 samples

1	0.58	0.97	0.64
2	0.37	0.61	0.17
3	0.85	0.36	0.74
4	0.28	0.18	0.23
5	0.72	0.21	0.08
6	0.18	0.01	0.99
7	0.48	0.46	0.87
8	0.68	0.59	0.37
9	0.98	0.87	0.56
10	0.06	0.76	0.45

## 4.3 Software

This section has been created to list the programs used in the present thesis. In the following they are briefly described.

### 4.3.1 Matlab

Matlab (**matrix laboratory**) is a programming language which allows matrix manipulation, implementation of algorithms, plotting of functions and creation of user interfaces both for numerical and symbolic computing. It can be interfaced with other programs written in other language like C, C++, Java, etc.

Matlab will be used to implement the entire workflow.

### 4.3.2 Nastran

Nastran is a finite element analysis (FEA) solver, developed for NASA in the 1960s. Among the different options available, which depend on the original

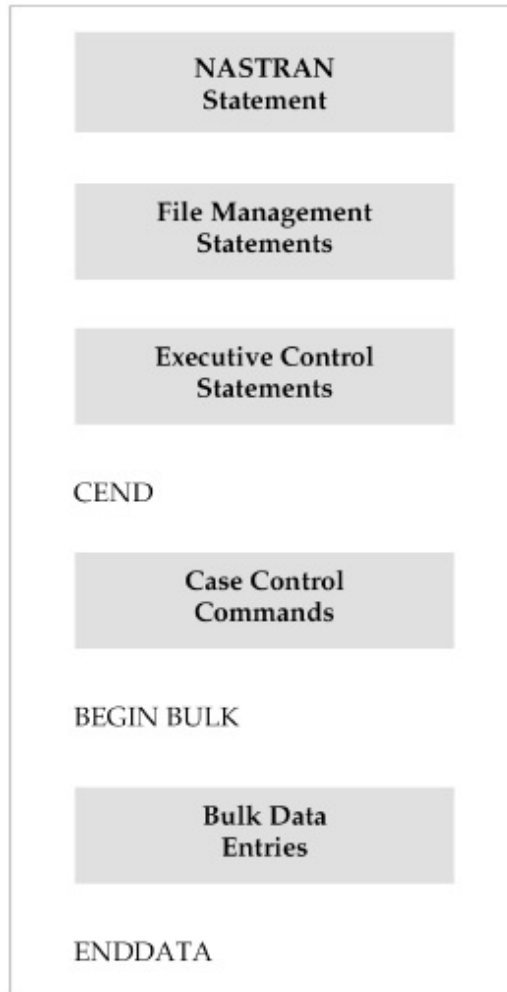


Figure 4.2: Bulk data structure

source code, the NX Nastran has been used. It hasn't a graphical interface for building a model or meshing and both inputs and outputs have to be provided by proper text file.

Due to the importance of the bulk data files for the model study, a picture of the structure is showed (fig.4.2).

Of a particular interest is the case control section and the bulk data entries.

The first one is used to:

- Define subcases analysis (multiple loadings in a single job execution).
- Specify and control the type of analysis outputs produced (forces, stress, etc..).
- Select output requests for printing, punching and plotting.
- Define titles, subtitles and labels for documenting the analysis.

The bulk data section contains all the information necessary to define the finite element model, such as coordinate systems, element properties, loads, boundary conditions and material properties.

For more details see [13].

Nastran will be used to run the study of the model, thus the output files from this solver to get the results will be exploited. Among them, only two type will be used: the punch file, that has the output results packed in row and column according to a special syntax (for example the values are written with 8 digits), and the f06 file, that contains both model information, like number of grid points or elements, and output information, but in a complex way. Obviously it has an ordinary and repeated syntax as well, but its parsing is more complicated. It is worth to point out that both of them are printed by Nastran only if requested by the user in the case control section.

### 4.3.3 Femap

Femap (**F**inite **E**lement **M**odelling **A**nd **P**ostprocessing) is an engineering analysis program, originally developed as a pre- and post-processor. It provides CAD import, modeling and meshing tools for FEM models, as well as the postprocessing functionality. It is essential to understand graphically the solution of the Nastran solver especially for complex structures and particular analysis. Femap will be employed to import CAD models, create the FEM ones and generate the bulk data files, which are written according a Nastran syntax. Thus, before starting the run of the Matlab script, the user must set up properly the model by using this graphic interface. It is really essential to create the mesh, impose constraints, loads, create the analysis, etc, because the model is displayed clearly and a direct manipulation can be performed on it, especially for complex structure. Obviously an alternative GUI can be used as long as it can generate Nastran bulk data.

# Chapter 5

## Workflow development

In the last decades, the development of more powerful and rapid calculators has led the mechanical engineering sector to a relentless dependence; in particular, the mechanical design field can't exclude the employment of this source. In a design process a fundamental step is the FEM analysis; it is essential because on one hand it avoids the physical construction of the model, and on the other it avoids the employment of mathematical models, which are often far from the reality.

Computers allowed to make repetitive calculations automatically which, otherwise, would require much more time. This aspect has led to the automatic workflow fulfillment.

The purpose of this program is to realize automatically a response surface of a structural model, in order to evaluate a solution to improve its performances or qualities. As affirmed at section 3.2, a screening method will be implemented, instead of a mathematical optimization. It employs graphical tool, such as two and three-dimensional diagrams.

In the present chapter the structure will be analyzed and all the files created will be deeply described.

## 5.1 Structure

The workflow has been created using the programming language Matlab for Nastran FEM models and it implements the optimization strategy described in section 3.3. Obviously, it realizes an optimization on customized models, which means that the user has to know how to set up the program, what raw materials provide, where the optimization is leading and how to read results.

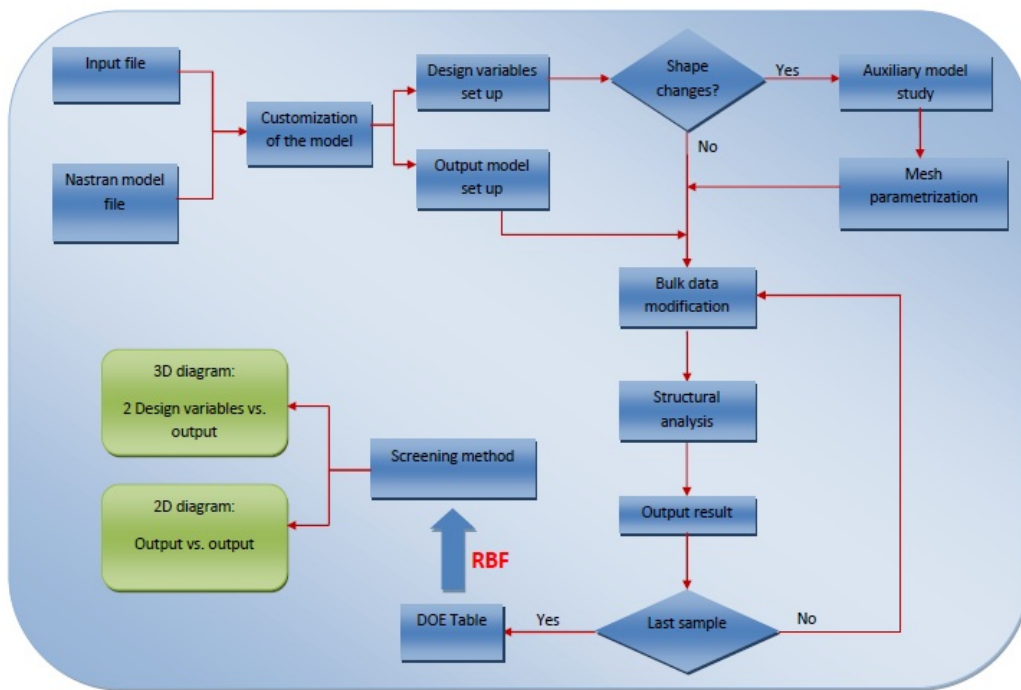


Figure 5.1: Workflow structure

First of all, as showed in figure 5.1, the input file and the Nastran model

file have to be provided to the Matlab script. These are the files that customize the analysis and both of them must be set up by the user before the run of the program, to properly analyze the model, without errors.

The first one is the *input.DOE.txt* file, which consists of all the essential data for the program; it is deeply described in section 5.2.1.

The second file is the bulk data. It includes the Nastran information of the structural model: geometry, mesh, forces, constrained, type of analysis and the desired outputs of the structural study. It must be called `<<"model name" _structural.dat>>`, to distinguish it from the auxiliary model (see sec.5.3) and to allow the program to read it. It must contain, in the executive control section (for more details see sec.4.3.2), the following entries:

- " DISPLACEMENT(PUNCH,PLOT) = ALL ", in order to obtain the displacements in the structure both as a punch file and a postprocess file. The first one is essential to evaluate the maximum stress in the Matlab script, the second one is necessary to display the results using FEMAP.
- " STRESS(PUNCH,PLOT) = ALL ", in order to obtain the same output files just described but with reference to the stresses.
- " ELSUM(EIDSUM) = ALL ", in order to print the value of the total mass into the \*.f06 file, which is another type of output Nastran file.

To create the bulk data file, the user has first to create a 3D model with a CAD tool, to import it in a FEM modeling software and to set up the structural study. Later, he has to extract the bulk data file from that FEM

model by means of Femap (sec.4.3), using the "export" command. It is important to underline that it could not be the only bulk data file requested by the Matlab script; in fact, if some shape changes have to be imposed over the structure, a second bulk data file, containing information of the auxiliary model, has be provided in addition (see section 5.3.2).

Once these files are created, the program can be run. First, it reads the information of the study from the *input\_DOE.txt* file, in order to impose the settings essential to customize the problem; indeed, that file represents the tool to communicate with the program because there is no direct interaction between the user and the Matlab script.

It is worth to point out that this customization can impose two type of design variables to parametrize the model, the thickness and the shape changes (see section 5.3), and it has been decided to implement three type of outputs obtainable from FEM study, listed in the following:

1. **Maximum stress.** The maximum value of the stress can be evaluated either for the entire structure or for each property of the model. It's important to underline that the stresses must be evaluated at the centroid of the elements; indeed, by default the results are given at the corners of them, where the values are interpolated and not directly evaluated, and it can lead to differences, between centroid and corners values, that can reach approximately 40%.
2. **Mass.** It is the entire mass of the structure.
3. **Maximum displacement.** Essentially the same consideration made for the stresses can be done for the maximum displacement.



Obviously, future improvements can concern the implementation of different kind of objective functions, such as maximum strain evaluation.

Once the customization of the study has been completed, it is necessary to decide how to explore the space of variables, i.e. choose how to distribute the samples. For this purpose, the Latin hypercube sampling method has been used (details at section 4.2).

Later, all the settings (design variables and outputs) have to be imposed and the model's bulk data file has to be modified as many times as the number of samples are. Thus a for-cycle has been created, containing the following actions:

1. Modification of the requested thicknesses.
2. Modification of the grid points positions, in order to impose the shape changes.
3. Study of the actual structural model.
4. Parsing of the Nastran files, in order to evaluate the requested outputs.

Once out from this cycle, the DOE table is obtained. An example is shown in fig.5.2. On the left side all the design variables are placed and on the right one all the outputs requested. Each row represents a study and a point of the response surface.

The results are then post processed. It is indispensable to obtain a packed surface for a proper analysis and this process is performed by means of RBF (see sec.4.1). This interpolating tool can enhance a lot the number of points of the response surface; it is worth to remember that the results are deeply

Design variable 1 [mm]	Design variable 2 [mm]	Maximum Stress [MPa]	Overall mass [Kg]	Maximum displacement [mm]
0.6077	0.2681	31	0.631	0.0192
0.6846	0.8278	30.8	0.640	0.0203
0.1189	0.7327	33.3	0.630	0.0186
0.6985	0.9575	30.6	0.642	0.0208
0.4566	0.3965	33.3	0.630	0.0192
0.0427	0.1148	34.5	0.620	0.0181
0.2531	0.8050	33.3	0.633	0.0193
0.4353	0.9063	31.5	0.638	0.0197
0.8476	0.4400	31.6	0.637	0.0199
0.9136	0.6662	30.1	0.641	0.0205

Figure 5.2: Example of DOE table with 10 samples

influenced by the number of samples which compose the DOE campaign, because the interpolation will be more accurate.

At that point, it is possible to perform the screening method, and to display the objective functions using the following type of diagram:

1. One-dimensional diagram, showing one objective function with respect to one design variable in order to understand the sensibility of the different variables, i.e. how much that variable influences that output.
2. Three-dimensional diagram, that shows the dependence of a selected output function from two selected design variables. It is thus graphically possible to evaluate the point, on that plot, that leads to an optimal solution for that output.
3. Two-dimensional contour plot, that is pretty the same as the previous

diagram, but with isoline instead of the surface.

4. Two or three-dimensional scattered diagrams, that relates two selected objective functions. With this tool it is possible to find a point that represents a good compromise between two objective functions, by extracting the Pareto frontier from the cloud of points.

Once the optimum point is obtained, it has to be verified. Indeed, this set of variables is evaluated by an interpolation. Hence, a new study has to be performed with this set, and the results obtained will be compared with that one previously interpolated. At this stage, this comparison has to be realized manually, depending on the output that has to be optimized and the constraints that have to be imposed.

This optimization strategy may have some obstacles while analyzing studies with more than three objective functions, because the scattered diagram, employed to evaluate the optimum value, cannot be displayed. At least, instead of having a scatter diagram with one optimum point, more plots can be superimposed for a multi-response contour plot (see fig.3.2), in order to obtain an optimum area.

Most of the engineering optimization analysis are characterized by tens or hundreds of objective functions and a possible solution could be implementing a mathematical optimization method, able to manage with much more design variables and objective functions.

## 5.2 Program components

In the present section all the component of the program will be described: the input text file and the Matlab scripts.

### 5.2.1 Input file

All the input data, essential for the right analysis of the problem and to customize the study, are gathered in the input file. It has a specific syntax that must be followed to allow the Matlab script to read it properly. The fundamental information are:

- **Name of the model.** It is necessary because each model has its own name and its files are recalled in the script.
- **Samples.** They are the number of studies requested by the user. It influences a lot the computational time of the script and the accuracy of the response surface. The entry must be a number.
- **Thickness switch.** It is the item designate to inform the program that thickness changes are active or not. The entry is a string: 'on' if active, otherwise 'off'. If the thickness changes are 'on', a set of values must be provided simultaneously: the boundaries of the thickness range, maximum and minimum value, and the IDs of the property to change(see section 5.3.2). It is important to underline that a single thickness change can be referred to more than one ID; hence at the entry *Thickness ID PSHELL* more than one value can be provided but they must be separated by a white space.

It is also possible to give a name to the thicknesses, in order to avoid confusion, putting it between brackets.

- **Shape switch.** It is the item designate to inform the program that shape changes are active or not. The entry is a string: 'on' if active, otherwise 'off'. If the shape changes are 'on', it must be provided the boundaries of each shape change, i.e. the maximum and minimum values of the parameter representing that shape change.
- **Stress switch.** It is the item necessary to inform the program if the maximum stress is requested as an output or not. The entry is a string: 'on' if it has to be evaluated otherwise 'off'. In addition it can be specified if its maximum value has to be evaluated generally or for each property. Its entry is a string as well and it could be 'general' or 'property'. Indeed, often the engineer is interested to the behavior of a specific part of the entire structure to check for example how much it is exploited.
- **Mass switch.** It is the item used to inform the program if the mass is a requested output. The entry is a string: 'on' if it has to be evaluated otherwise 'off'.
- **Displacement switch.** It is the item necessary to inform the program if the maximum displacement of the structure is requested as an output or not. The entry is a string: 'on' if it has to be evaluated otherwise 'off'.
- **Work storage.** It is the item used to decide if all process components

developed have to be kept, i.e. the Matlab workspace, the pictures, the \*.dat and the \*.op2 files. The first one is the bulk data file while the second contains the postprocessing information. They can be important especially when the studies have to be checked singularly using FEMAP because they can be reloaded and analyzed graphically, for example to see where the peak of the stress is located or how does the structure deform. Due to the high weight of the output files produced by Nastran, only the \*.dat and the \*.op2 files are stored.

- **Yielding stress** and **Safety factor**. They are respectively the limit stress condition and the parameter that describes the structural capacity of a system beyond the expected loads. These two parameters has to be provided in addition because the models are subjected to structural analysis.

### 5.2.2 Matlab scripts

The entire structure has been developed in the "Main" script with the addition of some functions. Indeed with Matlab is possible to create separate scripts with specific objective, that are recall every time they are necessary. In the following the functions are described.

- ***DOE***. It is the main script and the body of the program. It contains the preprocessing to impose the settings of the problem, the for cycle to run the studies and the postprocessing to realize and display the results.
- ***get\_disp***. It is the function created to evaluate the displacement at the nodes, essential to impose shape changes. This purpose is reached parsing the punch file from the model study, obtained by Nastran. It is appropriate for parsing files with thousands of numbers, because its syntax is specific and repeated. The input parameters are the name of the punch file, the number of grid points and the number of shape changes.

It is worth to point out that this function is not aimed to obtain the maximum displacement, but the displacement of all nodes. It is employed to read the deformation of the auxiliary file solution and then to impose the shape change. More details in section 5.3.2.

- ***get\_Elem\_Matrix***. It is a function aimed at storing all the needful information about each element: the type of element, its identification number (ID) and the ID of the related property.

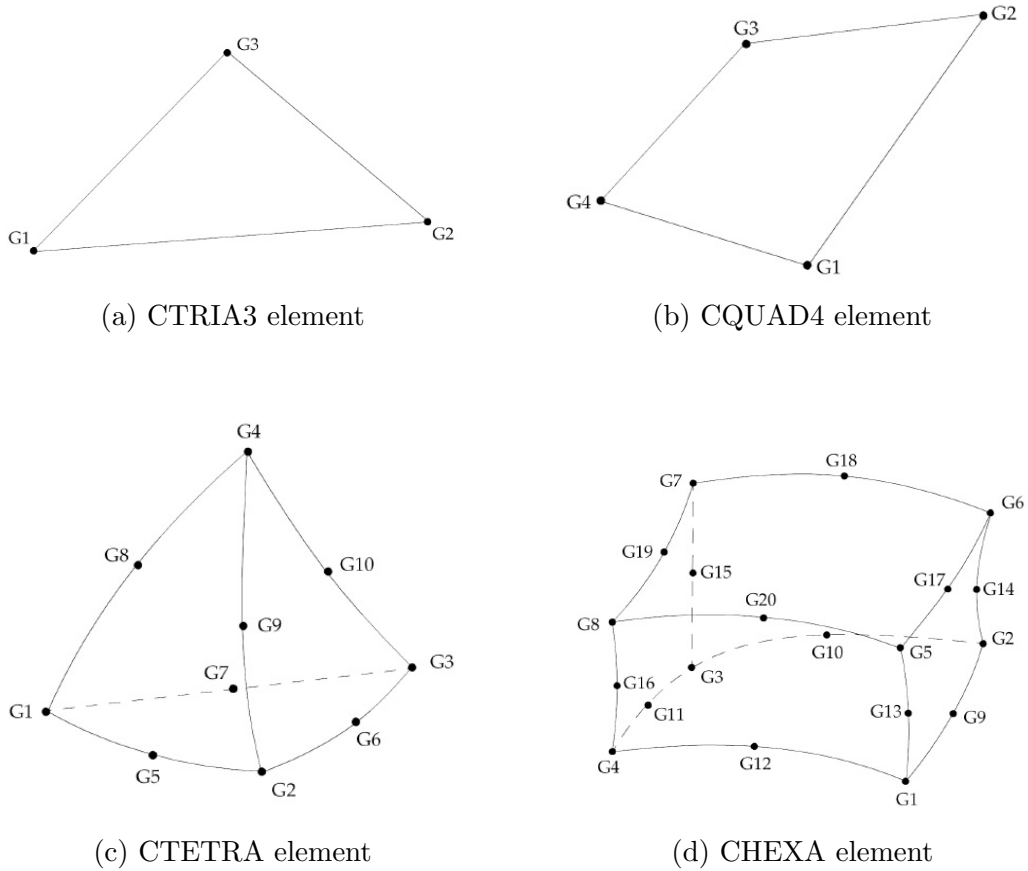


Figure 5.3: Database elements

It is worth to point out that not all the element types can be stored, due to the fact that each of them has a particular syntax. At the moment, the database contains:

1. CTRIA3, an isoparametric membrane-bending or plane strain triangular plate element (fig.5.3a).
2. CQUAD4, an isoparametric membrane-bending or plane strain quadrilateral plate element (fig.5.3b).



3. CTETRA, a four-sided solid element with four to ten grid points (fig.5.3c).
4. CHEXA, a six-sided solid element with eight to twenty grid points (fig.5.3d).

For more information see [12].

The input variables that must be provided are the name of the bulk data file and the number of element.

- ***get\_mass***. It is the function to obtain the value of the model's overall mass. This output is printed at the entry "TOTAL MASS" in the f06 file only if "ELSUM(EIDSUM) = ALL" is written in the case control section of the bulk data file. It is necessary to provide the name of the f06 file that has to be parsed.
- ***get\_max\_disp***. It is the function to evaluate the maximum displacement of the structure. All the displacements are stored in the punch file; this function reads the displacement of each nodes, one by one, comparing the first two values; it discards the lower one and then it proceeds comparing the remaining one with a third value, choosing the higher. It goes on in the same way until all the displacements are compared.  
The input variables are the punch file and the number of nodes of the model.
- ***get\_max\_sigma***. It is the equivalent function to *get\_max\_disp* but for the stress evaluation. Differently from the previous function, the ele-

ment matrix of the problem, obtained from the *get\_Elem\_Matrix* function, has to be provided.

- ***initial\_data***. It is the function that reads the input information of the study by parsing the input text file (section 5.2.1).

It doesn't need any inputs, but in the same folder the input file has to be provided.

- ***lhsdesign***. It is the Matlab function containing the latin hypercube sample (see 4.2) of  $n$  values on each of the design variables.

- ***metadata***. It is the function able to evaluate the entire RS. Through the values of  $\beta$  and  $\gamma$ , it is possible to choose the number of points to interpolate the RS. It is worth to point out that evaluate the entire RS can be pretty heavy, computational talking. Indeed, if we want to interpolate  $N$  points over each range of the design variable (DS), the total number of points of the fitted RS will be  $N^{DS}$ . It means that if we want 50 samples for each parameter, with just 5 parameters, the RS will have 312500000 points! Hence, can be better display part of the RS through *Surf\_2in*.

- ***norm\_doe***. It is the function which takes a vector as input and gives it back normalized.

- ***num\_elem***. It is the function that reads the f06 file of the model and returns the number of each type of elements and its total sum.

- ***num\_grid\_point***. It is the function that read the number of nodes. Its input is only the f06 file.

- ***num\_shapechange***. It is the file aimed to evaluate the number of shape changes from the bulk data file of the auxiliary model. Indeed, if at least one shape change is imposed to the model, an auxiliary model has to be provided, with one subcase for each shape change. The script reads the number of this type of modification by reading the number of subcases.
- ***overwrite\_position***. It is the script essential to impose the modification of the shape due to the shape changes: it overwrites the position of all grid points in the file that has to be studied.  
It has as input the file to overwrite, the matrix containing the new positions of the nodes and the number of them.
- ***pos\_nodes***. It is the function which evaluates the original position of the nodes. It reads the bulk data file and returns a matrix containing the positions of nodes.
- ***RBF\_fit***. It is the function necessary to evaluate the coefficients  $\gamma$  and  $\beta$  for the RBF (see equation 4.4). It needs the DOE table and the number of outputs.
- ***Surf\_2in***. It is the function to display the two-dimensional diagram described in section 5.1.

The inputs are: the  $\gamma$  and  $\beta$  coefficients, obtained from the *RBF\_fit* function, the DOE table, the objective function among the outputs, the design variables that have to be fixed, if present, and those with respect to which the diagram depends from.

- *thickness\_change*. It is the function aimed to impose the modification of the thickness. To work properly, it needs the bulk data file of the model, to modify, the DOE table, where the thicknesses are stored and the ID of the property to be change.

## 5.3 Model parametrization

For a proper analysis of the problem, each study has to be customized depending on the type of design variables. In the following, these types will be widely discussed, highlighting how they work and which elements they need.

### 5.3.1 Thickness change

In Nastran the elements information are gathered in a specific entry called property. Depending on the type of element, its name will change: for thin shell elements there is the PSHELL property, while for solid elements there is PSOLID one.

Since the former is related to shell elements, it contains the value of their thickness. In table 5.1 it is shown the format; it has a fixed and repetitive syntax.

Table 5.1: PSHELL format

1	2	3	4	5	6	7	8	9
<i>PSHELL</i>	<i>PID</i>	<i>MID</i> <sub>1</sub>	<i>T</i>	<i>MID</i> <sub>2</sub>	<i>12I/T</i> <sup>3</sup>	<i>MID</i> <sub>3</sub>	<i>TS/T</i>	<i>NSM</i>
	<i>Z</i> <sub>1</sub>	<i>Z</i> <sub>2</sub>	<i>MID</i> <sub>4</sub>					

Where:

*PID* Property identification number

*MID*<sub>1</sub> Material identification number for the membrane

*T* Membrane thickness

For the other entries see [12].

Hence, to impose the thickness change it is necessary to overwrite the value in the related field. Furthermore, the user must provide the ID of the property as well, to decide to which group of elements need to be changed their thickness. An example is showed in figure 5.4; in this case the property number three is the objective of the modification.

```

$ Femap with NX Nastran Property 1 : long. anteriore
PSHELL      1      6 5.75-4      6      6      0.
$ Femap with NX Nastran Property 2 : long. posteriore
PSHELL      2      6 9.75-4      6      6      0.
$ Femap with NX Nastran Property 3 : centina 1
PSHELL      3      6 8.75-4      6      6      0.
$ Femap with NX Nastran Property 4 : centina 2
PSHELL      4      6 7.75-4      6      6      0.
$ Femap with NX Nastran Property 5 : centina 3
PSHELL      5      6 .001425      6      6      0.
$ Femap with NX Nastran Property 6 : centina 4
PSHELL      6      6 .001425      6      6      0.
    
```

Figure 5.4: Bulk data structure

### 5.3.2 Shape change

Much more tricky is the modification of the shape of the model. That is due to the necessity to modify the position of all nodes involved in the change. It is worth to point out that with this parametrization of the model, its topology won't be influenced, but only the shape dimensions, and for this reason it is called mesh morphing. In the following, the customization of the model will be described to impose the shape changes.

First of all the user has to know which kind of modifications are suitable for the model and he has to provide them through an **auxiliary model**. It is a bulk data file that will be studied to obtain the displacement of the nodes of the mesh and thus has to be geometrically identical to real model. However

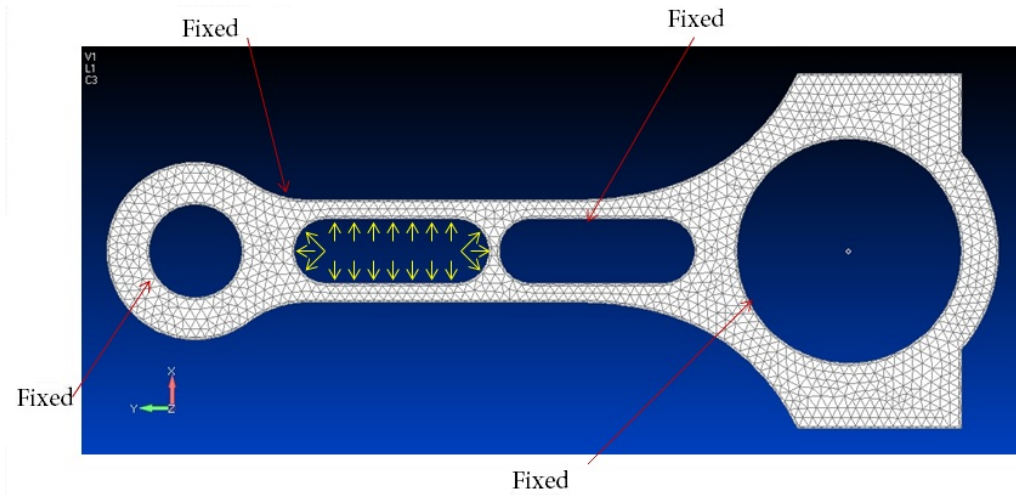
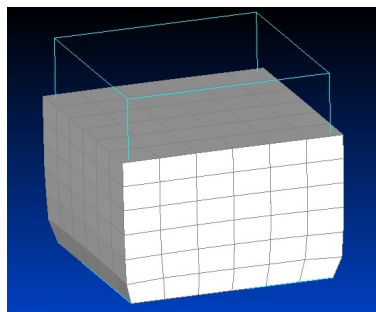
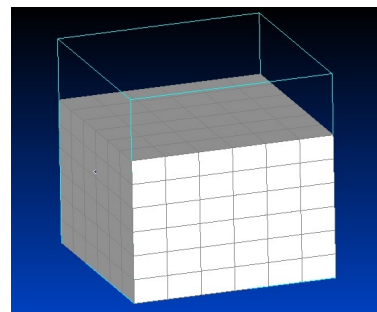


Figure 5.5: Example of an auxiliary model for the shape change

it has completely different loads and constraints; the loads that have to be imposed will be structural yieldings in the region where the model has to change shape. As regard the constraints, all the boundaries, which have not to be influenced by the modification, must be fixed.



(a) Carbon steel material



(b) Material with  $\nu = 0.01$

Figure 5.6: Cube reduction of height

For example in figure 5.5, the central left hole has to be enlarged, and all the other boundaries are thus fixed.

It is worth to point out that the single purpose of this study is to obtain the desired change of the shape; to this end, every artifice can be imposed on the auxiliary model: for example, if we want to reduce the height of the cube in figure 5.6, it is better to impose a fake material than a real one. A carbon steel cube, with the Poisson's ratio  $\nu = 0.3$ , deforms as 5.6a shows, because  $\nu$  determines the phenomenon of transverse expansion or contraction when the material is respectively compressed or stretched in a specific direction. For this reason, if this value is ideally null, this effect will disappear, as shown in figure 5.6b. Obviously a material with  $\nu = 0.01$  doesn't exist but it isn't important because it leads to the right modification of the structure and it doesn't influence the following real structural study at all.

This file is essential if the shape changes have to be imposed over the structure and it is fundamental remember to write in the executive control section " DISPLACEMENT(PUNCH) = ALL ", in order to obtain the displacements of this auxiliary study in a punch file, that will be parsed by the program. Furthermore for each shape change, a relative set of loads, constraints and results are requested; this means that it is necessary to create a subcase for each of these modifications.

Once the auxiliary model is ready, it has to be called <<"model name" *\_auxiliary.dat*>>, to allow the program to read it. This file is then run with Nastran; the modified positions of grid points are obtained through the *get\_disp* function and stored in a matrix called "Disp". To impose the morphing of the shape, these displacements are summed up with the original positions and multiplied by a coefficient, called shape change parameter. It represents the amplitude of the modification, thus it allows



to control the mesh morphing, and it is a design variable. There will be a parameter for each shape change, and the related ranges. Both of them have to be provided by the user depending on the geometric conditions.

All the processes just described occur automatically once the problem has been set up.

An example of the auxiliary model result, and than of the shape changes, is shown in figure 5.7. Here the value of the reduction of the hole is 1 *mm*, but with the shape parameter it is possible to enhance this deformation. It is clear that the maximum value that can be imposed are influenced by the geometry; this is the reason why the range of the shape parameters must be given by the user.

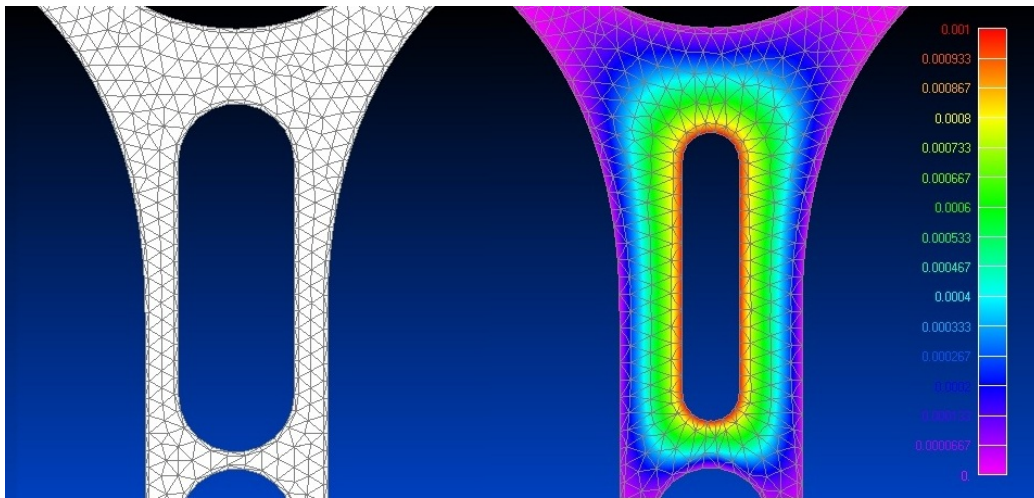


Figure 5.7: Example of shape change results

# Chapter 6

## Results

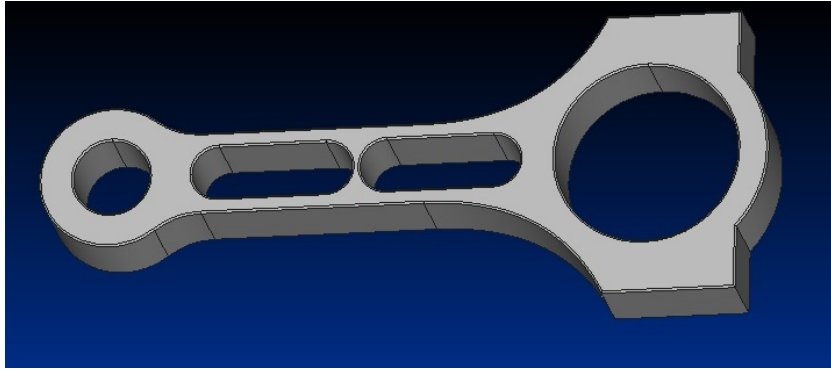
In the present chapter all the results obtained will be presented.

### 6.1 Connecting rod case

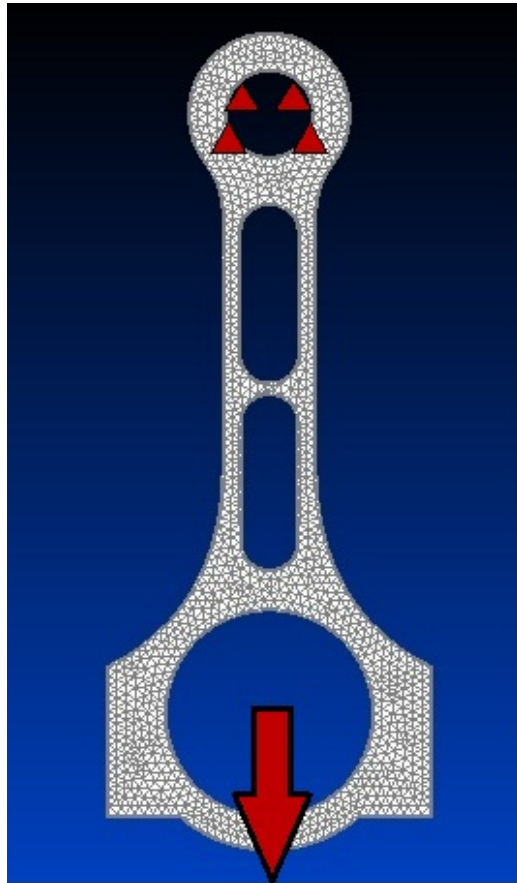
The connecting rod has been studied to try the proper operation of the shape changes, since no reduction of thickness can be imposed to the structure because it is made of solid elements.

Table 6.1: Connecting rod data

<b>Length</b>	<i>14 cm</i>
<b>Width</b>	<i>7 cm</i>
<b>Material</b>	<i>Carbonsteel</i>
<b>Young modulus (<math>E</math>)</b>	<i>206 GPa</i>
<b>Density (<math>\rho</math>)</b>	<i>7800 Kg/m<sup>3</sup></i>
<b>Yielding stress (<math>R_s</math>)</b>	<i>550 MPa</i>
<b>Bearing load</b>	<i>2.4 kN</i>
<b>Grid points</b>	<i>97361</i>
<b>Elements</b>	<i>62701 (CTETRA)</i>
<b>Samples</b>	<i>16</i>



(a) Model



(b) Boundary conditions

Figure 6.1: Connecting rod

The data of the model are summarized at table 6.1. The object of the analysis is to optimize the structure by reducing the two lightening holes in the middle (fig.6.1a). Thus an auxiliary model has to be created: two structural yielding, one for each shape change, are imposed orthogonally to the relative internal surfaces with a maximum value of 1 *mm*; the first hole enlargement occurs on the top central hole while the second on the bottom central one. Two different subcases are created in order to obtain a single punch file with both the studies results inside, instead of one punch file for each modification.

As regards the settings of the real physical study, it has been made by fixing the smaller circular hole and imposing on the bigger one a bearing load (see fig.6.1b).

The optimization has been performed using a DOE table of 16 samples with two design variables, two parameters of the shape changes, and two objective functions, the maximum stress and the mass of the structure.

In the following the optimization results will be displayed. Picture 6.2 shows the maximum stress in the structure while fig.6.3 the mass, both of them with respect to the designs variables. As expected, with an enlargement of both holes the stress decreases from 34 *MPa* to 28.5 *MPa* and the mass increases linearly. Because the admissible stress of the material, with a safety factor of 2, is  $\sigma_{amm} = R_s/2 = 225$  *MPa*, much higher than the maximum stress developed in the structure, the optimal design point is the one which minimizes the mass.

It is worth to point out that, more than an optimization, this analysis has been performed to understand if the program worked well using the shape

changes. A better employed of this workflow has been made on the WT model.

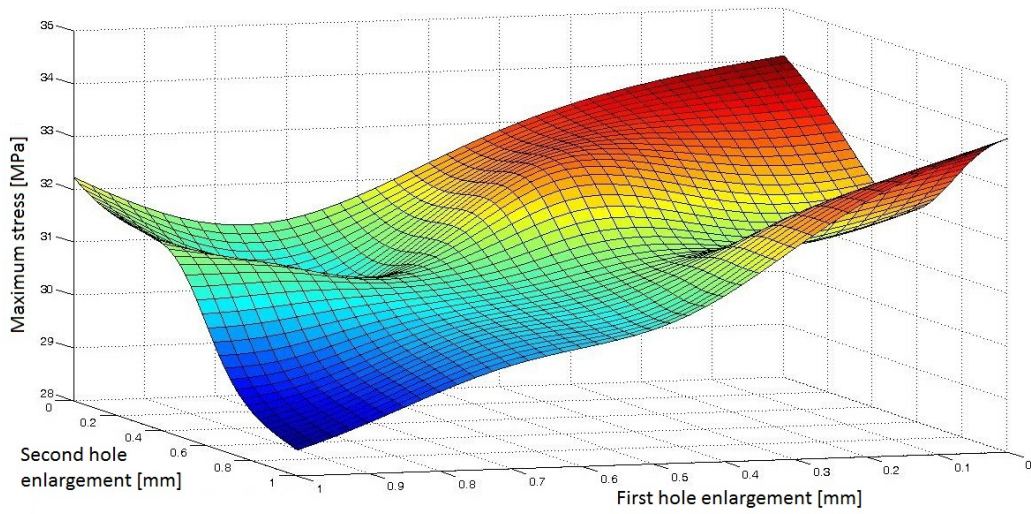


Figure 6.2: Maximum stress diagram

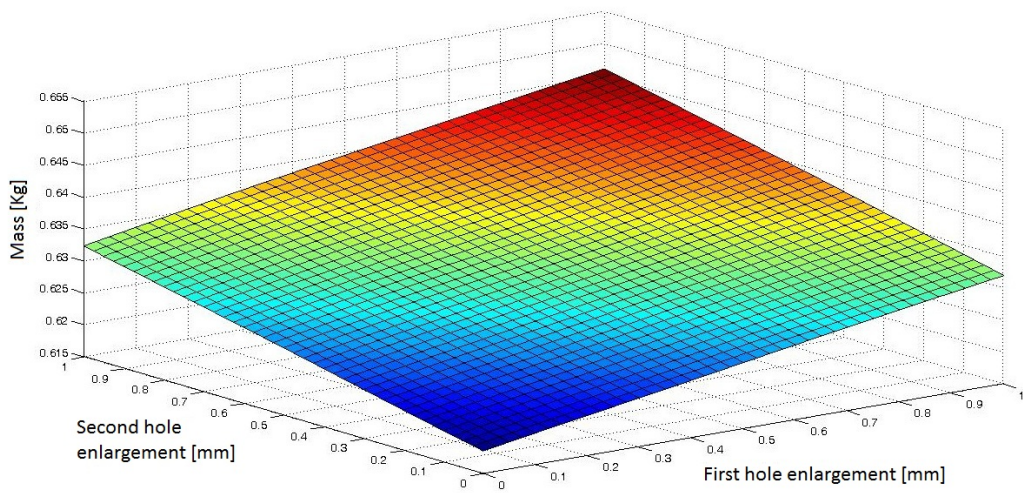


Figure 6.3: Mass diagram

## 6.2 WT model case

### 6.2.1 Description and previous results

An application of this workflow optimization process is the WT model, described in chapter 2.2. The results of this finite element model will be compared with the real model in the wind tunnel at the university "Federico II" of Naples.

The layout of the wing is pretty simple and economic (fig.2.2), because its purpose is just to validate the tools used to transfer loads from the CFD model to the FEM one. It is composed by two spars, the structural components that should sustain the lift of the entire wing, i.e. the bending load, eleven ribs, necessary to avoid unwanted deformations of the skin under the lift force, and the skin, the part of the wing that determines the fluid dynamic of the structure and counteract torsion stresses.

The chosen material is the easy available aluminum 6061, because it is economic, suitable for hot forging and highly weldable; it is commonly used for construction of homebuilt aircraft structures such as wings and fuselages.

A previous structural analysis has been realized by Marco Fontanella in his thesis [5]. The model has the data collected in table 6.2 and 6.3.

Table 6.2: WT CFD model data

<b>Wind speed</b>	45 <i>m/s</i>
<b>Wing twist</b>	6°
$C_L$	1.1
<b>Lift</b>	105 <i>Kg</i>

It is worth to point out that they are obtained in order to avoid the

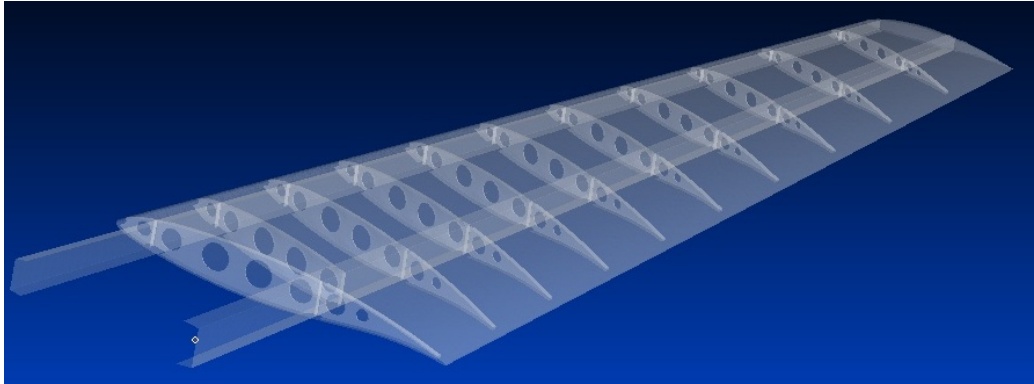


Figure 6.4: Wing model

Table 6.3: WT mesh model data

<b>Length</b>	1600 <i>mm</i>
<b>Material</b>	<i>Aluminium 6061</i>
<b>Young modulus (<math>E</math>)</b>	70 <i>GPa</i>
<b>Density (<math>\rho</math>)</b>	2700 <i>Kg/m<sup>3</sup></i>
<b>Yielding stress (<math>R_s</math>)</b>	85 <i>MPa</i>
<b>Rear spar thickness</b>	1 <i>mm</i>
<b>Front spar thickness</b>	0.5 <i>mm</i>
<b>Skin thickness</b>	0.5 <i>mm</i>
<b>Rib thicknesses</b>	[1.5 ; 0.5] <i>mm</i>
<b>Elements</b>	126040 ( <i>CTRIA3</i> )

buckling of the skin and optimizing. Indeed its thickness is 0.5 *mm* and the lift produces a compressed area on the top of the wing. Some structural results are shown below (fig.6.5, 6.6 and 6.7).

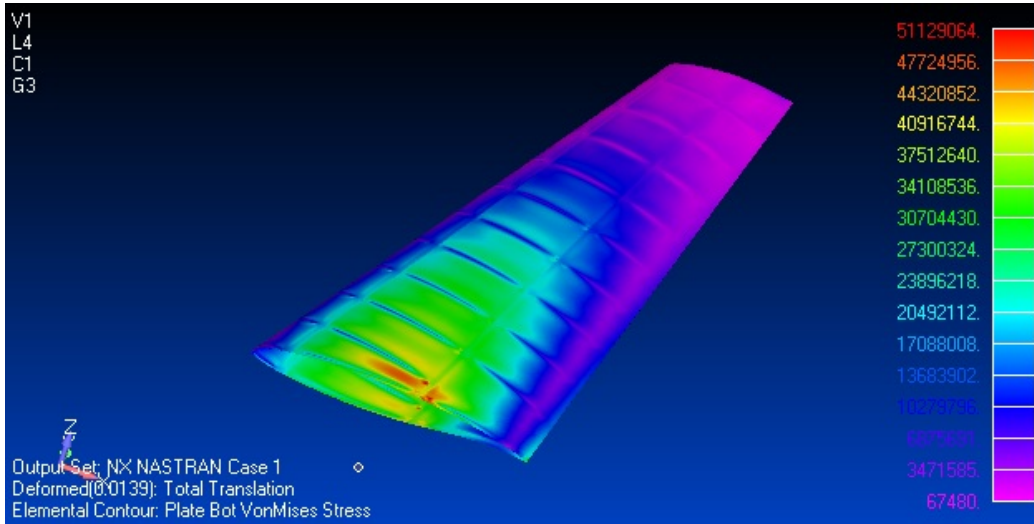


Figure 6.5: Skin stresses; maximum value: 51 MPa

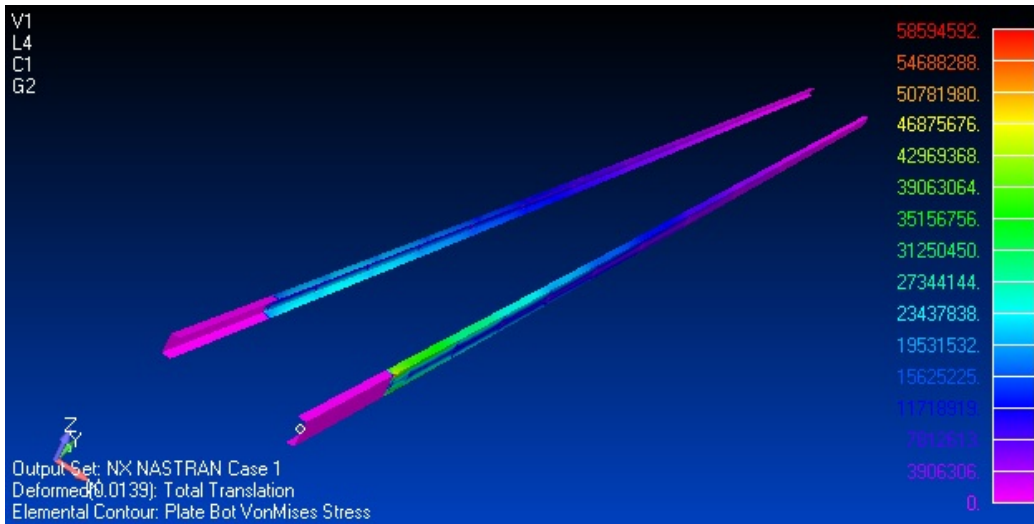


Figure 6.6: Spars stresses; maximum value: 58.6 MPa



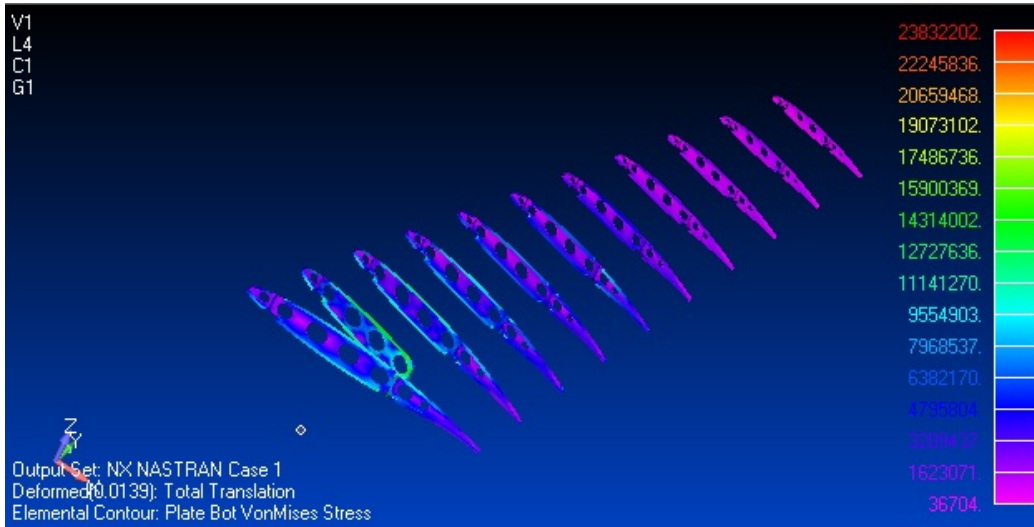


Figure 6.7: Ribs stresses; maximum value: 23.8 MPa

As expected, the ribs are not well-stressed, because their role is to avoid unwanted deformations of the skin, such as buckling. The maximum values is reached in the spars, near the fixed edge of the skin, and it is higher than the  $\sigma_{amm}$ , which, using a safety factor of 2, is 42.5 MPa.

For this reason, an optimization process will be carried out, in order to obtain the best compromise between maximum stress and mass, modifying the width of the rear spar slab and the thicknesses of skin, spars and ribs.

## 6.2.2 WT model optimization

Before starting the optimization, it is necessary to customize the problem. In the *Input\_DOE* text file, the information gathered in table 6.4 are provided, according to section 5.2.1.

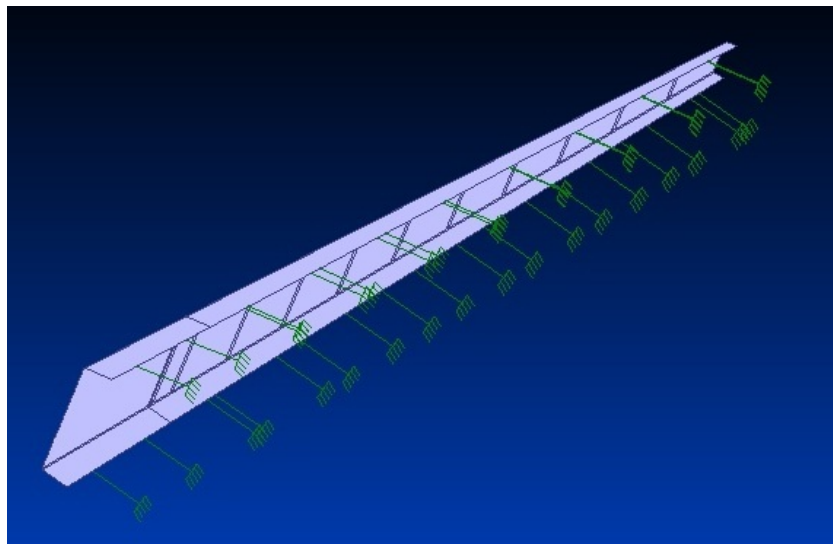
This study has the purpose to reduce, at the same time, the mass of the structure and the maximum stress at the fixed section of the rear spar, where most of the forces are unloaded. Then, an increase of the width of its slabs can lead to a decrease of internal tensions. At the same time, the thicknesses of skin, spars and ribs are increased in order to strengthen the structure. It is important to highlight that, structurally talking, the two spars act differently, because usually the rear spar is loaded more than the front one; hence the thicknesses of the spars are analyzed separately, each one represented with a design variable.

The properties of the shell elements, where the thickness values are placed, are fourteen, and they will be modified according to the part of the wing to which they are related to (see tab.6.4).

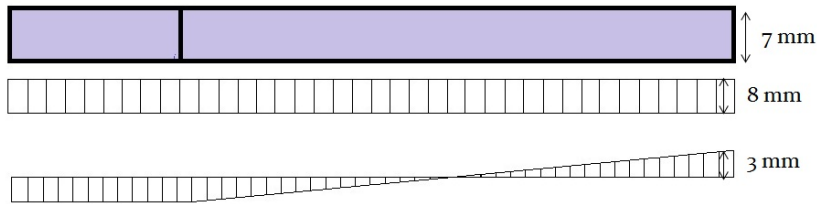
Table 6.4: WT optimization study data

<b>Samples</b>	120
<b>Thickness 1 (skin)</b>	$[0.5 ; 1] \text{ mm}$
<b>ID skin property</b>	14
<b>Thickness 2 (rear spar)</b>	$[0.5 ; 1.5] \text{ mm}$
<b>ID skin property</b>	2
<b>Thickness 3 (front spar)</b>	$[0.5 ; 1.5] \text{ mm}$
<b>ID skin property</b>	1
<b>Thickness 4 (ribs)</b>	$[0.5 ; 1.5] \text{ mm}$
<b>ID skin property</b>	3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13
<b>Linear shape morphing</b>	$[0 ; 3] \text{ mm}$
<b>Constant shape morphing</b>	$[0 ; 8] \text{ mm}$

The auxiliary model has to be created due to the presence of two shape changes. These modifications are aimed to enlarge the rear spar slabs, both the upper and the lower one, in order to reduce the stress on it. Two yieldings are imposed on the slabs using Femap, as shown in figure 6.8a: a linear deformation, which tapers the spar, and a constant one, which enlarges the slabs themselves (fig.6.8b).



(a) Yielding imposed in Femap



(b) Boundary conditions

Figure 6.8: Shape changes customization

It is worth to point out that the linear modification has the objective to reduce the width of the spar at the tip as well; the reason is linked to the

fact that tapering the rear spar, it is probably possible to reach a condition with an homogeneous distribution of the stresses, making a better employ of the entire part. Moreover, it has been set its maximum at  $3\text{ mm}$  in order to avoid the contact between the ribs and the slab of the rear spar.

The mesh morphing are displayed in figure 6.9; actually, it shows only the tip of the spar, because at the fixed section the result is an enlargement of the slabs in both cases, like figure 6.9b shows.

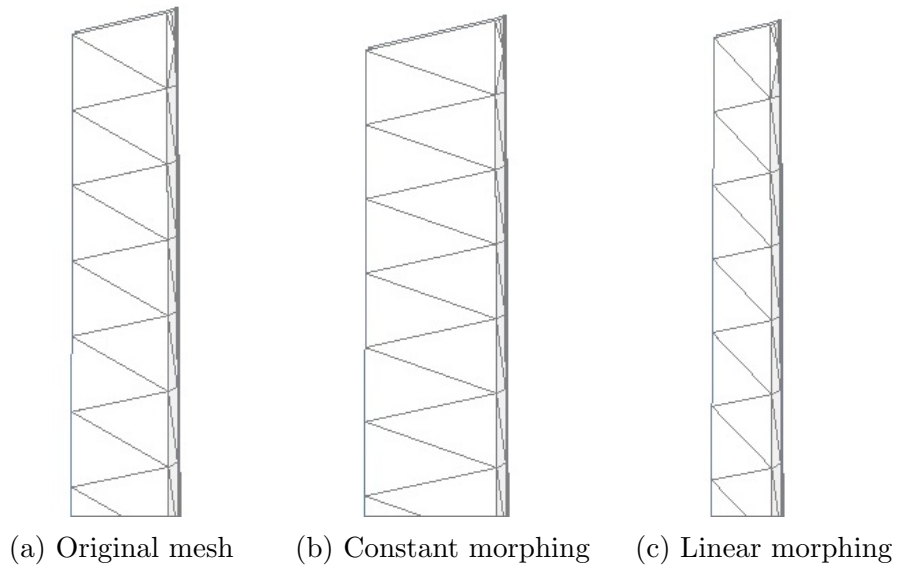


Figure 6.9: Mesh morphing of the spar tip due to shape changes

Recapping the settings, this optimization is composed by six design variables and two objective functions, maximum stress and mass.

Of particular interest is the one-dimensional diagram that shows the dependence of the maximum stress from a design variable, evidencing the sensitivity of that objective function. Creating as many diagrams as the design

variables are, it is possible to understand the most critical ones (fig.6.10).

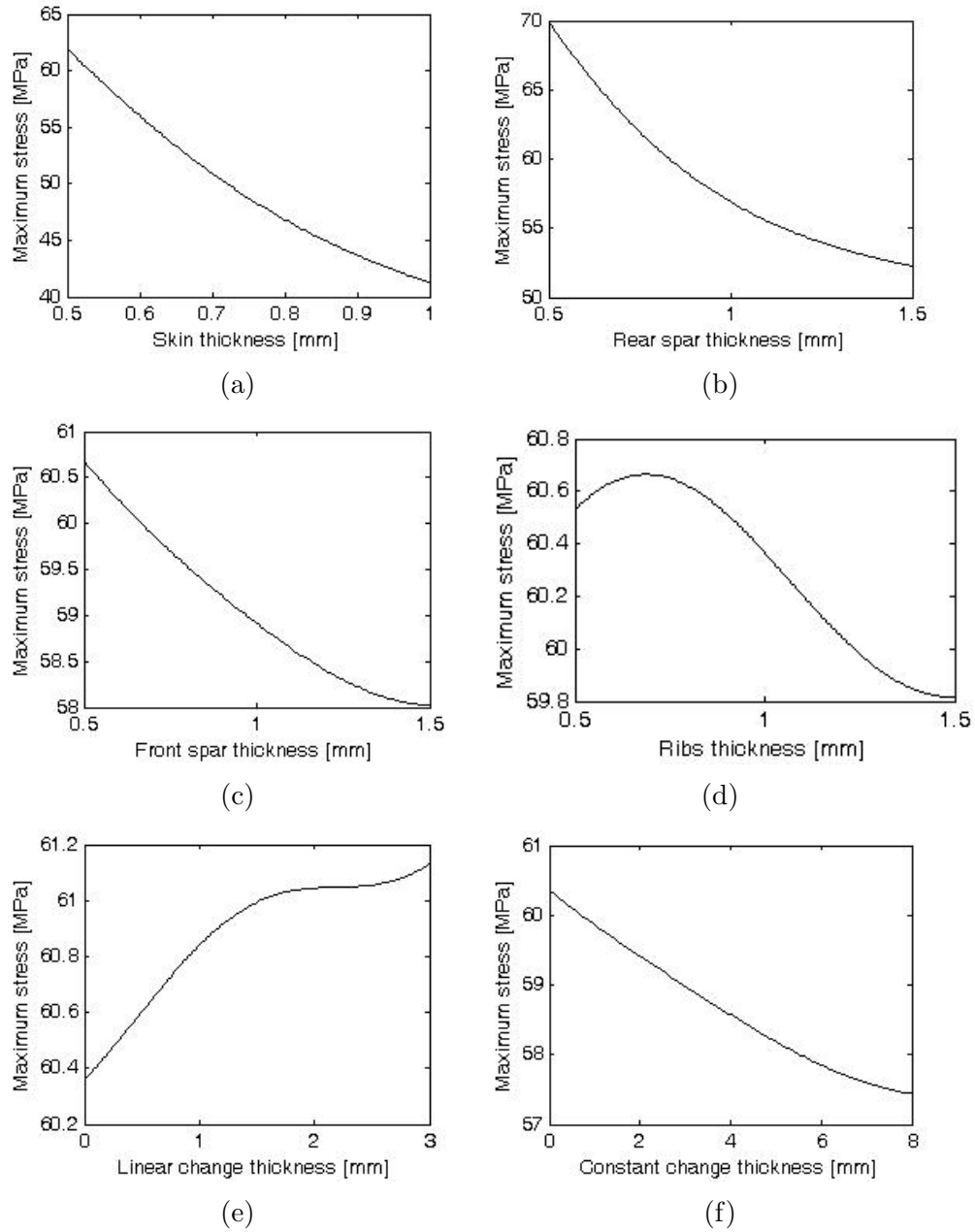


Figure 6.10: 1D plot representing the maximum stress with respect to all the design variables

In our case, as expected, the skin and the rear spar thicknesses are the most determinant. Their influence leads to a variation of the stress of about  $20\text{ MPa}$  (fig.6.10a) with regard to the skin thickness and around  $15\text{ MPa}$  for the rear spar's one (fig.6.10b). The maximum value of stress is almost independent from ribs thickness and the linear shape change; in the first case, because the ribs aren't structural elements but they have to maintain the shape of the skin, while in the second one, the linear change modification is really small (maximum  $3\text{ mm}$ ), due to geometric conditions.

To evaluate the best set of design variables, the scatter diagram, that relates the maximum stress and the total mass, has been displayed (see figure 6.11a); the frontier has been extracted (fig.6.11b) and, imposing the constraint of the maximum available stress, the best solution has been obtained. Moreover, in green the DOE table samples have been showed.

In table 6.5 this set of parameters are collected:

Table 6.5: Set of the optimum design variables

<b>Thickness 1 (skin)</b>	$0.72\text{ mm}$
<b>Thickness 2 (rear spar)</b>	$1.28\text{ mm}$
<b>Thickness 3 (front spar)</b>	$0.5\text{ mm}$
<b>Thickness 4 (ribs)</b>	$0.5\text{ mm}$
<b>Linear shape morphing (linear)</b>	$3\text{ mm}$
<b>Constant shape morphing (constant)</b>	$7.11\text{ mm}$

Later, these values have been imposed on the model and then it has been studied, in order to verify the accuracy of the results obtained. In table 6.6 the data are compared.

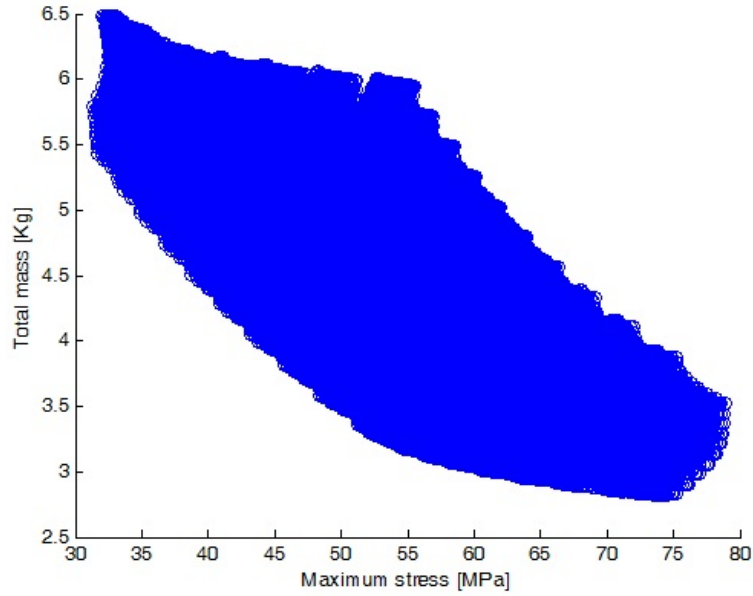
These results show a difference of  $2.8\%$  for the maximum stress and with regard to the mass their values are almost equal. The displacement result has

Table 6.6: Comparison between automatic workflow, direct study and previous study results

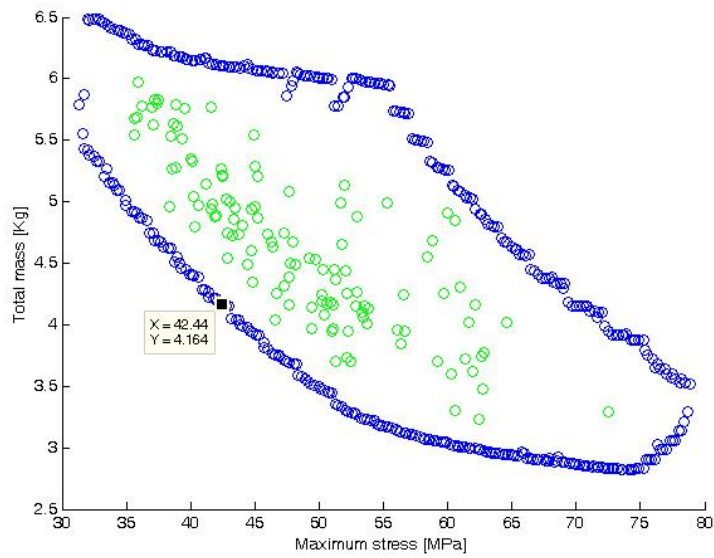
	Interpolation results	Verification results	Previous results
<b>Maximum stress</b>	<i>42.37 MPa</i>	<i>43.58 MPa</i>	<i>58.6 MPa</i>
<b>Total mass</b>	<i>4.164 Kg</i>	<i>4.167 Kg</i>	<i>3.464 Kg</i>
<b>Maximum displacement</b>	<i>9.96 mm</i>	<i>9.83 mm</i>	<i>15 mm</i>

been shown as well to enforce the verification of the interpolation performed. The optimum solution can then be accepted.

This process has let to a reduction of the maximum stress of 27% and an increase of the mass of 20.6%. In the following some figures of the FEM results will be displayed (fig.6.12, 6.13 and 6.14). Comparing the figures of the previous study, the essential reductions are obtained in the spars, from *58.6 MPa* to *43.58 MPa*, and in the skin, from *51 MPa* to *37.4 MPa*.



(a) Entire diagram



(b) Pareto frontier (blue) and the DOE table points

Figure 6.11: 2D plot representing the relation between maximum stress and the total mass



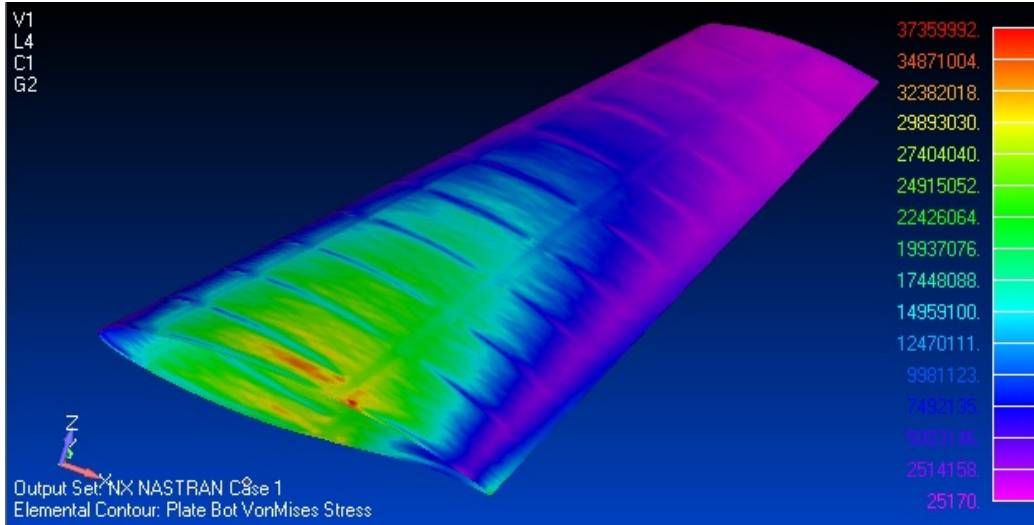


Figure 6.12: Skin stresses; maximum value: 37.4 MPa

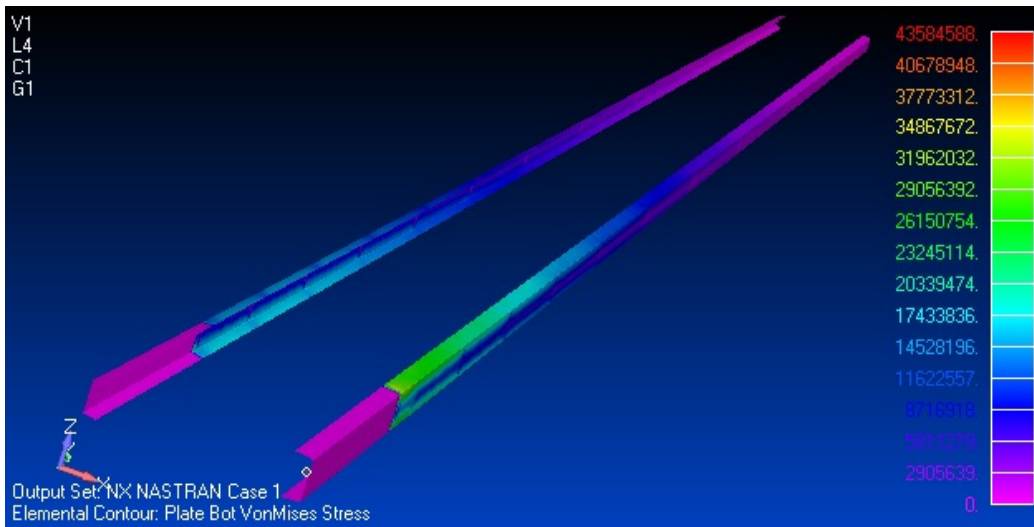


Figure 6.13: Spars stresses; maximum value: 43.58 MPa

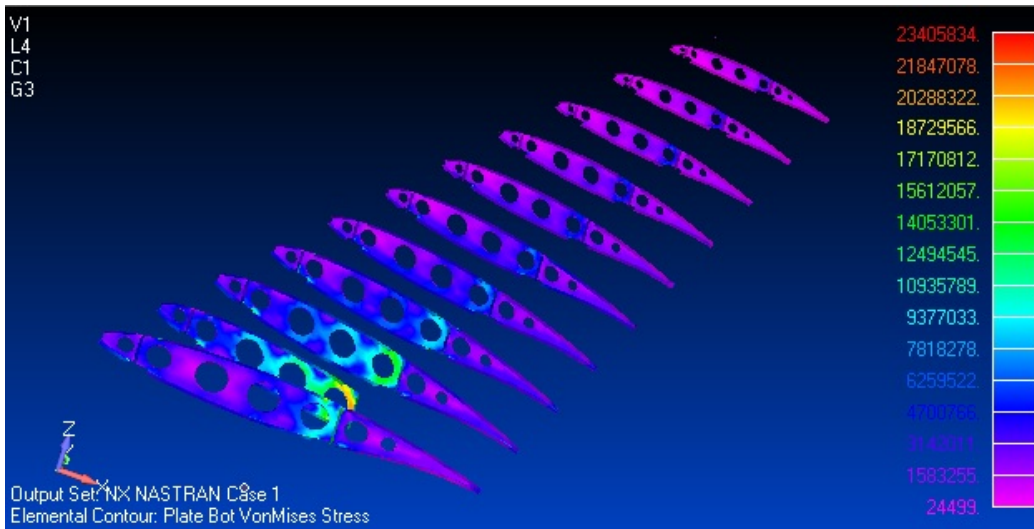


Figure 6.14: Ribs stresses; maximum value:  $23.4 \text{ MPa}$

# Chapter 7

## Conclusion

This thesis deals with the realization of an optimization workflow. The model has to be provided by the user using Femap or whatever else graphical interface that prints the bulk data files in Nastran. The program, then, studies the model using Nastran and create the DOE table with all the design variables and the requested objective functions, according to the customization of the model furnished.

A metamodel is then created by means of the RBF functions and it is analyzed graphically in order to obtain the best set of variables for the best performances. The graphical tools available to the user are a three-dimensional diagram, that can display the dependence of an objective function from two design variables, and a two- or three-dimensional scatter diagram, able to compare directly all the objective functions in order to choose the best compromise between them.

As stated before this way of working may present some problems especially when the number of design variables grows too much. Indeed, the

response surface could reach easily  $1e9$  points, which would request consequent devices due to the heavy computational weight.

A possible solution would be then a mathematical optimization, exploiting the steepest ascent, which would lead to an easier, accurate and automatic optimization process but, at the same time, it would reduce the possibility to manually analyze the solution.

Moreover, it would be necessary to improve the postprocessing analysis, for instance implementing new kind of diagrams, and to implement new type of objective functions.

# Bibliography

- [1] E Abbasciano and ME Biancolini. “Shape morphing of CFD models”. In: *Ingegneria dell'autoveicolo* 64.1-2 (2011), pp. 30–40.
- [2] Marco Evangelos Biancolini. “Mesh Morphing and Smoothing by Means of Radial Basis Functions (RBF): A Practical Example Using”. In: *Handbook of Research on Computational Science and Engineering: Theory and Practice: Theory and Practice* (2011), p. 347.
- [3] Marco Evangelos Biancolini, Carlo Brutti, and Eugenio Pezzuti. “Shape optimisation for structural design by means of Finite Elements Method”. In: *ADM* 12 (2001).
- [4] A De Boer, MS Van der Schoot, and H Bijl. “Mesh deformation based on radial basis function interpolation”. In: *Computers & structures* 85.11 (2007), pp. 784–795.
- [5] Marco Fontanella. “NX Nastran user’s guide”. In: (2013).
- [6] Rajeev K Jaiman et al. “Conservative load transfer along curved fluid–solid interface with non-matching meshes”. In: *Journal of Computational Physics* 218.1 (2006), pp. 372–397.

- [7] Stefan Jakobsson and Olivier Amoignon. “Mesh deformation using radial basis functions for gradient-based aerodynamic shape optimization”. In: *Computers & Fluids* 36.6 (2007), pp. 1119–1136.
- [8] Xiangmin Jiao and Michael T Heath. “Common-refinement-based data transfer between non-matching meshes in multiphysics simulations”. In: *International Journal for Numerical Methods in Engineering* 61.14 (2004), pp. 2402–2427.
- [9] Ruichen Jin, Wei Chen, and Timothy W Simpson. “Comparative studies of metamodelling techniques under multiple modelling criteria”. In: *Structural and Multidisciplinary Optimization* 23.1 (2001), pp. 1–13.
- [10] Mansour Keramat and Richard Kielbasa. “Latin hypercube sampling Monte Carlo estimation of average quality index for integrated circuits”. In: *Analog Design Issues in Digital VLSI Circuits and Systems*. Springer, 1997, pp. 131–142.
- [11] Raymond H Myers, Douglas C Montgomery, and Christine M Anderson-Cook. *Response surface methodology: process and product optimization using designed experiments*. Vol. 705. John Wiley & Sons, 2009.
- [12] MSC Nastran. “NX Nastran quick reference guide”. In: (2012).
- [13] MSC Nastran. “NX Nastran user’s guide”. In: (2012).