



UNIVERSITÀ DELLA CALABRIA
Dipartimento di MECCANICA

DOTTORATO DI RICERCA IN INGEGNERIA MECCANICA

CICLO XXIII (2007-2010)

ING-IND/15 DESIGN AND METHODS FOR INDUSTRIAL ENGINEERING

Methods and tools for Topology Optimization systems integration within the Product Development Process

By Loris Barbieri

A Dissertation submitted in partial fulfilment of the requirements for the
Doctoral Research degree in Mechanical Engineering

Doctoral Research Director

Prof. Sergio Rizzuti

Supervisors

Prof. Maurizio Muzzupappa

Dr. J.P. Pernot

Candidate

Dr. Loris Barbieri

Acknowledgments

First and foremost, I sincerely thank Prof. Maurizio Muzzupappa and Prof. Fabio Bruno for their immense guidance and support, for having me in their research group, and most of all, for giving me the opportunity to learn, to grow, and to explore. Not only do I respect them as my research advisors, I also look up to them as my mentors.

I also want to express my gratitude to Prof. Cugini, Prof. Bordegoni, Prof. Cascini, Ing. Rotini, Prof. Nigrelli and Ing. Ingrassia with whom I had the opportunity and the good fortune to be member of the group responsible for the PROSIT project. The dialogue and exchange of ideas amongst this group was a constant source of inspiration as well as wisdom that often encouraged me to stretch the boundaries of my knowledge and experience in innumerable ways.

A large debt is also owed to Dr. J.P.Pernot for many helpful and fruitful discussions and suggestions especially on the second stage of my work. There is no doubt that the work is much stronger than it might have been for this input.

I gratefully acknowledge Prof. Luchi and Prof. Poggialini for valuable suggestions and comments that helped me to a much better writing and preparation of my papers. Thanks are also due to Dr. De Napoli who took the time to review my thesis.

I cannot forget all the people with whom I spend many hours and which are much more than colleagues: Marco, Agostino, Rosella, Francesco Caruso, Francesco Cosco, Stefano, Giorgio, Gianfranco, Alessandro. Finally, I thank Abdou Fadel and Ruding for their friendly sympathy and nice chats during my stay in France.

Loris Barbieri

Table of contents

Introduction.....	1
-------------------	---

– Part I –

<i>Toward the definition of a methodology for an efficient integration of Topology Optimization tools in the Product Development Process</i>	<i>6</i>
--	----------

Chapter 1

TO – CAD systems integration.....	7
1.1. Topology Optimization and the integration issue.....	7
1.2. Literature review.....	12
1.2.1. <i>Integration CAD/CAE/TO systems.....</i>	<i>13</i>
1.2.2. <i>Interpretation of 3D topology optimization results.....</i>	<i>14</i>
1.3. Proposed methodology.....	17
1.3.1. <i>Initial modelling.....</i>	<i>19</i>
1.3.2. <i>Mesh Generation and Model Optimization</i>	<i>21</i>
1.3.2.1. <i>Topology Optimization in Altair Optistruct.....</i>	<i>21</i>
1.3.3. <i>Model Checking and Geometrical Analysis</i>	<i>24</i>
1.3.4. <i>Final modelling.....</i>	<i>26</i>
1.4. Case study: plastic moped wheel.....	28
1.4.1. <i>Geometry Modeling:</i>	<i>29</i>
1.4.2. <i>Mesh generation and model optimization</i>	<i>30</i>
1.4.3. <i>Model checking.....</i>	<i>31</i>
1.4.4. <i>Feature based re-design:.....</i>	<i>33</i>
1.5. Conclusion and synthesis of the needs.....	34

Chapter 2

MB-TO-CAD systems integration.....	36
2.1. Extension of the methodology to Multi-Body systems.....	36

2.2.	Related works	39
2.2.1.	<i>Integration MBS-TO systems</i>	39
2.3.	Methodology.....	40
2.3.1.	<i>Design of a suspension component: the Upright</i>	40
2.3.2.	<i>MB-TO-CAD integration</i>	43
2.4.	Case study: FSAE upright design	46
2.4.1.	<i>Geometric problem definition</i>	47
2.4.2.	<i>Determination of dynamic loads</i>	48
2.4.3.	<i>Topological optimization analysis</i>	50
2.4.4.	<i>Model redesign</i>	52
2.4.5.	<i>Finite element analysis</i>	54
2.5.	Conclusions and perspectives.....	55

- Part II -

<i>Improvement of the methodology by introducing curve fitting tools and parametric archetype.</i>	58
--	----

Chapter 3

Constrained Force Density B-spline Curve Fitting implementation for the manipulation of the optimized profiles.	59
3.1. Introduction	59
3.2. Development of the algorithm for Curve Fitting based on the Force Density Method (FDM)	61
3.2.1. <i>Force Density Method</i>	63
3.2.1.1. <i>Mechanical model of a bar network</i>	64
3.2.1.2. <i>Bar network static equilibrium</i>	65
3.2.2. <i>Definition of the network</i>	67
3.2.3. <i>B-spline Force Density Fitting Curve</i>	72
3.3. State of the art.....	76
3.4. Constrained B-Spline Force Density Fitting Method	77
3.4.1. <i>Initial Force Density Fitting</i>	79

3.4.2. Curvature evaluation 79

3.4.2.1. Curvature 79

3.4.2.2. Leyton Process Grammar..... 81

3.4.3. Splitting ToolBox 83

3.4.4. Constrained Force Density B-spline Fitting Toolbox 87

3.4.4.1. Position Constraint 89

3.4.4.2. Distance Constraint 90

3.4.4.3. Straight Constraint..... 90

3.4.4.4. Curvature Constraint 91

3.5. Conclusion and limits of the fitting 91

Chapter 4

Parametric archetypes as support for interpreting topological optimization results.....94

4.1. Parametric archetype inception 94

4.2. Extension of the methodology with the introduction of the parametric archetypes.. 97

4.3. Case study: connecting rod 98

Synthesis and conclusions 103

Bibliography 105

Introduction

Product Development Process (PDP) includes many critical aspects, which depend on the increasing demands of the market pushed by global competition. To improve their competitiveness, companies are continuously researching on product innovation, supporting this complex process with the introduction of new manufacturing technologies that are progressively available and more efficient collaborative design environments based on the interoperability of data and applications.

These tools have demonstrated to increase extremely the effectiveness of the design activities where they are used; however they are not still integrated and interoperable, therefore not used optimally. Nevertheless, in a concurrent engineering view in order to reduce development times and increase activities effectiveness, it is required that the various phases of conceptual design, optimization and detailed design could be integrated as far as possible; that is because the design and product development process is a continuous iteration among these phases.

IEEE defined interoperability as the ability of two or more systems or components to exchange information and to use the information that has been exchanged [IEEE 1990]. The lack of interoperability among these innovative tools creates artificial barriers for communication and information sharing within the design process.

Achievement of efficient and effective interoperability among the software tools is needed not only to eliminate problems related to file format conversion and data exchange, but also to improve knowledge management, defined as the formalization, preservation and transmission of the actors' knowledge along the entire process. The effort must start from the consideration that it is necessary to formalize and store any design choice that the designers and analysts make during their activities. The most relevant choices have to be automatically transmitted to the interested actors in order to reduce informal collaboration - which leaves no trace and causes only delays - and to optimize the integration of new tools in the PDP. Therefore, there is a need, clearly expressed by manufacturing companies, for methods and tools capable of acquiring the design knowledge, in order to make it formalized and available in subsequent stages of the PDP and in future projects.

There are a lot of examples in which the introduction of an innovative system within the product development process cause unavoidable problems of communication and transmission of data and information among the different systems. For example, even if currently virtual reality is a design support technique widely recognized by industries, the efficiency of VR applications into the PDP still finds an obstacle in the poor integration of the instruments employed. The use of VR applications still needs long and quite hard procedures to work effectively: the models have to be converted into a format compatible with VR systems and have to be enriched with all the information needed to solve a specific task. In fact virtual assembly and virtual cabling represent two of the most common applications of VR within the PDP which lead the user to do a lot of manual work in order to prepare the virtual environment or to post-process data generated in VR. These integration problems can be resolved by developing specific software toolbox and interfaces, which are able to create an appropriate data exchange link between CAx and VR [Barbieri et al. 2008c, 2006].

The research activity described in this manuscript deals with the problems and limits of the introduction of Topological Optimization (TO) tools within the PDP.

Topology Optimization is a mathematical technique that better mechanical properties through a global optimization of the product in terms of weight, stiffness, resistance and cost. Besides, TO tools systems have already demonstrated their relevant potentialities to increase the effectiveness of specific design activities. Nevertheless, as stated before, the potential of a smooth and effective integration of all these computer-based systems has not been fully exploited. This is primarily due to firstly a poor integration between different applications, and secondly by the difficulty to fully understand how to take advantages of these tools and how to effectively use them in the context of the Product Development Process. Indeed, it has been estimated that the United States industry spends billions of dollars because of poor interoperability between Computer-Aided Engineering software tools [Szykman et al. 2001].

A full integration of these technologies is still far to be reached [Mervyn et al. 2004], and big efforts are required to set up successful collaborations and to push companies to focus their attention on the adoption of new organizational paradigms to better coordinate the design activity in such a context. The capability to support in a more integrated way all the

stages of the Product Development Process will be one of the most important competitive factors for these systems in the next years. In a concurrent engineering view, in fact, it is required that the phases of conceptual design, optimization and detailed design would be integrated as far as possible. In order to reduce development time and increase activities' effectiveness, design and Product Development Process have to be considered as a continuous iteration among these phases.

In particular the research activity starts taking into consideration three critical problems regarding the interoperability between the TOs and Product Lifecycle Management (PLM) systems. The first one is the definition of the problem of submitting to the TO, starting from a CAD model which defines maximum dimensions and invariant geometries. The second one concerns the problem of data exchange between CAD and CAE systems. The third one regards the interpretation of the results given by the TO, in order to realize the final CAD model, taking into consideration the manufacturability constraints of the product. This work addresses such complex problems.

More precisely, two parts have been distinguished. Part I describes the main issues related to the integration of these complementary instrument and the solutions proposed by the author. It gathers together two chapters that are crucial since they define the methodology which is at the basis of the next improvements and extensions realized in the second part of the manuscript:

- the first chapter corresponds to the study and individuation of these interoperability problems from which a methodology takes origin in order to obtain an efficient use of the topological optimizers in the PDP. All the arguments and reasoning described are supported and validated with case studies that demonstrate how the use of Software Development Kits (SDK), allowed by the CAx tools employed in the process, improves the efficiency of the methodology. In fact, the manuscript shows how it can be possible to create custom procedures and interfaces that automate operation sequences, thus fastening the designer's work, according to the typical CAD-Automation approach. More in details, they provide means to analyze and extrapolate useful geometrical information from the results provided by the optimizer, as well as semi-automatic modelling features

for some specific geometries. A detailed example related to the design of a plastic moped wheel clarifies the whole procedure.

- the second chapter proposes an extension of the methodology in order to increase the level of integration with other CAx systems and, most of all, to reduce decision makings of the user. In particular a CAD, a multi-body simulator and a topological optimization tool are synergically employed in order to support the design of a suspension component. In order to illustrate this capability, the process is applied to the conceptual design of the Upright for a Formula SAE prototype. The results show that the integrated design approach can efficiently support the selection of the optimum conceptual design of a mechanical component with complex dynamic behaviour, in particular when very little previous experience on the system is available. For the sake of brevity, the proposed methodology enables the designer to shorten design cycle time, improve design quality, and reduce cost by allocating less design iterations and by incorporating higher fidelity tools at the conceptual design stage.

Once the initial methodology has been defined it is possible go deeper into the development of methods and tools that allows to overcomes the limits and drawbacks of the approach proposed in the first part of the manuscript. Part II is structured into two chapters:

- the third chapter highlights the critical points of the initial methodology and develop specific tools that allows to improve the efficiency and objectivity of the whole process. More in detail this chapter is meant to provide an over-view of the mathematics that underlies the curve fitting technique implemented. This approach aims to translate the optimized profiles and curves into B-spline curves that could be used directly into a CAD system without users' intervention. As described in detail in the chapter, the fitting tool has been implemented upon the mechanical model of bar networks that allows to develop a more flexible tool than the traditional least squared approach usually adopted for fitting problems. B-splines are mathematically more sophisticated than other types of splines

(although it is not the most general type of this curve), but their flexibility is necessary in order to develop an efficient and versatile tool.

- the last chapter integrates and extends the methodology proposed in the first part of the manuscript with the tools and methods developed in the third chapter and introduces the parametric archetype. It corresponds to a 2D functional description that contains a base of knowledge common to all versions of the same mechanical component. This knowledge can be shared and reused in different stages of the proposed methodology but especially during the redesign phase in order to ease and fast the interpretation of the projections of the optimized model. The approach has been tested with a case study.

This research activity has been inspired by the involvement in the “MUR/PRIN-Prosit” project [Cugini et al. 2008; Prosit 2008] which purpose is to study and prove the possibility of integrating innovative tools, like Product Lifecycle Management and Knowledge Based Engineering systems, with CAI (Computer Aided Innovation) tools and with topological optimizers, in order to offer formalized and validated routines, which allow a systematic adoption and an integrated utilization of such tools in the PDP [Cugini et al. 2007].

- Part I –

Toward the definition of a methodology for an efficient integration of Topology Optimization tools in the Product Development Process

This first part aims at presenting the proposed methods and tools which have been developed to ease the introduction of TO tools within the PDP. It gathers together two chapters.

The first one, starting from the limits risen from the state of the art and the lacks of integration between TO tools and the other CAx systems, introduces a methodology that allows to achieve a high interoperability among the various product lifecycle support tools.

The second chapter extend the methodology proposed in the first chapter to enable an interoperability integration among a multi-body simulator, a topological optimization tool and a CAD system.

The first part ends with a synthetic description of all the advantages of the proposed methodology but also of the crucial points on which is possible operate in order to enhance the whole methodology. One will notice that the conclusions of these chapters introduce and inspire the goal of the second part of the manuscript.

Chapter 1

TO – CAD systems integration

This chapter synthesizes the results of the bibliographical analysis of the approaches and techniques actually adopted in order to improve the integration of TO tools in the PDP and, in particular, to support the post-processing of optimizer results (section 1.2). This study brings to the development of a methodology (section 1.3) which defines the guidelines to implement an efficient use of TO tools in PDP. In CAD environment these guidelines are translated into knowledge based user interfaces that are able to simplify the designer's job. The last section concludes about the validation of the proposed methodology with an industrial case study of a plastic moped wheel (section 1.4).

This chapter is crucial since it defines the methodology which is at the basis of the next improvements and extensions realized in the next chapters.

1.1. Topology Optimization and the integration issue

The design process can be viewed as an optimization process to find structures, mechanical systems, and structural parts that fulfill certain expectations towards their economy, functionality, and appearance. Generally, the design process is an iterative procedure consisting of the following components:

- Conceptual design
- Design
- Testing
- Optimization

Today's testing ground is usually the computer. Finite element analysis (FEA) and Multi-body dynamics analysis (MBD) are the most used tools for computational design testing. The results of computational analyses are used to determine design improvements.

Changes to the design are introduced in all phases of the process. At a certain stage of this process, changes to the concept become prohibitive. The concept phase plays a fundamental role concerning overall efficiency of the design and the cost of the overall development process. Indeed a well-known fact is that more than half of the production cost is committed during the conceptual design stage. It is believed that it is during this early design stage that one has the greatest opportunity for leverage and change. This is perhaps the time and the place where design changes can be made rapidly with little cost. The right decisions in this stage ensure a successful end product. In the conceptual design phase of a design process the freedom of the designer is limited only by the specifications of the design, this notion has gain general acceptance over the past decade and may be depicted graphically by the illustration in figure 1 [ROED 1996]:

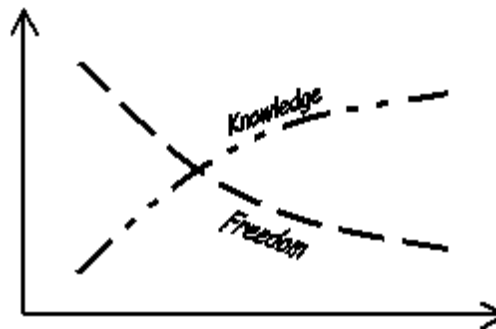


Figure 1: Decision making in the design process.

Today, the decision on how a new design should look is based largely upon a benchmark design or on previous designs. The decision making is based on the experience of those involved in the design process. Conceptual design tools such as *topology optimization* can be introduced to enhance the process.

Topological optimization [Bendsoe et al. 2003] is an innovative design technique, which drives the design process in the search of the optimal shape of a mechanical structure within a pre-defined design space. The method generates a structural shape, thus providing a first idea of an efficient geometry [Schramm 2005; Schumacher 2005], in the attempt to reach the optimization goal [Schramm 2003]. Thence the concept can be based on results

of a computational optimization rather than on estimations (fig.2). Using topology optimization, the initial design step is already based on input generated using computational analysis. Topology optimization redefine the role of computational analysis and simulation in the design process. Finite element analysis has matured from a testing tool to a design tool.

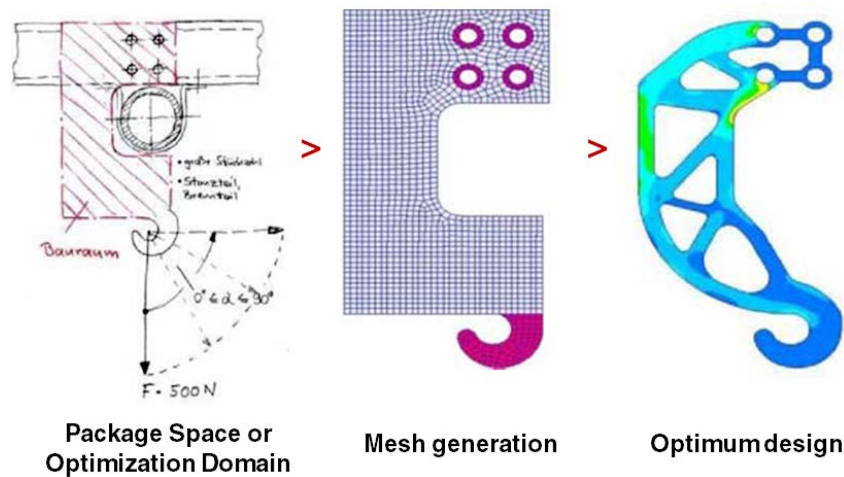


Figure 2: traditional optimization process.

In the real world, the design process is not as straightforward as described above. The design is not just driven by one performance measure - it has to be viewed as a multidisciplinary task. Today, the different disciplines work more or less independently. Analysis and optimization is performed for single phenomena such as linear static behavior or noise, vibration and harshness. Still, the idea persists that if one performance measure improves, the whole performance improves. A simple example shows that this is not quite true. Take the design of a car - a high stiffness is necessary for good driving and handling, and high deformability is important for the crashworthiness of the design. This shows that improving one measure may result in degrading another. Therefore, compromises must go into the formulation of the optimization problem. The definition of the design problem and of the design target is most important. The solution can be left to computational means. Multidisciplinary considerations, especially in the conceptual design, are, in many ways, still active research topics and are being covered by future developments of topology optimization.

Figure 3 compares the design process using topology optimization with the conventional method of leaving the concept entirely to experience and intuition. The overall cost of

design development can be reduced substantially by avoiding concept changes introduced in the testing phase of the design. This is the major benefit of modifying the design process by introducing topology and topography optimization.

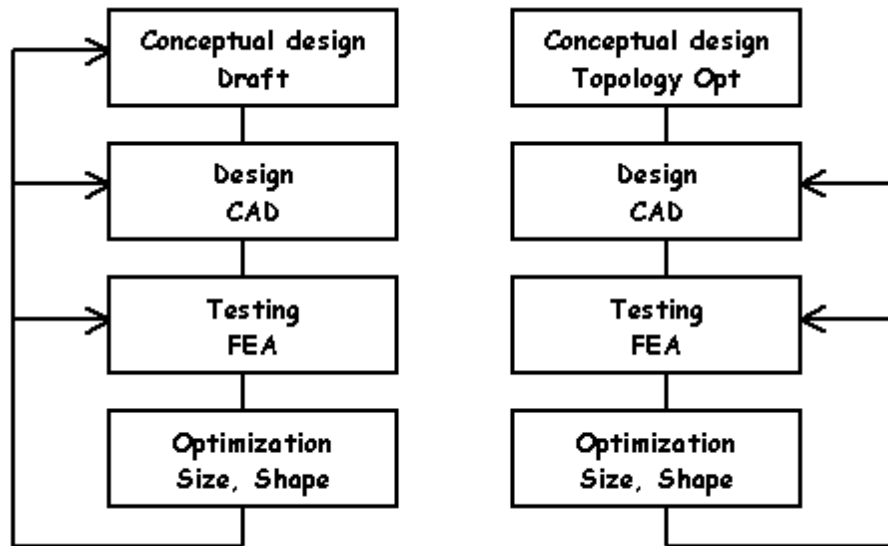


Figure 3: The design process without and with the use of topology optimization.

Topological optimization tools are put on the market as integrated modules within FEA (Finite Element Analysis) systems like Altair Hyperworks [AE 2010], Tosca [FED 2010], MSC Nastran [MSCN 2010], which allow engineers to analyze and optimize the product. Thanks to their particular features, these tools are employed in several engineering fields (automotive, civil, aeronautics, aerospace) but, whether we refer to commercial systems or to solutions proposed in the literature (section 1.2), the introduction of TO systems in PDP intensifies the need of managing design knowledge and integrating CAx systems in PDP. In particular, at present, there are no tools able to:

1. **support the correct formulation of a topological optimization problem.** If the TO problem is not well defined, none of the results will lead to the optimal solution. During this phase, in fact, it is necessary to define both an objective function (typically in terms of volume, mass, uniformity of distribution of stress or of deformation energy) and the project constraints (load system, maximum dimensions, functional surfaces, etc.). Moreover, it is important to know which volumes are to be considered “invariant” (not modifiable during the optimization process) and which volumes are on the contrary to be optimized.

2. *guarantee an efficient data exchange between CAx and TO systems.* The introduction of TO systems in PDP involves a redesign of the whole process because of the poor integration with CAx systems.

3. *translate TO results in order to realize a CAD model,* taking into consideration the manufacturability constraints of the product. All the geometric information are lost when a virtual model of the product undergoes an optimization cycle (fig.4). The consequence is that the optimized model does not take into account either functional and technological features, or rules and attributes associated to the various parts of the product.

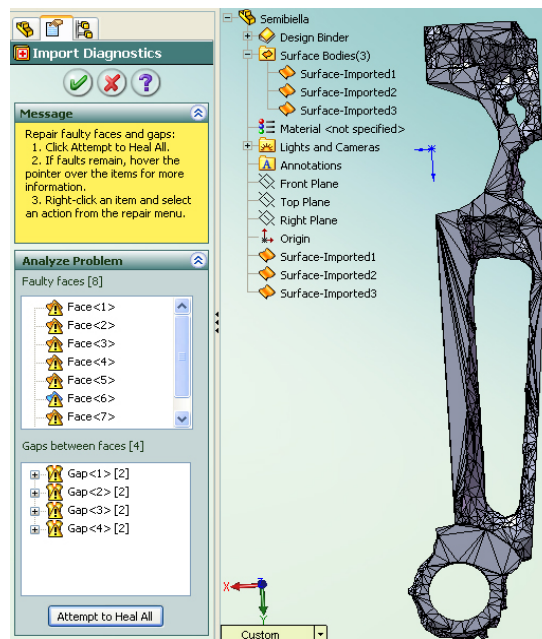


Figure 4: Optimized model imported into a CAD system and feature-tree that collects only the initial reference planes and imported surfaces with many errors.

At present, neither commercial systems nor solutions proposed in the literature can respond to all these issues, it is therefore particularly important to develop methodologies and tools to rationalize the integration process between TOs and CAx systems. Only two approaches seem to be practicable:

1. develop a specific tool able to translate TO results into a feature model through an automatic interpretation of the voxel model;
2. develop CAD automation tools in order to extend the potential (and to facilitate the interoperability) of current CAx systems.

The second approach is the most suitable for a rational integration of TO systems in PDP, because it allows the engineers to interpret the TO solution while taking into account other

non-formalized constraints (e.g.: aesthetics, manufacturability, etc.). In order to accomplish this goal it's important to pick out that the most important PLM vendors have developed specific tools to achieve the interoperability between software of different typologies. These tools allow the user to implement the software modules needed to build an efficient communication pipeline between the various PLM components. Generally, these Software Development Kits (SDK) are used by the same software house which produces them, in order to implement the interoperability between the several components of its own PLM. In some cases, the SDKs are available to the customers who need to integrate third-part software products, or their own code, on the pipeline.

On the basis of these consideration the research activity has been focused on defining a methodology whose aim is to obtain an efficient use of topological optimizers in the PDP. The development of this methodology has been reached by means of the use of SDK tool for the implementation of Knowledge Based (KB) custom user interfaces able to guide and simplify the designer's job.

1.2. Literature review

As stated before, an effective introduction of new design techniques, like topological optimization, requires the adoption of methodologies and tools able to formalize, store and make the knowledge and the experience acquired by engineers involved in the PDP available to other designers. There have been some commercial KBE (Knowledge Based Engineering) Systems for over 15 years, like SeaShell, KTI - The ICAD System, Behavioral Modeling-PTC, RuleStream, etc. Such softwares support the formalization of knowledge by incorporating the design choices (design intent), the engineering and manufacturing rules and the best practices, in order to make such knowledge available for a rapid development of new products. These systems usually work on high-level product representation schemes [Mandorli et al. 2001], which enclose both the geometry of the component, described in terms of features, and other attributes and rules that implement the knowledge base associated to that component [Ma et al. 2003].

The necessity of KB tools prove the lack of interoperability between TOs and the other CAx systems, in particular the need of integration between CAD and TOs is fundamental

in fact one of the main problems regards the interpretation of optimized models due to the loss of all features data in the optimization process.

1.2.1. Integration CAD/CAE/TO systems

As stated in the introduction, whether we refer to commercial systems or to solutions proposed in the literature, the interoperability between CAx systems and topological optimizers is, at present, a particularly critical aspect.

Actually TO tools have reached wide spread acceptance as integrated applications in the CAE environment, but a closer integration in the product development process is still desirable. At present, in the literature there are few works about these topics.

A first proposal is presented by FE-Design [FED 2010; Sauter et al. 2001a] that offers with the optimization system CAOSS [Sauter et al. 2001b], an integrated solution for structural optimization. The integration of topology optimization in the design process starts from the definition of the design space in the CAD system. Mesh generation and topology optimization follow as next steps. The result of the optimization has to be transferred into a smooth structure where a remodelling due to manufacturing constraints can take place. In the last step the optimized design is converted into a geometry based CAD model [Spath et al. 2001].

Differently from FE-Design, Vaidya et al. [Hawthorne et al. 2005; Vaidya et al., 2006] have created an integration between geometric, FE and design optimization models in which the designer is supported by a GUI-based module used to build the finite element analysis and design optimization databases from the CAD database and eventually rebuild this latter from the results obtained by topological optimization. The software suite is a collection of independent modules that can be plugged together to create the required set of FE analyses and design optimization capabilities. No support is available for the designer to interpret the results of the topological optimization.

Besides what is proposed in the literature, nowadays the most important software vendors have developed specific tools to achieve the interoperability between systems of different typologies [HM 2010; PT 2010; PXS 2010; SWOA 2010]. These tools allow the user to implement the software modules needed to build an efficient communication pipeline between the various systems. Generally, these Software Development Kits (SDK)

are used by the same software house which produces them in order to implement the interoperability between the several tools. In some cases, the SDKs are available to the customers who need to integrate third-part software products, or their own code, on the pipeline.

1.2.2. Interpretation of 3D topology optimization results

A further problem related to the use of TOs in PDP lies in the extraction of a feature based 3D model from the results of the topological analysis. As stated before this problem is due to the loss of all features data in the optimization process. The output of topological optimizers, i.e.: the optimized model is a voxel model, so it takes into account neither functional and technological features, nor rules and attributes associated to the various parts of the product. In practice, the geometry can be useless because the optimized model:

1. may not be manufacturable or its manufacturing costs may be very high;
2. violates design constraints, standards or functional aspects;
3. turns out to be very complex to rebuild in terms of functional and technological features.

In all these cases, the designer has to model the geometry again, interpreting the results obtained by the optimization process, and considering standards and manufacturing constraints [Rosen et al.1992]. At present, the only tool offered by PLM systems as a support for the user are feature recognition modules; among others the most widespread are feature recognition 1 of Catia, Holemaking of UG/NX, Feature Recognizer of SolidEdge and Feature Works of Solid Works, but this tool are ineffective if used on topological optimization results.

There are many interpretation techniques available in the literature, most of them only deal with the two dimensional problems [Corney 1993; Kyprianou 1980; Vandenbrande et al. 1993; Yildiz et al. 2003; Ozturk et al 2006]. For example, the image interpretation approach commonly seen in the literature can only represent the boundaries of the two-dimensional gray level topology result. Moreover, some of the interpreting techniques must be performed manually. Thus, a technique that can handle both two- and three-dimensional problems and can be performed without human intervention is desired.

All these techniques can be roughly divided into three categories: image interpretation approach [Papalambros et al. 1990; Bendsoe et al. 1991; Olhoff et al. 1991; Bremicker et al. 1991; Chirehdast et al. 1994; Lin et al. 2000], density contour approach [Kumar et al. 1996; Youn et al. 1997; Hsu et al. 2001], and geometric reconstruction approach [Tang et al. 2001]. The image interpretation approach uses graphic facilities or computer vision technologies to represent the boundary of the black-and-white finite element topology optimization result. The density contour approach generates the boundaries of the structure by redistributing densities from the topology optimization result. In the geometric reconstruction approach, the boundaries are represented by the mathematical geometric reconstruction technique. Topology optimization result is usually a gray level image in discrete finite elements, which is hard to interpret from a design point of view. It is especially difficult to interpret three-dimensional topology optimization result indeed where 3D topological optimization results are concerned, there are few studies available in literature.

Hsu et al. [Hsu et al. 2001, 2005], adopting the density contour approach, present an automation process for translating three-dimensional topology optimization results into a smooth CAD representation. In the interpretation process the three-dimensional component is described by representative cross-sections. On each cross-section a density redistribution algorithm translates the black-and-white voxel data resulting from topology optimization into a smooth density contour represented by B-spline curves. A three-dimensional CAD model is obtained by sweeping through all the cross-sections (fig.5).

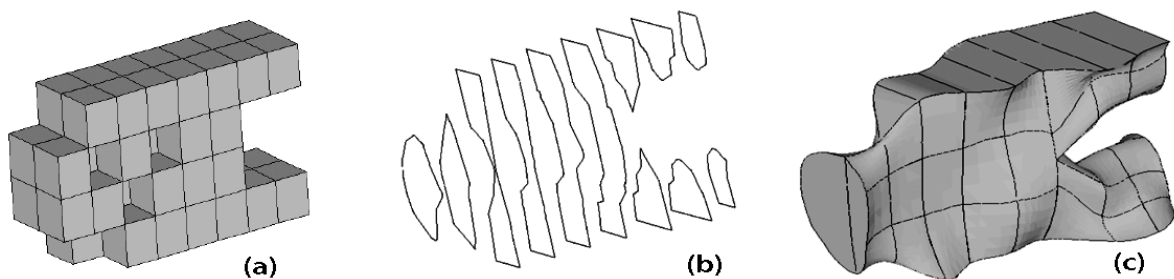


Figure 5: (a) Topology optimization result of a three-dimensional cantilever beam; (b) sections extracted from the optimized model; (c) the shape optimization result of three-dimensional cantilever beam.

It should be noted that the selected number of representative cross-section is related to the resolution of three-dimensional CAD model. More representative cross-sections are needed when the resolution of three-dimensional model is required. On the other hand, the

direction of selected representative cross-section will affect the efficiency in reconstructing the three-dimensional CAD model (fig.6).

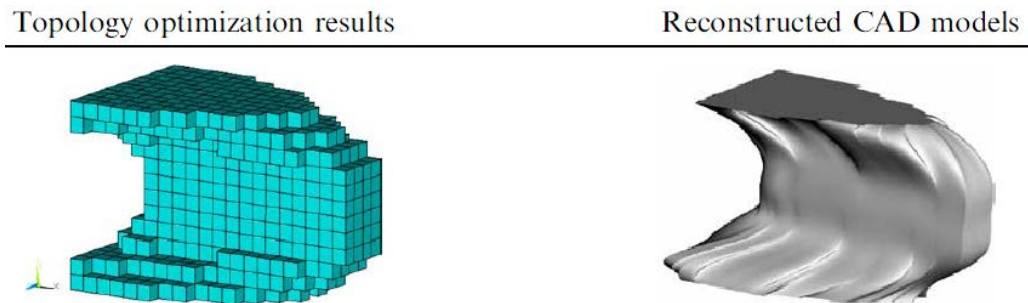


Figure 6: Topology optimization results and the reconstructed CAD models using sections extracted along y direction.

Tang and Chang [Tang et al. 2001] present an integrated approach that supports the topological optimization and CAD-based shape optimization. This integration is carried out by first converting the geometry of the topologically optimized structure into B-spline curves and surfaces. In particular these parametric cubic B-spline curves and surfaces are utilized for curve fitting and surface skinning, respectively, which approximate the boundary edges and surfaces of the coarse structural layout. During this approximation process, the control points, which govern the shape of the B-spline curves and surfaces, as well as basis functions are acquired (fig.7).

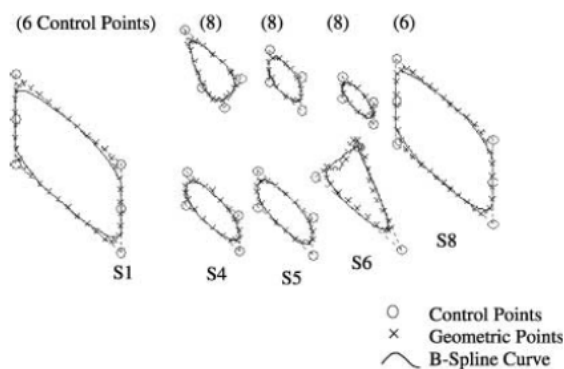


Figure 7: Section contours and fitting curves.

Then, through CAD API (Application Programming Interface), these control points and basis functions are imported into the CAD environment to reconstruct solid models that are enclosed by the B-spline curves or surfaces (fig.8). For structures with branches, Boolean

operations are employed. These imported control points also parameterize the boundary curves or surfaces of the reconstructed solid model and later serve as design variables for CAD-based shape optimization.

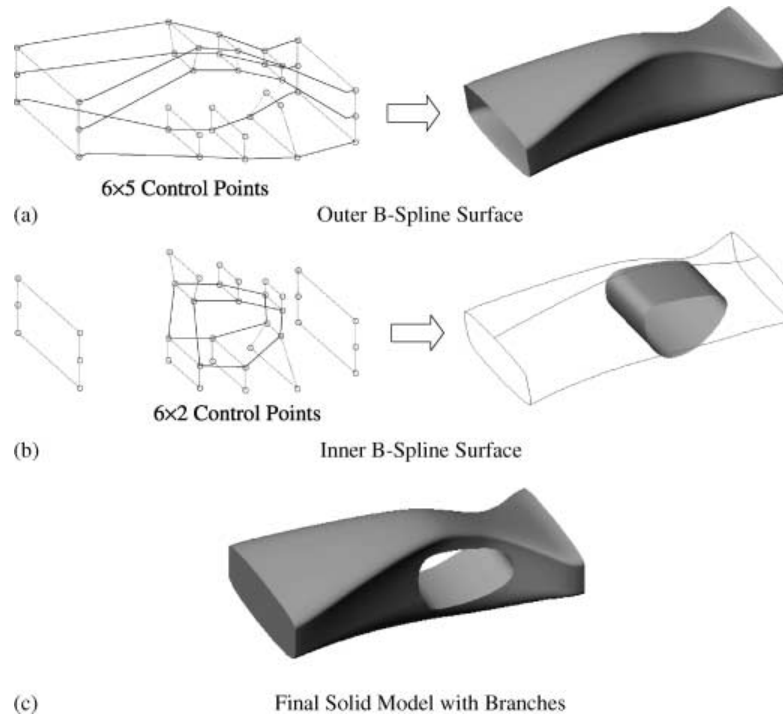


Figure 8: Solid model created using Boolean subtraction.

1.3. Proposed methodology

The introduction of TO systems in the design process requires a new definition of design methodology, because the introduction causes complications in the communication between tools. The introduction of TO systems in PDP affects two aspects of the process: the data exchange between TO and CAx systems and the interpretation of the results of the optimization, in order to realize the final CAD model (Fig.9).

On the basis of the considerations expressed in section 1.1, a methodology has been defined that does not modify the phases of the traditional process (fig.9: modeling – optimization – remodeling) but revisits this process in order to achieve some improvements through the use of custom user interfaces and the introduction of new CAx systems.

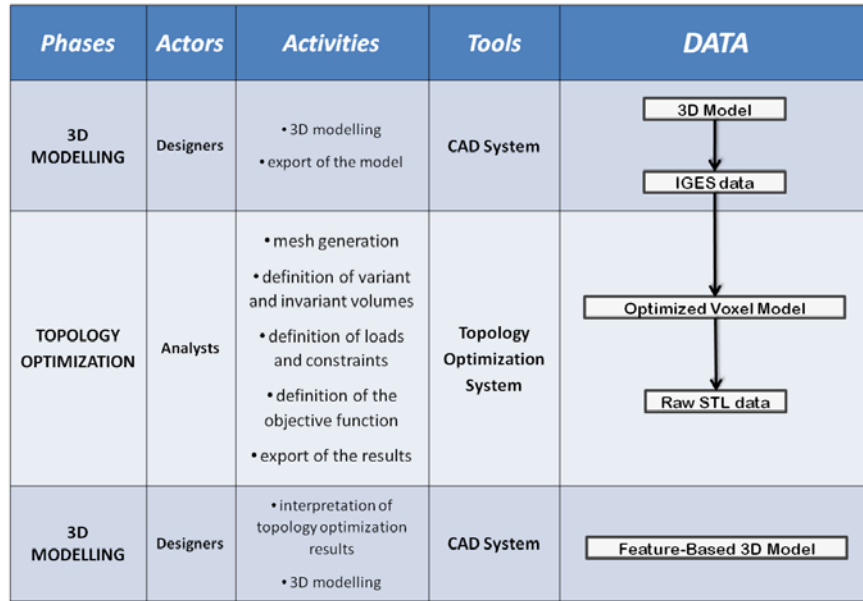


Figure 9: Traditional methodology.

This methodology facilitates the designer’s job and the communication with the analysts during the whole process and provides a knowledge base to make decision less subjective. The steps of such a methodology and the implemented tools are shown in figure 10.

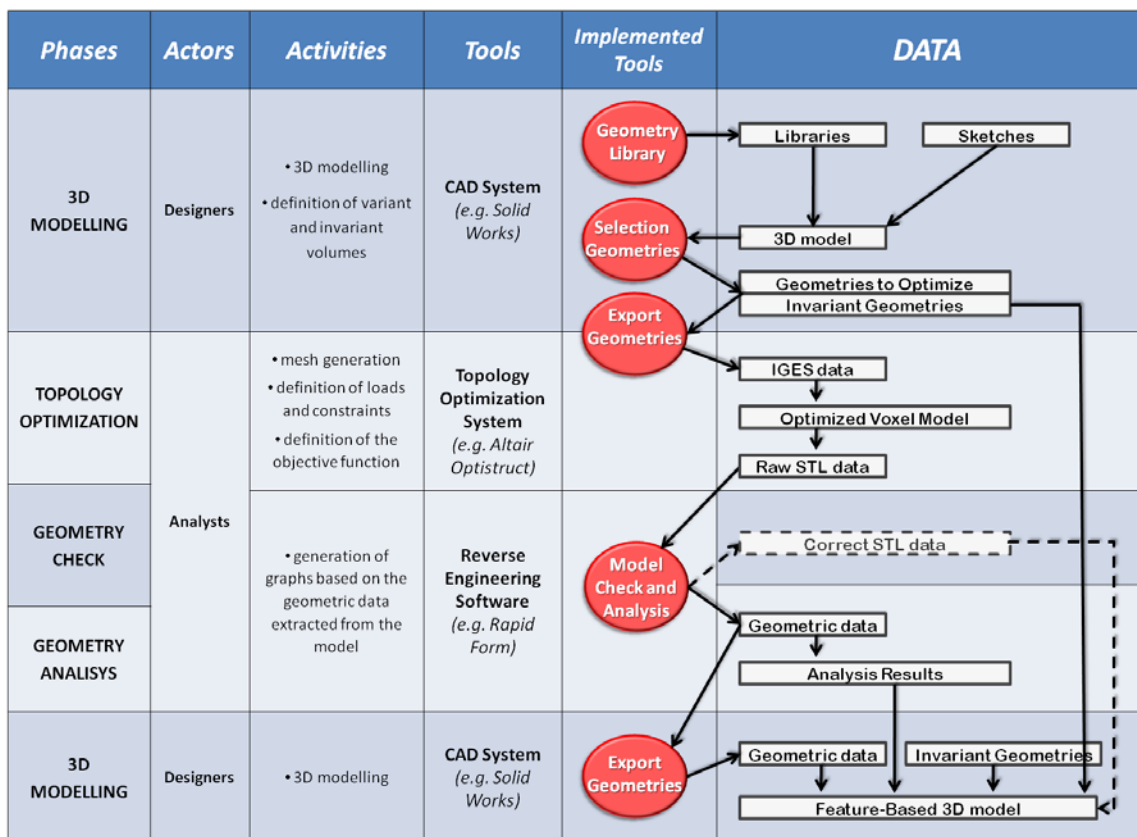


Figure 10: The methodology and implemented tools.

In the following sections the each phase of the proposed methodology will be explained in detail.

1.3.1. Initial modelling

The first step of the methodology is the preparation of the model to be optimized. According to the statements made in the previous section (it is always possible to subdivide the geometric model into a set of “characteristic” volumes that can be considered invariant volumes or volumes to be optimized), a tool is provided that allows the designer to model the component in a CAD environment by joining primitive geometries. This approach is justified because the introduction of TO systems in the design process simplifies the modelling phase, considering that the optimal geometry of the component will be defined by a TO tool. At this stage, in fact, the CAD model can be roughly defined to specify the overall dimensions and volumes that will remain unchanged during the optimization process. Therefore, the designer should model the component by joining primitive geometries: either through a simple geometry sketch or by taking it from custom libraries (fig.11). For the second case, it is necessary to define a specific 3D model library formed by “characteristic volumes” of the component to be analyzed (this solution is recommended if it is necessary to optimize a whole family of parts).

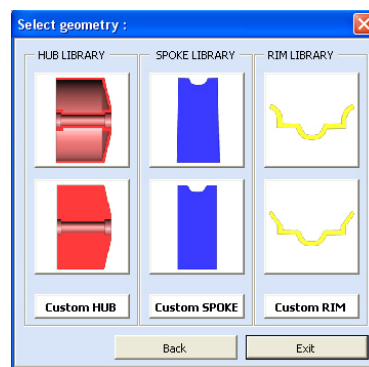


Figure 11: Interface for the modelling of the optimization model.

For the implementation of this interface, the libraries available with the API of SolidWorks have been used. In particular, the following routine have been utilized: the properties of ModelDoc2, PartDoc and Body2 for the creation, loading and modification of the native files of SolidWorks, as well as for the option of accessing properties of the geometries, such as colour. The functions of the SelectionMgr Object served to make the

interface interactive, while the numerous properties of Feature and Sketch Objects allow the designer to insert, modify and eliminate the parts which populate the library.

The creation of a model through the union of different volumes easily distinguishes geometries subject to optimization from the invariant ones. This method offers several advantages. It facilitates geometric modelling operations, simplifies the creation of the model to be optimized and, most of all, it allows the preservation of the knowledge about the identification of each geometry throughout the optimization process. The formal identification of the geometries is useful for two reasons:

- the analyst immediately recognizes constrained and variable parts before starting the TO;
- the designer limits the reconstruction phase of the geometrical model only to the optimized volumes.

Moreover, such a modelling procedure makes the data transfer from the CAD to the CAE environment much more effective because it simplifies the instantiation of various volumes to which one may apply different collectors (during optimization analysis). In order to support the data exchange between CAD and TO systems, two other interfaces are provided. The first allows us the selection of invariant geometries which can be stored in separate files and reloaded later. These interfaces were implemented by using the SolidWorks API and, in particular, the PartDoc and Feature Objects routines. The second interface allows the designer to transfer the data to the HyperMesh module directly from SolidWorks. This functionality is implemented through Visual Basic 6 routines and, by means of Tcl/Tk script, allows the analyst to automatically import the model into Altair (fig.12).

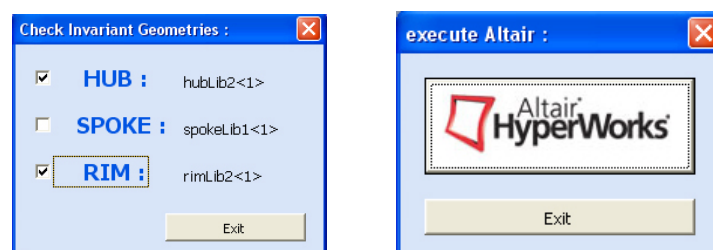


Figure 12: Interface for the selection of invariant geometries and window for loading Altair.

1.3.2. Mesh Generation and Model Optimization

The optimization phase primarily requires the definition of collectors, which are the physical properties of the model and of the boundary conditions. A collector can specify the type of material, loads and constraints, or also define variant and/or invariant volumes. When the analyst starts to work in the TO software, he/she has the model exported from the CAD immediately available, having already defined which parts are invariants and which have to be optimized. This is a very useful and practical application of Knowledge Management, because the STL file loaded in the TO is unable to convey any metadata that may support the communication from the designer to the analyst. The implemented tools support the automatic transfer of semantic data about the variant/invariant parts.

Once the topological optimization phase is completed, it is possible to export the model in an STL file.

1.3.2.1. Topology Optimization in Altair OptiStruct

In commerce various solvers exist in order to perform topological optimization analysis, as stated before Altair OptiStruct has been chosen as solver for the proposed methodology.

The history of this software has undergone several stages of evolution towards its maturity today. The first version of this code was a research code that was developed in university research labs by a group led by Profs. Diaz and Kikuchi in 1991. Altair started to market it to industrial users as Altair OptiStruct 1.0 in 1993.

Altair OptiStruct is a finite element and multi-body dynamics software which can be used to design and optimize structures and mechanical systems. Topology optimization analysis, performed by means of OptiStruct, generates an optimized material distribution for a set of loads and constraints within a given design space. The design space can be defined using shell or solid elements, or both. The classical topology optimization set up solving the minimum compliance problem, as well as the dual formulation with multiple constraints are available. Constraints on von Mises stress and buckling factor are available with limitations. Manufacturing constraints can be imposed using a minimum member size constraint, draw direction constraints, extrusion constraints, symmetry planes, pattern grouping, and pattern repetition. A conceptual design can be imported in a CAD system using an iso-surface generated with OSSmooth, which is part of the OptiStruct package.

Optistruct solves the following structural optimization problem:

$$\min f(x) = f(x_1, x_2, \dots, x_n)$$

$$\text{Subject to: } g_j(x) \leq 0 \quad j = 1, \dots, m$$

$$x_i^L \leq x_i \leq x_i^U \quad i = 1, \dots, n$$

The objective function $f(x)$ and the functions $g(x)$ in the constraint function are structural responses obtained from a finite element analysis. A constraint is considered active if it is satisfied exactly ($g = 0$); it is considered inactive if $g < 0$; it is considered violated if $g > 0$.

The selection of the vector of design variables x depends on the type of optimization being performed. In topology optimization, the design variables are element densities.

In order to solve these structural optimization problems OptiStruct uses the density method [Mlejnek 1992; Yang et al. 1994], also known as SIMP (Solid Isotropic Microstructure with Penalty method) [Rozvany et al. 1995] in the research community.

Under topology optimization, the material density of each element should take a value of either 0 or 1, defining the element as being either void or solid, respectively. Unfortunately, optimization of a large number of discrete variables is computationally prohibitive. Therefore, representation of the material distribution problem in terms of continuous variables has to be used. With the density method, the material density of each element is directly used as the design variable, and varies continuously between 0 and 1; these represent the state of void and solid, respectively. Intermediate values of density represent fictitious material.

In general, the optimal solution of problems involves large gray areas of intermediate densities in the structural domain. Such solutions are not meaningful when we are looking for the topology of a given material, and not meaningful when considering the use of different materials within the design space.

Therefore, techniques need to be introduced to penalize intermediate densities and to force the final design to be represented by densities of 0 or 1 for each element. The penalization technique used is the "power law representation of elasticity properties," which can be expressed for any solid 3-D or 2-D element as follows: $\underline{K}(\rho) = \rho^p \underline{K}$ where \underline{K} and \underline{K} represent the penalized and the real stiffness matrix of an element, respectively, ρ is the density and p the penalization factor which is always greater than 1.

The case studies faced during this work refers to three-dimensional models, then for what concern 3D models, according to the homogenization method, the material is perforated by an infinite number of periodically distributed voids. In each of the solid cells containing a void, the void is shaped like a rectangular prism with a length, width, and height determined by variables a , b , and c (fig.13). The orientation of the void is determined by angles Θ , Φ , and Ψ . These angles are determined by orienting the void in the direction that is most effective in reducing strain energy density.

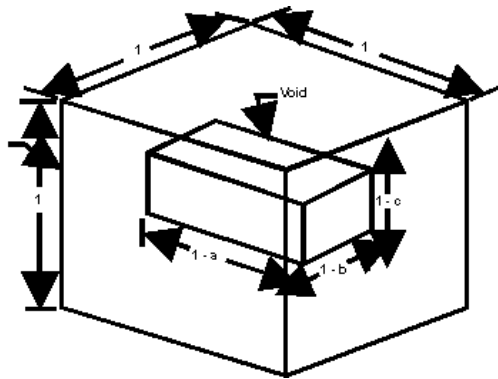


Figure 13: A cell in a solid element. Design variables a , b , and c for a solid design element.

The effective material property for a given void size (a , b , and c) is calculated for the perforated material, assuming that there are an infinite number of such voided cells. The equivalent an-isotropic material property is rotated to the optimal void direction and is then used in the calculation of the elemental stiffness matrix.

Therefore, each solid element has three void size design variables and three void angle design variables. The material density for a solid element is equal to 1, minus the volume of the void:

$$\rho_{MAT} = 1 - (1 - a)(1 - b)(1 - c)$$

This term is dimensionless during optimization and may vary within the range of the *mindens* parameter (default = 0.01) to 1. A material density of zero indicates there is no material in the element and results in an ill-conditioned stiffness matrix. The *mindens* should never be set to zero for models with solid design elements. A material density of 1 indicates the void size is zero. In the density method, the material's density is directly used as the density design variable. The effective elasticity property is equal to a scalar function

of the density, times the original material property. In this case, there are no angle design variables.

1.3.3. Model Checking and Geometrical Analysis

As already mentioned, the translation of optimization results into a feature-based model, usable for successive analysis phases, is not an operation adequately supported by CAD systems. One of the main problems pertains to the fact that all the geometric information of the voxel model generated by the TO are lost when a virtual model of the product undergoes an optimization cycle. The consequence is that the optimized model does not take into account either functional and technological features, or rules and attributes associated with the various parts of the product.

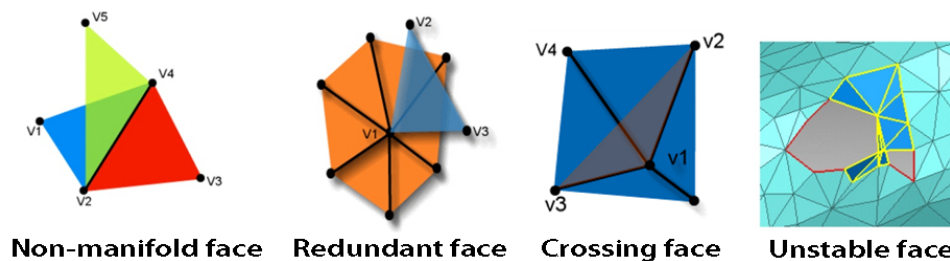


Figure 14: Typical errors present in a STL model.

Reverse Engineering software (i.e. RapidForm) is able to support the analyst in resolving different problems such as geometry checking, mesh simplification and reference geometry extraction.

According to the proposed methodology, the analyst manages and interprets the result of the TO with the aim of extracting some useful information that will guide the designer in the final modelling phase. In order to facilitate these operations, a KB custom user interface (fig.15):

1. simplifies the navigation through several RapidForm menus (a selection of RapidForm functionalities that are required for carrying out model checking and analysis);
2. represents a logical sequence of operations to carry out, thus supplying the user with a basic knowledge concerning the number and the succession of operations to be carried out.

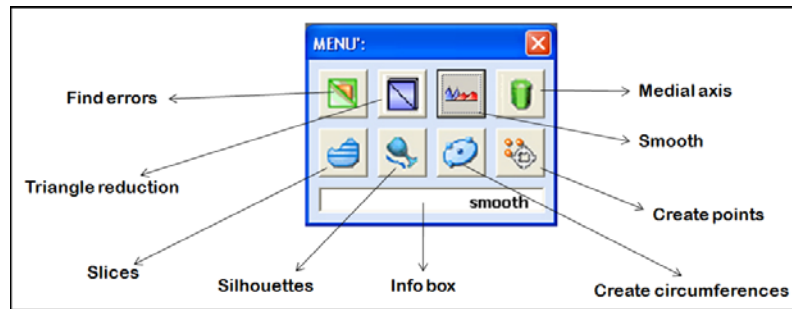


Figure 15: Interface for the rapid execution of tools present in RapidForm.

These functionalities are very useful, because the analyst can clean the voxel model from all faulty triangles (fig.14) and can extract useful information from the model on the basis of his/her own knowledge. In particular, the analyst, starting from his/her knowledge related to specific case study, can:

- extract profiles, guidelines, trajectory, points and/or reference curves from the voxel model (fig.16);
- calculate different section properties (fig.17-18);
- measure distances;
- draw graphs.

This interface was realized by using RapidForm API, in particular the functions belonging to RFCurveTool, RFDocument, RFMeshTools, RFModel, RFNurbsCurve, RFPointTools and RFRefGeomTools Objects.

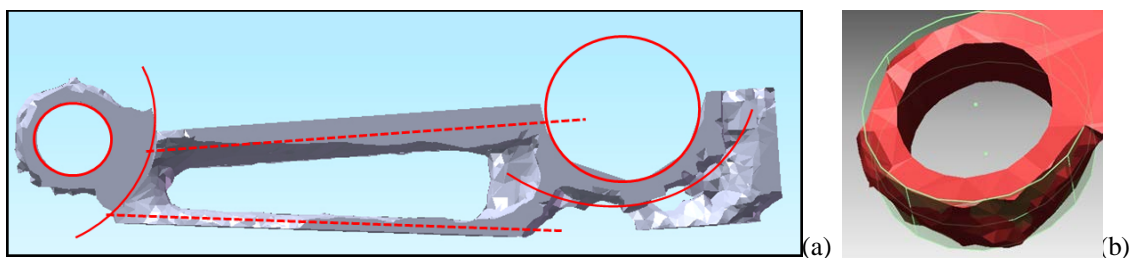


Figure 16: Reference curves, axis (a) and surfaces (b) extraction from the optimized model of a connecting rod.

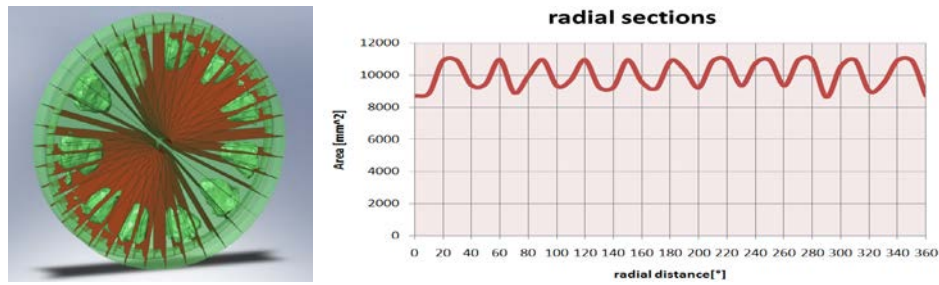


Figure 17: Graph of the radial section areas.

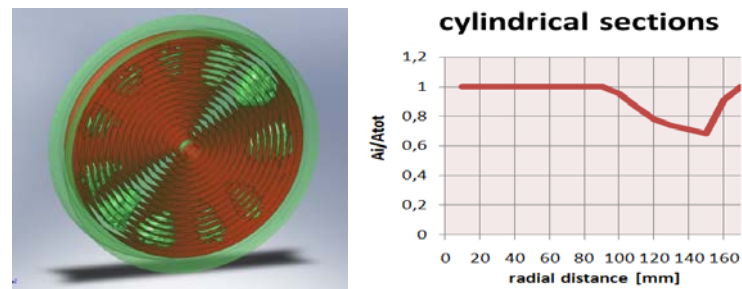


Figure 18: Graph of the cylindrical sections areas.

1.3.4. Final modelling

As discussed before, the topological optimization tools are not adequately integrated with CAD systems; the main problems being the interpretation of the results generated by TO. Generally, the analyst provides the complex geometry to the designer who must remodel the component without any support, subjectively interpreting the results. According to the methodology presented, the designer should be able to operate within a CAD environment using two different types of data: invariant geometries/volumes and reference geometries extracted from the optimized volumes. The availability of such data significantly simplifies the reconstruction of the feature based model, which will be further processed. Moreover, the availability of the reference geometries makes the reconstruction of the model shape less dependent on the interpretation of the designer because he/she can rely on the analyst's knowledge.

If the operations of data export are carried out manually by the analyst, a good knowledge of functionalities and import/export options of the software is required. In fact, a lot of time will be spent managing a large number of files. Such considerations led to the implementation of a module that allows the user to select reference geometries realized in RapidForm and export them interactively to SolidWorks.

The module uses the APIs of both Rapidform and Solidworks. The module is run in RapidForm and its interface is subdivided into three parts (fig.19). In the first part, the user selects the type of object to export: NURBS (profiles, guidelines, projections of silhouettes of the model, etc.) or reference geometries (points, lines, axes, circumferences, etc.). The interface uses the functions found in the RFSelectTools library and, in particular, the SelectEntity function, which requires the entity type to be defined.

The second part of the module concerns the operations of importing data to SolidWorks. The user may choose to import a curve, a profile or a line as a 3D curve or as a 3D sketch. This choice allows the user to import data in the most suitable way, according to the features he/she wants to use during the remodelling phase. For example, if one wants to realize a Loft or Sweep operation, it is better to import profiles as 3D curves; if one wants to realize an Extrude or Cut operation, it is better to import the profile as a 3D sketch. The implementation of such import options uses the SolidWorks API, and, in particular, the ImportIgesData Object functions. This object allows one to specify levels and values when importing IGES data and its CurvesAsSketches property allows the user to get or set whether the curves are imported as sketches or reference curve features.

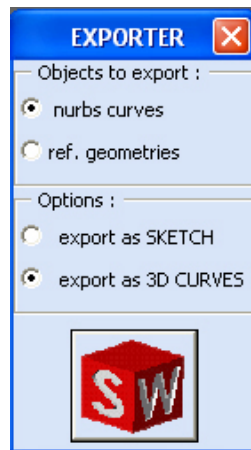


Figure 19: Interface for the selection and automatic export of data.

Finally, in the third part of the module, there is a single button (marked by the SolidWorks icon) which automatically activates the process of exporting data from RapidForm to SolidWorks. The implementation of this part of the interface, uses the RapidForm export functions, such as ExportCurves and ExportRefGeoms belonging to the RFDocument Object; and the SolidWorks import functions, such as LoadFile of SldWorks Object and the functions of the above mentioned ImportIgesData. For a correct functioning

of the interface, it is necessary to assure the right synchronization of the operations carried out by the two different software applications. Such a characteristic is implemented by using specific SldWorks Object functions. Through the RapidForm window, the user extracts, creates and selects data useful for the remodelling and, by using the implemented module, he/she obtains an export and immediate visualization of the data within the SolidWorks window (fig.20). This operation may be repeated every time the user needs to export data to SolidWorks and it also allows him/her to work exploiting functionalities and advantages of two software programs simultaneously.

Finally, a macro, implemented in SolidWorks, allows one to automatically load invariant geometries instantiated during the initial phases of the methodology. The macros use the SolidWorks API, and, in particular, the InsertPart of PartDoc Object routine.

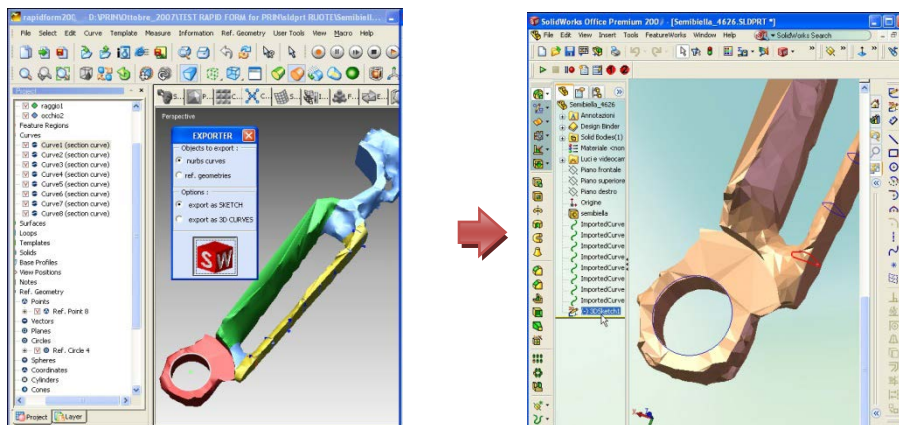


Figure 20: RapidForm and SolidWorks session works.

1.4. Case study: plastic moped wheel

A specific case study which concerns the design of a plastic moped wheel is used to test and validate the efficacy of the methodology. The implementation of the knowledge, in the procedure proposed as a test case, consists especially in:

1. identification of the parts of the moped wheel that can be considered “homogeneous” with reference to an optimization problem; in the case study the rim, the hub and the spoke are identified;
2. creation of a geometrical model DB for each of the three parts;
3. optimization analysis of spokes of the wheel;

4. extraction of profiles, guidelines, points and reference curves of the optimized model;
5. feature based modeling of the wheel.

1.4.1. Geometry Modeling:

The KB interface, implemented like a SolidWorks plug-in, supports, in an automatic manner, the generation of the three above-mentioned volumes, through a dialogue window. For this test case, we have developed one DB related to the plastic moped wheel.

The rim and the hub of the wheel have been identified as invariant volumes and the spokes as volume to be optimized. Consequently, a simplified DB related to the plastic moped wheel contains pre-defined geometries for each one of the “characteristic volumes” (fig.21b). All geometries in the DB are parametric. So, it is possible to modify the dimensions, according to specific needs

The designer can utilize this DB through the KB interface, implemented as a SolidWorks plug-in, and in an automatic way, the generation of the wheel by joining these volumes through a dialogue window (fig.21b). All geometries in the DB are parametric in order to modify the dimensions according to specific needs. It is always possible to sketch the wheel, taking care to model the wheel by union of the different volumes (fig.21a).

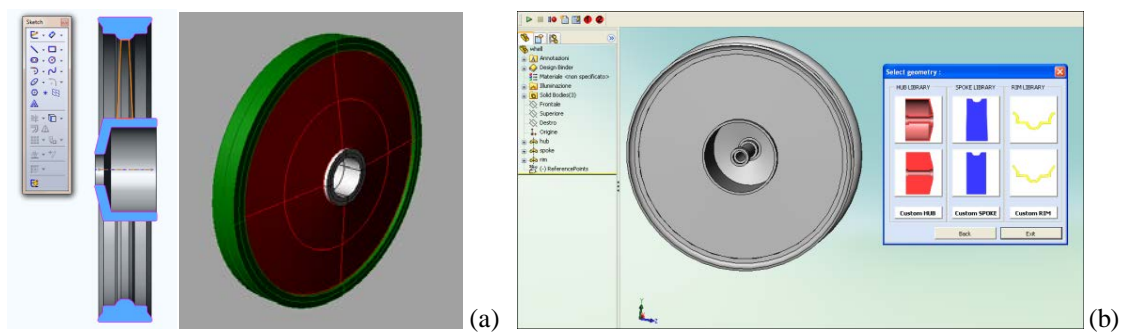


Figure 21: Interface for the creation of the optimization model.

By means of “*Selection of Invariant Geometries Interface*” (fig.22a), it is possible to make data management efficient. In fact, both the invariant volumes and the volume to be optimized are automatically stored in different files. The “*Export Interface*” (fig.22b) enables the user to export the geometry model into the CAE environment by means of a direct call to the optimization software (Altair HyperMesh).

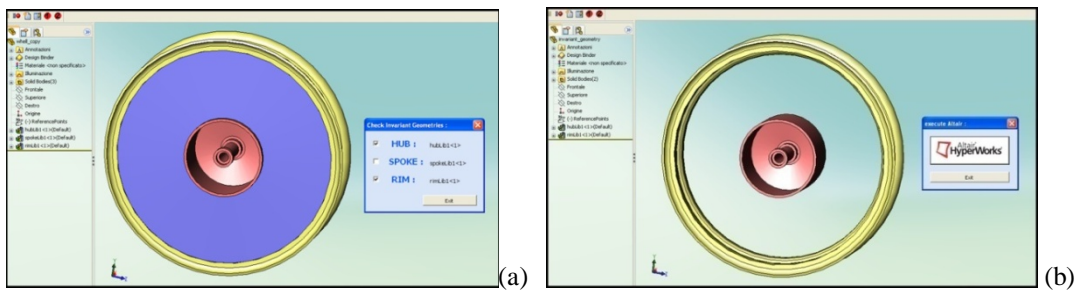


Figure 22: Creating the model and managing information through the implemented interfaces.

1.4.2. Mesh generation and model optimization

According to the test case, the optimization analysis of spoke contours is carried out with a concentrated force applied on the wheel as hypothesis (impact wheel simulation) and the mass minimization is the objective function. The hub and the rim are considered invariant volumes.

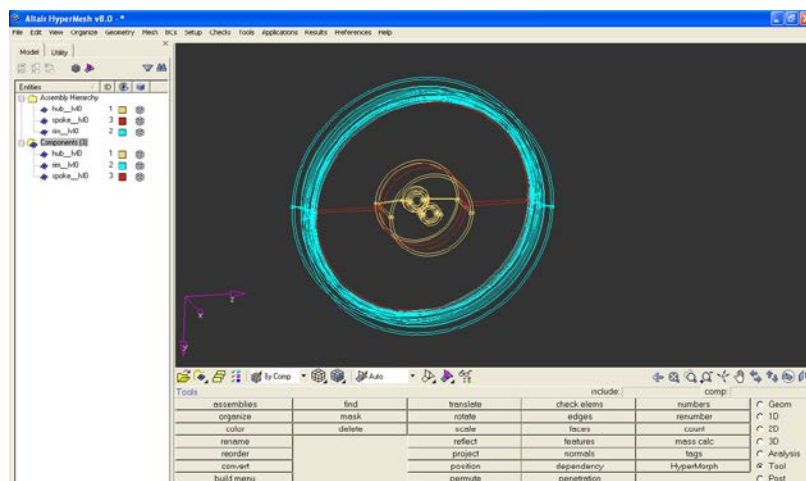


Figure 23: The model, imported in Optistruct, preserves the information about the variant and invariant volumes.

As stated before, the information about variant and invariant volumes has been defined during the geometry modeling phase, this kind of information are transferred automatically by the “*Export Interface*” in Altair and are very useful because the wheel is not imported as a single IGES file but is split in various IGES files, as depicted in figure 24, and each one keep the geometry of a specific variant/invariant volume. In a nutshell, the CAD automation tools developed support automatic data exchange and knowledge transfer between designers and analysts which work is simplified, especially during the generation

of the mesh, by the automatic splitting of the initial model in variant and invariant IGES files.

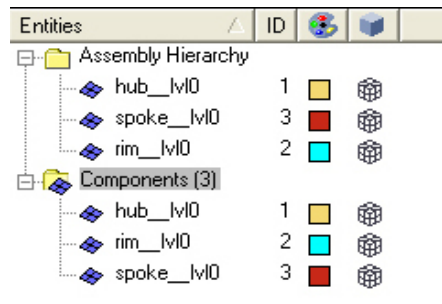


Figure 24: The wheel is imported in various IGES files, each one corresponding to a specific variant/invariant volume.

At the end of the optimization process, it is necessary to execute the OssMooth module, present in Hyper Works, in order to translate the topological optimization results into an STL format file.

1.4.3. Model checking

The STL model of the optimized wheel is formed by approximately 32,000 triangles, which leads to the inevitable presence of errors and faults in the triangles themselves. As we have already mentioned in the previous section, the model is loaded in RapidForm and through the implemented interface (fig.15) the user may carry out all the required functionalities for an optimal management of STL files. The case study involved:

1. a reduction of the number of triangles (from 32,000 to 20,000), in order to allow an easier importation of the model in a CAD environment;
2. a check for the individuation and rapid resolution of the defects of the triangles (approximately 200 defects eliminated);
3. an extraction of profiles and guidelines and instantiation of points and reference curves/planes (fig.25).

In order to increase the level of knowledge and extract further information (to be used during the final phase) from the optimized geometry, a geometrical analysis of the model may be particularly useful. In this way, it is possible to reconstruct the model shape in a

less subjective way. The following figures show how the optimized spokes of the wheel were analyzed through RapidForm.

In particular, the figure 25 shows: the result of the optimization on the spoke of the wheel; the profiles and the guidelines extracted from the optimized model; and, finally, the trend line of the area within the profiles according to their distance from the centre.

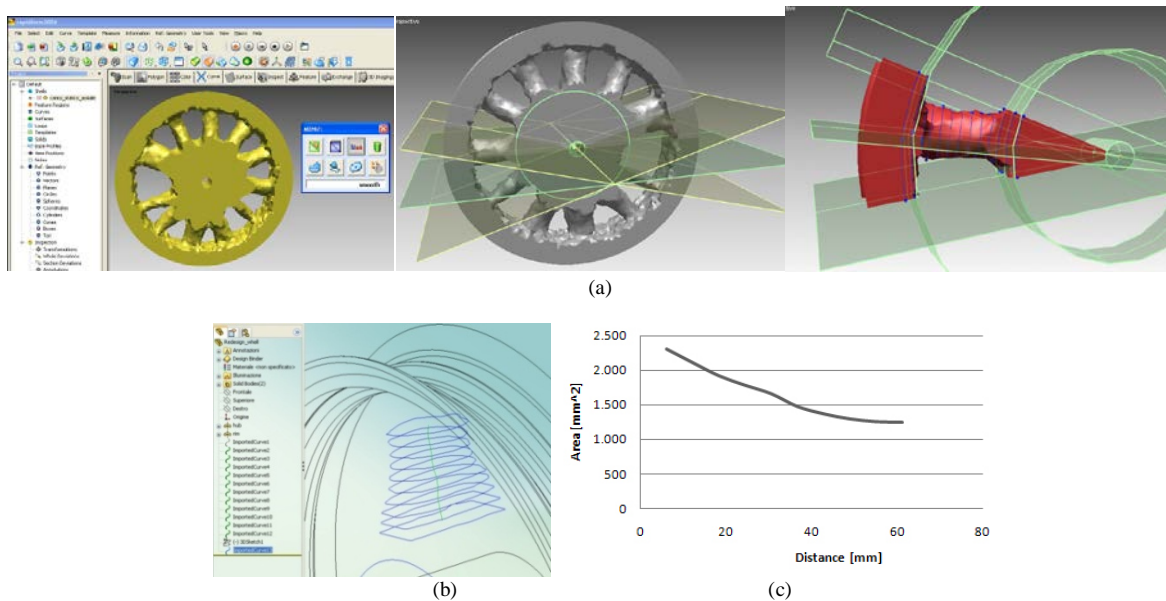


Figure 25: (a) Checking the optimized model and extracting reference planes and curves; (b) extraction of profiles; (c) data analysis (section area vs radial distance).

Figure 26 shows the trend lines of the distances from the centroid (of a generic section belonging to the spoke of the wheel) to a series of points (set on the profile itself and calculated every 10 degrees). Such a trend line is then compared to a circle with a radius equal to the average distance, and to an ellipse with a major and a minor axis equal respectively to the maximum and the minimum distance calculated into the section (fig.26a). Two cases were considered: the case called “*ellipse 1*”, where the major axis coincides with the major distance, and the case called “*ellipse 2*”, where the minor axis coincides with the minor distance (fig.26b). Figure 26c shows the graphs reporting such trend lines; the table reports a statistical dispersion index of data.

This information was very useful in the redesign of the wheel; in particular, a wheel has been identified with spokes with an elliptical profile (fig.26, case: “*ellipse 1*”), decreasing towards the outside of the wheel (fig.25c).

The possibilities to analyze the optimized volume are endless. The designer has to choose how and what he/she wants to investigate referring to a specific test case. The proposed improvements can be of great help in clarifying the problem and simplifying the modelling phase. It is possible to generalize the proposed methodology because of the capabilities of the implemented interfaces to manage a generic STL model.

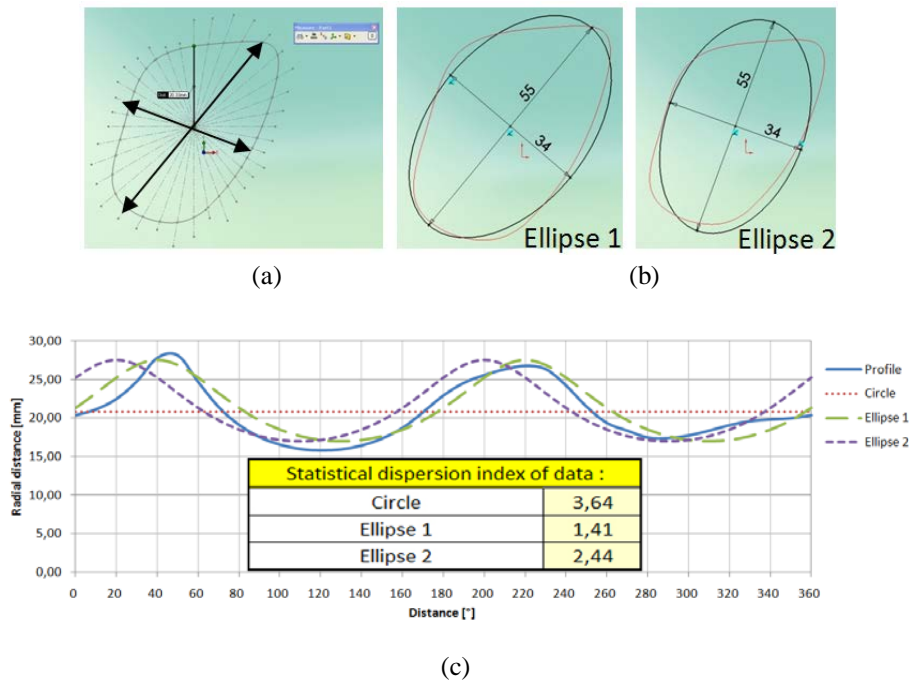


Figure 26: Analysis of optimized section. (a) Generic section with minor and maximum distances; (b) “ellipse 1” (its major axis coincides with the major distance), “ellipse 2” (its minor axis coincides with the minor distance); (c) graph and table of the data analyzed.

1.4.4. Feature based re-design:

The last step of the methodology requires the redesign of the optimized volumes, in order to obtain a feature based manufacturable product model. Both the invariant volumes and the data extracted from RapidForm simplify this operation. The export data interface allows a safe and rapid export of curves and profiles in SolidWorks (fig.27a). As shown in figure 27b, loaded in SolidWorks are the invariant geometries (defined during the initial phases) and ten elliptic profiles (defined during the analysis phase). Starting from these elements and considering eleven spokes (as suggested by TO results), the wheel is remodelled in a symmetric way (the symmetry imposition is driven by the knowledge and the experience of the designer). Figure 27c shows the final result of the methodology.

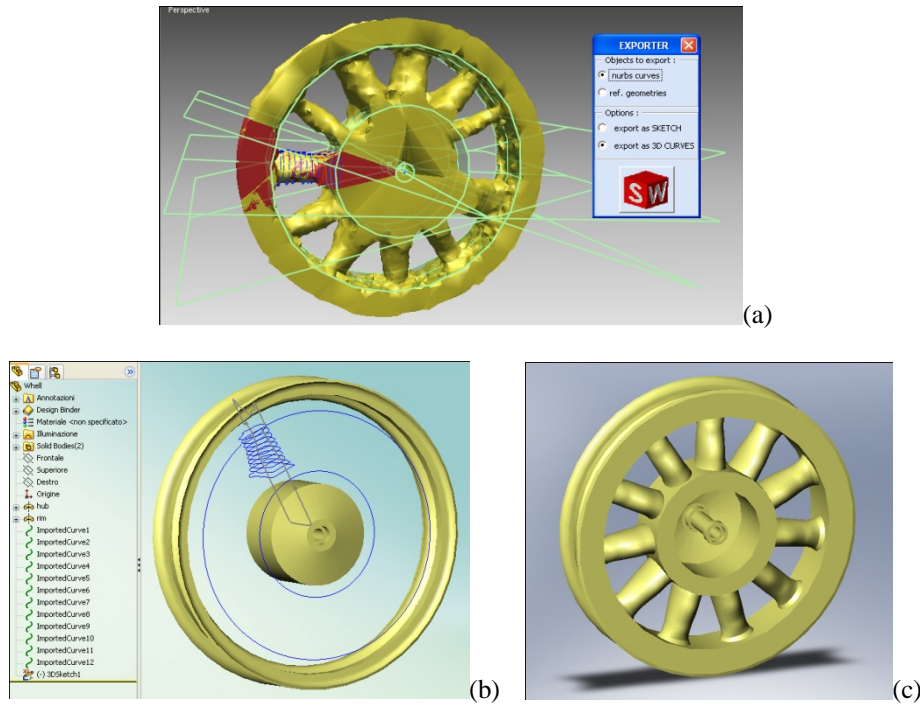


Figure 27: (a) Exporting data to SolidWorks; (b) invariant geometries, sections and profiles in CAD environment; (c) feature based modelling of the wheel.

1.5. Conclusion and synthesis of the needs

In this chapter a methodology has been described that clearly defines the roles, the activities, the data exchanged and the software used by designers and analysts when companies try to integrate TO tools in the PDP.

The research is not aimed at the creation of a fully automatic system for design embodiment, but provides a common base for knowledge sharing and for a better interconnection between different systems and applications. In particular, the CAD automation tools developed support automatic data exchange and knowledge transfer between designers and analysts, and define an innovative procedure to support the interpretation of the TO results.

The proposed methodology has been tested with the plastic moped wheel case study employing some custom KB interfaces implemented in SolidWorks and in RapidForm in order to allow an efficient interoperability between CAx systems and OptiStruct.

The methodology described in this first chapter has been implemented in order to overcome the main problems due to the lack of integration between TO tools and the other

CAX systems in particular with the CAD systems. In the next chapter improvements will be proposed in order to increase the level of integration with other CAX systems and, most of all, to reduce decision makings of the user.

Chapter 2

MB-TO-CAD systems integration

This chapter presents the enhancement of the methodology described in the section 1.3 with the introduction of a multi-body simulator in the design process. The introduction of MB systems allows to reduce decision makings and increase the objectivity of the proposed methodology because the results of MB analysis are useful information for the definition of the topological optimization problem. In synthesis the parameters for the set-up of the TO analysis are based on input generated using computational analysis rather than on estimations or on previous designs. Next the motivations and the state of the art written in section 2.1 and 2.2, a methodology in which a CAD, a multi-body simulator and a topological optimization tool are synergically employed in order to support the designer's job is described in section 2.3. In order to illustrate this capability, the process is applied to the conceptual design of the Upright for a Formula SAE prototype (section 2.4). The results show that the integrated design approach can efficiently support the selection of the optimum conceptual design of a mechanical component with complex dynamic behaviour, in particular when very little previous experience on the system is available. The last section (section 2.5) gathers together the results of the methodology developed in the first and second chapters and introduces the goal of the next part of the manuscript.

2.1. Extension of the methodology to Multi-Body systems

The first chapter deals with the main problems related to the introduction of topological optimization systems within the product development process. Once these problems have

been identified a methodology has been developed in order to provide the designer guide lines for an efficient use of TO tools with the other CAx systems. Where it was necessary custom user interfaces and cad automation modules have been implemented to simplify the designer job, reduce decision makings and, in particular, to catch and keep designer knowledge along the whole PDP in order to reuse this in subsequent design phases or future reworks. The methodology (section 1.3) has been tested with case studies in which the aim was the design of a single mechanical model without taking into account its interaction with other components. But mechanical engineers face one further and important requirement. Most components they design have to assemble with other components, they need to fit together. This means designers have to work with a package space within which the component needs to fit, and assembly points that cannot be varied since they are decided by other components. In mathematics, the package space is referred to as the design space or the optimization domain. Finally, engineers may not be allowed to change every possible parameter. For example, the loads, their values and directions, applied to a component may be restricted by factors beyond the users' control: the loads adopted in the design of a vehicle upright should be specified on three points where the upright is connected to the tie rod, the upper control arm and the lower control arm, respectively, also its dimensions depend by a packaging analysis. The parameters that analysts have the freedom to vary are called design variables. The dependence of the objective on the design variables is expressed as an equation, called the objective function.

The statement of the design optimization problem then, consists of the package space, the design variables, the constraints and the objectives. If the designer has any of these wrong, it's pretty likely his design proposals will be useless!

The next picture shows decision making activities for the methodology proposed in the first chapter. As stated before there are various decisions based upon the experience of the designer therefore the complexity of the design process increase.

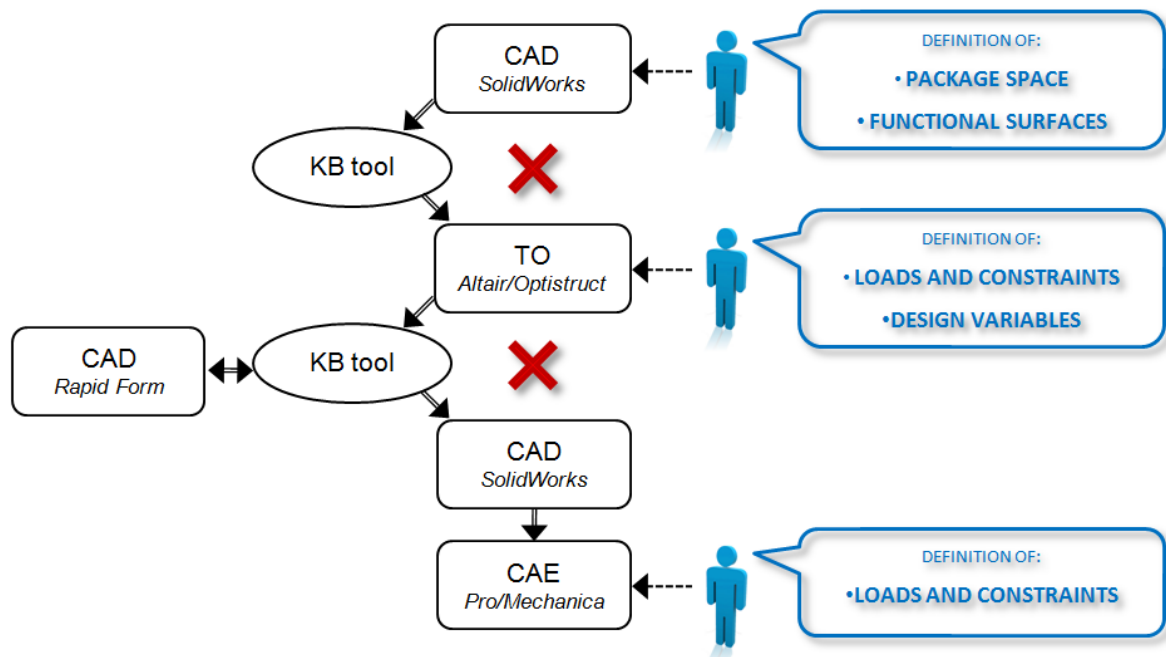


Figure 28: Decision makings activities for the proposed methodology.

From these considerations come out that an intelligent choice of the design variables, loads and constraints helps the optimizer find the solution faster if the user restrains himself from taking the brute force approach and leaving all decisions to the optimizer. However a wrong choice of design variables and constraints can completely destroy engineer's design effort, since it may result in no feasible solution, or, even worse, a wrong solution.

Then the aim of this chapter is the definition of an extended version of the methodology proposed in the first chapter in order to be an efficient support in the optimum design of a mechanical component with complex dynamic behaviour particularly when almost no experience on the system is available.

A solution to this problem can be found in the introduction in the proposed methodology of a multi-body simulator (MB) whose analysis' results provide the data and information necessary to chose and set, in a correct manner, the design variables of a topology optimization problem and the loads and constraints of a finite element analysis.

Hence, in order to increase the efficiency in the use of the TO within PDP, in this chapter a methodology has been developed in which CAD, a multi-body simulator and a topological optimization tool are sinergically employed. In order to illustrate this

capability, the process is applied to the conceptual design of the Upright for a Formula SAE prototype. In particular, for the correct formulation of the topological problem, the TO system (Altair Optistruct [AE 2010]) has been integrated with the Multi-Body (MB) system (MSC/ADAMS/Car [MSCN 2010]) in order to determine the topological information, and the loads and constraints related to the suspension components [Albers et al. 2002, 2007; Haussler et al. 2001; Milliken et al. 1995]. The topological information of these components are used to define, by the CAD system (SolidWorks [SW 2010]), the overall dimensions and the functional surfaces of the upright.

It is important to underline that, thanks to the integrated use of MB and TO systems, our approach allows to evaluate how much different topological optimizations (related to different boundary conditions) can influence the behavior of a complex system. In fact, while the MB system allows to determine, in different load conditions, the load distributions between the parts, on the other hand, the TO system allows to optimize every component for each different configuration. Thus, the designer can easily identify the worst case and determine the best (optimized) system configuration.

2.2. Related works

2.2.1. *Integration MBS-TO systems*

When only a limited experience about the analyzed system is available, the integrated approach of MBS and TO systems can be of major interest to support on one hand the formulation of realistic boundary and loading conditions, on the other the definition of optimum shape and topology. In the literature, many papers present various researches about coupling MBS with structural and topological optimization.

Albers et al. [Albers et al. 2002] describe the structural optimization of dynamically loaded flexible components modelled by finite elements, embedded in a multi-body system by means of an automated coupling of MSC.ADAMS with MSC.Nastran Sol200 as optimizer. Different analysis domains, namely hybrid multi-body dynamic systems (MBS), finite element analysis (FEA), control system simulation and topology optimization are integrated in a straightforward, automatic way in [Albers et al. 2007]. The approach is applied to the design of a humanoid lightweight robot. Cumnuantip et al. present an

improved approach for an interdisciplinary connection between MBS and, other CAE and TO tools [Cumnuantip et al. 2005]. In order to illustrate its capability, the process has been applied to the conceptual design of a landing gear system of the Blended Wing Body (BWB) aircraft. The integrated approach (MB and TO) applied to automotive components is described in [Chiandussi et al. 2005; Murali et al. 2006]. In [Chiandussi et al. 2004], the authors show the results obtained by using a topology optimization code to solve a three-dimensional problem concerning a real automotive component (McPherson rear suspension sub-frame). In [Murali et al. 2006], the authors describe the design of a lower control arm of a sport utility vehicle. The case study illustrates the complete process of topology and shape optimization to get the final design of the lower control arm.

2.3. Methodology

In this section, the proposed methodology has been developed with reference to the specific case study of the design of a suspension component. To better understand the new features introduced by this methodology, the traditional approach is first described and its critical points highlighted.

2.3.1. *Design of a suspension component: the Upright*

The design of an automotive component has always been a complex process strictly dependent on the functions that the component must perform within the vehicle system. Therefore the final solution is almost never obtained directly but it is rather the result of an iterated optimization process which incorporates the capacity, the intuition, the experience and the know-how of the designer [Smith 1978; Staniforth 2006; Milliken et al. 1995].

The development process of a vehicle (fig.29) consists of four main phases starting from the definition of the vehicle specifications and ending with the identification of the characteristics required to the singular component.



Figure 29: Vehicle development process.

In the first phase the vehicle concept is defined by identifying the characteristics and functionalities that the designer wants it to possess. In the next phase, the functional and performance macro-requirements, established in the previous phase, are objectified. In the third phase, the characteristics of the sub-systems are defined, taking into account the objective characteristics of the entire vehicle.

During the last phase the design process is utterly refined focusing on the single components of the different sub-systems. It is hence possible to define three design levels (component, sub-system, vehicle) for each of which an iterative design process is identified within an analogous higher level design process; every level sends inputs to and receives feedbacks from the upper level (fig.30).

The design of the upright is performed in the design process of the suspension subsystem and, indirectly, in the design process of the whole vehicle.

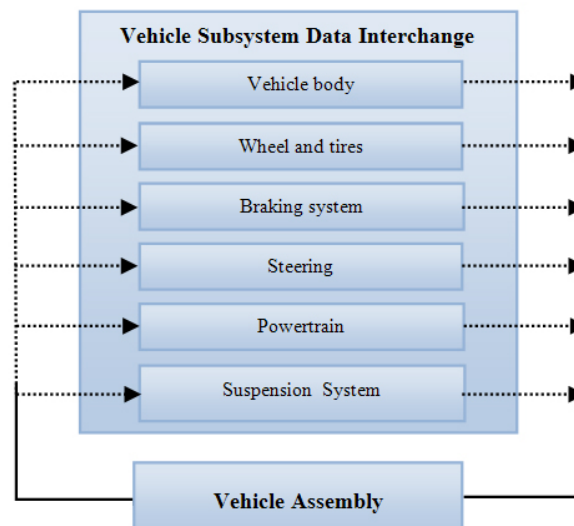


Figure 30: Chart flow subsystem design.

The identification of requirements and constraints for the upright design descends from the analysis of the dynamic behaviour of the whole vehicle. In particular, the handling

analysis allows to determine the Kinematic and Compliance (K&C) characteristics of the suspension that are the basis for the structural component design. Suspensions are generally modelled in multi-body environments through kinematic constraints and shapeless elements, depending on the appointed architecture.

The use of these environments enables the designer to submit the system to K&C and optimization analyses in order to find a topology of the system that satisfies both the kinematic requests and the component stiffness both related to the compliance required to the whole suspension.

The topological information are used in CAD systems in conjunction with the data inferred from multi-body kinematic analysis in order to carry out a packaging analysis. On the other hand, the definition of the geometry of the components leads the kinematic optimization in MB systems. Whereas, information concerning the component compliance are evaluated by FEM analyses when performing the structural test. Depending on the success or unsuccess of the structural test, these results are transferred to multi-body systems to either confirm or correct the values of the compliance. Data exchange occurs between CAD and FEM systems: the geometrical modelling systems allow to define a geometry of the components compatible with packaging analysis, while structural simulators pick out a geometry that satisfies structural requirements. This iterative data exchange between MBS, FEM and CAD systems, well supported by current tools, enables the optimal solution to be found for the entire system (fig.31).

Design requirements and constraints of each component ensue from the analysis of the specific vehicle sub-system to which the component belongs. Depending on the functionalities that the upright has to execute in the suspension sub-system, the requested properties are:

- an assembling geometry and a kinematic behaviour into a predefined volume;
- specific compliance;
- light-weight structure.

The upright compliance requirement is defined by means of the compliance analysis of the suspension sub-system (the suspension must undergo a given deflection under specified loads).

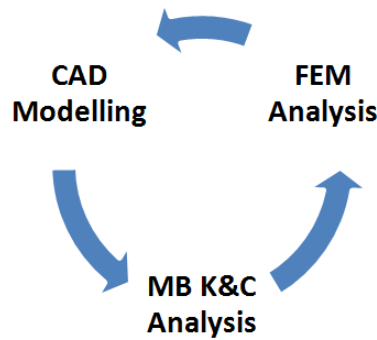


Figure 31: Tools for “traditional” approach of vehicle suspension design.

The kinematic and packaging analyses allow to determine the volume necessary to achieve optimal kinematics, functional surfaces essential for assembly, and overall dimensions such to avoid interferences during the dry run. In this way it is possible to specify geometric constraints that determine the maximum dimensions and the initial shape of the component.

Usually the material of the component is defined a priori on the basis of know-how, survey of competitors and economic valuations [Erickson et al. 2004; Jawad et al. 2002a, 2002b]. Similar considerations allow to choose an appropriate factor of safety. The loads, i.e. their values and directions, are the last characteristic to be determined. The final optimal result will be a component with the minimum mass that satisfies the specified requirements.

Establishing loads and optimal geometry is the critical point of this approach. In fact, without a physical prototype it is necessary to calculate accurately the normal operation loads. As a consequence, a robust virtual model must be implemented in the multi-body system in order to simulate the normal operation of the suspension. Concerning the structural optimization, the traditional trial and error method enables the designer to make subsequent and iterative modifications motivated by geometry, stiffness and safety requirements.

The rate of convergence and the convergence itself depend strictly on the designer experience; the time could be very long; and the result could not be the best one.

2.3.2. MB-TO-CAD integration

The introduction of TO systems during the design process of a suspension component allows to overcome the crucial point about optimal geometry definition. On the other hand,

this introduction requires to redefine the design methodology because it makes more complicate the communication between tools.

As already pointed out, the introduction of TO systems in PDP stresses three aspects of the process: the first aspect concerns the definition of the problem to be optimized; the second the data exchange between TO and CAx systems; the third the interpretation of the results of the optimization, in order to realize the final CAD model.

In the proposed approach, CAD automation tools have been adopted to improve the integration and interoperability between MBS, CAD and TO system [Barbieri et al. 2010, 2009, 2008a, 2008b, 2007]. The scheme of the methodology is shown in figure 32.

The methodology integrates four different systems and supports the phases of data exchanging through KB tools.

Why the first KB tool? At the present, the lack of adequate tools for supporting the communication among CAD and TO systems forces the designer to generally operate in one of the two following ways. A possible procedure consists in realizing a geometrical model in a CAD environment and exporting it in a compatible format (generally IGES format) into the optimization code. This operative procedure, if ,on one hand, allows the user to exploit the advantages of the CAD systems for the construction of the geometry, on the other hand makes more difficult the definition of the so-called “*collectors*” which store the different properties of the material, the different load and constraint conditions, etc. Another important task of the collectors is the storage of the user defined subdivision of the component in “volumes to be optimized” (which will be subjected of the optimization process) and in “invariant volumes” (which will remain unchanged throughout the process).

The second procedure implies modelling the part within the topological optimization system by using the few modelling tools present in such systems. Following this approach, the modelling phase and the following phase of collector definition become more and more difficult as the complexity of the model and the number of components increase.

In the proposed methodology, the designer operates according to the first procedure supported by a KB tool. Through this interface, the designer can import/export the model from/to Altair Optistruct and manage the upright geometries as two different sets of volumes (invariant volumes and volumes to be optimized). In this way, it is possible to

simplify the collector definition phase and to keep track of these volumes throughout the entire optimization process in order to simplify the designer’s job in the redesign phase.

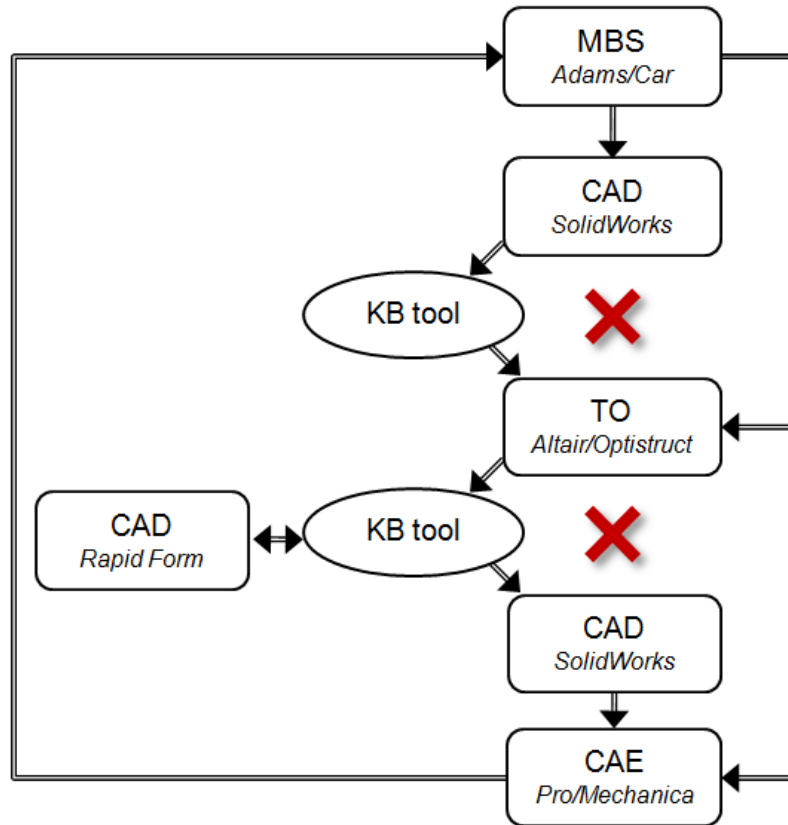


Figure 32: Proposed methodology.

This operative method involves the need to change the approach to 3D modelling. On one hand, in fact, the introduction of TO systems in the design process simplifies the modelling phase because it is no longer necessary to specify the geometry in detail, whereas it is sufficient to define the component overall dimensions. Moreover, the designer will model the component by differentiating the invariant volumes from the volumes to be optimized. Such modelling procedure makes the data transfer from the CAD to the TO environment much more effective because it simplifies the choice of the various volumes to which different collectors can be applied.

A second difficulty of the process, is encountered at the end of the optimization phase. One of the main problems concerns the voxel model generated by Altair. The format used is usually an STL format, which presents a high number of triangles with frequent errors on the triangles themselves such as non-manifold faces, redundant faces, crossing faces

and unstable faces. This model is usually unmanageable by CAD systems; the designer, at present, may only observe it because of the limited tools he/she possesses. To evaluate the optimization results by a FEM system, the designer must remodel the component without any support, interpreting subjectively the results.

The second KB interface (section 1.3.3) allows the designer to interpret and extract useful geometrical information from the results provided by the optimizer. In particular, this interface runs in RapidForm software which has a wide range of tools and specific functionalities to manipulate STL files. The designer can use the KB interface to reduce the number of triangles, in order to allow an efficient data exporting to the CAD environment. Moreover, the KB interface is able to identify and to extract, from the optimized volumes, all geometrical elements considered to be useful during the successive modelling phase (planes, axes, points, profiles, trajectories, etc.). According to the proposed methodology, during the remodelling phase, the designer is able to operate within a CAD environment using two different typologies of data: invariant geometries/volumes (obtained from the first KB interface) and reference geometries extracted from the optimized volumes. The availability of such data simplifies significantly the reconstruction of the feature based model, which will be further processed. In order to illustrate its capability, in the following section, the approach is applied to the design of a FSAE upright.

2.4. Case study: FSAE upright design

The vehicle taken as reference for the design of an upright is a prototype for Formula SAE competition realized in the University of Calabria.

Formula SAE® is a student design competition organized by SAE International. The concept behind Formula SAE is that a fictional manufacturing company has contracted a design team to develop a small Formula-style race car. Each student team designs, builds and tests a prototype race car based on a series of rules whose purpose is both to ensure onsite event operations and promote clever problem solving [FSAEI 2010].

2.4.1. Geometric problem definition

The 3D model and the physical prototype of the upright are shown in figure 33. The goal is the optimization of weight and stiffness.

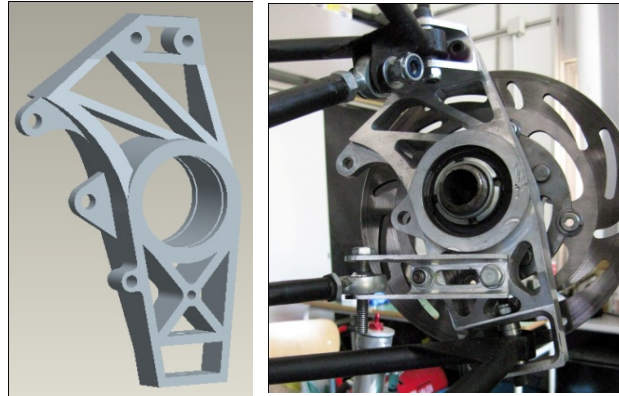


Figure 33: 3D model of the upright 2008 and physical prototype of the suspension.

Following the approach previously described, the multi-body models of the suspensions (and of the entire vehicle) have been realized by the commercial software MSC-ADAMS/Car (fig.34).

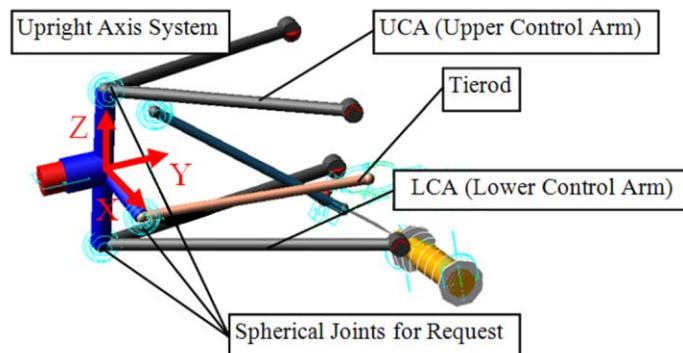


Figure 34: Multibody ADAMS/Car front-left suspension model.

In order to define the functional surfaces, all the components which must be directly assembled with the upright have been considered. The dimensions of the wheel have been defined in relation to the pneumatic characteristics. These dimensions allow to determine the volume into which the upright has to be positioned. The selection of the bearing for the coupling with the hub and of its blocking system has permitted to establish the dimension of the seat of the bearing at the centre of the upright. The rod end bearing for the coupling with the lower control arm has determined the diameter of the hole and the dimensions of

the slot located on the bottom part of the upright. The vertical distance between the opposite faces of the slot has been defined on the basis of the dimensions and of the mounting of the bearing; the space occupied during the kinematic movements of the suspension has dictated the horizontal distance between the opposite sides of the slot. The location of the hole has been established by kinematic considerations. The connection of the upright with the upper control arm has determined the position of two more holes. In fact, a separate component, joined to the upright by two screws, has been added to accommodate the bearing seat. A similar solution has been adopted to join the upright to the tie rod; the position of the holes for this connection has also been identified. The selection of the brake gripper has permitted to define the holes and the surface for its assembly.

The introduction of TO systems in the design process simplifies the modelling phase because the optimal geometry of the upright will be defined in the next TO phase. At this stage, the CAD upright model can be defined in a “rough” manner (fig.35a). The model must contain information about the overall dimensions and the “functional surfaces”, resulting from the packaging analysis, that will remain unchanged during optimization (fig. 35b).

To support these modelling operations, a KB interface (fig.12) has been implemented in SolidWorks which allows to specify the invariant geometrical elements (starting from the functional surfaces) and to store these entities and the model to be optimized in separate files [Barbieri et al. 2007; 2008a; 2008b]. Moreover, through this interface, the HyperMesh module can be directly executed in SolidWorks. This functionality is implemented in Visual Basic 6 by using API of SolidWorks; by means of Tcl/Tk script, it is possible to import automatically the model into Altair.

2.4.2. Determination of dynamic loads

The loads adopted in the design of the upright have been defined making reference to the worst conditions in relation to the maximum lateral and longitudinal accelerations recorded during standard manoeuvres in a Formula SAE race. Dealing with a frontal upright, braking and cornering simulations have been performed, considering the combination of the two as the worst operative condition [Jawad et al. 2002].

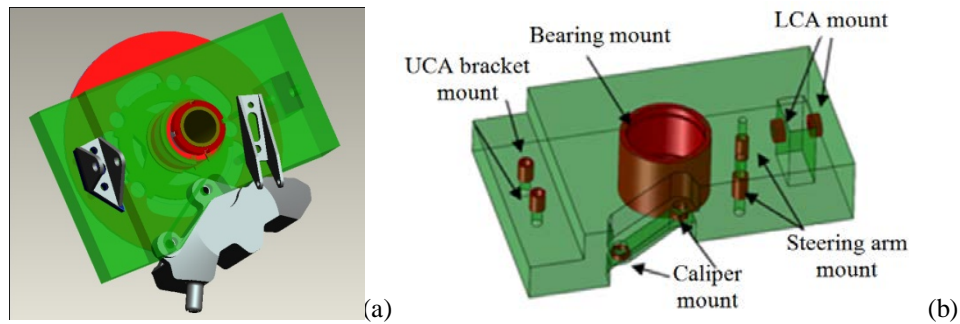


Figure 35: (a) Upright and related sub-assembly; (b) model to optimize and definition of invariant volumes.

The loads taken as references in the analysis refer to the mostly overloaded wheel. The loads are given through their components in the SAE coordinate reference system of the upright and they are specified on the three points where the upright is connected to the tie rod, the upper control arm and the lower control arm, respectively.

Where cornering was concerned, the model of the vehicle travelled along a 30 meters diameter circular trajectory, at a reduced longitudinal acceleration, starting from the first gear and changing up to the limit of the cornering. The braking manoeuvre started from a velocity of 100 Km/h, in fourth gear, jamming on the brakes until the vehicle came to a complete stop.

For sake of brevity, only some of the charts of the analysis results are presented on the following (fig.36-37):

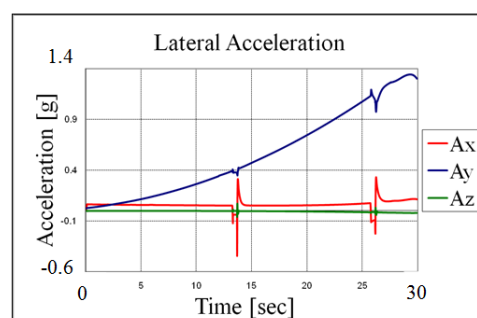


Figure 36: Chassis acceleration during cornering analysis (Vehicle Axis System).

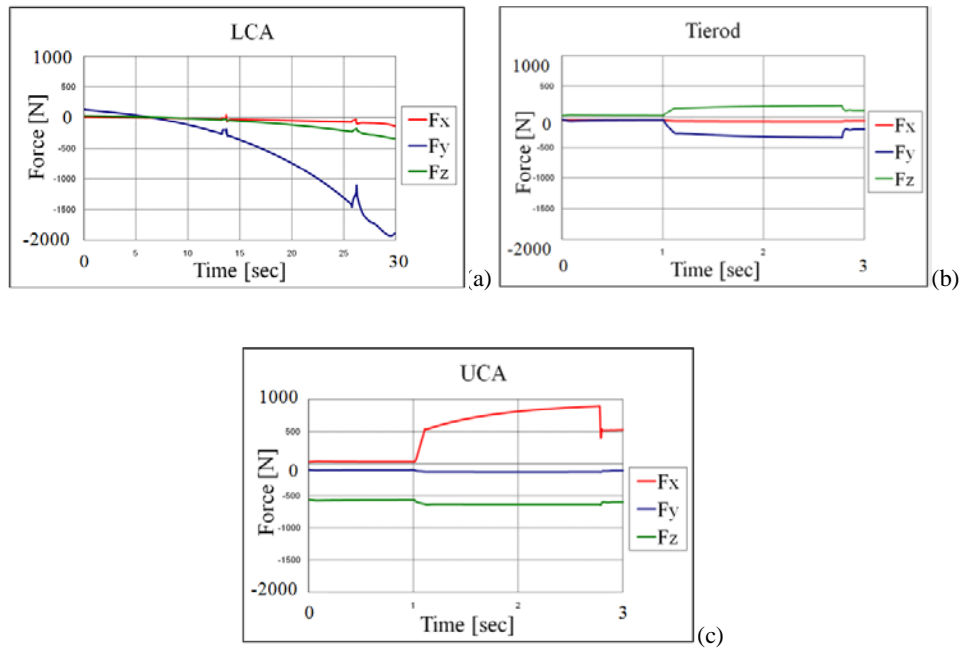


Figure 37: Time-history (a) LCA loads during cornering analysis, (b) Tie rod and (c) UCA loads during braking analysis (Upright Axis System).

The maximum upright load values are shown in table 1.

<i>Braking and cornering combination loads value</i>			
	Fx [N]	Fy [N]	Fz [N]
Upper Control Arm	936.6	1357.5	-765.7
Lower Control Arm	-1620.0	-1926.9	-341.3
Tierod Control Arm	153.1	-539.9	279.2

Table 1: Upright load values.

2.4.3. Topological optimization analysis

The topological optimization analysis has been performed using Altair Optistruct. Before starting the analysis, it is necessary to create the mesh of the model. Altair offers, in the Geometry panel, various tools that allow the designer to create the model but, as pointed out before, it is also possible to import the upright surfaces in IGES format. In this phase, the designer must simply define the collectors, i.e. declare the physical properties of the model and the boundary conditions identified during the multi-body analysis.

With regard to the material, Aluminum 7075-T6 was selected because of its excellent weight/strength ratio. The safety factor was chosen equal to 2.5 taking into consideration the contemporary presence of braking and cornering [Erickson et al. 2004; Jawad et al. 2002a; Jawad et al. 2002b].

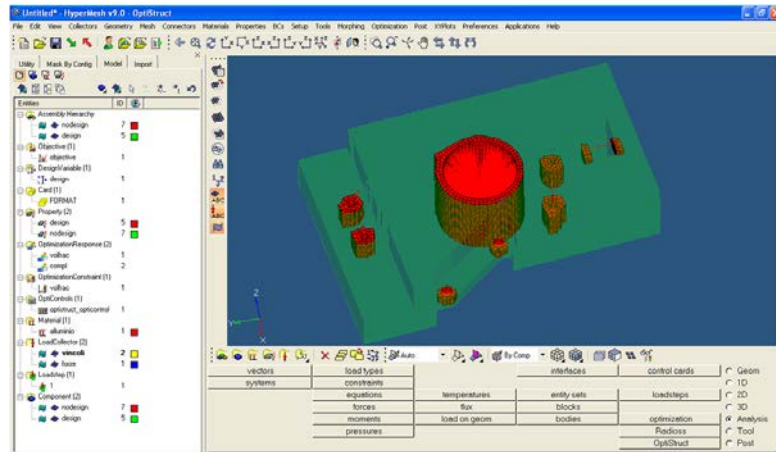


Figure 38: Functional surfaces translated in invariant volumes for the topological optimization analysis.

It is possible to create a collector for specifying the type of material, another for loads and constraints, other ones to define variant and/or invariant volumes. Importing invariant volumes from CAD system facilitates the operation of selection of hexahedral elements assigned to these collectors (fig.39).

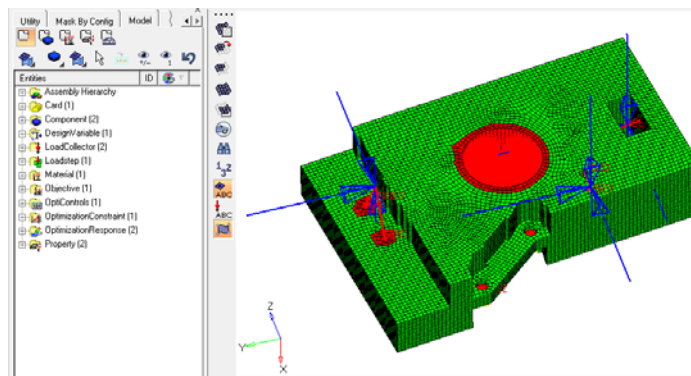


Figure 39: Mesh generation and assignment of loads and constraints.

The topological optimization problem is then defined as follows: minimize the volume of the upright.

At the end of the optimization process it is necessary to execute the OssMooth module, present in Altair Hyper Works, in order to translate the topological optimization results into an STL format file and export it in CAD environment (fig.40).

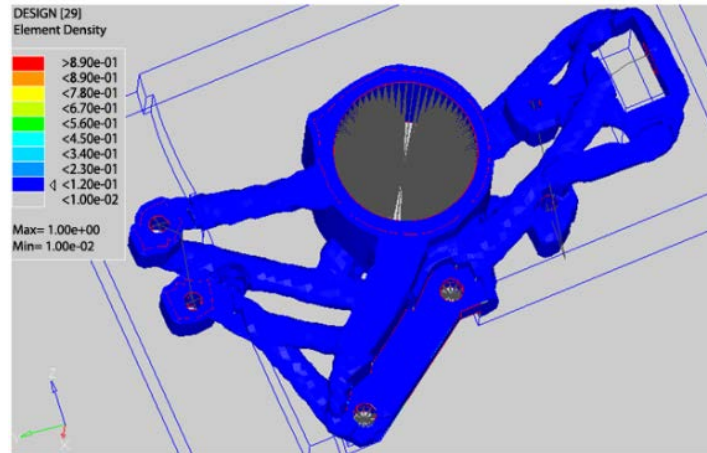


Figure 40: Topology optimization results.

2.4.4. *Model redesign*

As evidenced before, the topological optimization tools are not adequately integrated within the PDP especially with CAD systems. One of the main problems regards, in fact, the interpretation of the results generated by TO.

Through the second KB interface (fig.15), implemented in Rapid Form, the voxel model can be managed in order to reduce the number of triangles, and identify and solve problems concerning faulty triangles. The reduction of triangles and the correction of the faulty triangles have allowed the following model processing (fig.41).

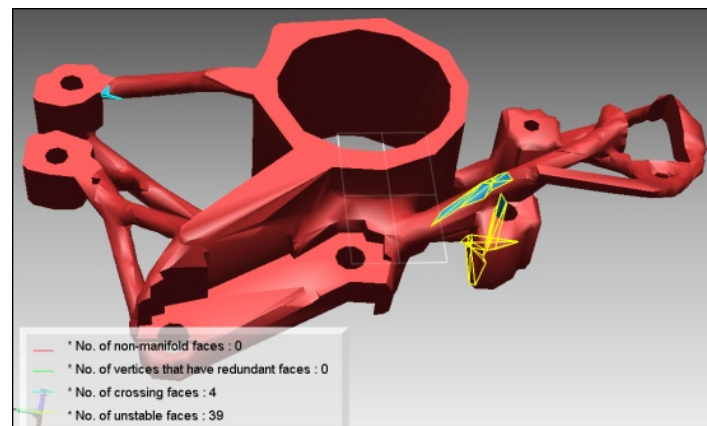


Figure 41: Individuation and resolution of errors in STL file.

At the end of the STL file cleaning, it is possible to extract, from the optimized model, projections, silhouettes and reference curves. The second KB interface, in fact, proposes a selection of RapidForm functionalities (fig.15) that are required for carrying out a model analysis.

For this specific case, in order to redesign the model, it has been considered more profitable to extract two orthogonal projections of the upright (fig.42). These two projections allow the designer to rapidly model the component taking into account the manufacture constraints (CNC machining).

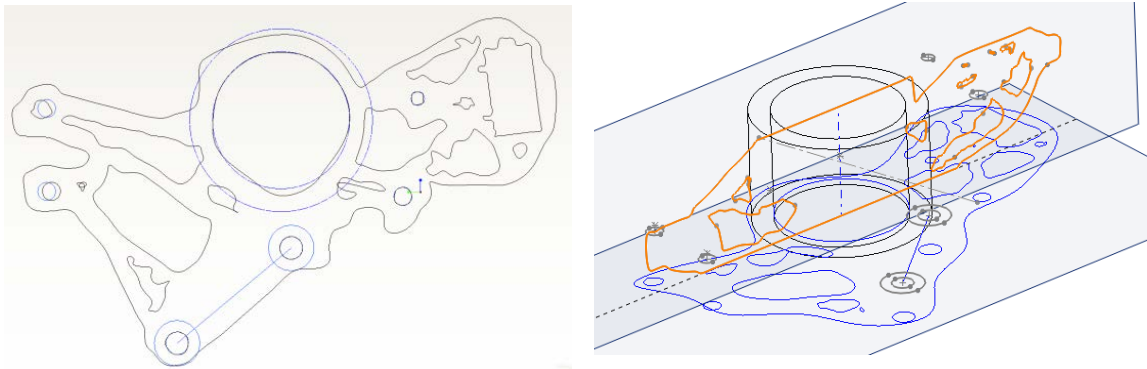


Figure 42: Optimized upright projections of the model.

This set of information and the geometrical elements defined (fig.43) are the essential knowledge base to realize an upright able to meet both design and manufacture requirements.

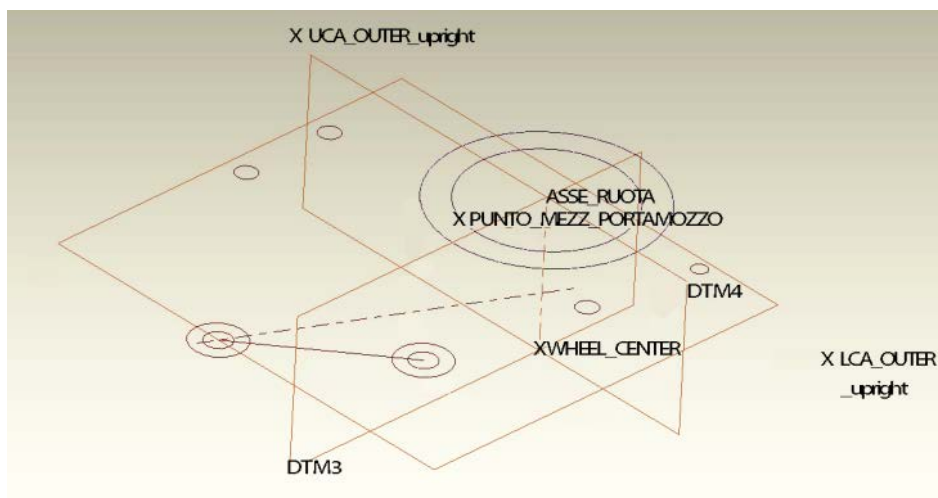


Figure 43: Reference geometries import.

The operations of data exporting may be carried out manually by the user; this would however require a good knowledge of functionalities and options of import/export routines of both software tools (Rapid Form and SolidWorks). Furthermore, in terms of the quality of information extracted from the optimized model, a large number of files will have to be managed, thus causing a great amount of time to be consumed. Such considerations led to realize a module implemented in RapidForm that allows to select reference geometries and export them interactively to SolidWorks. In order to realize such a module, the APIs of both software tools had to be used. The interface of the module is subdivided into three parts (fig.19), for more details read section 1.3.4. For a correct functioning of the interface, it is necessary to assure the right synchronization of the operations carried out by the two different software programs [Barbieri et al. 2008a].

The data exported from Rapid Form are the starting point of the modelling phase in SolidWorks. Figure 44 shows the result of this phase.

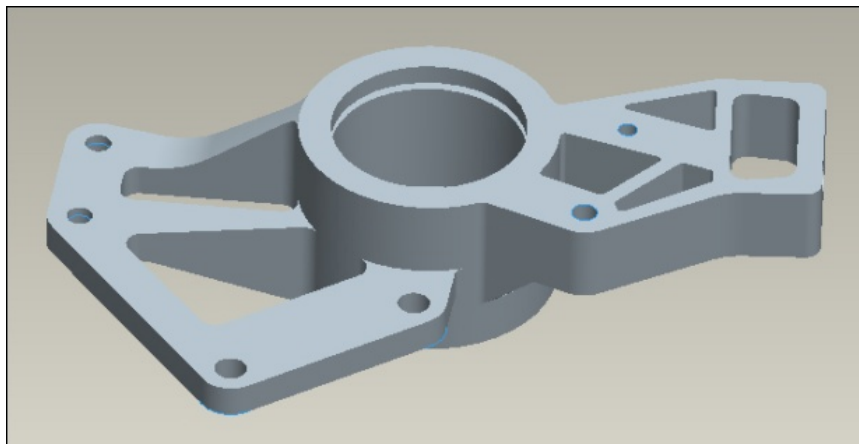


Figure 44: Final feature-based model.

2.4.5. *Finite element analysis*

In the last phase of the proposed methodology, a standard finite element analysis has been performed to test and validate the new shape of the optimized model (fig.45). The FEM code Pro/Mechanica by PTC was used. Loads and constraints were the same as those adopted during the optimization phase.

The comparison, between this model and the upright currently present in the FSAE prototype, shows a significant improvement in terms of weight and stiffness (Table 2).

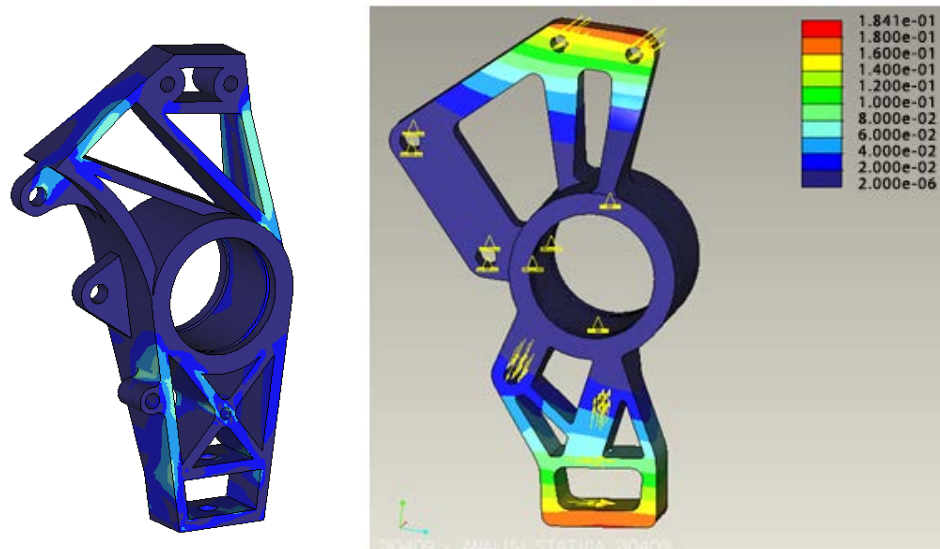


Figure 45: Comparison among FEM model of the upright 2008 and the new upright.

	Upright 2008	Upright 2009
Maximum Stress (Mpa)	94.2	168
Maximum Deflection (mm)	0.24	0.18
Weight (Kg)	0.966	0.721

Table 2: Data comparison.

2.5. Conclusions and perspectives

The introduction of TO systems during the design of a suspension component requires to redefine the design methodology because it makes more complicate the communication between tools. Starting from the literature and thanks to the matured experience in the PROSIT project [Cugini et al. 2008], some critical points have been individuated that currently limit such process reaching the following conclusions:

1. it is always possible to subdivide the geometric model into a set of “characteristic” volumes that can be considered invariant volumes or volumes to be optimized;
2. since the exchange format between CAD and TO tools is STL, there is no possibility to include in the model any metadata that can store the knowledge about the design choice, the history, the meaning of each part, etc.;

3. the invariant geometries cannot be modified during the optimization process. Maintaining the knowledge of these geometries across the optimization process simplifies the designer's job in the final modelling phase;
4. STL files storing optimized models, are usually not manageable by CAD systems because they contain a high number of faulty triangles;
5. it is impossible to use current feature recognition modules for the reconstruction of feature based models, starting from voxel models outgoing TOs;
6. it is more convenient to manage the TO results with the typical Reverse Engineering software rather than with a CAD system;
7. it is possible to extract - from voxel models - useful information on sections and/or profiles, which may be used during the feature based reconstruction phase.

At the present, neither commercial systems nor solutions proposed in literature can respond to all these questions. The proposed approach seeks to answer to the above considerations, going beyond the current state of the art. The manuscript proposes a methodology in which multi-body simulation, CAD and topological optimization are synergically employed to support the design of a suspension component.

The difficulties due to the poor integration of TO system in PDP have been overcome by implementing KB interfaces that support the designer in importing/exporting the model from/to TO, and facilitate the extraction of useful geometrical information from the results provided by the optimizer.

It is worth noticing that the aim of this research it's not the creation of a fully automatic system for design embodiment, but a common base for knowledge sharing and for a better interconnection between different systems and applications. The methodology has been extended to the individuation of the optimal design of a component part of an assembly, for the validation of the new version of the proposed approach the design of the Upright for a Formula SAE prototype has been adopted. The results have shown that the integrated design approach can be an efficient support in the optimum design of a mechanical component with complex dynamic behaviour, particularly when almost no experience on the system is available. The comparison between FSAE Upright version 2008 (designed by the traditional approach), and FSAE Upright version 2009 (designed by the proposed approach) shows a significant improvement in the mechanical properties.

As stated in the introduction the first two chapters are crucial since they define the methodology which is at the basis of the next improvements and extensions realized in the second part of the manuscript. At moment the case studies prove that the proposed methodology allow an efficient introduction of TOs in PDP but presents crucial points and limits that are a base of inspirations for following improvements.

- Part II -

Improvement of the methodology by introducing curve fitting tools and parametric archetype.

The second part goes deeper into the development of methods and tools that allows to overcome the limits and drawbacks of the methodology proposed in the first part of the manuscript. This part is structured into two chapters.

The third chapter is meant to provide an over-view of the mathematics that underlies the curve fitting technique implemented. This approach aims to translate the optimized profiles and curves into B-spline curves that could be used directly into a CAD system without users intervention. As described in detail in the chapter, the fitting tool has been implemented upon the mechanical model of bar networks that allows to develop a more flexible tool than the traditional least squared approach usually adopted for fitting problems.

The manuscript ends with the fourth chapter that extend the methodology, presented in the first part, with the introduction of the parametric archetype, that represent a base of knowledge that can be used in different stages of the methodology, but it is particularly useful for interpreting projections extracted from the optimized model.

Chapter 3

Constrained Force Density B-spline Curve Fitting implementation for the manipulation of the optimized profiles.

The chapter starts with the individuation of the drawbacks of the approach proposed in the first part of the manuscript that suggest the critical points on which is possible to operate in order to enhance and make more efficient and objective the full process. Then the chapter continues with the study and the implementation of a mathematical approach that allows to translate an optimized profile into a B-spline curve that could be directly imported into a CAD system. In particular, after a brief overview of the mechanical model of bar networks, the algorithm implemented for the curve fitting based on the force density method is depicted (section 3.2). After the state of the art described in section 3.3, in section 3.4 the methodology and the characteristics of the constrained force density fitting method are detailed. The last section (section 3.5) synthesizes the benefits and the limits of the proposed method.

3.1. Introduction

The methodology, proposed in the first part of the manuscript, provides a geometric analysis phase in order to support the designer during the interpretation of the geometric data extracted from the optimized model cause these data cannot be reutilized directly in a CAD system. These kind of analysis are very useful especially to infer graphs and diagrams for the evaluation of different sections or volumes and also of the trend lines of trajectories, profiles and guidelines. Then the outputs of this phase are information that should be reinterpreted by the engineer in order to redesign, with the sketch tools of a CAD

system, the final profiles and curves necessary to model the final 3D feature based component. As depicted in the next figure, the presence of the engineer necessary for the interpretation of information yield by the topology optimization analysis represents critical points of the proposed methodology (fig.10). In order to enhance the methodology is necessary to operate on these critical points and implement specific tools that allows to improve the efficiency and objectivity of the whole process.

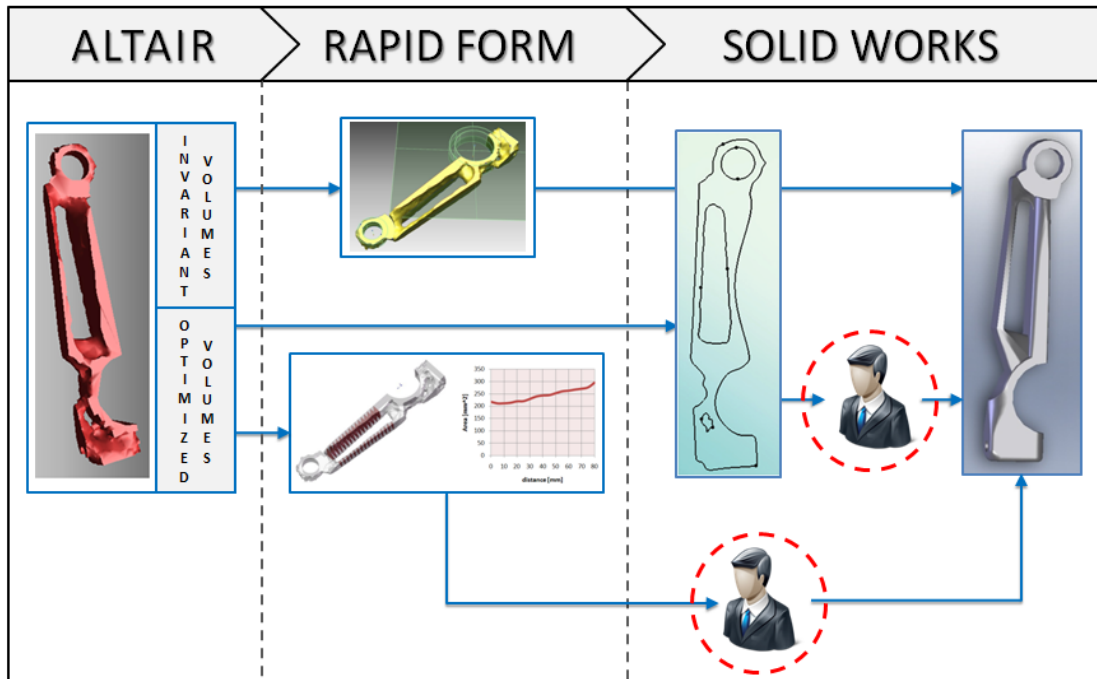


Figure 46: Critical points of the proposed methodology.

To account for these emerging needs the aim is the development of a tool that takes as input the curve and profiles extracted from the optimized model and give in output a B-spline curve that could be directly imported and manipulated into a CAD system. The translation of the optimized curve into the final B-spline curve occurs by fitting curve techniques. In particular, in order to improve the flexibility of the tool the fitting problem is not solved adopting common Lest Squares Method but the strategy relies on the mechanical model of the Force Density Method (FDM) introduced by Scheck [Scheck 1974] who describes the behaviour of a network with tensile bars (or springs). At this point, the geometric and mechanical problems are coupled. The novelty consists in the developing of a constrained fitting technique based on the Force Density Method.

To experiment with curves there are various commercial and free software [Blender 2010; Wings3D 2010; DesignMentor 2010; GIMP 2010] but those who wants can either write their own software (most likely in OpenGL). About the specific goal that this part of the manuscript intents on reaching, two powerful commercial software packages are available:

- Mathematica [Wolfram Research 2010] is the granddaddy of all mathematical software. It has facilities for numerical computations, symbolic manipulations, and graphics. It also has all the features of a very high-level programming language.
- Matlab (matrix lab) [Mathworks 2010] is a similar powerful package that many find easier to use.

For the implementation of the Constrained Force Density Fitting B-spline toolbox Matlab has been preferred for its user-friendly tools and especially for the great background and experience of the LSIS laboratory in which this research topic has been handle.

3.2. Development of the algorithm for Curve Fitting based on the Force Density Method (FDM)

Engineers often have the necessity to fit data using a model based on mathematical equations. In order to choose the right model one needs a good understanding of the physics, chemistry, electronics, and other properties of a problem. Then, at least some agreement is needed between the data and the model curve. If the model is supposed to represent exponential growth and the data are monotonically decreasing, the model is obviously wrong. To find the values of the model's parameters that yield the curve closest to the data points, one must define a function that measures the closeness between the data and the model. This function depends on the method used to do the fitting, and the function is typically chosen as the sum of the squares of the vertical deviations from each data point to the curve. However the true distance, rather than vertical distance, might have been used but it is more difficult to handle [Arlinghaus 1994].

There are two types of fitting, interpolation and approximation. Interpolation constructs a curve or surface which satisfies the given data precisely, e.g., the curve passes through the given points and assumes the given derivatives at the prescribed points. Approximation

entails curves and surfaces which do not necessarily satisfy the given data precisely, but only approximately. In approximation it is often desirable to specify a maximum bound on the deviation of the curve or surface from the given data, and to specify certain constraints, i.e., data which is to be satisfied precisely. Clearly, approximation is also more expansive than interpolation. The objective function to be minimized must measure the error in some way, e.g., least squares or maximum deviation [Laurent et al. 1993].

There are many subtleties in fitting, and literally hundreds of papers have been written on this topic. Many of the techniques are heuristic, and there are usually no unique or clear-cut "right" answers. A fundamental problem is that the given data never specifies a unique solution. How often has the implementor of a fitting algorithm been told by a designer using the software: "but that's not the curve I wanted!". And the reply is often: "well, it's mathematically correct; it satisfies the data you gave me!". [Piegl et al. 1997].

Input to a fitting problem generally consists of geometric data, such as points. Output is a curve or surface, i.e., control points, knots and weights. Furthermore, either the curve's degree must be input or the algorithm must select an appropriate degree. The system of equations, corresponding to the constraints (data points) specified by the user, is often under-constrained. It means that there is an infinite number of solutions and that an optimization problem has to be written with a functional to minimize.

The main theory behind curve fitting data revolves around minimizing the sum of the squares of the residuals (where the residual of a curve fit for each data point is the difference between the observed data point and the predicted value as given by the function of the curve). This approach is known as the method of the least squares. This method assumes that the measurement errors are independent and normally distributed with constant standard deviation. For linear functions, the solution for a best fit curve is a defined closed solution that can be directly solved. However, for non-linear functions, an iterative non-linear least squares approach is utilized to converge to the best fit curve. Several algorithms have been formulated to aid in converging the solution to non-linear curve fitting [Chernov et al. 2004a, 2004b; Yang et al. 2008].

In order to avoid these non-linear problems related to the adoption of the least squared method a solution could be found in the introduction of the force density method in fitting problems.

3.2.1. Force Density Method

In 1971 Linkwitz and Schek [Linkwitz et al. 1971] discovered a new formulation of the figure of equilibrium of forces, the force-density formulation. They realised that this was more appropriate for solving the problem, especially that of finding good initial geometry.

The properties of the Force Density Method were subsequently studied thoroughly by Schek [Schek 1974] and the method could be implemented in an efficient way by applying special sparse matrix techniques for solving the resulting equations. It proved to be a powerful tool for setting up and solving the static equilibrium state of any bar network. This method has been applied in various domains including lightweight tension and tensile integrity structures [Gründig 1985; Gründig et al. 2000], the optimization of tensile textile structures [Véron et al. 1998], the auto-adaptative meshing [Noel 1994], the correction of surfaces [Véron et al. 1997], the shape optimization according to mechanical criteria [Guillet et al. 1996] and finally to the deformation of free form surfaces [Guillet 1999; Guillet et al. 1998, 2000; Léon et al. 1995; Pernot et al. 2003, 2004a, 2004b].

The main advantages of the FDM are:

1. Any state of equilibrium of a general net structure can be obtained by the solution of one system of linear equations. This system is constructed using the force-length ratios or “force densities” in the branches as network description parameters. In other words one single force density is prescribed in order to obtain a unique result for the appropriate state of equilibrium.
2. It is possible to extend the linear approach to the non-linear force density method. The number of non-linear equations is identical with the number of additional conditions and is independent of the number of net nodes. Therefore this approach may be simpler and less expensive than the conventional solutions.
3. Another more theoretical advantage of the force densities as net description parameters lies in the fact that it is possible to reveal an interesting cross-connection: geometrical minimum way nets can easily be interpreted as pre-stressed nets and vice versa.

3.2.1.1. Mechanical model of a bar network

The FDM enables geometric manipulations of a bar network through the modification of external forces applied to its nodes. The position of the nodes is automatically updated to compensate the external forces variation and maintain the static equilibrium of the structure.

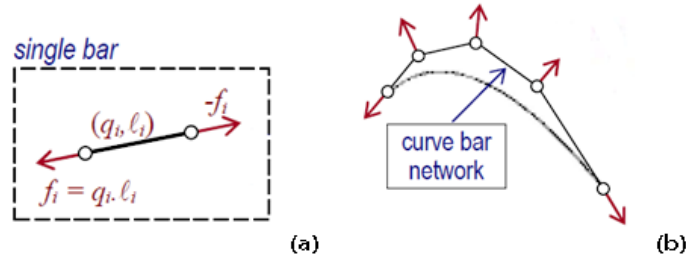


Figure 47: Parameters of a single bar's network (a) and example of a simple network (b).

Each bar can be seen as a spring with a null initial length and a stiffness q_i (more precisely a force density). To preserve the static equilibrium state of bars of length l_i , f_i external forces have to be applied to the endpoints of the bar: $f_i = q_i \cdot l_i$ (fig.47a). The set of external forces applied to the initial bar network can be then obtained through the static equilibrium equations at each node (fig.47b). Whether these external forces were not applied to the networks, all the nodes would be gathered together at a single point.

The shape of a network depends strongly on the given rule or list determining which nodes have to be connected by bars (springs). For this topological description the usual branch-node matrix [Argyris 1964; Fenves et al. 1963] is applied.

The connectivity of the r^{th} bar network can be expressed by a single matrix C^r of size $(N_b^r \times N_n^r)$ where N_n^r represents the number of nodes and N_b^r the number of bars. Each bar B_j^r of end points N_i^r and N_k^r ($i < k$) represents the j^{th} row of the C^r matrix, and each node N_m^r the m^{th} column. Thus, the matrices C^r can be filled according to the following rules:

$$c_{jm}^r = \begin{cases} 1 & \text{when } B_j^r \text{ is connected to the first node } (m = 1), \\ -1 & \text{when } B_j^r \text{ is connected to the second node } (m = k), \\ 0 & \text{otherwise} \end{cases} \quad (3.1)$$

Being δ_{ij} the Kroenecker's symbol and $q_j^r = f_j^r / l_j^r$ the force density into the j^{th} bar of length l_j^r , the force density matrix Q^r is defined such that $Q_{ij}^r = q_j^r \cdot \delta_{ij}$. The force density q_j^r is restricted to positive values ($q_j^r > 0$) to produce a network under tension everywhere

and ensure the robustness of the approach [Léon et al. 1995]. The following figure depicts a simple bar network and the corresponding connectivity matrix:

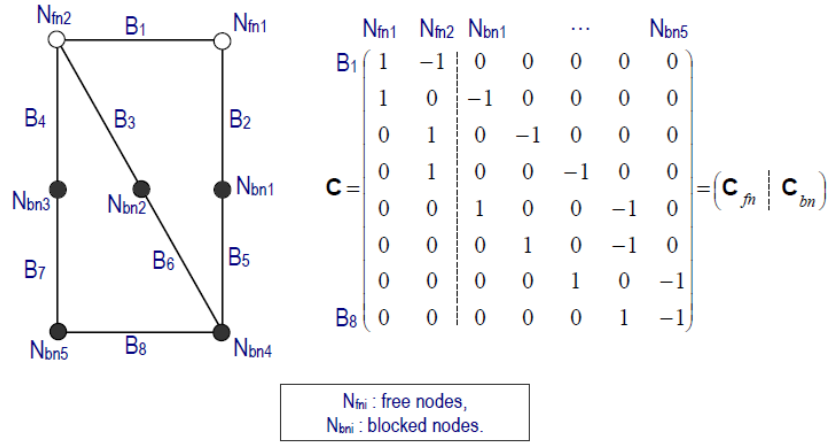


Figure 48: Decomposition of the branch-node matrix of a bar network composed of 7 nodes and 8 bars.

3.2.1.2. Bar network static equilibrium

Given \mathbf{x}^r , \mathbf{y}^r and \mathbf{z}^r , the three vectors containing the components of the 3D coordinates of the N_n^r nodes of the bar network coupled to the control vertices of the r^{th} geometric model, the f_x^r , f_y^r and f_z^r components of the external forces applied to these nodes can be obtained by using the following $(3 \times N_n^r)$ equations expressing the r^{th} bar network static equilibrium:

$$\begin{aligned} \mathbf{f}_x^r &= ({}^t C^r \cdot Q^r \cdot C^r) \cdot \mathbf{x}^r, \\ \mathbf{f}_y^r &= ({}^t C^r \cdot Q^r \cdot C^r) \cdot \mathbf{y}^r, \\ \mathbf{f}_z^r &= ({}^t C^r \cdot Q^r \cdot C^r) \cdot \mathbf{z}^r, \end{aligned} \quad (3.2)$$

The C_r matrices can be decomposed into two matrices $C_{fn}^r (N_b^r \times N_{fn}^r)$ and $C_{bn}^r (N_b^r \times N_{bn}^r)$ representing the connections related respectively to the N_{fn}^r free nodes and the N_{bn}^r blocked nodes (fig. ss). In the same way both external force and position vectors can be decomposed into ${}^t f_x^r = [{}^t f_{fn_x}^r, {}^t f_{bn_x}^r]$ and ${}^t \mathbf{x}^r = [{}^t x_{fn}^r, {}^t x_{bn}^r]$ for the x-components. By performing similar decompositions for the ${}^t f_y^r, {}^t f_z^r, {}^t \mathbf{y}^r$ and ${}^t \mathbf{z}^r$ vectors, force's equations can be written as:

$$\begin{aligned}
f_{fn_x}^r &= D_f^r \cdot x_{fn}^r + D_{bf}^r \cdot x_{bn}^r, \\
f_{fn_y}^r &= D_f^r \cdot y_{fn}^r + D_{bf}^r \cdot y_{bn}^r, \\
f_{fn_z}^r &= D_f^r \cdot z_{fn}^r + D_{bf}^r \cdot z_{bn}^r,
\end{aligned} \tag{3.3}$$

for the external forces applied to free nodes and as:

$$\begin{aligned}
f_{bn_x}^r &= {}^t D_{bf}^r \cdot x_{fn}^r + D_b^r \cdot x_{bn}^r, \\
f_{bn_y}^r &= {}^t D_{bf}^r \cdot y_{fn}^r + D_b^r \cdot y_{bn}^r, \\
f_{bn_z}^r &= {}^t D_{bf}^r \cdot z_{fn}^r + D_b^r \cdot z_{bn}^r,
\end{aligned} \tag{3.4}$$

for the external forces applied at blocked nodes. The different D_i^r matrices are obtained through the decompositions:

$$\begin{aligned}
[{}^t C^r Q^r C^r] &= [{}^t C_{fn}^r \quad {}^t C_{bn}^r] Q^r \begin{bmatrix} {}^t C_{fn}^r \\ {}^t C_{bn}^r \end{bmatrix} = \\
&= \begin{bmatrix} {}^t C_{fn}^r Q^r C_{fn}^r & {}^t C_{fn}^r Q^r C_{bn}^r \\ {}^t C_{bn}^r Q^r C_{fn}^r & {}^t C_{bn}^r Q^r C_{bn}^r \end{bmatrix} = \begin{bmatrix} D_f^r & D_{bf}^r \\ {}^t D_{bf}^r & D_b^r \end{bmatrix}
\end{aligned} \tag{3.5}$$

By summing up the equilibrium equations expressed at each free (3.3) and fixed (3.4) node, a global equilibrium condition of the r^{th} network can be formulated with the three following scalar equations:

$$\sum_{i=1}^{N_{fn}^r} \underline{f}_{fn_i}^r + \sum_{j=1}^{N_{bn}^r} \underline{f}_{bn_j}^r = \underline{0}, \tag{3.6}$$

where $\underline{f}_{fn_i}^r$ and $\underline{f}_{bn_j}^r$ represents respectively the external force applied to the i^{th} free node and the external forces applied to the j^{th} blocked node. Of course, this equation is a necessary but not sufficient condition of static equilibrium.

Conversely, being given a set of external forces applied to the nodes of the r^{th} bar network, the position of the free nodes are given by:

$$\begin{aligned}
x_{fn}^r &= (D_f^r)^{-1} \cdot (f_{fn_x}^r - D_{bf}^r \cdot x_{bn}^r), \\
y_{fn}^r &= (D_f^r)^{-1} \cdot (f_{fn_y}^r - D_{bf}^r \cdot y_{bn}^r), \\
z_{fn}^r &= (D_f^r)^{-1} \cdot (f_{fn_z}^r - D_{bf}^r \cdot z_{bn}^r),
\end{aligned} \tag{3.7}$$

These last equations show how it is possible to manipulate indirectly the node positions through the manipulation of external forces applied to that nodes. The unknowns of the deformation process are not anymore the positions but the external forces themselves. The

force densities are constant during the modification and the different D_i^r and $(D_f^r)^{-1}$ matrices stay constant during this process. One can notice that the force densities q_j^r could also be the unknowns used to access a new static equilibrium state. In that case, the external forces would be constant and the D_i^r and $(D_f^r)^{-1}$ matrices will vary during the possibly iterative resolution process thus requiring further computations slowing down the hybrid geometry manipulations.

Even if only the external forces applied at the free nodes are necessary to compute the new free nodes positions, some external forces applied to the blocked nodes connected to at least one free bar, i.e. a bar that changes of length during the process, may vary during the deformation process and can therefore take part to the definition of the objective function to be minimized. They are linearly dependent on the external forces applied at free nodes and can be obtained using equations (3.4) and (3.7):

$$\begin{aligned}
 f_{bn_x}^r &= {}^tD_{bf}^r \cdot (D_f^r)^{-1} \cdot f_{fn_x}^r + \left(D_b^r - {}^tD_{bf}^r \cdot (D_f^r)^{-1} \cdot D_{bf}^r \right) \cdot x_{bn}^r, \\
 f_{bn_y}^r &= {}^tD_{bf}^r \cdot (D_f^r)^{-1} \cdot f_{fn_y}^r + \left(D_b^r - {}^tD_{bf}^r \cdot (D_f^r)^{-1} \cdot D_{bf}^r \right) \cdot y_{bn}^r, \\
 f_{bn_z}^r &= {}^tD_{bf}^r \cdot (D_f^r)^{-1} \cdot f_{fn_z}^r + \left(D_b^r - {}^tD_{bf}^r \cdot (D_f^r)^{-1} \cdot D_{bf}^r \right) \cdot z_{bn}^r.
 \end{aligned} \tag{3.8}$$

3.2.2. Definition of the network

Given a pin-joint network consisting of springs or bars, the first problem is the definition of the network considering all possible shapes of equilibrium which the network can attain. Initially the possible network forms are not constrained by any geometrical restrictions. Only one important requirement has been imposed: the shape shall be an equilibrium state, this means that the sum of all forces is zero at each node.

In order to define the network for the fitting problem two main problems have to be considered:

1. individuation of points (network's nodes) in which calculate the equilibrium state, that are also the points on which insert the springs;
2. disposition and number of spring to insert among nodes and initial points to fit.

As suggested by C.H. Thornton and C. Birnstiel [Thornton et al. 1967] the computation of the general initial network has defined by means of trial and error methods.

For what concern the first problem the most appropriate solution is that both the initial points and points defined directly on the curve are the nodes on which construct the network. In particular, as showed by the simple example depicted in the next figure, except for the first and last initial points for each ones there is a correspondent point on the curve.

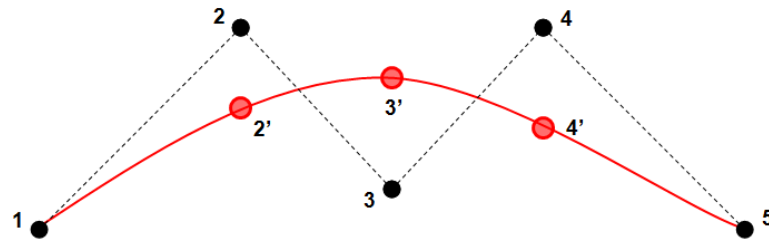


Figure 49: Definition of the nodes for the construction of the network.

One of the solutions taken in consideration was the choice of the control points of the curve as nodes, but in this case the fitted curve could not never interpolate the initial points and, most of all, the definition of the network is really hard when the number of the control points is smaller than the number of the points to fit.

Then one of the advantages on apply the nodes directly on the curve is that the network is independent from the number of the control points of the final fitted curve, in a nutshell, for m initial points Q_k there are $2(m - 1)$ nodes on which construct the network.

About the second problem, various and different layouts of the springs have been tested. The following figures show some of these tests: in the first figure the springs are linked only between the initial points and the nodes on the curve then for m initial points there are $m - 2$ springs, in this case the approach is very similar to the Least Squared Method with the advantage that the equations of the equilibrium calculated on each nodes, by means of the Force Density Method, yields always linear equations. In the second figure the network has been realized by the use of two kind of springs:

- link between an initial point $(1, 2 \dots m)$ and its correspondent node on the curve $(2', 3' \dots m - 1)$.
- link between two consecutive nodes on the curve: $x'_i x'_{i+1}$ for $1 \leq i \leq m - 1$.

This kind of network yields $2m - 3$ springs.

The figure 50c depicts a network composed by the springs that link two nodes on the curve $x'_i x'_{i+1}$ for $1 \leq i \leq m - 1$, and the other ones are due to the links of each initial point with three nodes on the curve, this connection follows the subsequent rule: *each x_{i+1} is linked with x'_i, x'_{i+1}, x'_{i+2} for $1 \leq i \leq m - 2$* . Then, on the whole, there are $3(m - 2)$ springs.

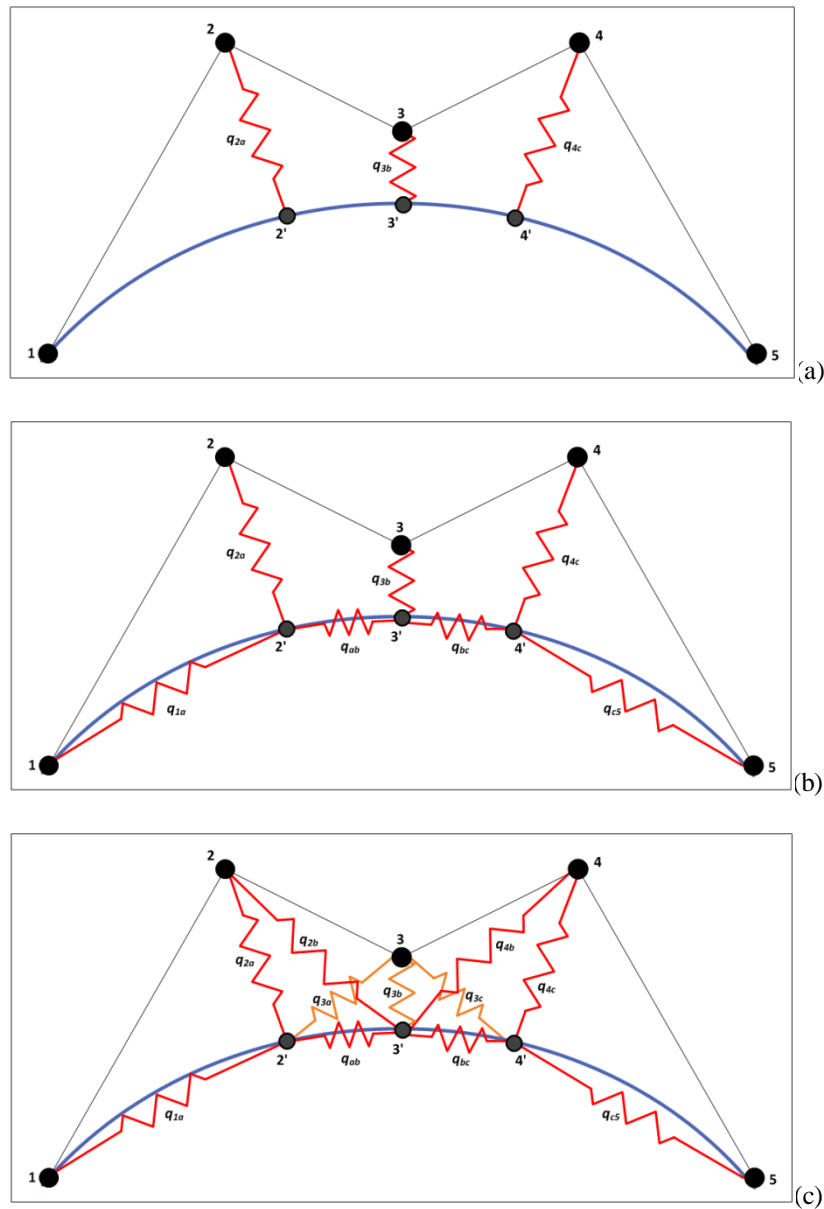


Figure 50: Different layout of the network with (a) $m-2$ springs, (b) $2m-3$ springs and (c) $3(m-2)$ springs.

The network adopted for the force density fitting problem is depicted in the next figure:

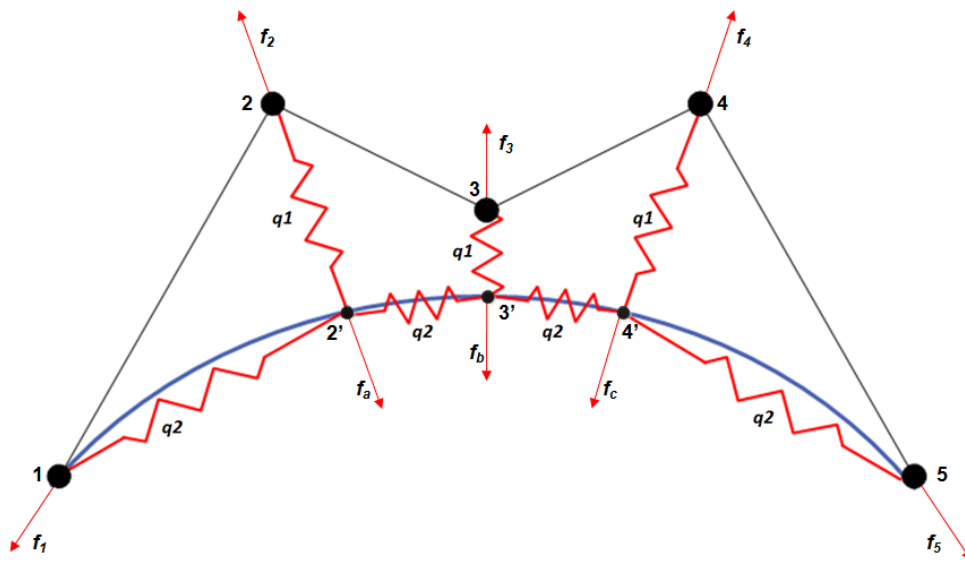


Figure 51: Final network defined for the Force Density Fitting.

Where 1 to 5 are the Q_k points to fit, $2'$, $3'$ and $4'$ are the nodes located directly on the curve. There are totally $2m - 3$ springs that can be distinguished in those that link two consecutive nodes on the curve with the same force density q_2 , and the other ones, with force density q_1 , that link an initial point and the correspondent node on the curve.

The force densities q_i are parameters that influence the shape of the final fitted B-spline, as example the next figures represent the same B-spline with different force density values:

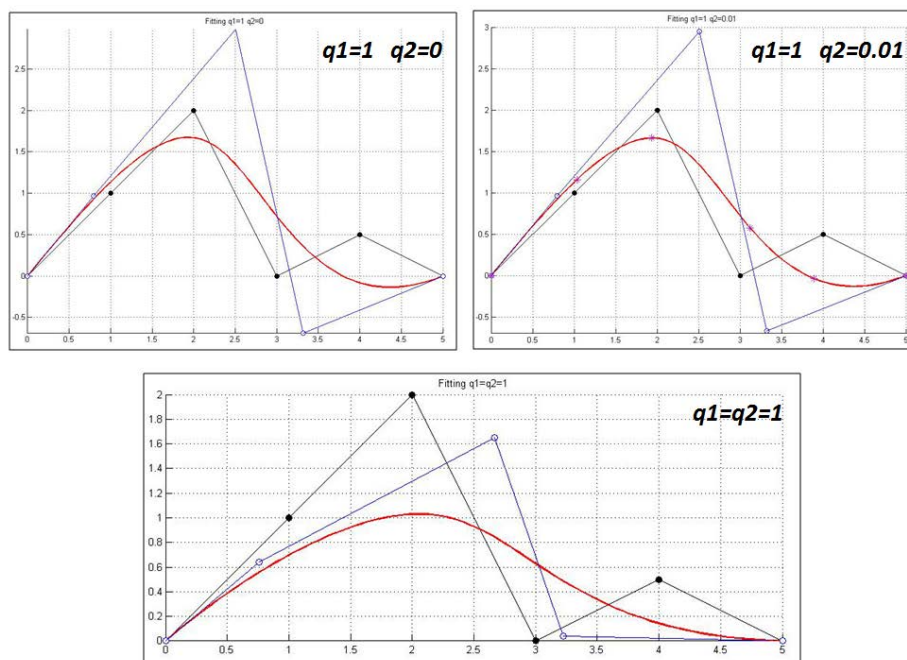


Figure 52: Comparison among fitting curve with different force density values.

The preference of this network compared to the other ones comes out evident if the characteristics of this network, as explained in section 3.2.1.1, are calculated. Ad example, the connectivity matrix for the network of the figure 51 is:

$$C = (springs \times nodes) = (springs_{P-B} + springs_{B-B}) \times (nodes_B + nodes_F)$$

Where:

- $springs_{P-B} = S_{P-B} = m - 2$ are the springs between the initial points and the correspondent point on the B-spline curve;
- $springs_{B-B} = S_{B-B} = m - 1$ are the springs between consecutive points on the B-spline curve;
- $nodes_B = N_B = m$ are the blocked nodes;
- $nodes_F = N_F = m - 2$ are the free nodes.

then the connectivity matrix $C = (springs \times nodes) = ((2m - 3) \times (2m - 2))$ is:

$$C = \begin{matrix} & N_1 & N_2 & N_3 & N_4 & N_5 & N_{2'} & N_{3'} & N_{4'} \\ \begin{matrix} S_{2-2'} \\ S_{3-3'} \\ S_{4-4'} \\ S_{1-2'} \\ S_{2'-3'} \\ S_{3'-4'} \\ S_{4'-5} \end{matrix} & \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \end{bmatrix} & = & \begin{bmatrix} A_1(m-2 \times m) & A_2(m-2 \times m-2) \\ A_3(m-1 \times m) & A_4(m-1 \times m-2) \end{bmatrix} \end{matrix}$$

in particular, if $A_3 = A_4 = 0$ and $q_1 = 1$ this approach give the same results of the Least Squared Method. Then the great flexibility of this method, due to the various parameters that can be manipulated, contains also the solutions obtain by means of the Least Squared Method.

Obviously the equilibrium is calculated only on the free nodes, then, for a fitting problem with m points to fit, $m - 2$ equations will be calculated in order to compose the equation of the objective function to minimize.

3.2.3. B-spline Force Density Fitting Curve

This section describes in details the mathematical procedure defined for finding the best-fitting curve to a given set of points by minimizing an objective function. Among the different curves that can be adopted the choice has been for B-spline curves that require more information and a more complex theory than Bézier and Spline curves but they present various advantages:

- a B-spline curve can be a Bézier curve;
- B-spline curves satisfy all important properties that Bézier curves have and provide more control flexibility;
- it is possible change the position of a control point without globally changing the shape of the whole curve.

Consequently, some skills on B-spline is required to understand the rest of the manuscript. The B-spline curve is an approximating curve and is therefore defined by control points. However, in addition to the control points, the user has to specify the values of certain quantities called knots. They are real numbers that offer additional control over the shape of the curve. Then a B-spline curve of degree p ($p+1$ is the order of the b-spline) is defined by:

- n control points: $p_0 \dots p_{n-1}$
- $n+p+1$ number of knots

Letting $B(u)$ be the position vector along the curve as a function of the parameter u , a

B-spline curve is given by:

$$B(u) = \sum_{i=0}^{n-1} N_{i,p}(u) \cdot P_i \quad u_{min} \leq u < u_{max} \quad 1 \leq p \leq n - 1$$

The theory for B-splines was first suggested by Schoenberg [Schoenberg 1946]. A recursive definition useful for numerical computation was independently discovered by Cox [Cox 1971] and by de Boor [De Boor 1972], in particular de Boor's algorithm [De Boor 1972, 1978] is a generalization of the de Casteljaou's algorithm [de Casteljaou 1986] for Bézier curves. Riesenfeld [Riesenfeld 1973] and Gordon and Riesenfeld [Gordon et al. 1974] applied the B-spline basis to curve definition.

The basis functions $N_{i,p}(u)$ are defined by the Cox-de Boor recursion formulas on $[u_i, u_{i+p+1}]$; specifically:

$$N_{i,1}(u) = \begin{cases} 1 & \text{if } x_i \leq u < x_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1} + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}$$

The B-spline curve is the model adopted for the fitting of initial Q_k points, the coordinates of the control points of the fitted B-spline curve are the solution of an optimization problem in which the objective function is defined by means of the Force Density Method. In order to expose more clearly these topics the following flow chart depicts the mathematical approach defined to reach these goals.

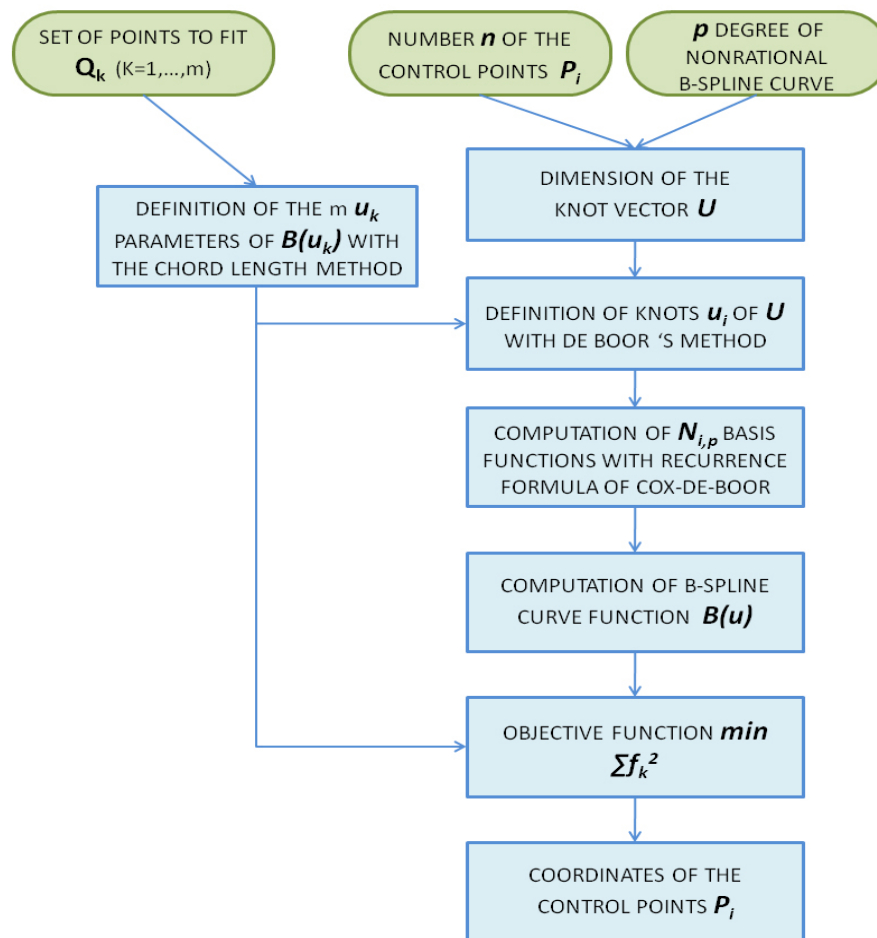


Figure 53: Flow chart of the force density fitting B-spline approach.

Except for the initial data, represented in oval frames, for each step specific functions have been programmed in Matlab in order to automate the whole process.

Suppose a set of points $\{Q_k\}, k = 1, \dots, m$ is given, the user has to specify the number n of the control points P_i of the fitted B-spline curve and its degree p .

From the initial points is possible to derive the $m \bar{u}_k$ values necessary to find the position of the nodes on which construct the network, as described in section 3.2.2. These parameters are really important because their choice affects the shape and parameterization of the curve. In order to calculate the \bar{u}_k values, lying in the range $u \in [0,1]$ the user can utilize the function $[uk]=chord_length_method(Q_k, n, p)$ based on the *Chord Length Method*. This is the most widely used method, and it is generally adequate. It also gives a "good" parameterization to the curve, in the sense that it approximates a uniform parameterization. In details, let d the total chord length:

$$d = \sum_{k=1}^m |Q_k - Q_{k-1}|$$

then $\bar{u}_0 = 0$ and $\bar{u}_{m-1} = 1$

$$\bar{u}_k = \bar{u}_{k-1} + \frac{|Q_k - Q_{k-1}|}{d} \quad k = 1, \dots, m$$

The $m \bar{u}_k$ parameters, with n and p values, are necessary for the computation of the knots vector U . The B-spline curves adopted during the unconstrained initial fitting problem are open (or non periodic or clamped) uniform B-spline that impose that the initial and final $(p + 1)$ knots are, respectively, equal to 0 and 1. In presence of internal knots, which number is equal to $(n - 1) - p$, their definition is obtained through the adoption of specific methods. In order to calculate automatically the knots vector U two functions have been programmed. The first one is $[U] = averaging_method(Q_k, n, p, u_k)$ based on the *Avering Method* that allows knots to reflect the distribution of \bar{u}_k :

$$u_0 = \dots = u_p = 0 \quad u_{n+p-2} = \dots = u_{n+p} = 1$$

$$u_{j+p} = \frac{1}{p} \sum_{i=j}^{j+p-1} \bar{u}_i \quad j = 1, \dots, n - p - 1$$

The second one is $[U] = deBoor_method(Q_k, n, p, u_k)$ that, for a big number of control points, is more efficient than the first one. If d is a positive real number, denote by $i = int(d)$ the largest integer such that $i \leq d$:

$$d = \frac{m}{n - p}$$

then define the internal knots by: $i = int(jd) \leq d \quad \alpha = jd - 1$

$$u_{p+j} = (1 - \alpha)\bar{u}_{i-1} + \alpha\bar{u}_i \quad j = 1, \dots, n - p$$

For the computation of the basis functions the $[matrNipUk] = NipUk (n, p, U, uk)$ function has been developed in order to return a matrix of dimension $(n \times m)$:

$$\begin{matrix} & \bar{u}_1 & \bar{u}_2 & \dots & \bar{u}_m \\ N_{0p} & & & & \\ N_{1p} & & & & \\ \dots & & & & \\ N_{(n-1)p} & & & & \end{matrix} \quad [\ n \times m \]$$

each element of the matrix allows the evaluation of a basis function N_{ip} for a specific value of \bar{u}_k . In this manner it is possible to compute automatically the equation of the B-spline and the (x, y) coordinates of a specific point of the B-spline $B(u)$.

In order to select one solution among all the possible ones, an *objective function* is added and a criterion has to be chosen. Here, the minimization of the sum of the square external forces, compute on the free nodes of the network, has been used:

$$\Phi(F) = \sum_{j=1}^{m-2} f_j^2 = \begin{cases} \sum_{j=1}^{m-2} f_{xj}^2 \\ \sum_{j=1}^{m-2} f_{yj}^2 \end{cases}$$

The equation of the objective function $\Phi(F)$ is calculated automatically by means of functions, programmed in Matlab, which formulation is based on the considerations of the sections 3.2.1 and 3.2.2.

The solver used is “*fmincon*” that attempts to find a constrained minimum of a scalar function of several variables starting at an initial estimate. The solver return a vector of dimension $2n$ that are the (x, y) coordinates of the n control points.

3.3. State of the art

The constrained curve fitting process fits equations of approximating curves to the raw field data. Nevertheless, for a given set of data, the fitting curves of a given type are generally not unique. Thus, a curve with a minimal deviation from all data points is desired. As stated in section 3.2, usually constrained and unconstrained curve fitting problems find solutions using algorithms based on the least squared method (Ordinary Least Squares for linear problem and Total Least Squares for non-linear problems). Indeed, even if the Force Density Method has been applied in different and various engineering field (section 3.2.1.), for what concern curve fitting problems the state of the art proposes only two researches.

Eric Saux [Saux 1998] proposes an application to cartographic generalization of maritime lines. His attempt is to include the fitting method for geographic data reduction in cartographic maps; in particular the use of B-spline curves for modeling smooth lines such as roads, railways or waterways. With his research Saux proposes two approaches for the implementation of the working tools in generalization [Ruas et al. 1993] of cartographic maps. One is automatic while the other is interactive. A catalogue of cartographic generalization operators has been proposed, including selection/elimination, aggregation, structuring, compression (or filtering), smoothing, exaggeration, caricaturing, enlargement and displacement. The curve deformation is obtained through mechanical parameter modifications which lead to a shape modification, the strategy is that suggested by Léon and Trompette [Léon et al. 1994] based on the force density method. Each equilibrium position of the cable network (or the control polygon) can be determined solving a linear system of equations. The strategy relies on a mechanical approach permitting fast calculations as well as local and global deformations. The principal drawback of this research is the limitation of the implemented tool to the cartographic map context.

Sanchez [Sanchez et al. 2008] proposes NURBS fitting techniques to represent tensile structures. The first step of the method consists into an initial grid generation, this structural net has the minimum nodes and elements needed to obtain the double curvature of the membrane. On the initial grid the force density method is applied in order to obtain the equilibrium shape of the membrane.

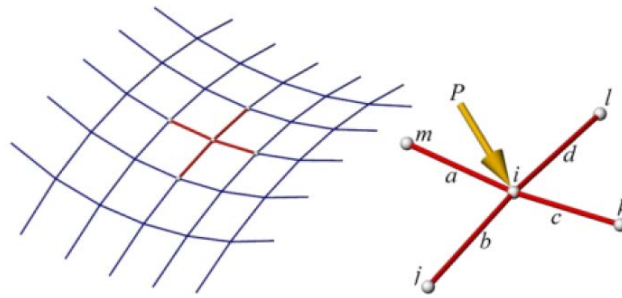


Figure 54: Part of a cable network.

Once the equilibrium shape is obtained, NURBS fitting techniques [Rogers et al. 1989; Piegl et al. 1997] are used to represent a smooth parametric surface that passes through the points obtained in the previous form finding step. The values of the force density parameters and the applied forces may also be changed, obtaining different equilibrium shapes in real time. With this method they have test different algorithms but these are not described in details in the paper. The main limit of the proposed techniques is that they use an unconstrained fitting curve method useful only to find the equilibrium state of tensile structures.

3.4. Constrained B-Spline Force Density Fitting Method

Once the general mathematical approach has been defined it is necessary to specify the constraints to which the fitting B-spline has to satisfy. These constraints should be introduced directly by the designer that, based on the own experience, knows the type of constraints and specifies the points or the parts of the curve on which impose them.

The main problem in this case is the possibility to introduce, for the same curve, more than one constraint. As stated before the B-spline are piecewise polynomial curves then a first constrain applied on the left side of a curve could influence also the right side. In consequence, it's necessary to introduce into the methodology a tool able to split the initial curve in different parts that are not subjected to the constraints applied on the previous and next parts. Then the *splitting toolbox* should be able to introduce G^0 *geometric continuities* (clearly a G^0 geometric continuity implies a C^0 parametric continuity, and vice versa) by means of specific tools, implemented in Matlab, based on the curve editing techniques that will be exhaustively described in section 3.4.3.

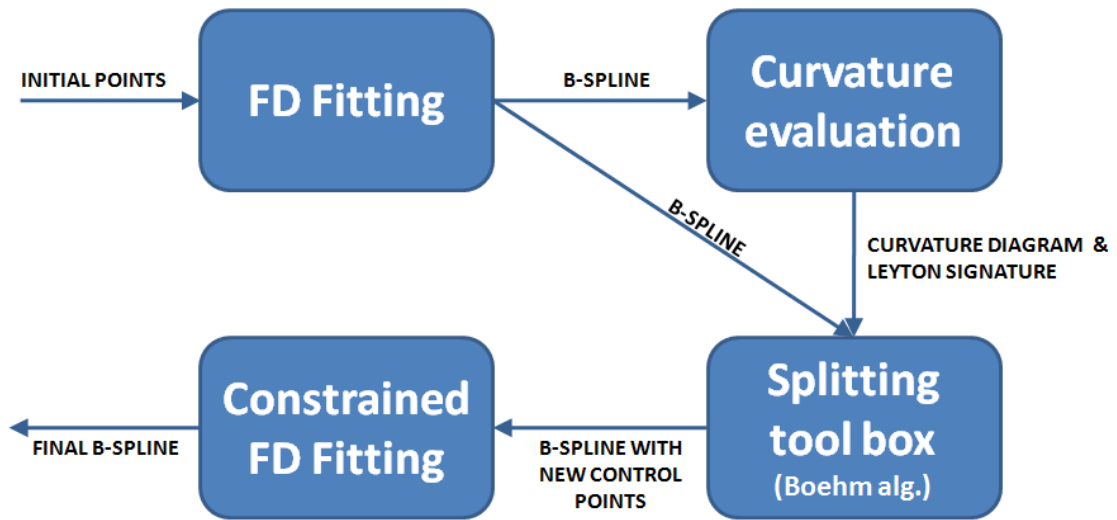


Figure 55: Constrained Force Density B-Spline Fitting methodology.

The implementation of the splitting toolbox could be sufficient for the calculation of the final constrained fitting curve, but in order to improve the objectivity of the entire methodology the process has been enriched by the introduction of another *toolbox for the curvature analysis and evaluation*. This toolbox calculates the curve’s curvature (section 3.4.2.1.) and depicts this information on diagrams that allows the designer to identify and select, in a more objective manner, the principal and characteristic points of the curve on which introduce G^0 discontinuities. In addition to the curve’s curvature graphs, the curvature evaluation toolbox is able to highlight the most important points belonging to the B-spline by means of the *Process-Grammar* (the principle is presented in the section 3.4.2.2.) ideated by Leyton [Leyton 1987a].

The process ends with the last toolbox that recalculates the fitting curve taking into account the points to fit, the curvature information and the characteristic points inferred from the initial fitting curve, the new geometric parameters output of the split toolbox.

From a mathematical point of view a constrained fitting problem could be translated into the subsequent system:

$$\begin{cases} \min \Phi(F) = \min \sum_k f_k^2 & \text{objective function} \\ G(F) = 0 & \text{constraints} \end{cases}$$

3.4.1. Initial Force Density Fitting

The first toolbox requires the Q_k initial points, the number n of control points and the degree p on the B-spline curve as input data. FD fitting toolbox use the functions described in detail in section 3.2.3 and return the coordinates of the control points of the fitted (open uniform) B-spline curve.

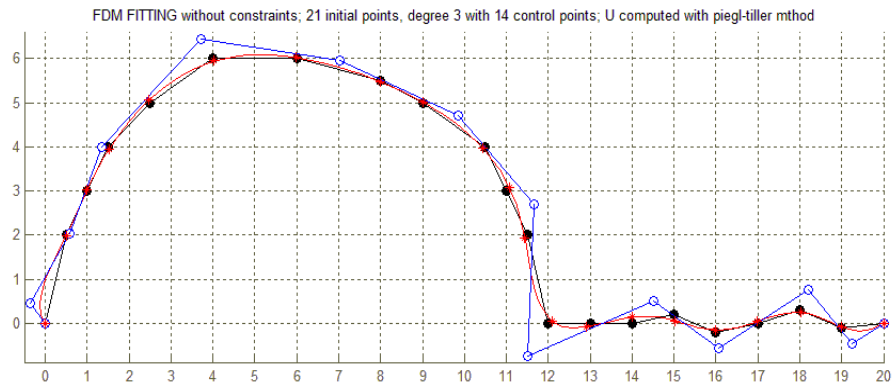


Figure 56: Initial B-spline fitted curve.

Ad example, in figure 56 the initial fitting of 21 initial points (black circles) is depicted, the fitted B-spline (red curve) has 14 control points (blue polyline) and the degree is equal to 3. On the B-spline is possible to notice the nodes of the network represented with red stars.

3.4.2. Curvature evaluation

The aim of this phase is the individuation, in an objective manner, of the principal characteristics of a curve. In order to reach this goal the initial fitting curve (3.4.1 section) is subject to a curvature analysis that consist of two main steps: the curvature computation of a shape and its Leyton signature that allow to index each curve through the adoption of the Leyton process grammar.

3.4.2.1. Curvature

The curvature of a curve is a useful entity, so it deserves to be rigorously defined. Intuitively, the curvature should be a number that measures how much the curve deviates from a straight line at any point. It should be large in areas where the curve wiggles,

oscillates, or makes a sudden direction change; it should be small in areas where the curve is close to a straight line. It is also useful to associate a direction with the curvature, i.e., to make it a vector. The curvature is defined as the vector k whose magnitude is the reciprocal of the radius of the circle [Salomon 1999, 2006]. Using differential geometry, it can be shown that the vector k is defines as follow:

$$k = \left| \frac{\dot{x}\ddot{y} - \dot{y}\ddot{x}}{(\dot{x}^2 + \dot{y}^2)^{3/2}} \right|$$

The curvature function of a B-spline, programmed in Matlab with the name “*curv*”, is written as it follows [Grimaud 2008]:

$$k(u) = \frac{\det\left(\frac{dB}{du}, \frac{d^2B}{du^2}\right)}{\left\|\frac{dB}{du}\right\|^3}$$

The following graph depicts the curvature for the B-spline yields by the initial fitting toolbox showed in section 3.4.1., the point with the highest curvature is picked out:

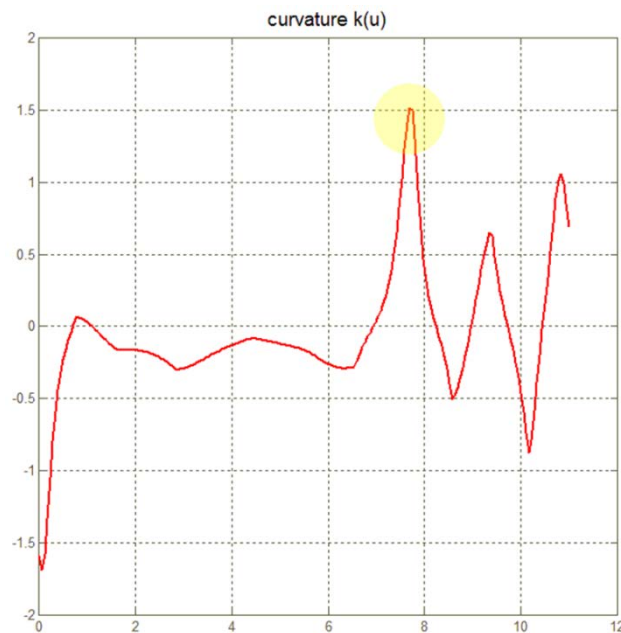


Figure 57: Plot of the curvature of the B-spline showed in figure 56.

3.4.2.2. Leyton Process Grammar

The process-grammar of Leyton [Leyton 1987a, 1987b, 1988, 1989, 2007] is a syntactic description based on curvature. After Leyton's publications, the grammar, and the mathematics on which it is based, was applied by scientists in many disciplines: radiology, meteorology, computer vision, chemical engineering, geology, computer-aided design [Pernot et al. 2003b], robotics, anatomy, botany, forensic science, architecture, abductive reasoning, linguistics, mechanical engineering, computer graphics, archaeology, etc.

The grammar was developed in the purpose to infer history from shape. The shape can be understood as the result of various processes such as protrusion, indentation, squashing, resistance. The Process Grammar relies on two structural factors in a shape: *symmetry* and *curvature*. Mathematically, symmetry and curvature are two very different descriptors of shape. However, the *symmetry-curvature duality theorem* proved in [Leyton 1987a] shows that there is an intimate relationship between these two descriptors. Indeed the theorem states that, to each curvature extremum, there is a unique symmetry axis terminating at that extremum. It is valuable to illustrate the theorem on a closed shape, for example, shown in figure 58, where the axis are shown as dashed lines:

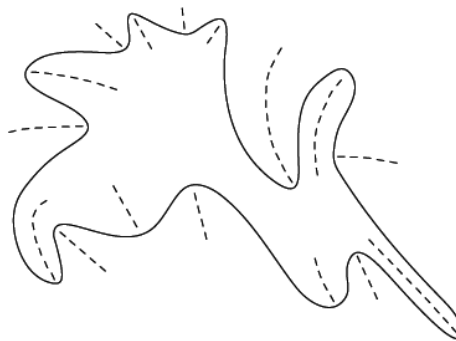


Figure 58: Illustration of the Symmetry-Curvature Duality Theorem

Furthermore the Interaction Principle states that each of the axes is a direction along which a process has acted. The implication is that the boundary was deformed along the axes; e.g. each protrusion was the result of pushing out along its axis, and each indentation was the result of pushing in along its axis.

Any individual outline, together with the inferred arrows, will be called a process-diagram. The reader should observe that, on each process-diagram in figures 59, 60 and 61, a letter-label has been placed at each extremum (the end of each arrow). There are four

alternative labels, $M+$, $m-$, $m+$, and $M-$, and these correspond to the four alternative types of curvature extrema.

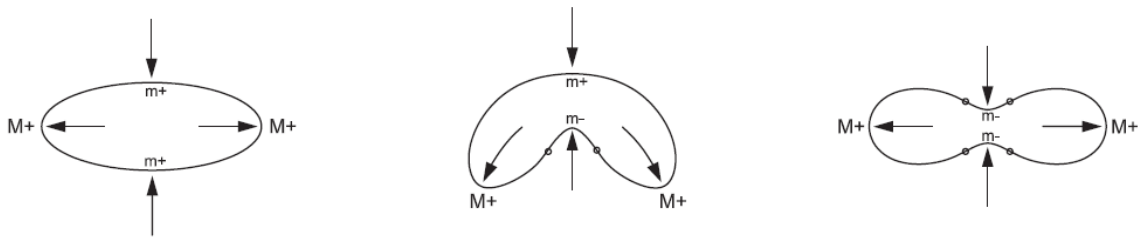


Figure 59: The inferred histories on the shapes with 4 extrema.

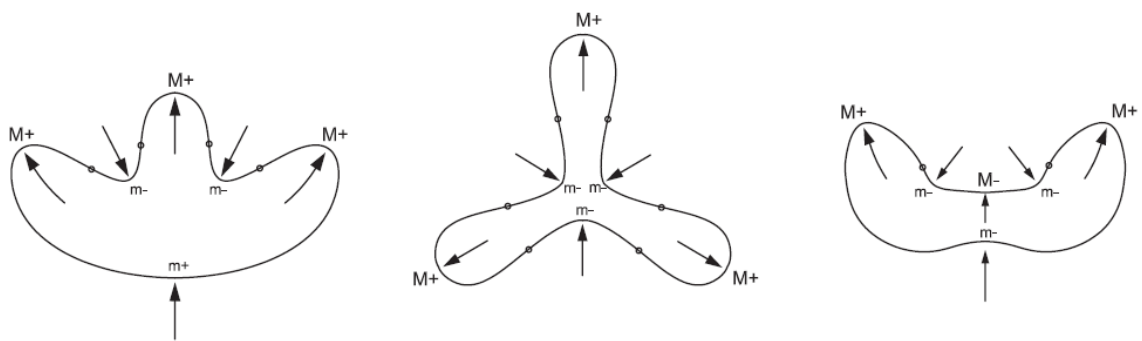


Figure 60: The inferred histories on the shapes with 6 extrema.

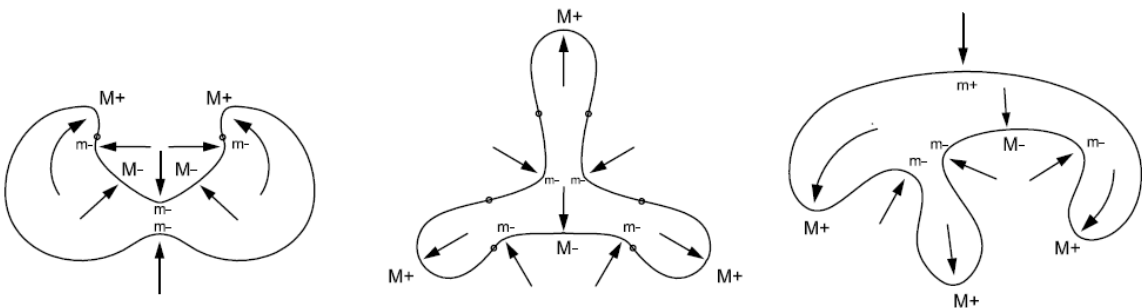


Figure 61: The inferred histories on the shapes with 8 extrema.

It becomes clear that the four extrema types correspond to four processes. Table 3 gives the correspondence:

EXTREMUM TYPE	↔	PROCESS TYPE
$M+$	↔	protusion
$m-$	↔	indentation
$m+$	↔	squashing
$M-$	↔	internal resistance

Table 3: Correspondence between extremum type and process type.

First choose the direction of traveling along the curve to be that which keeps the solid on the left side of the curve. Then define curvature as the rate of anti-clockwise rotation. Denote a curvature maximum and minimum by M and m , respectively; and denote positive and negative curvature by $+$ and $-$, respectively. Then, in the *curvature function* of figure 57, the extrema are illustrated by the graph in the following figure:

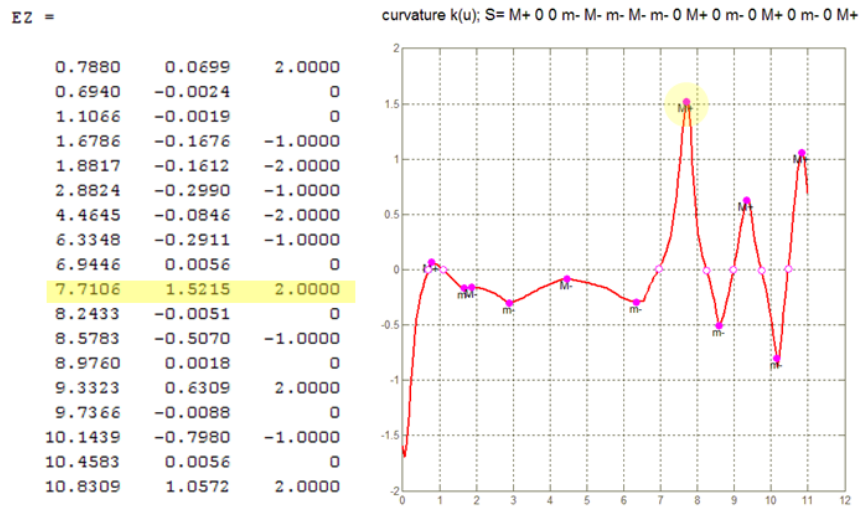


Figure 62: A curvature function showing the four kinds of extrema.

As stated before, the Leyton Grammar allows to assign a specific signature to a curve but, most of all, to know exactly the position of the characteristic points of the B-spline curve. In this case the highest curvature's value is $k = 1,5215$ for $B(u = 7,7106)$.

3.4.3. Splitting ToolBox

If reasonable distribution of computed points is desired the adoption of specific techniques and mathematical methods are necessary to control the shape of the curve.

For B-spline editing the literature proposes different and various algorithms [Piegl et al. 1995] that allows to increase or reduce the flexibility of a curve, these algorithms can be classified in:

- **degree elevation/reduction**: increase/decrease the b-spline's order ($p+1$) value; early degree elevation algorithms were developed by Cohen [Cohen et al. 1985] and Prautzsch [Prautzsch 1984b]. More efficient algorithms are given by

Prautzsch and Piper [Prautzsch et al. 1991] and Piegl and Tiller [Piegl et al. 1994, 1997].

- **knot insertion(subdivision)**: as an alternative to degree elevation, the flexibility of a B-spline curve is increased by inserting additional knot values into the knot vector U . Inserting a single knot value is referred to as **knot insertion**. Inserting multiple knot values is called **knot refinement**. The effect is to locally **split** a piecewise polynomial segment for a given knot value interval (parametric interval) into two piecewise polynomial segments over that interval. There are two basic methods for accomplishing knot value insertion. The first is the so-called Oslo algorithm developed by Cohen et al. [Cohen et al. 1980] and the one developed by Prautzsch [Prautzsch 1984a], which simultaneously insert multiple knot values into the knot vector. The second method, by Boehm [Boehm 1980; Boehm et al. 1985], sequentially inserts single knot values into the knot vector.
- **knot removal**: is the reverse of knot insertion. Boehm [Boehm 1980] briefly mentions knot removal but provides few details. Tiller [Tiller 1992] gives algorithms for removing a specified number of knot values and as many knot values as possible for B-spline. Tiller assumes that a knot is removable if and only if it leaves the original curve unchanged both parametrically and geometrically. However, as Tiller points out, knot removal, like degree reduction, results in an overspecified problem and thus can only be solved within some tolerance.
- **reparameterization**: is another technique for raising the degree of a B-spline curve. It consists in the redistribution and redefinition of the knot values in the knot vector U without changing the number of the knots [Piegl et al 1997].

The most important and used techniques are those for degree elevation and knot insertions. The basic idea behind these techniques is to increase the flexibility of the curve without changing the shape of the curve. The success of the idea depends on the fact that there are an infinite number of control polygons (control points) with more than the minimum number of vertices that represent identical B-spline curves. Subsequent manipulation of the new control polygon vertices is used to change the curve shape.

In order to develop a toolbox that allows to locally split a B-spline, that in the proposed approach means insert a G^0 continuity on the curve, is necessary increase the flexibility of

the B-spline by inserting additional knot values into the knot vector U that correspond to the **knot insertion** technique. As stated before in the knot insertion process, a knot is added to the knot vector of a given B-spline. This results in an additional control point and a modification of a few existing control points. The end result is a curve defined by a larger number of control points, but which defines exactly the same curve as before knot insertion. Knot insertion has several applications. One is the de Boor algorithm for evaluating a B-spline. Another application is to provide a designer with the ability to add local details to a B-spline. Knot insertion provides more local control by isolating a region to be modified from the rest of the curve, which thereby becomes immune from the local modification. In particular the development of the splitting toolbox is based on the **Boehm** algorithm. Consider the original curve $B(u)$ defined by:

$$B(u) = \sum_{i=0}^{n-1} N_{i,p}(u) \cdot P_i$$

with the knot vector $[U] = [u_1; u_2 \dots u_{n+p+1}]$

After knot insertion, the new curve is $C(t)$ defined by:

$$C(t) = \sum_{j=0}^{m-1} \bar{N}_{j,p}(t) \cdot P_j^*$$

with the knot vector $[U^*] = [u_1^*; u_2^* \dots u_{m+p+1}^*]$

where $m > n$. The objective is to determine the new control polygon vertices P_j^* such that $B(u) = C(t)$, as depicted in the figure 64.

It is important to notice that the curve represented in figure 64 is splitted in the point P_4^{***} , then it is composed of two cubic Bézier segments, cause the knot vector $[U^{***}] = [0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 2 \ 2 \ 2 \ 2]$ presents three multiple knot values equal to 1 (the multiplicity of the internal knots is equal to the degree of Bèzier curves that compose the B-spline curve).

Then the implementation of the **splitting toolbox** allows to insert G^0 continuities in the characteristic points detected during the **curvature analysis**; from a mathematical points of view, the outputs of the splitting toolbox are:

1. the modified knot vector $[U^*]$ with internal multiplicities;

2. the new set of \bar{u}_k parameters (their number is equal to the number of control points P^*) necessary to build the final Force Density network. In order to automatically calculate these new parameters the $[uk]=chord_length_after_Boehm(Q_i, n, p, U_boehm)$ function.

These outputs, with the equations of the constraints, represent the inputs for the last toolbox, the constrained force density fitting, that allow to find the final shape of the fitted B-spline curve.

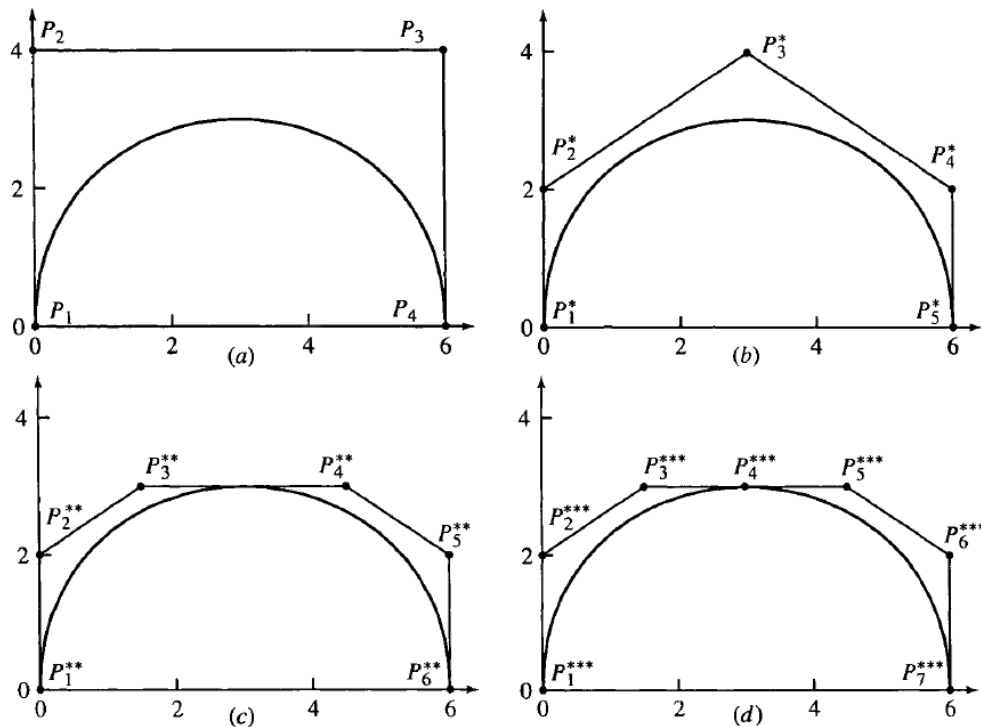


Figure 63: Knot insertion results. (a) Original control polygon of a fourth-order B-spline curve; (b) control polygon after insertion of one knot value; (c) control polygon after insertion of the second multiple knot value; (d) control polygon after insertion of the third multiple knot value.

The next figure shows the result of the splitting toolbox applied to the B-spline depicted in figure 56, the curve has been splitted in the point according the curvature analysis represented in figure 62. The G^0 continuity has obtained with a multiple internal knot that yields 3 new control point one of which is exactly on the curve $B(u = 7,7106)$:

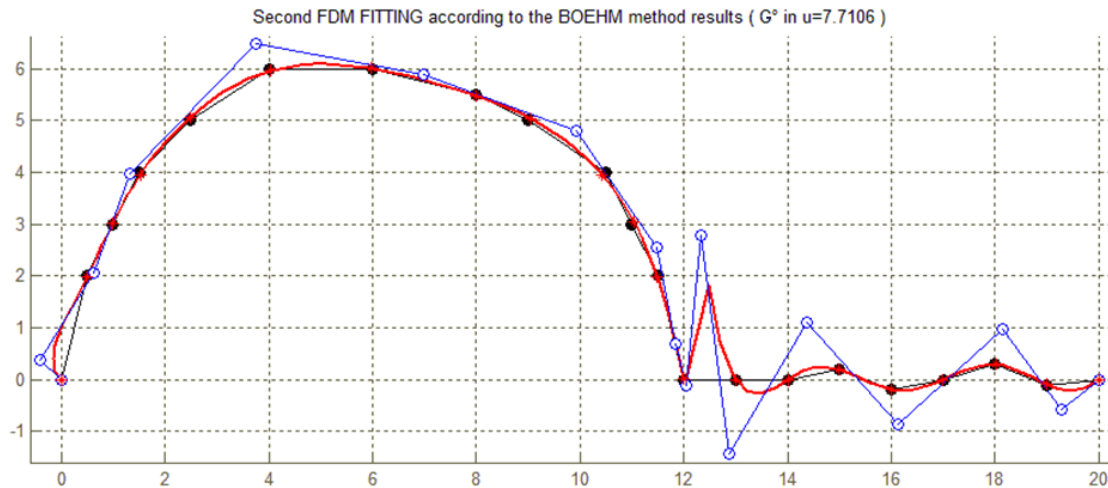


Figure 64: Curve splitting using Boehm algorithm.

3.4.4. Constrained Force Density B-spline Fitting Toolbox

While the first fitting toolbox yields a uniform B-spline, in order to utilize the output of the curvature and splitting toolbox and then perform a constrained fitting curve, in the last fitting procedure a different type of B-spline curve has been adopted. The nonuniform B-spline is more general than the uniform or open B-splines, although it is not the most general type of this curve. It is obtained when the knot values are not equally spaced. The only requirement is that the knots be nondecreasing. Adjusting the knot values (as well as having multiple values) is a feature that helps fine-tune the shape of the curve. Multiple knots can be used to pull the curve in a certain direction and to create a cusp or even a discontinuity at a join point.

In order to implement different kind of constraints in the final optimization problem, Kuhn-tucker condition with Lagrangian multipliers has been used. The condition has been implemented using the Matlab function "*fmincon*" (find minimum of constrained nonlinear multivariable function) belonging to the "*Optimization Toolbox*". The Kuhn-tucker conditions in mathematics are necessary for a solution in non-linear programming to be optimal. Provided some regularity conditions are satisfied. It is a generalization of the method of Lagrange multipliers to inequality constraints.

$$\min f(\varphi) \text{ such that } \begin{cases} lb \leq \varphi \leq ub \\ A_{eq} \cdot \varphi = b_{eq} \\ A \cdot \varphi \leq b \\ c_{eq}(\varphi) = 0 \\ c(\varphi) \leq 0 \end{cases}$$

where φ is the unknowns vector by means of is possible to calculate the coordinates (x,y) of the control points of the final fitting B-spline. lb, ub, b_{eq} , and b are vectors, A and A_{eq} are matrices, $c(\varphi)$ and $c_{eq}(\varphi)$ are functions that return vectors, and $f(\varphi)$ is a function that returns a scalar. $f(\varphi), c(\varphi)$, and $c_{eq}(\varphi)$ can be nonlinear functions. From the system above is possible to infer that the constraints are separated into the following types:

- **bound constraints:** lower lb and upper ub bounds on the components of the vector φ ;
- **linear inequality constraints:** linear inequality constraints are of the form $A \cdot \varphi \leq b$. When A is m -by- n , this represents m constraints on a variable φ with n components.
- **linear equality constraints:** linear equalities are of the form $A_{eq} \cdot \varphi = b_{eq}$. This represents m equations with n -component vector φ . You supply the m -by- n matrix A_{eq} and the m -component vector b_{eq} . The form of this type of constraint is exactly the same as for *Linear Inequality Constraints*.
- **nonlinear constraints:** nonlinear inequality constraints are of the form $c(\varphi) \leq 0$, where c is a vector of constraints, one component for each constraint. Similarly, nonlinear equality constraints are of the form $c_{eq}(\varphi) = 0$.

The following figure depicts the final FD fitting curve on which straight and curvature constraints have been imposed, as described in details in the next sections, they are both equality nonlinear constraints. As explained before, the *constrained FD fitting toolbox* take as input from the *splitting toolbox* the number of control points $p = 17$ and the knot vector U .

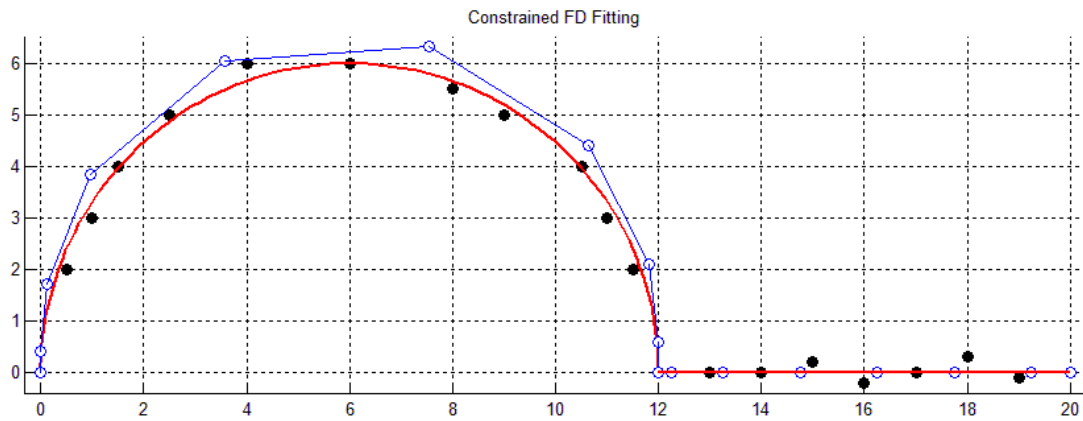


Figure 65: Final constrained FD fitting.

In the following sections the constraints programmed in Matlab, in order to achieve the constrained force density fitting goal, have been detailed:

3.4.4.1. Position Constraint

The *position constraint* is a *linear equality constraint* that can be imposed on:

- one or more control point p_i
- point/s of the b-spline $B(u)$

In the first case two equations are added to the system for each constrained control point. The constrain has been implemented in Matlab with the function `addPosConstrCtrlP(Aeq, beq, nCtrl, x, y)` through of which it's possible to specify the coordinates of the control point to constrain.

Also the position constraint applied on a specific point of the b-spline yields two linear equality equations to add to the system. Ad example the position constraint:

$B(u = 0,5) = (x, y)$ is resolved in

$$\begin{cases} B_x(u = 0,5) = \sum_{i=0}^{n-1} N_{i,p}(u = 0,5) \cdot x \\ B_y(u = 0,5) = \sum_{i=0}^{n-1} N_{i,p}(u = 0,5) \cdot y \end{cases}$$

3.4.4.2. Distance Constraint

The *distance constraint* is an *equality non-linear constraint* that can be applied on:

- Two control points of the b-spline: $\overline{P_i P_j} = d$
- Two points of the b-spline: $\overline{B(u_i) B(u_j)} = d$

In both cases the imposition of a distance constraint implies the addition of one equation to the system:

$$\begin{aligned}
 & \bullet \left((P_{xi} - P_{xj})^2 + (P_{yi} - P_{yj})^2 \right)^{1/2} - d = 0 \\
 & \bullet \left((B_x(u_i) - B_x(u_j))^2 + (B_y(u_i) - B_y(u_j))^2 \right)^{1/2} - d = \\
 & = \left(\left(\sum_{i=0}^{n-1} N_{i,p}(u_i) \cdot x - \sum_{i=0}^{n-1} N_{i,p}(u_j) \cdot x \right)^2 \right. \\
 & \quad \left. + \left(\sum_{i=0}^{n-1} N_{i,p}(u_i) \cdot y - \sum_{i=0}^{n-1} N_{i,p}(u_j) \cdot y \right)^2 \right)^{1/2} - d = 0
 \end{aligned}$$

The constraints' equation are computed automatically by means of the functions programmed in Matlab: `addDisConstrCtrlP (n, p, U, CtrlP1, CtrlP2, distance)` and `addDisConstr (n, p, U, u1, u2, distance)` in order to impose the distance constraint respectively between two control points and two points belonging to the curve.

3.4.4.3. Straight Constraint

In order to impose that a specific part of the b-spline is straight it's necessary apply the straight constraint on the control points p_i of the curve:

$$\overline{(P_{i-1} P_i)} \wedge \overline{(P_i P_{i+1})} = 0$$

that is an *equality non-linear constraint*. In order to introduce directly this type of constraint the `StraightConstraint` function has been implemented in Matlab in which the input parameters are the control points on which impose the constraint.

3.4.4.4. Curvature Constraint

Based on the equation described in section 3.4.2.1. the **curvature constraint** is a **equality non-linear constraint** that can be explicated in the subsequent equation:

$$\left| \frac{\dot{B}_x(\bar{u}_k) \cdot \ddot{B}_y(\bar{u}_k) - \dot{B}_y(\bar{u}_k) \cdot \ddot{B}_x(\bar{u}_k)}{\left((\dot{B}_x(\bar{u}_k))^2 + (\dot{B}_y(\bar{u}_k))^2 \right)^{3/2}} \right| = \left(\frac{1}{r} \right)$$

where \bar{u}_k is the points of the b-spline on which impose the curvature constraint, and r is the curvature's radius.

The derivatives \dot{B} and \ddot{B} of a B-spline curve at any point on the curve are obtained by formal differentiation. Specifically, the first derivative is:

$$\dot{B}(u) = \sum_{i=0}^{n-1} \dot{N}_{i,p}(u) \cdot P_i \quad \text{while the second derivative is:} \quad \ddot{B}(u) = \sum_{i=0}^{n-1} \ddot{N}_{i,p}(u) \cdot P_i$$

Here, the primes denote differentiation with respect to the parameter u . The derivatives of the basis functions are also obtained by formal differentiation:

$$\dot{N}_{i,p}(u) = \frac{N_{i,p-1}(u) + (u - x_i)\dot{N}_{i,p-1}(u)}{x_{i+p-1} - x_i} + \frac{(x_{i+p} - u)\dot{N}_{i+1,p-1}(u) - N_{i+1,p-1}(u)}{x_{i+p} - x_{i+1}}$$

Differentiating the previous equation yields the second derivative of the basis function:

$$\ddot{N}_{i,p}(u) = \frac{2\dot{N}_{i,p-1}(u) + (u - x_i)\ddot{N}_{i,p-1}(u)}{x_{i+p-1} - x_i} + \frac{(x_{i+p} - u)\ddot{N}_{i+1,p-1}(u) - 2\dot{N}_{i+1,p-1}(u)}{x_{i+p} - x_{i+1}}$$

3.5. Conclusion and limits of the fitting

As depicted in the previous sections the developed approach has many advantages due especially to the adoption of the Force Density Method that give a great flexibility to whole methodology. Ad example for specific force density values q_k of the developed network the system yields the same results of the Least Squared Method. Furthermore four different constraints have been developed but it's possible to implement any kind of linear/nonlinear and equality/inequality constraints.

Inevitably the methodology presents drawbacks, in particular there are two main limitations there are just mathematical limits belonging to the formulation of B-spline curves and of the optimization problems.

In fact the first limit is that B-spline curves are still polynomial curves hence cannot represent many useful simple curves such as circles and ellipses, broadly speaking they cannot represent conic sections. In order to overcome these kind of problems is necessary to adopt the Non-Uniform Rational B-Spline (NURBS) curves, generalizations of both B-splines and Bézier curves, that allows a correct representation of circles and rounds (fig.66).

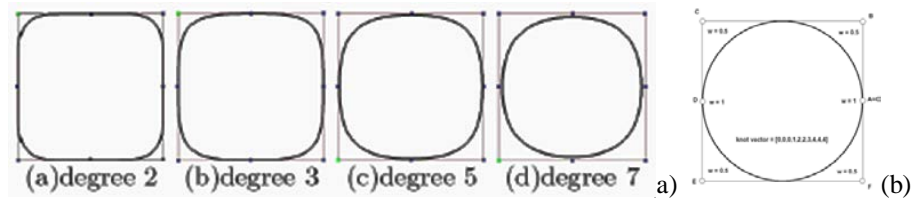


Figure 66: A seven degree B-spline curve cannot represent a circle (a), representation of a circle with NURBS (b).

The definition of the network, described in section 3.2.2, assumes that the nodes, on which calculate the equilibrium states necessary for the formulation of the objective function to minimize, are positioned directly on the curve. This decision is of fundamental importance because permits to readapt all the methodology to most versatile curves, the NURBS curves without change the phases and the mathematical approach defined.

For what concern the second limitation optimization problems can take many iterations to converge and can be sensitive to numerical problems such as truncation and round-off error in the calculation of finite-difference gradients. Most optimization problems benefit from good starting guesses. This improves the execution efficiency and can help locate the global minimum instead of a local minimum. Usually, the goal of an optimization is to find a local minimum of a function: a point where the function value is smaller than at nearby points, but possibly greater than at a distant point in the search space. Sometimes the goal of an optimization is to find the global minimum: a point where the function value is smaller than all others in the search space (fig.67).

Many numerical methods for optimization are based, in part, on the method of steepest descent (one the principal gradient method but also the simplest method). Obviously, steepest descent paths can be more complicated in more dimensions; in addition the introduction of constraints can break up one basin of attraction into several pieces, where a steepest descent path may be restricted from proceeding.

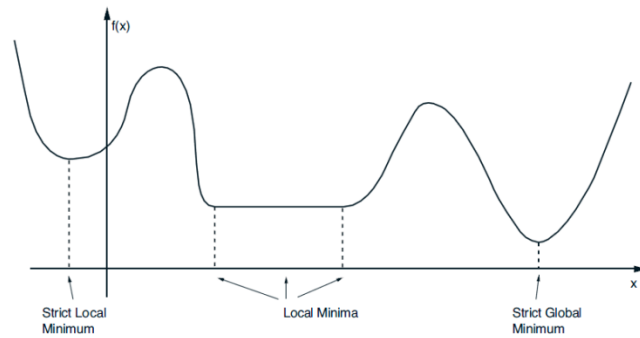


Figure 67: Unconstrained local and global minima in one dimension.

One of the main problem in the adoption of the Optimization Toolbox is that its solvers are not designed to find global optima, thence generally find the minimum point in the basin but leave the choice of starting point to the user. Then they find the optimum in the basin of attraction of the starting point. If a global optimum is requested, it is necessary to find an initial value contained in the basin of attraction of a global optimum. In conclusion, the more the user knows about possible initial points, the more focused and successful his search will be [Nocedal et al. 2006]. In particular the Matlab function “*fmincon*” adopted is a gradient-based method that is designed to work on problems where the objective and constraint functions are both continuous and have continuous first derivatives, this function uses algorithms based on Newton’s Method and its variations [Coleman et al. 1994, 1996]. When the problem is infeasible, “*fmincon*” attempts to minimize the maximum constraint value.

Chapter 4

Parametric archetypes as support for interpreting topological optimization results

This chapter aims at define the concept of the parametric archetype. It correspond to a high level functional description of mechanical components that can be introduced in different phases of the methodology, developed in the first part of the manuscript, as an alternative tool that helps and support the designers and the analysts during the PDP.

It is worth noticing that the aim of this chapter is not the creation of a fully automatic system for design embodiment, but it is the suggestion and experimentation of a common base for knowledge sharing and for a better interconnection between different systems and applications. In the section 4.1 the process that leads to the individuation of the archetype for different kind of commercial models has been described. The advantage of such a tool is the extension of the methodology (Part I) that has been readapted and upgraded in section 4.2. Finally section 4.3 proposes a case study in order to test and validate the capabilities of the parametric archetype.

4.1. Parametric archetype inception

The first and second chapters present a methodology supported by tools that allow an efficient data exchange between different systems and facilitate the designers' job by means of a wise and rational knowledge management. These tools are able to support the user even if very little experience on the systems is available, furthermore they allow to

exchange and share the knowledge, the experience and the comments of designers and analysts along the whole process.

The introduction of the parametric archetype can be considered a tool that brings into the methodology a new kind of knowledge because it is not related to the choices or experience of the users but it takes origin directly by the models. The parametric archetype can be defined as the 2D representation of the functional description of a mechanical component. In order to reach the individuation of a single parametric archetype for a specific typology of commercial model an approach has been proposed. The following figure illustrates the different stages that bring to the definition of the archetype:

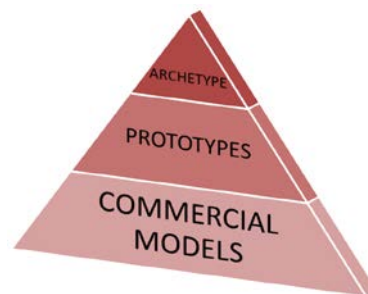


Figure 68: Stages for reaching to the definition of the archetype.

The first step is the gathering of commercial models belonging to the same typology, in consequence this stage has a high number of models. As it is possible to infer from the sequent illustration, for the same mechanical component is possible to find different shapes and dimensions:



Figure 69: Wide variety of geometries for the same mechanical component.

Once the first collection of commercial components is terminated the selection process goes on with the comparison of these models between themselves in order to identify the smallest number of components that differ in their topological features. These are the most characteristic models that can be defined *prototypes* and are represented by 2D sketches. For example, from the commercial models of the connecting rods is possible to identify the following prototypes:

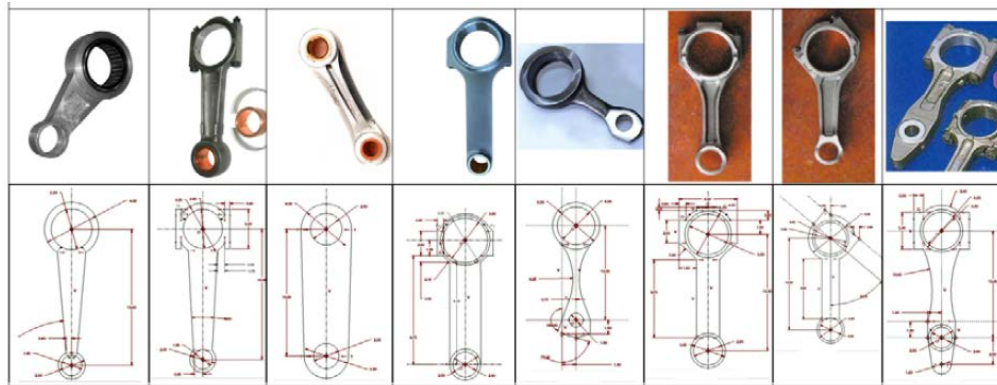


Figure 70: Prototypes of the commercial connecting rods.

The last stage of the selective process is occupied by only one element that is the parametric archetype. The definition of this archetype is reached through the individuation of the topological and functional characteristics in common with each prototype. Then the archetype can be considered a synthesis of those topological and functional information common to all the prototypes and commercial models from which it derives.

Starting from the prototypes of the commercial connecting rod models the parametric archetype depicted in the following figure is derived:

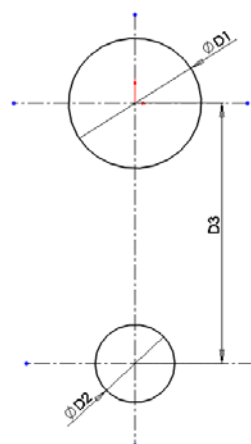


Figure 71: Parametric archetype of a connecting rod.

4.2. Extension of the methodology with the introduction of the parametric archetypes

The introduction of the parametric archetype allows to extend the capability of the methodology described in the first part of the manuscript. By definition the archetype is a base of knowledge, common to all components from which it derives, that can be shared and reused along the whole process. The extended methodology is shown in the figure 72:

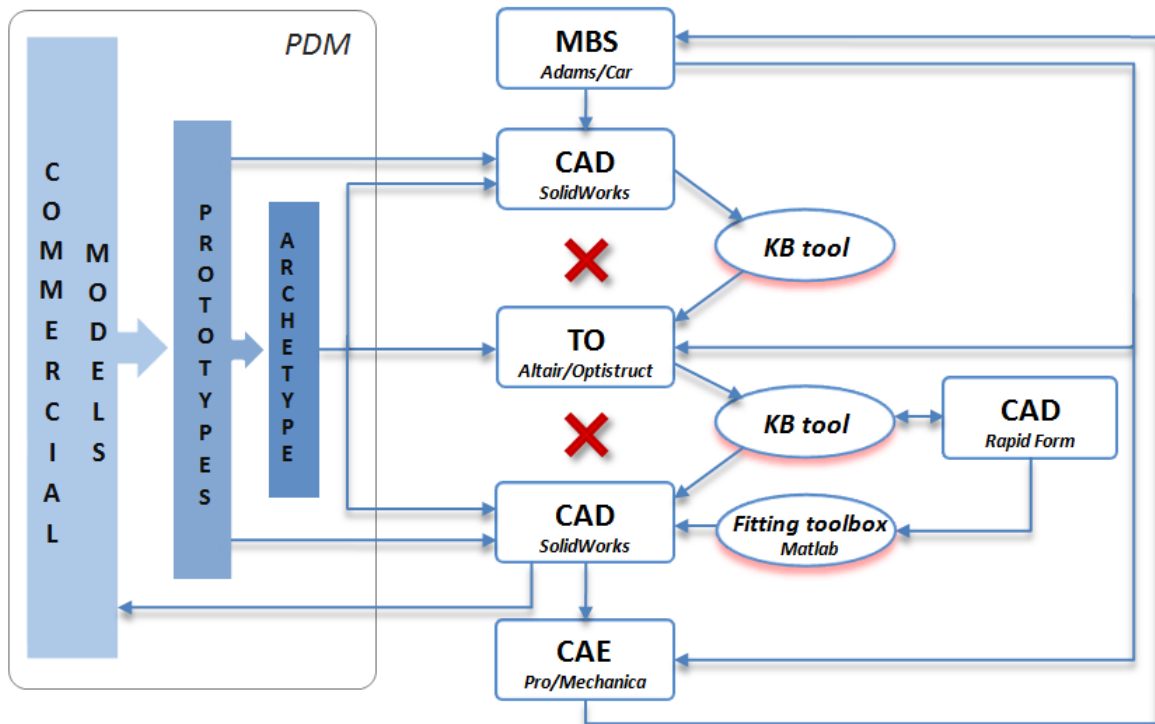


Figure 72: Extended methodology with the introduction of the parametric archetype.

During the modeling and preparation of the component to be optimized, it is possible to help and simplify the designer's job by implementing libraries of prototypes (fig.73) that can be inserted and used directly within the CAD environment. If no prototypes meet designer's needs certainly the parametric archetype can be use as base for the modeling of the initial component. The user interface (fig.73), that allows to manage prototypes and archetype, has been programmed in Visual Basic for Applications using the API of SolidWorks [SWOA 2010].

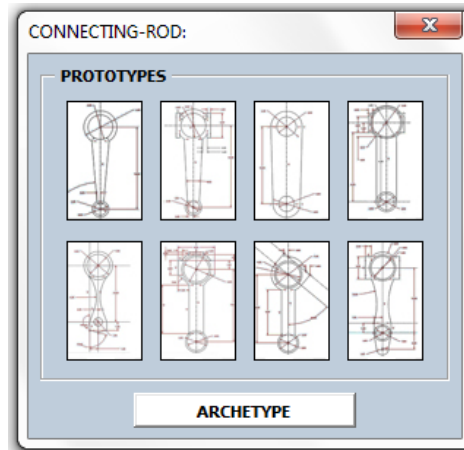


Figure 73: Connecting rod prototypes library.

As stated in the previous section the archetype is a 2D functional representation of mechanical components thence this kind of information are helpful also for the analyst, during the definition of the optimization problem, since the archetype suggests the invariant volumes of the model.

As the prototypes and archetype have been helpful during the initial 3D modeling, so they can be used in the redesign phase in order to ease and fast the interpretation of the results of the topology optimization analysis. In particular the designer can compare the projections of the optimized model with the prototypes, in case there is a similarity the user can edit and modify the desired prototype, according to the information suggested by the projection, in order to fast the redesign of the final component. Otherwise the archetype can be adopted as base of knowledge to start the redesign process.

4.3. Case study: connecting rod

For a better comprehension of the concepts stated in the previous section a case study which concern the design of a connecting rod is depicted.

As described in section 4.1, figure 74 illustrates the process that leads to the definition of the parametric archetype of the connecting rod:

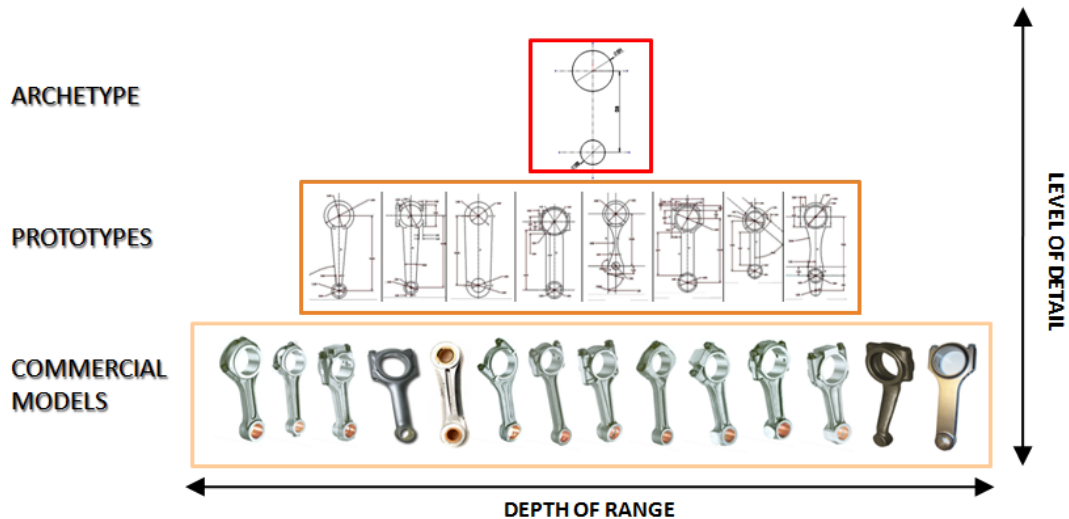


Figure 74: Stages for the individuation of the connecting rod archetype.

Starting from the gathering of commercial connecting rods, a first selection has been carried out in order to distinguish the different typologies that collect models with topological and functional common features. The prototypes allow an easier definition of the parametric archetype that for the connecting rod is composed by two circle representing the functional surfaces of the component (fig.71).

As stated in the previous section, during the initial modeling phase a prototype can be selected from the user-interface (fig.73) in order to support and fast the modeling. The prototype can be modified in order to satisfy the design requirements.

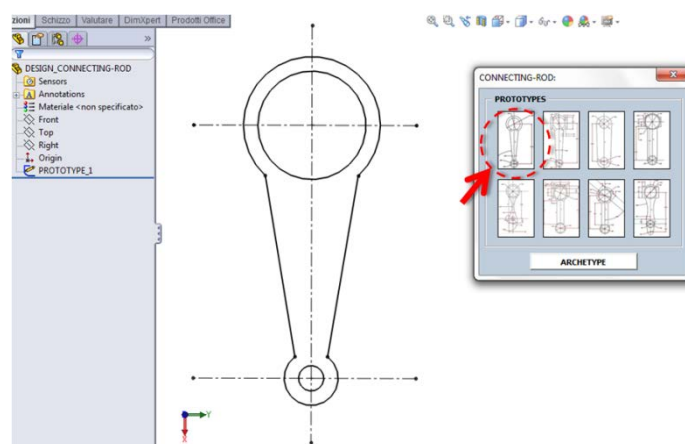


Figure 75: Initial modeling of the connecting rod choosing among different prototypes.

According to the proposed methodology, the topological optimization analysis is performed in Altair Optistruct. Before starting the analysis, it is necessary to create the

mesh of the model and then define variant/invariant volumes taking advantage of the data and information inherit from the previous phases (for more details read sections 1.3.2 and 1.4.2). But the analyst can make use also of the archetype because, as demonstrated in the following figure, it suggest to the analyst the invariant volumes:

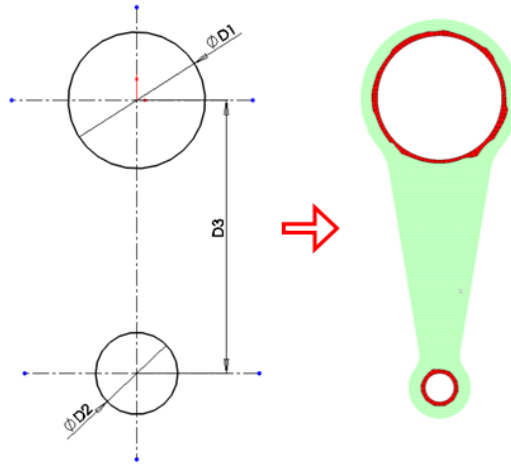


Figure 76: Analogy between archetype and invariant volumes.

Besides the generation of the mesh the analyst has to define the collectors which are the physical properties of the model and of the boundary conditions: material, loads and constraints, optimization constraints and objective function (fig.77).

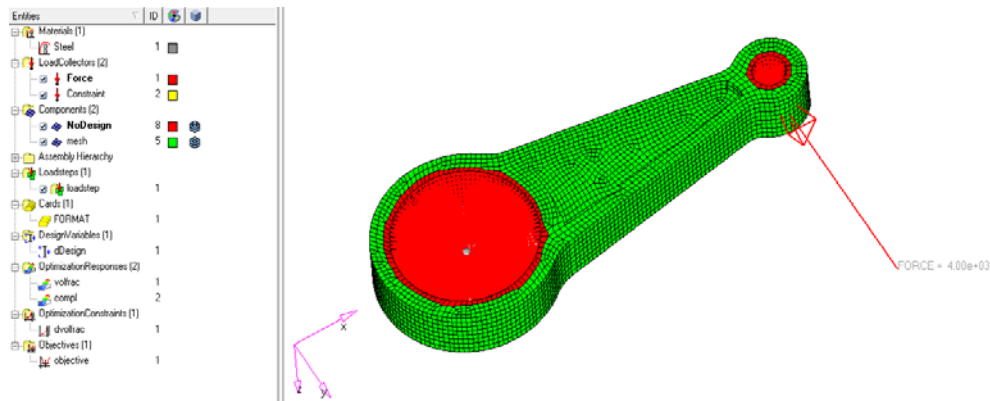


Figure 77: Definition of the material, loads and constraints, objective function.

Once the topological optimization phase is completed (fig.78a), it is possible to extract from the optimized model projections, profiles, points and reference planes and surfaces (fig.78b).

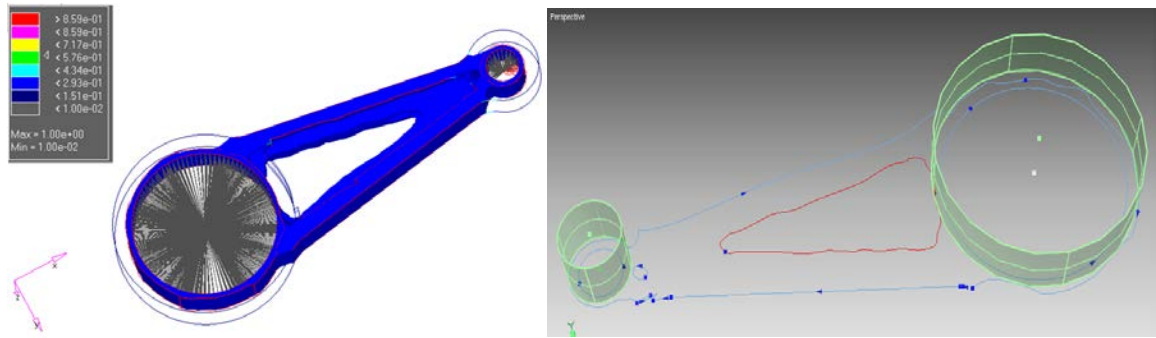


Figure 78: Topological optimization results (a); and projections and reference surfaces extracted from the optimized model (b).

As stated in section 4.2, during the redesign phase, the designer can use the archetype or the prototypes in order to ease and fast the interpretation of the results of the topology optimization analysis. First of all, as depicted in the following figure, he can compare the projections of the optimized model with the prototypes. In case there is a similarity the user can select the desired prototype and, in addition of the reference geometries extracted from the optimized model, takes advantage of this knowledge in order to fast the redesign of the final connecting rod (fig.80b).

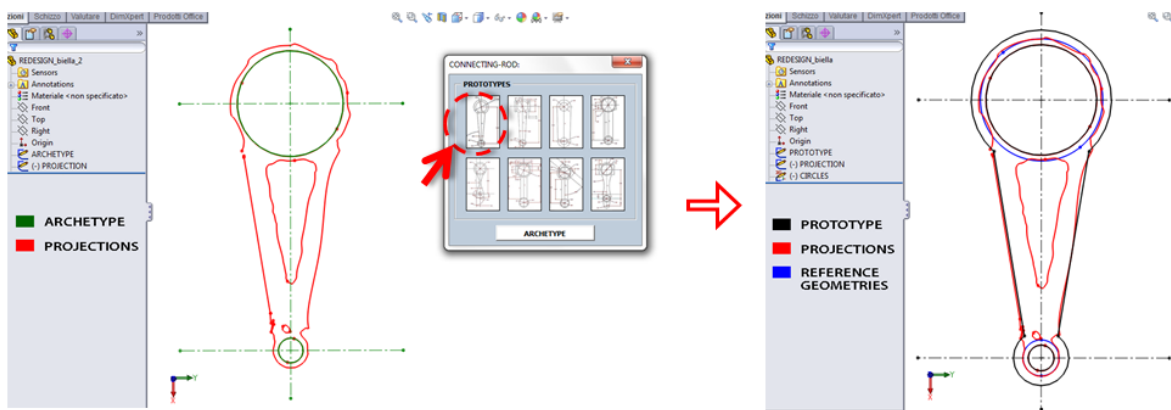


Figure 79: Use of the projections to interpret projections of the optimized model.

Otherwise the designer has the option to use the archetype as base of knowledge to start the redesign phase as depicted in figure 80a:

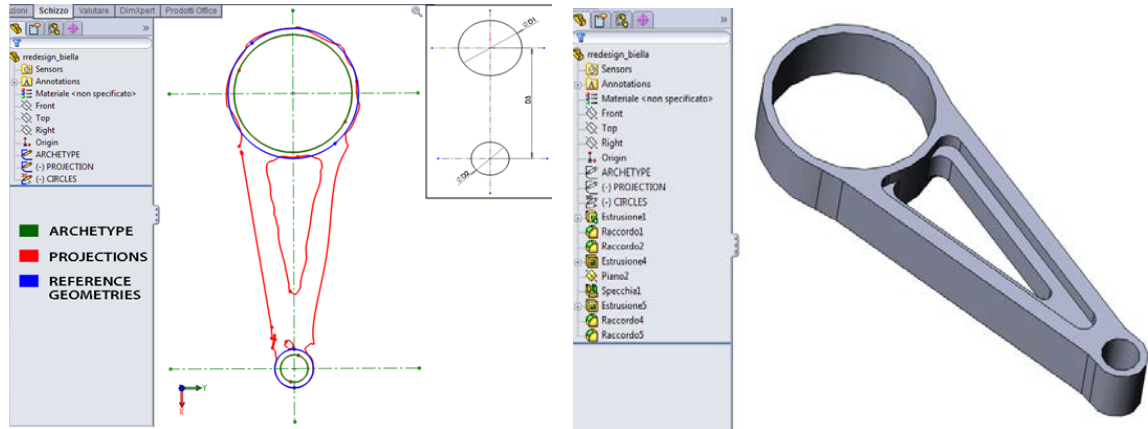


Figure 80: Use of the parametric archetype in the redesign phase (a). Final connecting rod 3D model (b).

The comparison between the results of FE analysis performed on the initial and final connecting rod shows a significant improvements in terms of weight and volume living almost invariant the stiffness of the mechanical component (table 4).

	Initial Connecting Rod	Final Connecting Rod
weight (g)	674.46	441.56
volume (mm³)	87592.82	57344.93
max stress(N/m²)	7,6989e+7	15,9373e+7
max displacement (mm)	0.2824	0.2935

Table 4: FE analysis results comparison.

Synthesis and conclusions

The present work has addressed the integration of Topology Optimization tools within the Product Development Process as a means to improve the innovation resources and the efficiency of a product development cycle. The rationale of this research is the lack of formalized and validated procedures allowing the systematic introduction and integration of these innovative tools in the design process. The development of these integration methods and tools, that are the most innovative aspect of the research, should provide advantages in terms of design costs reduction, errors reduction, product quality improvement, process execution time and more effective internal and external knowledge use and share.

A relevant aspect of the proposed methodology is that, through the definition of guidelines and best practices, can help overcoming classical interoperability problems affecting the design process, providing a common base for knowledge sharing and for a better interconnection between different systems and applications.

This work has been decomposed in two main categories: a methodology developed starting from the limits risen from the state of the art and the interoperability problems between TOs and the other CAx systems (part I) and the extension and improvement of this approach by introducing methods and tools that allow to overcome specific drawbacks of the initial methodology (part II).

In particular, in the first part, the difficulties due to the poor integration of TO system in PDP have been overcome by implementing different KB interfaces that automate operation sequences, thus fastening the designer's work, according to the typical CAD-Automation approach and, most of all, facilitate the extraction of useful geometrical information from the results provided by the optimizer in order to support and simplify the interpretation of the optimized models. It is important to underline that, thanks to the extension of the methodology with the introduction of a MB system (chapter 2), the proposed approach can efficiently support the selection of the optimum conceptual design of a mechanical component with complex dynamic behaviour, in particular when very little previous experience on the system is available.

Despite the promising preliminary results reported in the first part of the manuscript, the designer still plays a crucial role. Hence, in the second part of the manuscript, an improvement and extension of the methodology has been depicted in order to increase the level of integration with other CAx systems and, most of all, to reduce decision makings of the user. In detail, a novel algorithmic method for constrained B-spline fitting curves, based on a mechanical model of bar networks, has been presented in order to improve the efficiency and objectivity of the interpretation of curves and profiles extracted from the optimized models. The last chapter suggests the concept of the parametric archetype as a 2D functional description that contains a base of knowledge common to all versions of the same mechanical component. This knowledge can be shared and reused in different stages of the proposed methodology but especially during the redesign phase in order to ease and fast the interpretation of the projections of the optimized model.

All the arguments and reasoning described have been supported and validated with case studies. It is worth to note that the obtained results show better performances than those obtained by a classical optimization approach, thus confirming a contribution to the generation of an efficient solution that is ready to be adopted as a framework for the integration of actually available optimization software and PLM systems.

Bibliography

- [AE 2010] Altair Engineering. URL: <http://www.altair.com>.
- [Albers et al. 2002] Albers, A., Emmrich, D. and Haeussler, P. (2002) 'Automated Structural Optimization of Flexible Components using MSC.Adams/Flex and MSC.Nastran Sol200', 1th European MSC.ADAMS Users' Conf.
- [Albers et al. 2007] Albers, A., Ottnad, J., Weiler, H. and Haeussler, P. (2007) 'Methods for Lightweight Design of Mechanical Components in Humanoid Robots', IEEE-RAS 7th Int. Conf. on Humanoid Robots, Pennsylvania.
- [Argyris 1964] Argyris J.H., Recent advances in matrix methods of structural analysis Progress in Aeronautical Sciences (Pergamon Press, 1964).
- [Arlinghaus 1994] Arlinghaus S. L.; "Practical handbook of curve fitting". 1994
- [Barbieri et al. 2010] Barbieri, L., Bruno, F., Luchi, M.L., Malito, G., and Muzzupappa, M., 2010, "Knowledge Management in Topology Optimization of Automotive Components", Proceedings of the Eighth International Symposium on Tools and Methods of Competitive Engineering (TMCE 2010), Ancona, Italy.
- [Barbieri et al. 2009] Barbieri, L., Bruno, F., Luchi, M.L.; Malito, G., and Muzzupappa, M., 2009, "Design of Automotive Components Using Advanced CAE Systems", Proceeding of 21th European Modeling & Simulation Symposium (EMSS 2009), Tenerife, Canary Islands, September.
- [Barbieri et al. 2008a] Barbieri, L., Bruno, F., Cugini, U. and Muzzupappa, M. (2008) 'Design Automation tools as a support for knowledge management in topology optimization', Proceeding of the ASME 2008 International Design Engineering Technical Conferences & Computers and Information (IDETC08), New York.
- [Barbieri et al. 2008b] Barbieri, L., Bruno, F., Casini, G., Cugini, U., Muzzupappa, M. and Rotini, F. (2008) 'Guidelines for an efficient integration of Computer Aided Innovation with optimization and PLM systems', 2nd European HyperWorks Tech. Conf. (HTC08). Strasbourg, Germany.
- [Barbieri et al. 2008c] Barbieri, L; Bruno, F; Caruso, F and Muzzupappa, M. "Innovative integration techniques between Virtual Reality systems and CAx tools", The International Journal of Advanced Manufacturing Technology. Volume 38, Numbers 11-12 / October, 2008.

- [Barbieri et al. 2007] Barbieri, L., Bruno, F., Cugini, U. and Muzzupappa, M. (2007) 'CAD automation and KB user interfaces for an efficient integration of topological optimization tools in the product development process', 4th Int. Conf. on PLM, Brescia, Italy.
- [Barbieri et al. 2006] Barbieri L.; Bruno F.; Muzzupappa M.. "Towards the integration of Virtual Reality techniques in the Product Lifecycle Management systems"; In Proceedings of XVIII Congreso Internacional de Ingenieria Grafica, INGEGRAF 2006, Barcelona-Spain, June 2006.
- [Bendsoe et al. 1991] Bendsoe MP, Rodrigues HC. Integrated topology and boundary shape optimization of 2-D solid. *Comput Methods Appl Mech Eng* 1991;87:15–34.
- [Bendsoe et al. 2003] Bendsoe, M.P., Sigmund, O. (2003) 'Topology Optimisation. Theory, Methods and Applications', *Springer Verlag*.
- [Blender 2010] Blender. URL: <http://www.blender.org/>.
- [Boehm 1980] Boehm, W., Inserting new knots into B-spline curves, *CAD*, Vol. 12, pp. 199-201, 1980.
- [Boehm et al. 1985] Boehm, W., and Prautzsch, H., The insertion algorithm, *CAD*, Vol. 17, pp. 58-59, 1985.
- [Bremicker et al. 1991] Bremicker M, ChirehdastM, Kikuchi N, Papalambros PY. Integrated topology and shape optimization in structural design. *Mech Struct Mach* 1991;19:551–87.
- [Chernov et al. 2004a] N. Chernov , C. Lesort , N. Simányi, On the complexity of curve fitting algorithms, *Journal of Complexity*, v.20 n.4, p.484-492, August 2004.
- [Chernov et al. 2004b] N. Chernov, C. Lesort, Statistical efficiency of curve fitting algorithms, *Computational Statistics & Data Analysis*, p. 713-728, 2004.
- [Chiandussi et al. 2004] Chiandussi, G., Gaviglio, I. and Ibba, A. (2004) 'Topology optimisation of an automotive component without final volume constraint specification', *Advances in Engineering Software*, (pp. 609–617).
- [Chirehdast et al. 1994] Chirehdast M, Gea H-C, Kikuchi K, Papalambros PY. Structural configuration examples of an integrated optimal design process. *J Mech Des* 1994;116:997–1004.
- [Cohen et al. 1980] Cohen, E., Lyche, T., and Riesenfeld, R.F., Discrete B-splines and subdivision techniques in computer aided geometric design and computer graphics, *Comput. Graph. and Image Process.*, Vol. 14, pp. 87-111, 1980.

- [Cohen et al. 1985] Cohen, E., Lyche, T., and Schumacher, L.L., Algorithms for degree-raising of splines, ACM TOG, Vol. 4, pp. 171-181, 1985.
- [Coleman et al. 1994] Coleman, T.F. and Y. Li, "On the Convergence of Reflective Newton Methods for Large-Scale Nonlinear Minimization Subject to Bounds," Mathematical Programming, Vol. 67, Number 2, pp. 189–224, 1994.
- [Coleman et al. 1996] Coleman, T.F. and Y. Li, "An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds," SIAM Journal on Optimization, Vol. 6, pp. 418–445, 1996.
- [Corney 1993] Corney, J.R. (1993) 'Graph-Based Feature Recognition', Ph.D. dis., Heriot-Watt Univ.
- [Cox 1971] Cox, M.G., The numerical evaluation of B-splines, National Physical Laboratory DNAC 4, August 1971.
- [Cugini et al. 2007] Cugini, U., Cascini, G., and Ugolotti, U. (2007). "Enhancing interoperability in the design process, the PROSIT approach", Trends in Computer Aided Innovation ed. Noel Leon-Rovira, Springer Verlag.
- [Cugini et al. 2008] Cugini U., Cascini G., Muzzupappa M., Nigrelli V. (2008) 'Integrated Computer-Aided Innovation: the PROSIT approach', Special Issue on Computer-Aided Innovation of the Journal of Computers in Industry.
- [Cumnuantip et al. 2005] Cumnuantip, S., Spieck, M. and Krueger, W.R. (2005) 'An Approach for Sizing and Topology Optimization Integrating Multi-body Simulation', 46th ASME Structures, Structural Dynamics & Materials Conference, Austin, Texas.
- [De Boor 1972] De Boor C. (1972) 'On calculating with B-splines'. Journal Approx. Theory, Vol. 6, pp. 50-62.
- [De Boor 1978] De Boor C. (1978) 'A Practical Guide to Splines'. New York: Springer-Verlag.
- [de Casteljau 1986] de Casteljau P. (1986) 'Shape Mathematics and CAD'. London: Kogan Page.
- [Erickson et al. 2004] Erickson, G.W., Heath, M.R., Woods, B., Dolan, D.F., Sears, J.W. (2004) 'Design and Manufacture of Titanium Formula SAE Uprights using Laser-Powder-Deposition', SAE Paper N. 2004-01-3546.
- [FED 2010] FE-Design. URL: <http://www.fe-design.com>.
- [Fenves et al. 1963] Fenves S.J. and Branin F.H., Network topological formulation of structural analysis, Proc. of the ASCE, Journ. of the Struct. Division (August 1963) 483.

- [FSAEI 2010] FSAE International. URL: <http://students.sae.org/>.
- [GIMP 2010] GIMP. URL: <http://www.gimp.org/>.
- [Gordon et al. 1974] Gordon, W.J., and Riesenfeld, R.F., Bernstein-Bézier methods for the computer aided design of free-form curves and surfaces, *J. A CM*, Vol. 21, pp. 293-310, 1974.
- [Grimaud 2008] Grimaud J. (2008). Free form features semantic description model. Master research thesis, Ecole Nationale Supérieure d'Arts et Métiers (ENSAM). Aix-en-Provence.
- [Gründig 1985] Gründig, L., (1985), The FORCE-DENSITY - Approach and Numerical Methods for the Calculation of Networks. Proc. of 3. Intern. Symposium Weitgespannte Flächentragwerke, Stuttgart, March 1985.
- [Gründig et al. 2000] Gründig, L. Moncrieff, E., Singer, P. and Ströbel, D., (2000), 'A history of the principal developments and applications of the force density method in Germany 1970-1999,' in Papadrakakis, E. (Ed.), Proc. IASS-IACM 2000 Fourth International Colloquium on Computation of Shell & Spatial Structures, June 4-7, 2000, Chania-Crete, Greece.
- [Guillet et al. 1996] S. Guillet, F. Noel and J.C. Léon, Structural shape optimization of part bounded by free-form surfaces, *Structural Optimisation*, vol. 11(3/4), pp. 159–169, 1996. 19, 44, 75.
- [Guillet et al. 1998] S. Guillet and J.C. Léon, Parametrically deformed freeform surface as part of a variational model, *Computer Aided Design*, vol. 30(8), pp. 621–630, 1998.
- [Guillet 1999] S. Guillet, Modification et Construction de formes gauches soumises à des contraintes de conception, Ph. D. thesis, INP-Grenoble, 1999.
- [Guillet et al. 2000] S. Guillet and J.C. Léon, Deformation criteria for the direct manipulation of free form surfaces, *Curves & Surfaces*, edited by L. Schumaker and A. Le Mhaut, Saint-Malo, 2000.
- [Häussler et al. 2001] Häussler, P., Emmrich, D., Muller, O., Ilzhofer, B., Nowicki, L. and Albers, A. (2001) 'Automated Topology Optimization of Flexible Components in Hybrid Finite Element Multi-body Systems using ADAMS/Flex and MSC.Construct', ADAMS European User's Conf., Berchtesgaden, Germany.
- [Hawthorne 2005] Hawthorne and York, (2005) 'HYI-3D Design Optimization System.', Internal Documentation, Phoenix, AZ.
- [HM 2010] HyperMesh Tcl/Tk API. URL: <http://www.altairhyperworks.com>.

- [Hsu et al. 2001] Hsu YL, Hsu MS, Chen CT. Interpreting results from topology optimization using density contours. *Comput Struct* 2001;79:1049–58.
- [Hsu et al. 2005] Hsu, Y.L., Hsu, M.S. (2005) ‘Interpreting three-dimensional structural topology optimization results’. *Computers and Structures*, 83:327-337.
- [IEEE 1990] IEEE Standard Glossary of Software Engineering Terminology. 1990, IEEE Standard 610.12-1990, ISBN 1-55937-067-X.
- [Jawad 2002a] Jawad, B.A. and Baumann, J. (2002) ‘Design of Formula SAE Suspension’, SAE Paper N. 2002-01-3310.
- [Jawad 2002b] Jawad, B.A. and Polega, D.B. (2002) ‘Design of Formula SAE Suspension Components’, SAE Paper N. 2002-01-3308.
- [Kyprianou 1980] Kyprianou, L.K. (1980) ‘Shape Classification in Computer Aided Design’, Ph.D. dissertation, Christ College, Univ. Cambridge, Cambridge, U.K.
- [Kumar et al. 1996] Kumar AV, Gossard DC. Synthesis of optimal shape and topology of structures. *J Mech Des* 1996;118:68–74.
- [Laurent et al. 1993] Laurent-Gengoux, P., & Mekhilef, M. (1993). Optimization of a NURBS representation. *Computer-Aided Design*, 25(11), 699-710.
- [Léon et al. 1994] Léon J.C. and Trompette P. (1994). Controle de deformation de surfaces basé sur une approchemécanique. *Revue international de CFAO et d’infographie*, 9(1-2):41–55.
- [Léon et al. 1995] Léon J.C. and Trompette P. (1995). A new approach towards free-form surfaces control, *Computer Aided Geometric Design*, vol. 12, pp. 395–416.
- [Leyton 1987a] Leyton, M. (1987a) Symmetry-curvature duality. *Computer Vision, Graphics, and Image Processing*, 38, 327-341.
- [Leyton 1987b] Leyton, M. (1987b) A limitation theorem for the differential prototypification of shape. *Journal of Mathematical Psychology*, 31, 307-320.
- [Leyton 1988] Leyton, M. (1988) A process-grammar for shape. *Artificial Intelligence*, 34, 213-247.
- [Leyton 1989] Leyton, M. (1989) Inferring causal-history from shape. *Cognitive Science*, 13, 357-387.
- [Leyton 2007] Leyton, M. (2007) Process Grammar. DIMACS Technical Report 2007-09, Rutgers University, New Brunswick, NJ 08854, September 2007.

- [Lin et al. 2000] Lin C-Y, Chao L-S. Automated image interpretation for integrated topology and shape optimization. *Struct Multidiscip Optim* 2000;20:125–37.
- [Linkwitz et al. 1971] Linkwitz, K. and Schek, H.-J., (1971), 'Einige Bemerkungen zur Berechnung von vorgespannten Seilnetzkonstruktionen,' *Ingenieur-Archiv* 40, 145-158.
- [Ma et al. 2003] Ma, Y.S., Tong T., 2003, "Associative feature modelling for concurrent engineering integration", *Computers in Industry*, Vol. 51(1), pp.51-71.
- [Mandorli et al. 2001] Mandorli, F., Rizzi, C., Susca, L., Cugini, U., 2001, "How to Implement Feature-Based Applications using KBE Technology", *Proc of the International IFIP Conference on Feature Modeling and Advanced Design-For-The-Life-Cycle Systems, FEATS*.
- [Mathworks 2010] Mathworks. URL: <http://www.mathworks.com/>.
- [Mervyn et al. 2004] Mervyn F., Senthil Kumar A., Bok S.H., Nee A.Y.C. (2004). Developing distributed applications for integrated product and process design, *Computer-Aided Design* 36 (July (8)) 679–689.
- [Milliken et al. 1995] Milliken, W.F., Milliken, D.L. (1995) 'Race Car Vehicle Dynamics', SAE Publishing.
- [Mlejnek 1992] Mlejnek HP. Some aspects of the genesis of structures. *Struct Optim* 1992;5:64–9.
- [MSC 2010] MSC.Software. URL: <http://www.mscsoftware.com>.
- [Murali et al. 2006] Murali, M.R. and Krishn, Y. (2006) 'Topology and Classical Shape Optimization of a Lower Control Arm. A Case Study', *Proceedings of IDETC/CIE, Philadelphia, Pennsylvania*.
- [Nocedal et al. 2006] Nocedal J. and Wright S.J. (2006) *Numerical Optimization, Second Edition*. New York: Springer.
- [Noel 1994] F. Noel, *Mailleur auto-adaptatif pour des surfaces gauches en vue de la conception intégrée*, Ph. D. thesis, INP-Grenoble, 1994. 44, 75, 93, 123.
- [Olhoff et al. 1991] Olhoff N, Bends/e MP, Rasmussen J. On CAD-integrated structural topology and design optimization. *Comput Methods Appl Mech Eng* 1991;89:259–79.
- [Ozturk et al. 2006] Ozturk, N; Yıldız, AR; Kaya, N and Ozturk F: 2006, *Neuro-Genetic Design Optimization Framework to Support the Integrated Robust Design Optimization Process in CE, Concurrent Engineering: Research and Applications*, 14(1): 5–16.

- [Papalambros et al. 1990] Papalambros P, Chirehdast M. An integrated environment for structural configuration design. *J Eng Des* 1990;1: 73–96.
- [Pernot et al. 2003a] J.P. Pernot, S. Guillet, J.C. Lèon, B. Falcidieno, F. Giannini, A new approach to minimisations for shape control during freeform surface deformation, ASME 2003, DETC-DAC48777, Chicago, Illinois, USA, 2003.
- [Pernot et al. 2003b] Pernot, J-P., Guillet, S., Leon, J-C., Falcidieno, B., & Giannini, F. (2003). Interactive operators for free form features manipulation. In SIAM conference on CADG, Seattle, 2003.
- [Pernot et al. 2004a] J.P. Pernot, S. Guillet, J.C. Lèon, B. Falcidieno, F. Giannini, Multi-minimisations for shape control of fully free form deformation features, Proceeding of the IEEE Shape Modelling International Conference, Genova, Italy, pp. 53–62, 2004.
- [Pernot et al. 2004b] J.P. Pernot, S. Guillet, J.C. Lèon, B. Falcidieno, F. Giannini, Interactive operators for Free Form Features manipulation, book of the SIAM Conference on Geometric Design & Computing, 10-13 November 2003, Seattle, USA.
- [Piegl et al. 1994] Piegl, L., and Tiller, W., Software engineering approach to degree elevation of B-spline curves, *CAD*, Vol. 26, No. 1, pp. 17-28, 1994.
- [Piegl et al. 1995] Piegl, L., and Tiller, W., Algorithm for degree reduction of B-spline curves, *CAD*, Vol. 27, No. 2, pp. 101-110, 1995.
- [Piegl et al. 1997] L. Piegl and W. Tiller, *The NURBS Book*, Springer, New-York, 2nd edition, 1997.
- [Prautzsch 1984a] Prautzsch, H., A short proof of the Oslo algorithm, *Comput. Aid. Geom. Des.*, Vol. 1, pp. 95-96, 1984.
- [Prautzsch 1984b] Prautzsch, H., Degree elevation of B-spline curves, *Comput. Aid. Geom. Des.*, Vol. 1, No. 1, pp. 193-198, 1984.
- [Prautzsch et al. 1991] Prautzsch, H., and Piper, B., A fast algorithm to raise the degree of spline curves, *Comput. Aid. Geom. Des.*, Vol. 8, pp. 253-265, 1991.
- [Prosit 2008] “MUR/PRIN-Prosit” project. URL: www.kaemart.it/prosit.
- [PT 2010] Pro/TOOLKIT Customization API. URL: <http://www.ptc.com>.
- [PXS 2010] PLM XML SDK. URL: <http://www.plm.automation.siemens.com>.
- [Renka 2009] Robert J. Renka, Algorithm 893: TSPACK: tension spline package for curve design and data fitting, *ACM Transactions on Mathematical Software (TOMS)*, v.36 n.1, p.1-8, March 2009.

- [Renka 1993] R. J. Renka, Algorithm 716; TSPACK: tension spline curve-fitting package, ACM Transactions on Mathematical Software (TOMS), v.19 n.1, p.81-94, March 1993.
- [Riesenfeld 1973] Riesenfeld, R.F., "Application of B-spline Approximation to Geometric Problems of Computer Aided Design," Ph.D. dissertation, Syracuse University, Syracuse, NY, 1972. Also available as University of Utah UTEC-CSc-73-126, March 1973.
- [Rosen et al. 1992] Rosen, D.W., Grosse, I.R., 1992, "A feature based shape optimisation technique for the configuration and parametric design of flat plates", Eng. with Computers, Vol. 8, pp.81-91.
- [Rogers et al. 1989] Rogers, D.F. and Fog, N.G. (1989). 'Constrained B-Spline Curve and Surface Fitting', Computer Aided Design, 21[10], 641,648.
- [Rozvany et al. 1995] Rozvany, G.I.N.; Bendsøe, M.P.; Kirsch, U. 1995: Layout optimization of structures. Appl. Mech. Rev. (ASME) 48, 41–119.
- [RF 2010] RapidForm. URL: <http://www.rapidform.com>.
- [ROED 1996] Research Opportunities in Engineering Design: NSF Strategic Planning Workshop Final Report. 1996, National Science Foundation.
- [Ruas et al. 1993] Ruas A. and Lagrange J.P. (1993). Etat de l'art en généralisation & survey on generalization. Technical Report DT-93-0538, IGN-France.
- [Salomon 1999] Salomon D. (1999). Computer Graphics and Geometric Modeling. New York, Springer-Verlag.
- [Salomon 2006] Salomon D. (2006). Curves and surfaces for computer graphics. New York, Springer-Verlag.
- [Sanchez et al. 2006] Sanchez J., Serna M.A., Morer P. (2006) 'Using NURBS Fitting Techniques To Represent Tensile Structures'. Adaptables2006, TU/e, International Conference On Adaptable Building Structures. Eindhoven The Netherlands 03-05 July 2006.
- [Sauter et al. 2001a] Sauter, J., Meske, R. and Gungor, Z., (2001) 'Recent improvements in topology and shape optimization and the integration into the virtual product development process', Proc. of the NAFEMS World Congress, 24-28 April 2001.
- [Sauter et al. 2001b] Sauter, J., Starlinger, A., Golzer, P., Binderszewsky, J. (2001) 'Industrial Applications of Topology and Shape Optimization with Tosca and Abaqus', 2001 Abaqus users' conference.

- [Saux 1998] Saux E. (1998) 'B-spline Curve Fitting: Application to Cartographic Generalization of Maritime Lines. International Conference Graphicon 1998, Moscow, Russia.
- [Schek 1974] H.J. Schek, The force density method for form finding and computation of general networks, *Comp. Meth. in App. Mech. and Eng.*, vol. 3(2), pp. 115–134, 1974.
- [Schoenberg 1946] Schoenberg, I.J., Contributions to the problem of approximation of equidistant data by analytic functions, *Q. Appl. Math.*, Vol. 4, pp. 45-99; pp. 112-141, 1946.
- [Schramm 2003] Schramm, U. (2003) 'Recent Advances in the Application of Topology Optimiz. in Design'.
- [Schramm 2005] Schramm, U. (2005) 'Optimization Technology in Design – Trends, Direction, Gaps', 6th World Congresses of Structural and Multidisciplinary Optimization, Rio de Janeiro, Brazil.
- [Schumacher 2005] Schumacher, G. (2005) 'Concept Design by Optimization Methods', A350 Fuselage Tail Section 19 - German OptiStruct Users' Meeting.
- [Smith 1978] Smith, C., 1978. *Tune to Win*, Aero Publishers, Inc. Distributed by SAE.
- [Spath et al. 2001] Spath, D., Neithardt, W. and Bangert, C. (2001) 'Integration of topology and shape optimization in the design process'. In: *International CIRP Design Seminar*, Stockholm.
- [Staniforth 2006] Staniforth, A., 2006. *Competition Car Suspension: A practical handbook*, Haynes Publishing.
- [SW 2010] SolidWorks. URL: <http://www.solidworks.com>.
- [SWOA 2010] SolidWorks Open API and Catia CAA v5. URL: <http://www.3ds.com/>.
- [Szykman et al. 2001] Szykman S., Fenvesa S.J., Keirouzb W., Shooter S.B. (2001). A foundation for interoperability in next-generation product development systems, *Computer-Aided Design* 33 (June (7)) 545–559.
- [Tang et al. 2001] Tang P-S, Chang K-H. Integration of topology and shape optimization for design of structural components. *Struct Multidiscip Optim* 2001;22:65–82.
- [Thornton et al. 1967] Thornton C.M. and Birnstiel C., Three-dimensional suspension structures, *Proc. AXE, Journ. of the Struct. Div.* (1967) 247.
- [Tiller 1992] Tiller, W., Knot-removal algorithms for NURBS curves and surfaces, *CAD*, Vol. 24, No. 8, pp. 445-453, 1992.

- [Vaidya 2006] Vaidya, A., Yu, S.H., Ville, J.S., Nguyen, D.T., Rajan, S.D. (2006) 'Multiphysics CAD-Based Design Optimization', *Mechanics Based Design of Structures and Machines*, 34: 157–180.
- [Vandenbrande et al. 1993] Vandenbrande, J.H. and Requicha, A.A.G. (1993) 'Spatial reasoning for the automatic recognition of machinable features in solid models', *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 15 (12), pp.1-17.
- [Véron et al. 1997] P. Véron and J.C. Léon, Semi-global deformation and correction of free-form surfaces using a mechanical alternative, *The Visual Computer*, vol. 13, pp. 109–126, 1997.
- [Véron et al. 1998] P. Véron, P. Trompette and J.C. Léon, Integrated design and collaborative engineering of fabric structures, *Engineering with Computers*, vol. 14, pp. 23–35, 1998. 44, 75, 93.
- [Wings3D 2010] Wings3D. URL: <http://www.wings3d.com/>.
- [Wolfram Research 2010] Wolfram Research. URL: <http://www.wolfram.com>.
- [Yang et al. 1994] Yang R.J. and Chuang C.H. (1994). Optimal topology design using linear programming. *Comput Struct* 1994; 52(2):265–75.
- [Yang et al. 2008] Yang J. and Byun H. (2008). Curve fitting algorithm using iterative error minimization for sketch beautification. 19th International Conference on Pattern Recognition, 2008. ICPR 2008.
- [Yildiz et al. 2003] Yildiz, AR; Ozturk, N; Kaya, N and Ozturk, F: 2003, Integrated optimal topology design and shape optimization using neural networks. *Structural and Multidisciplinary Optimization*, 25(4): 251–260.
- [Youn et al. 1997] Youn S.K. and Park S.H. (1997). A study on the shape extraction process in the structural topology optimization using homogenized material. *Comput Struct* 1997;62(3): 527–38.