CAD-CAE integration and isogeometric analysis: trivariate multipatch and applications

Marco Brino

Politecnico di Torino, Department of Management and Production Engineering
Corso Duca degli Abruzzi, 24 – 10129 Torino, Italy

Final Dissertation

Doctoral School in
Production Systems and Industrial Design

Supervisor: Dr. Elvio Bonisoli

January 2015

Cycle XXVII

Ai miei genitori
per l'orgoglio che provano.

To my parents
for the pride they feel.

Index

Abstract

This PhD thesis is focused on the issues related to one of the critical steps during the lifecycle of a product design and manufacturing process: the transition between the geometrical and functional definition of a product, and the virtual prototyping with numerical simulations, also known as CAD (Computer-Aided Design) / CAE (Computer-Aided Engineering) transition.
The *isogeometric* methodologies developed by Hughes et Al. [1, 2] has the ambition to close the gap in CAD/CAE integration, allowing the two environments to underlay on the same framework, taking advantage of the *isoparametric* concept, widely used in Finite Elements world, coupled with the Non-Uniform Rational B-Splines (NURBS) that are a standard in CAD systems in the mathematical representation of geometries.
Even though the first paper was published 10 years ago, the method is not yet used in industrial applications and only few commercial software are able to handle isogeometric elements.
In this thesis a step towards the possibility of application in industry by developing a multi-patch coupling method where the geometry at the interface does not allow a compatible mesh.
This improvement opens new frontiers for applications in both static and dynamic solutions.
Another issue that is analysed in this thesis is the possibility to improve the geometry-to-analysis integration by conversion of the information that comes from CAD software, in terms of representation of the external surfaces, to solid information that is necessary to be suitable for a structural simulation.

Introduction

CAD-CAE integration is an applied research field where the aim is to tighten or, at best, eliminate the existing gap between the geometrical design of a component or structure and its virtual verification.
One emerging methodology that has shown an actual possibility of merging the two environments in the same framework is named isogeometric analysis (IGA) ([1], [2]) which takes advantage of the widely used NURBS ([3], [4]) mathematical definition of geometries, now a standard in the CAD programs.
IGA have shown several advantages, with respect to standard finite elements using Lagrange basis functions, that for example, in the structural mechanics field, are smoothness of the displacement and stress field across the element boundaries and the possibility of raising the order of the functions with the result of gaining smoothness [5], more stable and accurate results in the solution of contact problems [6]-[10] because of the exact (and smooth) computational domain, in particular due to the absence of gaps or overlaps and, thanks to its relationship with CAD, the possibility of allowing a direct topological optimisation [11]-[14].
The properties of gaining accuracy on the increasing of the order of the functions (an opposite behaviour with respect to standard finite elements) and the positiveness of the entries of the mass matrix (due to non-negative values in NURBS basis functions) are advantageous in the field of structural vibrations and transient dynamics problems [15]-[17].
An accurate study on the mathematical properties of IGA was carried out by Evans et al. [18].
Some of the disadvantages of NURBS-based IGA, which are topics of further study and development, are the lack of local refinement possibility and the need to represent complex models with a large number of patches. One possibility that emerged is represented by T-Splines [19]-[20], which is a mathematical representation of geometry that has the aim of replacing NURBS in CAD software packages, but a some research teams also developed a tri-variate version.

Since T-Splines are still not standard, other than mathematically hard to implement, several research activities are focused on using NURBS but the problem of connecting together a large number of patches that could happen to have non-matching mesh at the interface needs solution.
The parameterisation plays a fundamental role in the quality of the physical mesh, as stated in [21], [22] and in particular [23], where the concept of "analysis-aware modelling" is presented.
Nitsche's method [24] was developed for imposing in boundary conditions in weak form, and an extension can perform the coupling of different domains. The method is in between the Lagrange multipliers and the penalty method, and it takes the advantage of both, namely the consistency of the Lagrange multiplier approach, and the relative ease of implementation and parameter selection of the penalty method.

In this dissertation isogeometric analysis with Nitsche's coupling method is also applied to linear modal analysis for the investigation of *crossing* and *veering* phenomena [25], which occur among the frequency loci of the resonant conditions of a structure when a particular parameter, such as a dimension or the relative position among the components of a structure, is changed and this variation causes interaction among the different modes. These

phenomena are well nown in the aeroelasticity in long-span bridges [26], bladed discs [27], and aircraft wings [28]-[30].

The dissertation is organised with the statement of the CAD-CAE integration issue in Section 1, where the CAD geometrical representation is presented against to the mathematical framework of the Finite Element Method, in Section 2 both B-splines and NURBS are analysed with their properties, in Section 3 the isogeometric analysis concepts are presented and applied on examples composed by single-patch and with particular features.
Section 4 considers in detail the Nitsche's method, applied to the coupling of different domains, describing the case of the discretisation in finite elements and some implementation remarks, while in Section 5 the method is applied to case tests of increasing complexity and of increasing practical and industrial interest.
Section 6, analyses some added studies on parallel tests on the application of topological optimisation and the conversion from a boundary surface representation of a patch to a single tri-variate description.

The computations for the IGA NURBS models are performed in MATLAB using an isogeometric code developed by the author for these purposes, taking inspiration from both GeoPDEs [31] and MIGFEM [32], and with the help of V.P. Nguyen already developer of MIGFEM.
The geometries involved are structured data to fit the NURBS Toolbox [33], whose routines are mainly used to create or edit all the geometries used.

1. Isogeometric analysis concepts

1.1. The CAD/CAE integration issue

The main differences between CAD and CAE environments and thus the issues that cause the gap between them are summarised.

In CAD the goal is to mathematically define the external shape of an object. CAD can manage its dimensions and the interaction with other objects if, for example, the model is an assembly of components which form a system that, taking in consideration the mechanical field, performs a function. Then CAD could help designing geometrically functional systems. There are different ways to mathematically identify a shape in space and they are divided in three categories (as explained in detail in [34]):

- explicit functions;
- implicit functions;
- parametric functions.

In the first case the geometry is defined directly in the physical reference frame (x, y, z) and the functions are explicitly defined

$$y = f(x) \tag{1.1}$$
$$z = f(x,y) \tag{1.2}$$

but the main drawback of this choice is that equation (1.1) and (1.2) cannot represent closed or multivalued geometries. To overcome this drawback, implicit functions can be considered

$$F(x,y) = 0 \tag{1.3}$$
$$F(x,y,z) = 0 \tag{1.4}$$

Even though it is now possible to represent any geometry, other drawbacks have to be taken into account, such as the need of finding the roots, if a slope is near to be vertical, its value tends to infinity and the object is directly linked to the coordinate system.

In the parametric function case there is a *mapping* between two spaces, the *parametric space* and the *physical* (or *Cartesian*) *space*

$$C(x,y) = f(s) \tag{1.5}$$
$$S(x,y,z) = f(s,t) \tag{1.6}$$

where s and t are the parametric coordinates, in the ranges $s_{\min} < s \leq s_{\max}$ and $t_{\min} < t \leq t_{\max}$.

In computational geometry the parametric functions are more convenient and suitable for resolving situations like finding intersection conditions, and no need of roots computation is necessary.

B-Spline and NURBS are parametric functions and they are now widely used, if not even considered as a standard, in CAD environments.

In CAE the goal is to approximate the mechanical fields, such as displacement and stress, using some known functions that fit the *patch test* request (e.g. exactly represent the rigid body motions and the constant strain). These functions holds inside a certain domain called the *Finite Element*, and several finite elements connected together form the *mesh*.
These elements have polygonal shapes (such as triangles and quadrilaterals, tetrahedra and hexahedra) and are somehow distorted to approximate as close as possible the geometry of the object to be simulated.
The functions used are usually polynomials, where the simplest is the linear function

$$f(x) = c_0 + c_1 x \tag{1.7}$$

that interpolates two *nodes* at the beginning and the end of the element.
But for more complex elements (with eventually an increase of the order of the functions) the standard is the Lagrange interpolator function

$$f(x) = \prod_{\substack{0 \le m \le k \\ m \ne j}} \frac{x - x_m}{x_j - x_m} \tag{1.8}$$

These considerations lead to the conclusion that no general integration seems to be possible and then the two environments stay separated from the moment where the geometry is subdivided into finite elements.

From the practical point of view, considering the actual issue in an industrial scale, the workflow of a design-through-analysis process is represented by the scheme in Figure 1.1.

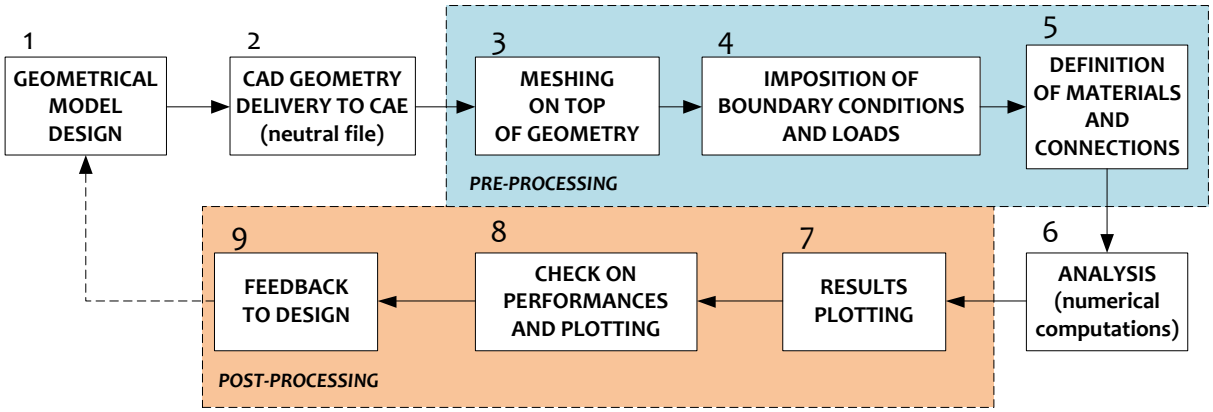


Figure 1.1 – Timeline of CAD and CAE during transition.

The scheme is commented in detail:
1) 3D geometrical model design – CAD systems are used to design the products from the geometrical point of view; both functional features and dimensions are defined taking into account performance properties;
2) CAD geometry delivered to CAE – A functionally-suitable version of the geometry is defined; this model can be delivered to the verification step through *neutral files*; in most cases CAD and CAE computer environments are different and native files usually cannot be directly read from each other, thus neutral files allow this exchange of information;

3) Meshing on top of geometry – The imported geometry is model on which the mesh is generated, where the mesh is the set of Finite Elements (triangles, quadrilaterals, tetrahedra, hexahedra) that constitute the domain of the mathematical model; the mesh is build defining a set of points on the edges of the geometry, then other point are placed on the external surfaces and connected together with planar elements, then other points are placed in the inside of the model and connected to generate the 3-dimensional elements;
4) Definition of materials and connections – After the domain of interest is defined, the necessary but not sufficient information to include is the one related to the elastic and inertial properties of the material involved; in case an assembly of components is considered, the relative positions and connections among the parts must be included as well;
5) Impositions of boundary constraints and loads – Besides some specific cases, such as free-free modal analysis, to solve a Finite Element model, all rigid body motions must be avoided and, thus, the model needs to be at least iso-statically constrained; usually performances under certain load conditions have to be verified, also these loads must be included in the finite element set of boundary conditions;
6) Analysis – The actual step where the computations occur and the system is solved in order to find the values of the unknown variables (Degrees of Freedom) are obtained as results, such as displacements, temperature, resonance frequencies …
7) Results plotting – Results are usually numbers arranged in vectors or matrices and even though a good engineer could be able to comment and come to conclusion only watching these numbers, the most practical and reliable way to interpret the results is to plot graphs, charts and colour contours; not only the rough results can be visualised, but also manipulated values such as stress and deformation fields from displacements, elastic strain energy from eigenvectors are produced, with the advantage of quick understanding;
8) Check of performances and comments – The graphical plots are not enough and engineers have to clearly point out comments on the results and choose whether a product or a system meets the required performance specifications or not; critical zones are emphasised to predict possible issues and focus the attention on those zones;
9) Feedback to design – The reports with results and comments are delivered to the design to confirm the required specifications are met or whether the component/system needs modifications, with suggestions on solutions.

Depending on the comments in the last step, a loop with the connection to the first step might be necessary, involving the need to modify the geometry and to follow all the steps of the workflow again. In terms of time, usually the transition from step 1 to step 3 is the most critical, even for simple components, and thus even more for complex parts or assemblies.

In Figure 1.2 a detail of the design-through-analysis transition is shown, with the example of a case where the CAD-CAE integration issue occurs.
The CAD team completes the modelling of a system and then delivers the final geometries to the CAE team, in the form of neutral files. Once CAE receives the data, the definition of the mesh can be started, but it does not only involve the generation of the domain with finite elements, but with the increase of complexity not all the geometrical features are suitable to be approximated with elements: a *defeaturing* step is applied, where come of the features are simplified or removed, and an equivalent mathematical entity is considered to take into account its contribution.

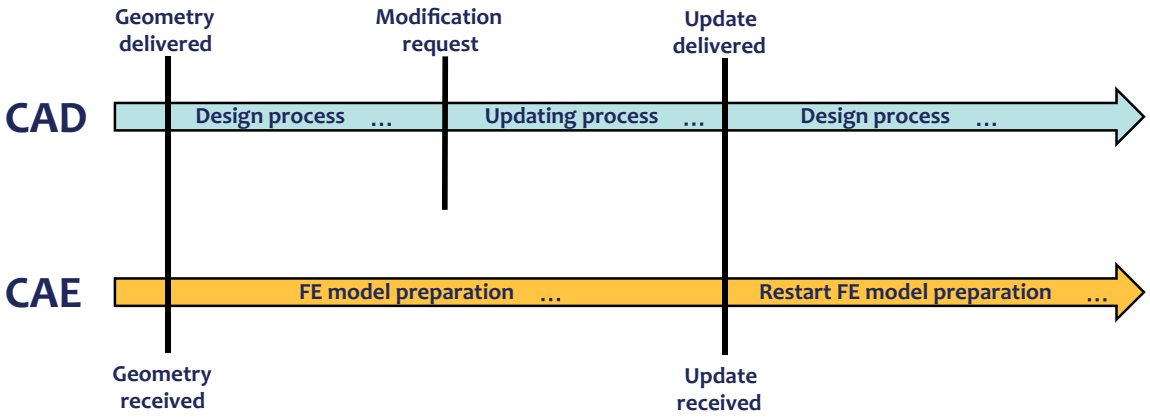


Figure 1.2 – Timeline of CAD and CAE during transition.

Classical cases are the removal of the fastening entity, that are substituted with rigid connections, small features like rounds and fillets and, in case of a modal analysis where only the global modes at lower frequency are interested, the simplification can be even deeper.
The advantage of defeaturing is to help the meshing process, to reduce the number of nodes and elements, to remove possible issues of small entities that, in order to fully replicate the geometry, feature severely distorted elements with the risk of obtaining bad results.
In the meantime these operations are performed, a modification on the geometrical model may occur, and thus a new geometry is delivered to the CAE team again.
In case the amount of modifications to be done on the mesh goes beyond the possibility of directly modify the already generated mesh to fit the modification requirements, the whole FE model generation should be started again from scratch.

The whole CAD-CAE integration issue resides on this drawback and then the need of the development of some methods that could, in the future, allow a geometrical design and a mathematical model for simulation to work with the same data and, eventually, in a unique model.

1.2. From isoparametric to isogeometric

A particular kind of element was developed with the intention to feature flexibility to the finite elements, other than taking advantage of the *Gauss quadrature numerical integration*: the *isoparametric element*.
This element is a mapping element from two different domains, which are the *parent element* where the quadrature points are defined and the physical element with its position in space (Figure 1.3).

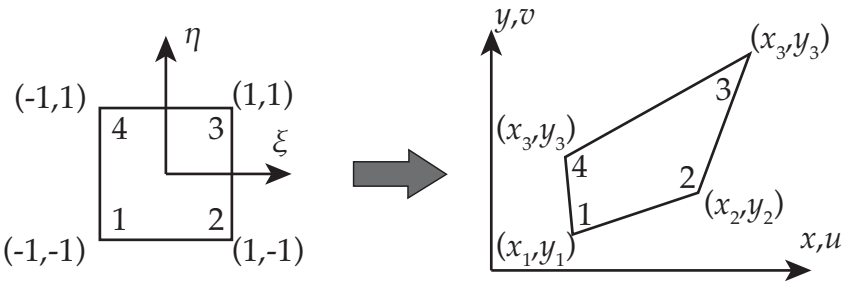


Figure 1.3 – Parametric (and parent) domain (left) and the relative Cartesian mapping.

This kind of element has an interesting propriety, which is the equality of the geometrical definition of the element and the function that approximate the displacement field.
A simple example to better understand the isoparametric concept is the 2-noded bar element in which is defined a mapping between the Cartesian coordinates and a *natural coordinate* that is the parametric space. The difference is illustrated in Figure 1.4.

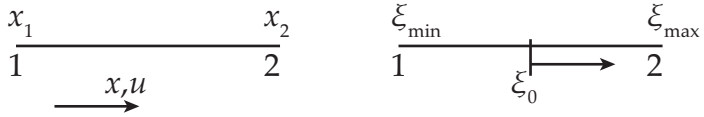


Figure 1.4 – Isoparametric representation of a 1-D bar element.

In the Cartesian system, the origin is placed in correspondence of node 1, and then node 2 has coordinates $(L,0)$.
In natural coordinates the origin is chosen to be placed in the middle of the bar, with a range $\xi_{\min} < \xi \leq \xi_{\max}$ that is chosen to be in the range $[-1,+1]$ to have a direct relationship with the position of the Gauss points for the numerical integration, that have the same domain range.
The mapping is then the transformation function that allows the translation from the coordinates in ξ to the coordinates in x, y

$$\begin{aligned} 0 &= f(\xi = 1) \\ L &= f(\xi = 1) \end{aligned} \tag{1.9}$$

and then the function for the transformation is

$$f(\xi) = \frac{L}{2}\xi + \frac{L}{2} \tag{1.10}$$

Even though Equation 1.10 is right, it is still case-specific. To obtain something that is general, the concept of *basis functions* in parametric space that are combined with coordinates of point in Cartesian space are introduced. Each basis function has the properties to result a value of 1 for the correspondent coordinate and a value of 0 for all the other coordinates. For the linear bar element the functions are trivially linear and then they have the form

$$f(\xi) = \sum_{i=1}^{n} N_i \mathbf{P}_i \tag{1.11}$$

that, in the specific case, is

$$\begin{aligned} N_1 &= \frac{1-\xi}{2} \\ N_2 &= \frac{1+\xi}{2} \end{aligned} \tag{1.12}$$

that is the geometrical definition of the elements in all its boundary and internal points.
The same function is used in finite element environment to approximate the displacement field by just substituting the coordinates of the points with the Degrees of Freedom, which are the displacements of that point in their Cartesian coordinates

$$\mathbf{u}(\xi) = \sum_{i=1}^{n} N_i \mathbf{d}_i \tag{1.13}$$

This concept is theoretically applied to any geometrical functions and, then, to B-spline and NURBS functions as well, as far as they fit the *patch test* requirements.

The general idea, then, is to take advantage of this concept to try integration between CAD and CAE, using directly the B-spline and NURBS functions that define mathematically the geometry of an object, as basis functions in a finite element analysis.

2. B-spline and NURBS

In this section B-splines and NURBS are presented with their properties and some examples of geometries with their mathematical representation, in order to understand their features and how these functions could help in the CAD-CAE integration.
NURBS are an evolution of B-splines, thus the latter is explained first.
B-splines curves are parametric functions that are defined by two ingredients: the *knot vector* (for the parametric domain) and the *control points* (for the physical domain).
The knot vector is a set of coordinates, in a non-decreasing order, in the parametric domain defined as $\Xi = \{\xi_1 \quad \xi_2 \quad \cdots \quad \xi_{n+p+1}\}$ where n is the number of basis functions and p is the polynomial degree. The space between two subsequent knots is called *knot span*. The basis functions are defined using the recursive formula

$$N_{i,0}(\xi) = \begin{cases} 1 & if\ \xi_i \leq \xi \leq \xi_{i+1} \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

$$N_{i,p}(\xi) = \frac{\xi \quad \xi_i}{\xi_{i+p} \quad \xi_i} N_{i,p\ 1}(\xi) + \frac{\xi_{i+p+1} \quad \xi}{\xi_{i+p+1} \quad \xi_{i+1}} N_{i+1,p\ 1}(\xi) \tag{2.2}$$

This formula is different from the order 0 (Equation 2.1), that involves the constant values, while for all subsequent orders the formula considered is Equation 2.2. It can be noticed that for each order the function depends by the functions at i and $i+1$, both at the previous order.
The most common knot vector used in CAD is the *non-uniform open knot vector*, which calls two properties in one:

- non-uniform – the space between the knots is not equal
- open – the first and last knots are repeated $p+1$ times.

One example of this kind of knot vector is

$$\Xi = \{0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 4 \quad 5 \quad 5 \quad 5\} \tag{2.3}$$

where it can be noticed that is not uniform and the feature of repeated first and last knots. Usually the knots are normalised such that they resides in the range [0, 1].
The dependency tree for the basis functions for the knot vector in 2.3 is the following

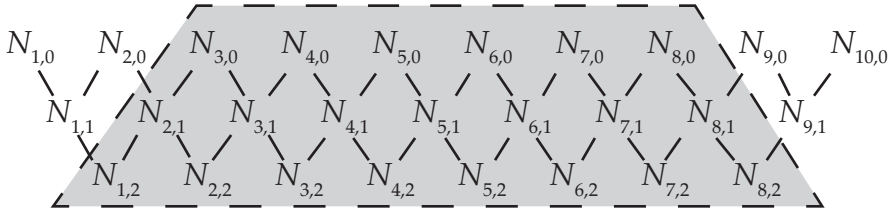


Figure 2.1 – B-spline basis functions dependency tree with highlighted non-zero functions.

$N_{i,0}$ are the constant values ad degree 0, and their support are in the range $[i, i+1]$. One example of the constant function is shown in Figure 2.2.

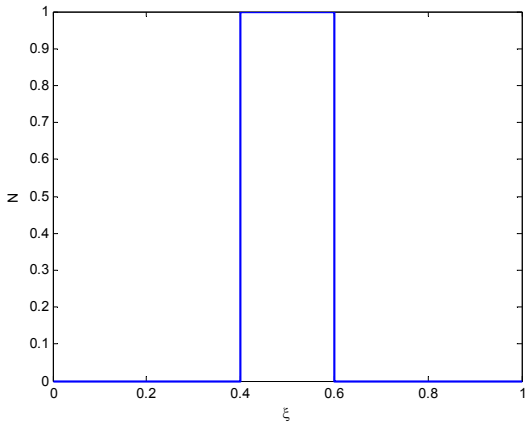


Figure 2.2 – Constant basis function $N_{5,0}$.

For all the functions $N_{i,1}$ the shape are linear and each function features the well-known 'hat' shape, that is the shape functions of the standard FEM linear element (example in Figure 2.3).

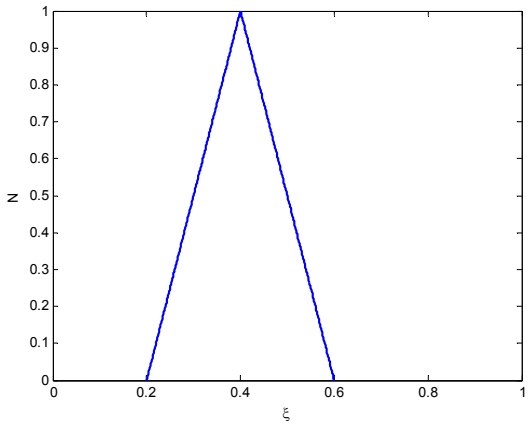


Figure 2.3 – Linear basis function $N_{4,1}$.

From the dependency tree in Figure 2.1, $N_{4,1}$ depends from $N_{4,0}$ and $N_{5,0}$, and then its support is from 1/5 to 3/5 ($[i, i+2]$).

The function of order greater than one feature a shape that is similar to the 'bell' curve, as is it shown in Figure 2.4. Even though also standard FEM has second-order functions, they have the property of return 1 for node $j = i$, and return 0 for any different node. This cause some functions to oscillate and then that the function goes below the value 0 and thus featuring negative values.
B-spline basis function, instead, have basis functions that are always non-negative regardless of the order of the functions.

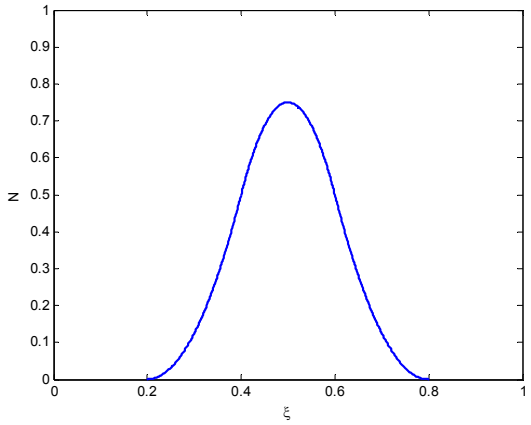


Figure 2.4 – Quadratic basis function $N_{3,2}$.

This time the dependency tree (Figure 2.1) shows that the most quadratic functions depend from two function of order one and three constants. In this case the support of these functions are in the range $[i, i+3]$, and obtaining generally that each function is supported in the range

$$\text{function support range} = [i, i+p+1] \tag{2.4}$$

where p is the degree of the function.
The dependency is different for the first, second, second to last and last function, because they depends from functions that are identically zero. This causes the shape of those functions to be different and this can be seen from Figure 2.5 and 2.6 .

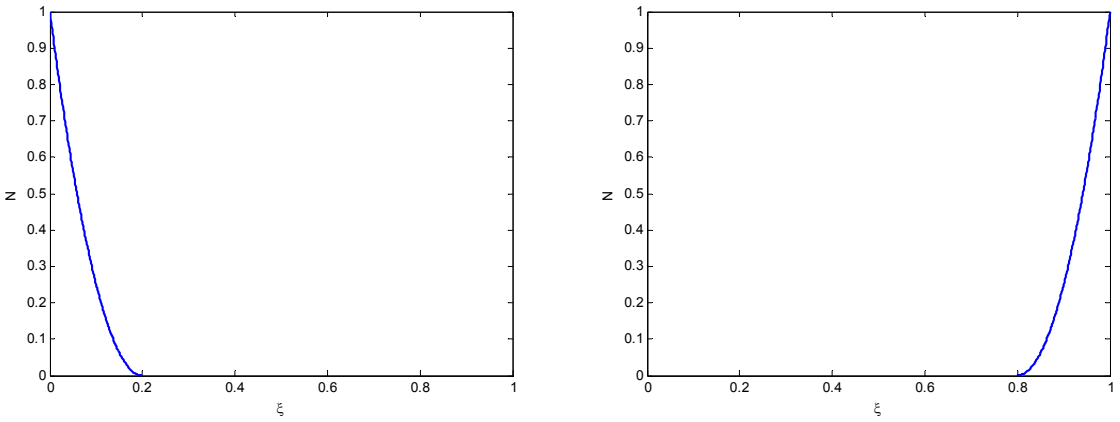


Figure 2.5 – Frist (left) and last (right) quadratic basis function shapes.

The knot vector that is taken into account presents a repeated knot (knot 4) which modifies the basis function shapes and in particular the continuity of the function across the knot span boundaries. The linear 'hat' functions are continuous just in position, because the sharp corner causes discontinuity of the tangent lines, thus the continuity is C^0. In the quadratic functions the knot span boundary is not well rendered, because the tangents are continuous, thus the continuity is C^1.

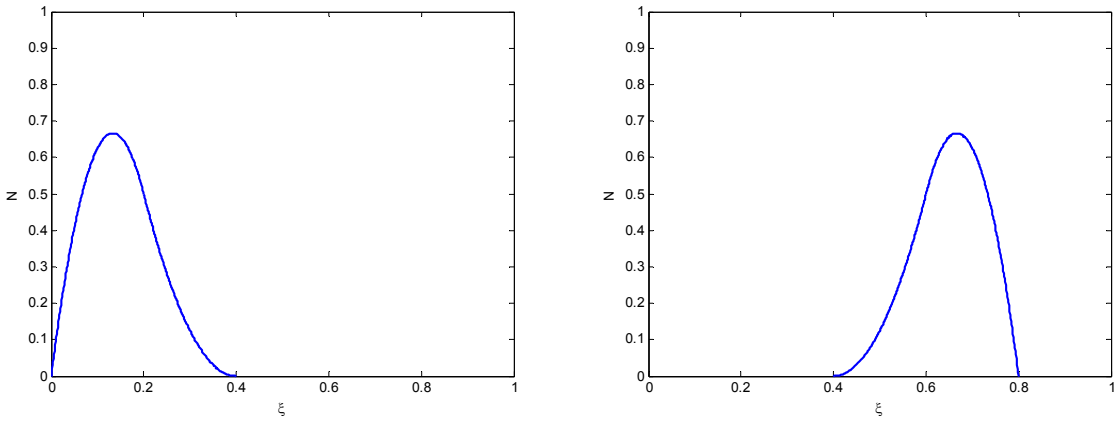


Figure 2.6 – Second (left) and second to last (right) quadratic basis functions.

In this case it is noticeable a change in concavity sign, which suggests that the curvatures are not continuous.

In the general case, for a function of order *p*, the continuity is C^{p-1}.

A repeated knot, instead, modifies this condition featuring a lowering in the continuity. In particular, each repeated knot removes one degree of continuity, defining the general condition that considering order *p* and multiplicity of knots *m*, the continuity is C^{p-m}.

In the knot vector considered, it generates a last knot span that is C^0 continuous and behaving in a similar way of standard FEM. In Figure 2.7 the functions of this last knot span are shown.

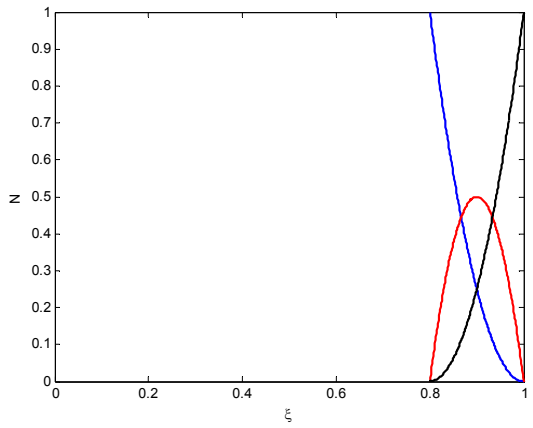


Figure 2.7 – Basis functions of last knot span.

All the basis functions are now plotted together in Figure 2.8 to have a general idea on their shape and summarise all their properties.

The basis functions are now combined with the positions of the *control points* to generate a curve and its complete function

$$C(\xi) = \sum_{i=1}^{n} \mathbf{P}_i N_i(\xi) \tag{2.5}$$

Figure 2.8 – B-spline basis functions for knot vector {0,0,0,1,2,3,4,4,5,5,5}.

which is the function to *map* the parametric coordinate ξ to the (x, y) or (x, y, z) Cartesian coordinates of the physical curve.
Different curves can be obtained from a different set of control points positions, as shown in Figure 2.9, without changing the knot vector and, thus, the functions involved.

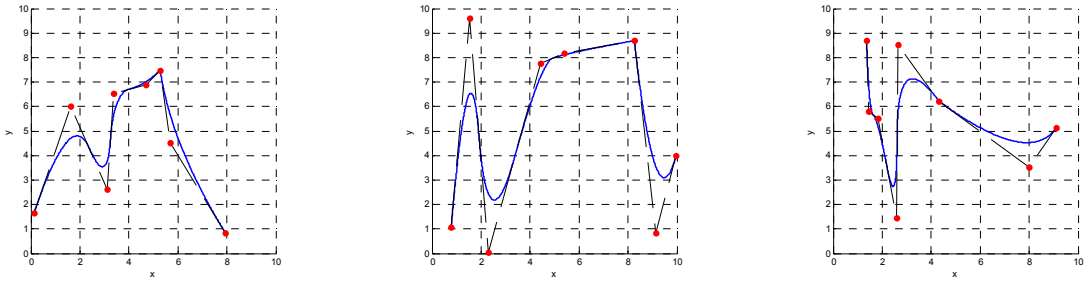


Figure 2.9 – B-spline curves from same knot vector and functions,
with different control points position.

In the dependency tree of Figure 2.1 the maximum order counts 8 functions, following the general rule that the number of basis functions that generates the curve is

$$n = k - p - 1 \tag{2.6}$$

where *k* is the number of knots in the knot vector.
This means that 8 coefficients $\mathbf{P}_i$ are necessary to complete the ingredients of Equation 2.5 , which means that 8 control points will drive the curve.

The tensor product between two separate parametric coordinates (thus knot vectors) $\Xi = \{\xi_1 \quad \xi_2 \quad \cdots \quad \xi_{n+p+1}\}$ and $\mathrm{H} = \{\eta_1 \quad \eta_2 \quad \cdots \quad \eta_{m+q+1}\}$ (where n is the number of control points in the first direction and m is the number of control points in the second) allow the representation of a surface, instead of a curve. In this case the control points form a *control net* where the control points are propagated from a direction to the other, in a matrix-like way.
Two independent set of basis functions (N_i and M_j) are then defined, one for each knot vector and the product of them forms two-dimensional basis functions.

Similarly to the case of a curve, the function for a surface is

$$S(\xi,\eta)=\sum_{i=1}^{n}\sum_{j=1}^{m}\mathbf{P}_{i,j}N_i(\xi)M_j(\eta) \tag{2.7}$$

which represents the *mapping* function. An example of a B-spline surface is shown in Figure 2.10.

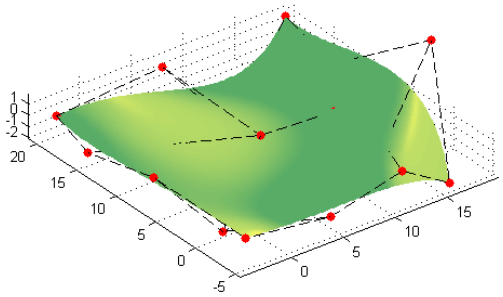


Figure 2.10 – Example of a generic B-spline surface.

In this example the patch surface is single-span bi-cubic, thus for a single element 4 control points are required for cubic functions. This leads to $n=4$ and $m=4$, which give a total number of control points of $n\times m=16$.

Following the same consideration, adding a third parametric direction (and then a third physical direction where to propagate the control points in a 3D matrix-like way) a full solid, with all its point inside, can be defined.
The third parametric coordinate, to add to the previous two, is $Z=\{\zeta_1 \quad \zeta_2 \quad \cdots \quad \zeta_{l+r+1}\}$ and adding its contributions to the computations of basis functions, the function of the full solid model is

$$V(\xi,\eta,\zeta)=\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l}\mathbf{P}_{i,j,k}N_i(\xi)M_j(\eta)L_k(\zeta) \tag{2.8}$$

A tri-variate patch is then obtained, and an example is shown in Figure 2.11.
Now each point on vertices, edges, surfaces and inside the volume can be traced, by setting a value to each parametric directions, as in Figure 2.12.
From this point, using the information on the surfaces, the two tangent vectors with respect to the two parametric coordinates can be computed, and from those vectors the normal direction can be computed as well, using the following relation

$$\mathbf{n} = \frac{\frac{\partial S}{\partial \xi} \times \frac{\partial S}{\partial \eta}}{\left| \frac{\partial S}{\partial \xi} \times \frac{\partial S}{\partial \eta} \right|} \tag{2.9}$$

where S is the surface formula in Equation 2.7 that is extracted from the volumetric data. An example of the three vectors is shown in Figure 2.13.

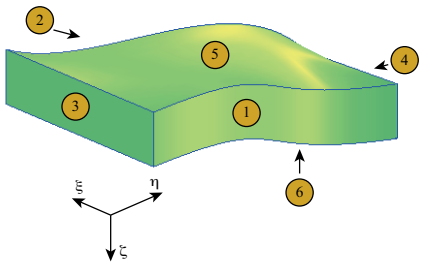


Figure 2.11 – B-spline volumetric tri-variate patch with parametric directions.

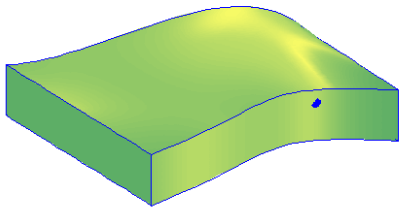


Figure 2.12 – Representation of a point in space from parametric coordinates (0, 0.75, 0.25).

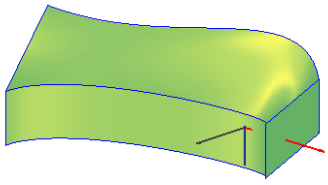


Figure 2.13 – Representation of the tangent vectors (red and blue) and the normal vector (grey).

The equation of the six boundary surfaces can be extracted from the volumetric information taking advantage of the property that setting the value of one of the parametric directions, obtaining the so called *isosurface* at that parametric value. The surface obtained is still a B-spline surface (as shown in Figure 2.14).

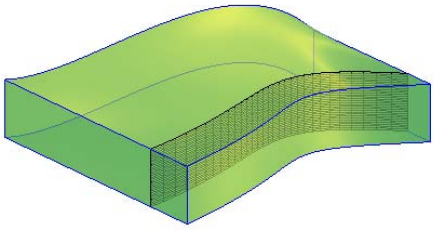


Figure 2.14 – Representation of an isosurface (in grey) obtained by setting the first parametric value at 0.25.

If the values of the parametric directions are set to 0 or 1 one at time, the external boundary surface is extracted and, as shown in Figure 2.11, the external surfaces are numbered according to the following sequence:

1) surface at $\xi = 0$;
2) surface at $\xi = 1$;
3) surface at $\eta = 0$;
4) surface at $\eta = 1$;
5) surface at $\zeta = 0$;
6) surface at $\zeta = 1$;

Because of the NURBS tensor product structure, this numbering is fixed and a consequence of the parameterisation of the geometry.

Even though B-splines are widely used, they have the limitation that, being polynomial they cannot represent conics as circles and ellipses, exactly.
To overcome this limitation, a rational form of B-splines was developed and this is the NURBS (in fact, Non-Uniform Rational B-Splines).
It represents a transformation from a space in $\Re^{d+1}$ to a space in $\Re^{d}$, where d is the number of physical spatial coordinates. This means, from a geometrical point of view, that a shape in a plane is the projection on a plane of a shape in 3D.

The additional dimension is called *weight* and used as a multiplication factor for the basis functions, that become

$$R_{i,p}(\xi) = \frac{N_{i,p}(\xi) w_i}{W(\xi)} = \frac{N_{i,p}(\xi) w_i}{\sum_{j=1}^{n} N_{j,p}(\xi) w_j} \tag{2.10}$$

and the case of solid NURBS the tri-variate basis functions and the volume functions become

$$R_{i,j,k}^{p,q,r}(\xi,\eta,\zeta)=\frac{N_i(\xi)M_j(\eta)L_k(\zeta)w_{i,j,k}}{\sum_{\hat{i}=1}^{n}\sum_{\hat{j}=1}^{m}\sum_{\hat{k}=1}^{l}N_{\hat{i}}(\xi)M_{\hat{j}}(\eta)L_{\hat{k}}(\zeta)w_{\hat{i},\hat{j},\hat{k}}} \tag{2.11}$$

$$\mathbf{V}(\xi,\eta,\zeta)=\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{l}\mathbf{P}_{i,j,k}R_{i,j,k}^{p,q,r}(\xi,\eta,\zeta) \tag{2.12}$$

An example of a NURBS volume is shown in Figure 2.15, where an hollow cylinder is considered. In the picture it is shown that only four knot spans (or *elements*, for analogy with FEM) are necessary to exactly model the cylinder without any geometrical approximations. The minimum number of knot spans which exactly represents geometry is called *coarsest mesh*.

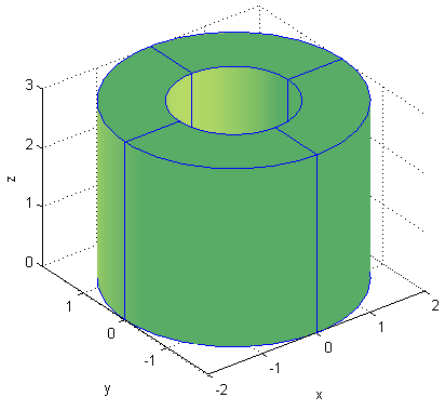


Figure 2.15 – Volumetric representation of an exact hollow cylinder.

Thanks to the isoparametric concept, the same NURBS functions that are used to define the geometry can be used to discretise the displacement field, in case of solid mechanics, substituting the control points coordinates with the unknown variables (degrees of freedom of the finite element model)

$$\mathbf{u}^e(\xi,\eta,\zeta)=\sum_{a=1}^{n}\mathbf{d}_a^eR_a^e(\xi,\eta,\zeta) \tag{2.13}$$

3. NURBS-based isogeometric analysis

To introduce how a NURBS function could be considered to build a finite element computation, the standard procedure is presented.
As anticipated in Equation 2.13, the relationship that links the displacements of the degrees of freedom of the system to the general displacement field at any point, considering a simple case of linear elasticity in static, is

$$u(x) = N(x)a \tag{3.1}$$

where $u(x)$ is the displacement field, obtained by $N(x)$ general shape function that is multiplied by the degrees of freedom a.
The deformation is generally defined as

$$\varepsilon(x) = \frac{d}{dx}u(x) \Rightarrow \boldsymbol{\varepsilon} = \partial\mathbf{N}\mathbf{a} = \mathbf{B}\mathbf{a} \tag{3.2}$$

where **B** is the matrix containing the derivatives of the shape functions, while the stress is obtained considering

$$\sigma = \mathbf{D}\boldsymbol{\varepsilon} = \mathbf{D}\mathbf{B}\mathbf{a} \tag{3.3}$$

where **D** is a matrix containing the material constitutive law, that varies depending by the analysis and the dimensions (1D, 2D or 3D) of the model involved.
Being the stiffness matrix in FEM usually defined as

$$\mathbf{K} = \int_V \boldsymbol{\varepsilon}^T \boldsymbol{\sigma} dV \tag{3.4}$$

and considering the relationships in Equations 3.2 and 3.3 , it becomes

$$\mathbf{K} = \int_V \mathbf{B}^T \mathbf{D}\mathbf{B} dV \tag{3.5}$$

that is the formula normally used in computations.
For known entities like bars and beams, the integral can be developed in closed form and used directly in the codes, but in case of isoparametric elements and even more in isogeometric analysis, the closed form is hard to obtain and then a numerical integration is performed.
This numerical integration uses the *Gauss points*, defined in the *parent element* space (a domain in the range [-1, 1]), where, depending by the nature of the function inside the integral (usually its order), a set of points and weights are defined, with the aim of simplifying the integral computation to a simple arithmetic sum

$$\int_a^b f(x)dx \cong \frac{b-a}{2}\sum_{i=1}^{n} w_i f\left(\frac{b-a}{2}g_i + \frac{a+b}{2}\right) \tag{3.6}$$

where a and b are the integration limits, g_i and w_i are respectively the *Gauss points* and *Gauss weights*, fixed tabular data that depends on the order of the function inside the integral. One important property of Gaussian numerical integration is that, if the function is polynomial, using n Gauss points warrant a polynomial function of a degree of $2n-1$ to be exactly computed.

The standard rough workflow in a finite element procedure is then the following:

- loading of the input geometry;
- definition of the connectivity (elements);
- computation of the basis functions and their derivatives;
- loop over the elements:
    - definition of constitutive matrix;
    - computation of the integral (loop over gauss points):
        - gauss point contribution computation;
        - contribution added to element matrices;
    - assembly element terms to global terms;
- imposition of boundary conditions (loads and constraints);
- solution of the system;

while the same workflow for a NURBS-based analysis is:

- loading of the input geometry;
- *refinement step to obtain final model;*
- definition of the connectivity (elements);
- loop over the elements:
    - definition of constitutive matrix;
    - computation of the integral (loop over gauss points):
        - *computation of the basis functions and their derivatives;*
        - gauss point contribution computation;
        - contribution added to element matrices;
    - assembly element terms to global terms;
- *imposition of boundary conditions* (loads and constraints);
- solution of the system;

where in italic font are highlighted the steps with differences with respect to the standard workflow.

Regarding the computation of the basis functions and the derivatives, the difference is in the not-standard and not-general nature of the B-splines/NURBS basis functions. In standard finite elements, even in the isoparametric case, the difference between the functions among the elements only depends by the kind of element involved. This means that for each kind of element, once the order of the Gauss rule is chosen, the basis functions and their derivatives are computed once for each point in parent space, before any loops.
For NURBS basis function this property does not exist, because at each Gauss points the function might be different and then the function and the derivatives must be computed specifically at each instance of the loop. This is evidently a drawback that can be overcome with the Bézier extraction technique [35], which manipulates the functions before the loops to have a standard-like look, compute functions and derivatives, perform the steps inside the

loop and restore the functions nature afterwards with a simple matrix computation. This technique drastically quickens the whole procedure.

The imposition of the boundary conditions (except from the ideal clamping case) cannot, in general, be solved in strong form, directly on the degrees of freedom, as it is usually done in standard FEM. This difference is due to the B-splines/NURBS property that their control points do not belong to the actual geometry (functions are not interpolator) and then a condition on the control points is not identically equivalent to the condition on the body. In order to apply values on the degrees of freedom such that the effect on the body fits the specification, the condition must be applied in weak form.

The last difference is one of the most important, and the one that have direct effect on the CAD-CAE integration possibility, considering that the final computation domain used to obtain the matrices that are then solved, is still a CAD model. In standard FEM once the mesh is built on top of the geometry, all links with the geometry is lost and only points (the nodes) and the connections among them (elements) remain. If the mesh is not fine enough to approximate well the behaviour of the part, a refinement must be applied, calling again the original geometry and restart the meshing procedure. In B-spline/NURBS the refinement is straightforward because both the cases of order elevation and element size reduction do not change the parameterisation of the geometry. There are well established and optimised algorithms and codes to obtain the new refined geometry and all the refined procedures can be performed anytime without losing connection with the original geometry.

3.1. Refinement: knot insertion

The analogy between the knot spans in the NURBS space and the elements in standard FEM allows the possibility of decreasing the size of an element (increasing then the number of elements) by *adding* subdivisions in the knot vectors, simply adding new knots. This can be referred to as the *h-refinement*.
In detail, if initially the original knot vector is $\Xi = \{\xi_1 , \xi_2 , \ldots \xi_{n+p+1}\}$, the new extended knot vector is then $\overline{\Xi} = \{\overline{\xi}_1 = \xi_1 , \overline{\xi}_2 , \ldots , \overline{\xi}_{n+m+p+1} = \xi_{n+p+1}\}$ taking into account that to maintain the parameterisation unchanged $\Xi \subset \overline{\Xi}$.
Being the knot vector changed in dimension, from Equation 2.6 it comes out that also the number of control points is changed.
The new control points are formed from a linear combination of the original control points thanks to a transformation matrix

$$\overline{P} = T^p P \tag{3.7}$$

where the transformation matrix is obtained, similarly to the computation of the basis functions, with a recursive formula

$$T_{ij}^0(\xi) = \begin{cases} 1 & \overline{\xi}_i \in \left[\xi_j , \xi_{j+1}\right) \\ 0 & otherwise. \end{cases} \tag{3.8}$$

$$T_{ij}^{q+1}(\xi)=\frac{\overline{\xi}_{i+q}-\xi_j}{\xi_{i+q}-\xi_j}T_{ij}^{q}(\xi)+\frac{\xi_{j+q+1}-\overline{\xi}_{i+q}}{\xi_{j+q+1}-\xi_{j+1}}T_{ij+1}^{q}(\xi)\quad for\quad q=0,1,2,\ldots,p-1 \tag{3.9}$$

Considering the example in Figure 3.2, a new set of knots are defined obtaining $\overline{\Xi}=\{0\quad 0\quad 0\quad 1/6\quad 1/3\quad 1/2\quad 2/3\quad 5/6\quad 1\quad 1\quad 1\}$ with the result in Figure 3.1.

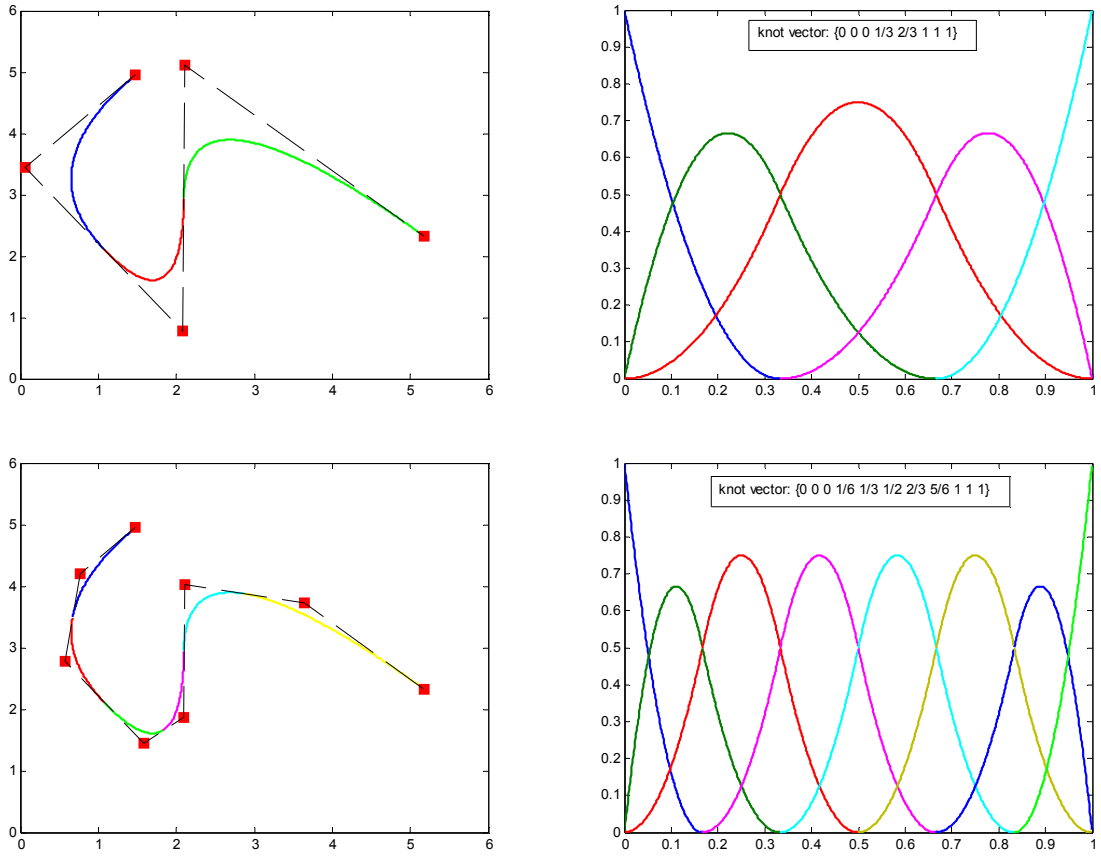


Figure 3.1 – Effect of knot insertion refinement on curve and basis functions.

The number of elements is then increased and the new control points position are obtained.

3.2. Refinement: order elevation

The order elevation is the definition of a higher degree of the basis functions involved, without changing the parameterisation of the geometry. Any curve can be represented exactly with another curve with a higher degree, but the original continuity across elements is preserved.

This has the analogy in standard FEM that is the *p-refinement*, with a difference involving the behaviour of the shape functions: being the Lagrange basis functions purely interpolator, raising the degree of the interpolation functions causes an increase on the oscillatory effect, while the NURBS basis functions cause an opposite effect of smoothening the curve as the degree raises.

The number of elements and their geometrical information remain unchanged, but the number of functions increases and thus the number of control points and their position. In Figure 3.2 an example of the effect of an order elevation is presented.

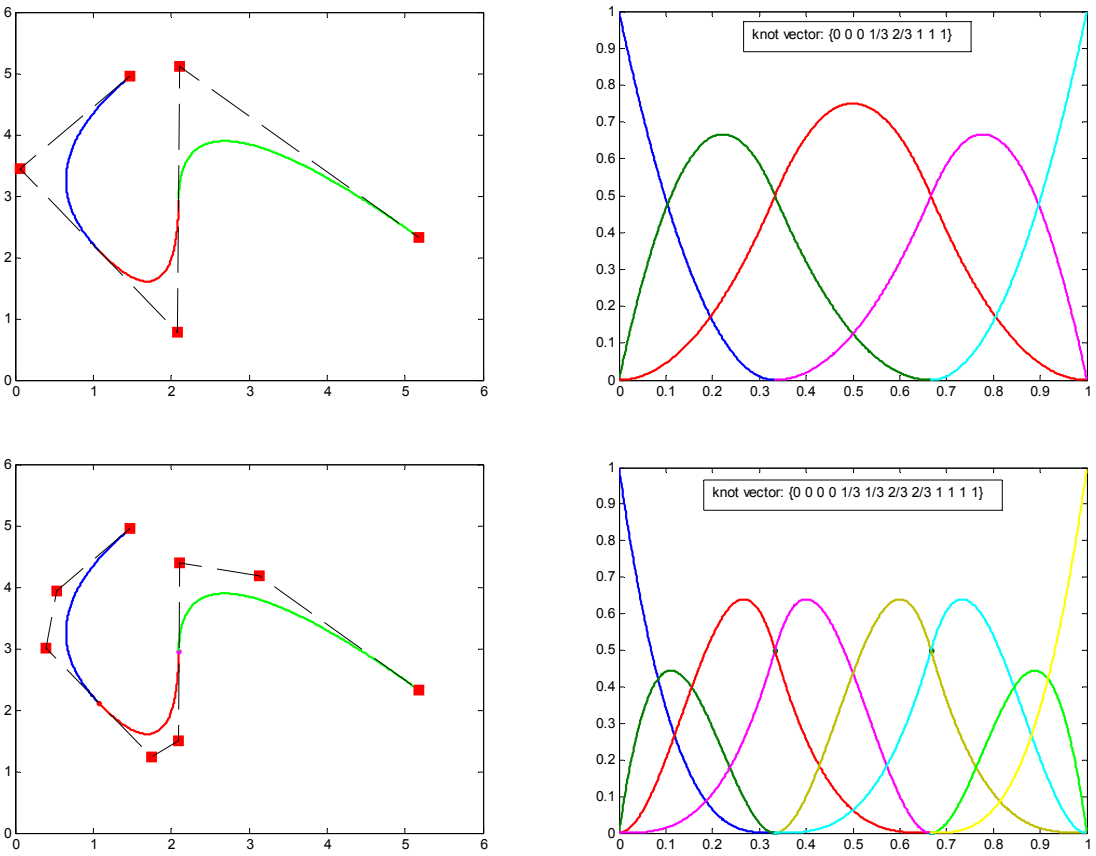


Figure 3.2 – Effect of order elevation refinement on curve and basis functions.

The knot repetition ensures that the continuity across the original elements is maintained.

3.3. Refinement: k-refinement

An important property of the *p-* and *h-refinement* in NURBS is that they are non-commutative, as they are instead in standard FEM.
The effect in the two cases, where knot insertion is performed before the order elevation and vice-versa, is shown in Figure 3.3.
Due to knot repetition property caused by the order elevation, applying it after the knot insertion leads to obtaining unnecessary repeated knots lowering the continuity in a point where it was C^{∞} because inside one of the original elements.
The second case, instead, allows the obtaining of the higher possible continuity across the added knots and thus obtaining better quality off the approximation with less control points (and, thus, degrees of freedom).
This effect is new and it is not analogue effect in standard FEM, then this procedure is referred to as *k-refinement*.

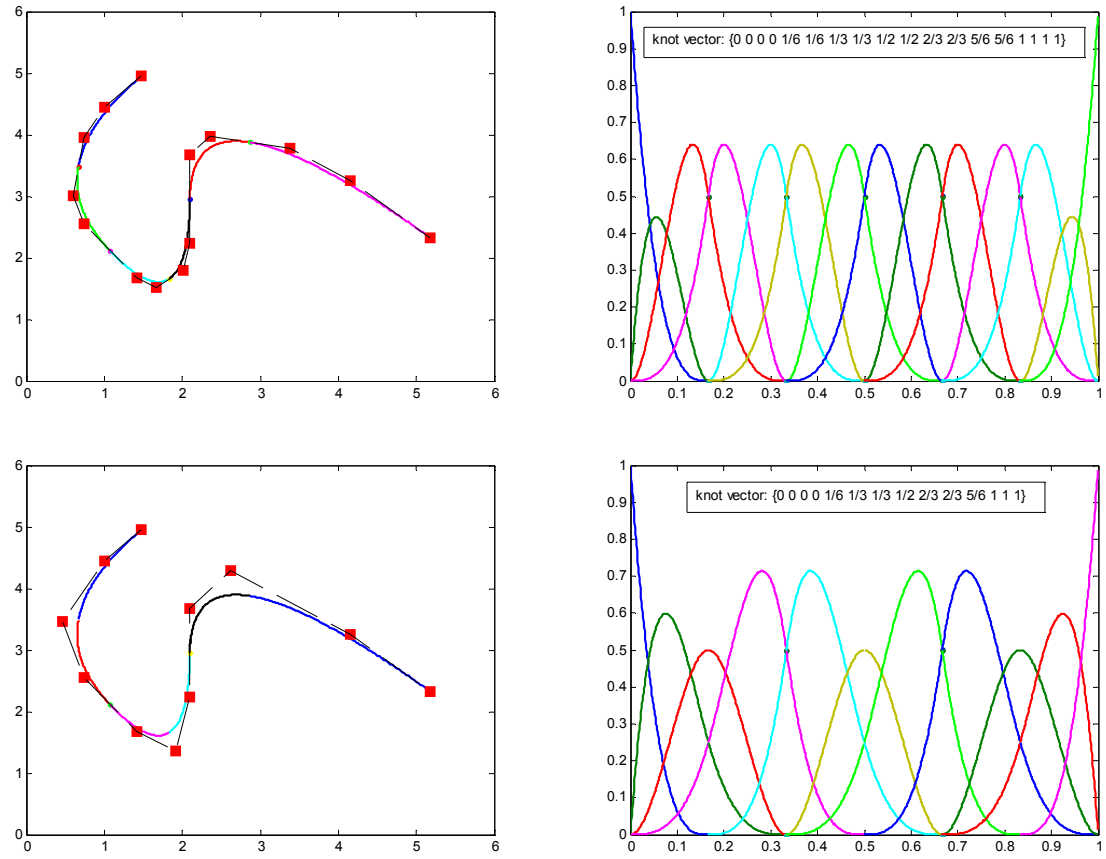


Figure 3.3 – Difference between knot insertion to order elevation and order elevation to knot insertion.

The sequences of order elevation and knot insertion must be considered in order of obtaining the maximum quality with the fewer possible number of function involved.
A common choice is a sequence of order elevations in order of reaching one order more of the desired continuity and then all the knot insertions.

3.4. Applications on single-patch

As first attempt to solve problems with B-spline/NURBS, the effects of boundary conditions can be neglected considering *free-free modal analysis*, which allows no constraints resulting that the first six natural frequencies are machine-zeros because they will be related to the six rigid body motions in space.
The modal analysis considered is undamped, thus only elastic and inertial contributions are involved obtaining the following equation of motion

$$\mathbf{M}\ddot{\mathbf{x}} + \mathbf{K}\mathbf{x} = \mathbf{f} \tag{3.10}$$

where $(\mathbf{x}, \ddot{\mathbf{x}})$ is the vector of the degrees of freedom in displacements and accelerations, **f** is the vector of the external loading, **K** is the stiffness matrix of Equation 3.6 and **M** is the mass matrix that is obtained from the kinetic energy equation, resulting in

$$\mathbf{M} = \int_V \mathbf{N}^T \rho \mathbf{N} dV \tag{3.11}$$

The mass matrix only involves the basis functions that in case of B-spline/NURBS are always non-negative, leading to the conclusion that all the entries of the mass matrix will be non-negative as well. This is an important property in terms of stability in dynamic analysis for the inversion of the matrix.
In modal analysis the natural frequencies and the deformed shapes at those frequencies can be computed considering the synchronous equation where the system responds at the same frequency of the load

$$\mathbf{f} = \mathbf{f}_0 \cos(\omega t); \qquad \mathbf{x} = \mathbf{x}_0 \cos(\omega t + \varphi) \tag{3.12}$$

where *t* is the time, ω is the oscillation frequency and φ is the delay in the response. Then Equation 3.10 becomes (considering analysing only the homogeneous equation)

$$-\omega^2 \mathbf{M}\mathbf{x}_0 \cos(\omega t + \varphi) + \mathbf{K}\mathbf{x}_0 \cos(\omega t + \varphi) = \mathbf{0} \tag{3.13}$$

which, considering to be valid at each time, it leads to the eigenvalue problem

$$(\mathbf{K} - \omega^2 \mathbf{M})\mathbf{x}_0 = \mathbf{0} \tag{3.14}$$

where the non-trivial solution is solved using

$$\det(\mathbf{K} - \omega^2 \mathbf{M}) = \mathbf{0} \tag{3.15}$$

For every eigenvalue ω_r, a corresponding eigenvector $\mathbf{x}_0$ is evaluated from equation (3.14), defined as mode-shape. Its scale is arbitrarily set, usually with normalisation with respect to unity modal mass.

3.4.1. Rectangular beam

A rectangular beam with the following characteristics is considered for the first analysis, which can be easily modelled with tri-linear functions, such that $\Xi = \mathrm{H} = \mathrm{Z} = \{0,0,1,1\}$, with two control points for each parametric direction, resulting a total number of control points of 8, which are the 8 vertices of the rectangular beam.
The original model is refined to tri-cubic elements with 8 subdivisions on the length and 2 subdivisions on the width.
In the formula presented in Equation (2.12) the term $\mathbf{P}_{i,j,k}$, which determines the coordinates of the positions of the control points, defines the control points in the *NURBS index*, which exploits the structured tensor product structure nature.

Table 3.1 – Beam properties

Characteristic	Property	Value
Beam geometry	Length x [mm]	1000
	Width y [mm]	50
	Height z [mm]	20
Beam material	Density ρ [kg/m^3]	2700
	Young's modulus E [GPa]	69
	Poisson's ratio ν [-]	0.33

A conversion for changing the (i, j, k) indexing, that are related to the (ξ,η,ζ) direction respectively, to the single index (as you would have a node number in standard FEM) is necessary for generating the array of degrees of freedom which is the unknown variable of the system.
A table of correspondence is defined and called *INN*, which links the triple indexing to the single, where the row index of the matrix is the single index and three columns will contain the values of the triple indexing. The arrangement order is first the index i, then the index j, and then the index k.
Another table necessary for the management of the model throughout the simulation is the association of the control points involved in each knot span, considered as *elements* in the computations. The table is arranged in columns where each column refer to an element, and the entries of the column are the control points numbers in single index.
Those table are important during both the elemental matrices computation (Jacobian and parametric to Cartesian mapping) and the assembly operation in the global matrices (from the numbers of the control points to the row/column correspondence in the global matrices.

To validate the method the results were compared to the ones obtained with the use of commercial software that uses second-order tetrahedra to discretise the geometry. The choice of the software is SolidWorks, because of its tight use of geometry to shorten the gap between its CAD and CAE environments.

The following Table 3.2 summarises the details of the two different discretisation.

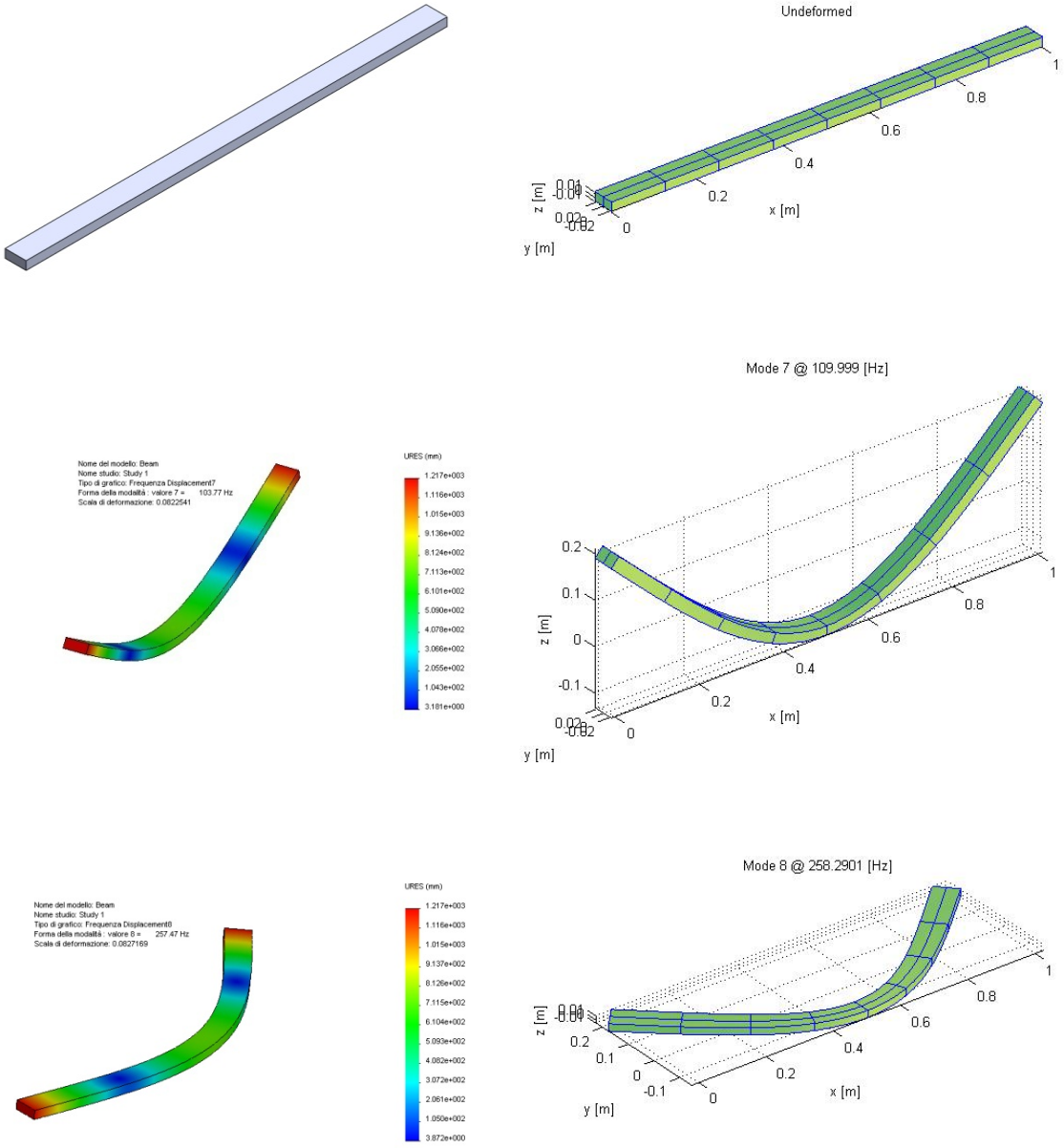
Table 3.2 – Discretisation details of FEM and IGA models

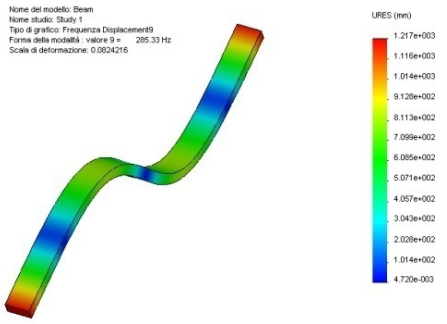
Property	SolidWorks FEM	IGA
Number of elements	8820	40
Number of nodes/ control points	14797	154
Number of Degrees of Freedom	44391	462

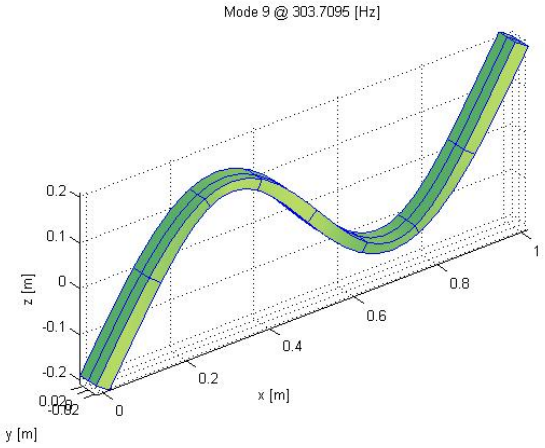
Even though the prismatic shape of the object would suggest the use of hexahetra to discretise the geometry, the comparison with NURBS would not be consistent because NURBS are general and with NURBS any kind of shape could be discretised, while not all

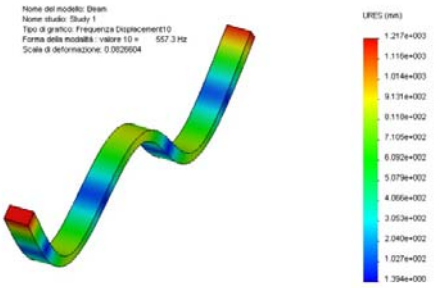
geometries could be discretised with hexahedra (or, at least, with difficulties). Tetrahedra are instead general and can discretise any geometry and could be compared with NURBS.
The drawback with the generality condition is that in the case of the considered beam, the shape is thin and then several small tetrahedra are necessary to model the beam with a sufficient goodness of the elements in terms of Jacobian and aspect ratio, and this justifies the large number of nodes and elements for the SolidWorks FEM case.
Moreover, the modal analysis is usually interested on investigate the first global modes at lower frequencies and thus a fewer number of elements are sufficient to catch this modes with good approximation.
The results are summarised in Table 3.3.

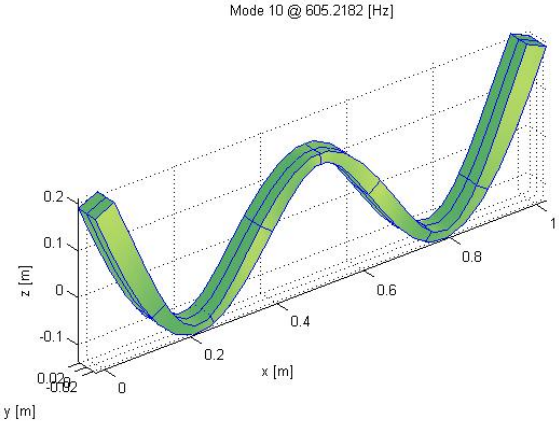
The mode-shapes comparison of these modes are shown in Figure 3.4.

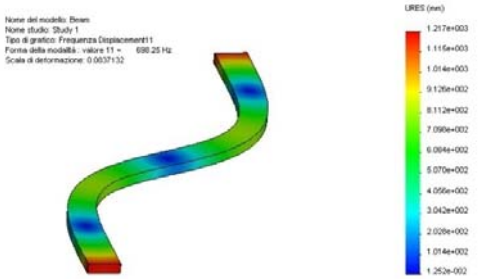


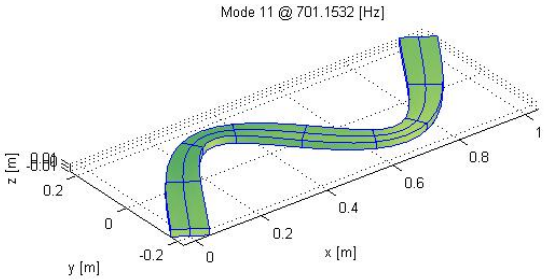

Nome del modello: Beam
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement9
Forma della modalità : valore 9 = 285.33 Hz
Scala di deformazione: 0.0824216
URES (mm)
1.217e+003
1.116e+003
1.014e+003
9.128e+002
8.113e+002
7.099e+002
6.085e+002
5.071e+002
4.057e+002
3.043e+002
2.028e+002
1.014e+002
4.720e-003

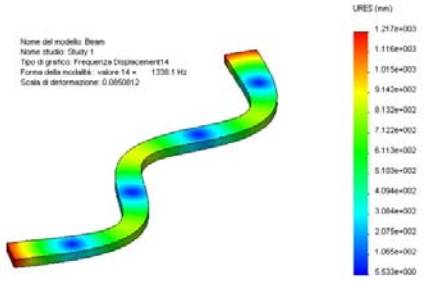

Mode 9 @ 303.7095 [Hz]
z [m]
x [m]
y [m]

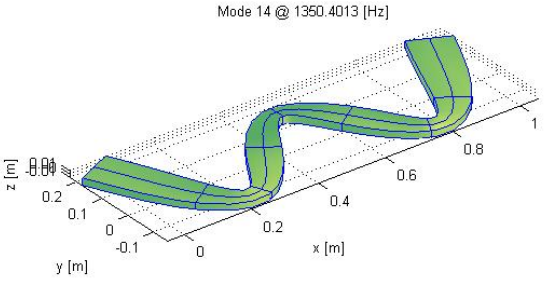

Nome del modello: Beam
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement10
Forma della modalità : valore 10 = 557.3 Hz
Scala di deformazione: 0.0826604
URES (mm)
1.217e+003
1.116e+003
1.014e+003
9.131e+002
8.119e+002
7.105e+002
6.092e+002
5.079e+002
4.066e+002
3.053e+002
2.040e+002
1.027e+002
1.394e+000


Mode 10 @ 605.2182 [Hz]
z [m]
x [m]
y [m]


Nome del modello: Beam
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement11
Forma della modalità : valore 11 = 698.25 Hz
Scala di deformazione: 0.0837132
URES (mm)
1.217e+003
1.115e+003
1.014e+003
9.126e+002
8.112e+002
7.098e+002
6.084e+002
5.070e+002
4.056e+002
3.042e+002
2.028e+002
1.014e+002
1.252e-002


Mode 11 @ 701.1532 [Hz]
z [m]
y [m]
x [m]


Nome del modello: Beam
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement14
Forma della modalità : valore 14 = 1338.1 Hz
Scala di deformazione: 0.0850812
URES (mm)
1.217e+003
1.116e+003
1.015e+003
9.142e+002
8.132e+002
7.122e+002
6.113e+002
5.103e+002
4.094e+002
3.084e+002
2.075e+002
1.065e+002
5.533e+000


Mode 14 @ 1350.4013 [Hz]
z [m]
y [m]
x [m]

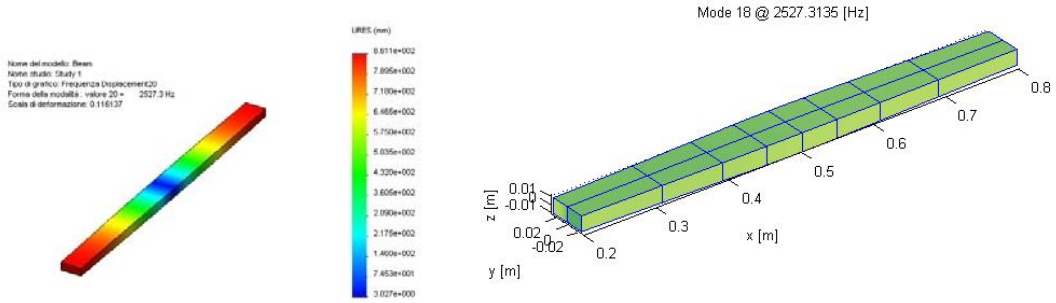


Figure 3.4 – Comparison of FEM (left) and IGA (right) mode-shapes for the beam model.

Table 3.3 – Results comparison between FEM and IGA models of beam.

Mode	FEM frequencies [Hz]	IGA frequencies [Hz]	Description
7	103,77	109,99	I bending xz
8	257,47	258,29	I bending xy
9	285,33	303,70	II bending xz
10	557,30	605,21	III bending xz
11	698,25	701,15	II bending xy
14	1338,10	1350,40	III bending xy
18	2527,30	2527,30	I axial

3.4.2. Hollow cylinder (pipe)

An application where NURBS-based analysis can show its particular advantages is the case of a shape that can exactly represent with a number of elements severely lower with respect to standard FEM with tetrahedra.
A hollow cylinder (or *pipe*) is then chosen as subject, with geometrical and material data summarised in Table 3.4.

Table 3.4 – Pipe properties

Characteristic	Property	Value
Pipe geometry	Length x [mm]	650
	External diameter $\varnothing_{ext}$ [mm]	65
	Internal diameter $\varnothing_{int}$ [mm]	35
Pipe material	Density ρ [kg/m^3]	8000
	Young's modulus E [GPa]	190
	Poisson's ratio ν [-]	0.29

In general a NURBS patch cannot represent bodies with cavities, but in some cases a workaround is possible and objects with thickness can be obtained, as for a circular pipe.

The first parametric direction is the circumferential (or tangential) direction and is composed by 4 circular arcs of 90° span that are C^0 at the boundaries. The generation of one of this circular arcs is presented in Figure 3.5.

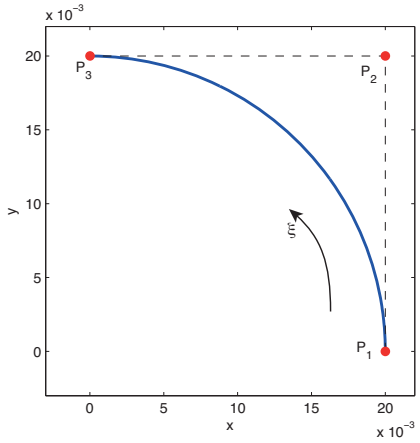


Figure 3.5 – Representation of a quarter of a circular arc with NURBS.

The knot vector considered is quadratic and with one non-zero span $\Xi = \{0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1\}$, which leads to 3 control points. Due to the open knot vector condition, the first and the last control points are the start and the end of the arc, respectively. The middle control point is placed at the intersection of the two tangent lines at the start and the end of the curve while, for the weight, the following relationship is considered

$$w_2 = \cos\left(\frac{\theta}{2}\right) \tag{3.16}$$

where ϑ is the span angle of the arc.
The four quarter of arc together will form the full circle, featuring a knot vector $\Xi = \{0 \quad 0 \quad 0 \quad 1/4 \quad 1/4 \quad 1/2 \quad 1/2 \quad 3/4 \quad 3/4 \quad 1 \quad 1 \quad 1\}$ with 9 control points, where the first and last control points are coincident.
The second parametric direction is the radial direction and can be represented as linear. The new 9 control points are related to the second circle (representing the internal or the external diameter of the pipe) and then placed with the same rule previously discussed.
The choice of the inwards or outwards direction with respect to the first circle will affect the direction of the normal vector of the surface and it is an important consideration because this choice will affect the third direction. An example of the resulting surface is shown in Figure 3.6.
The third parametric direction is towards the length of the pipe, and the choice of the upper of lower direction must consider that the direction must be concordant to the direction of the normal of the surface.

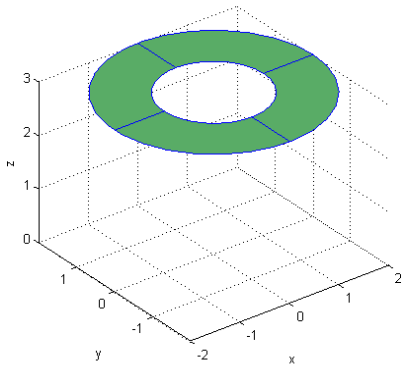


Figure 3.6 – Representation of the annular surface.

This final consideration is important because a discordant direction would lead to negative Jacobian in the computation of the matrices.
The final model is refined to tri-cubic functions, with 8 elements in the first parametric direction and 8 along the third. Similarly to the example of the beam above, the same geometry is discretised with second-order tetrahedra and solved with SolidWorks. The details of the two models are summarised in Table 3.5.

Table 3.5 – Discretisation details of FEM and IGA models.

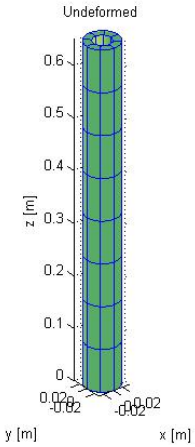
Property	**SolidWorks FEM**	**IGA**
Number of elements	8204	64
Number of nodes/ control points	13796	249
Number of Degrees of Freedom	41338	1287

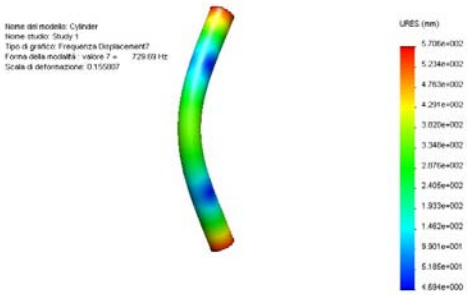
The results of the modal analysis is summarised in in Table 3.6.

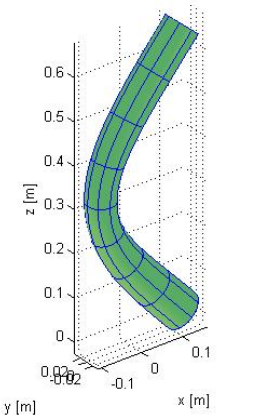
Table 3.6 – Results of natural frequencies and mode-shapes description.

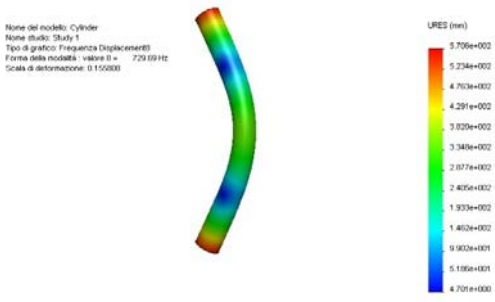
Mode	**FEM frequency [Hz]**	**IGA frequency [Hz]**	**Description**
7	729,69	730,25	I bending xz
8	729,69	730,25	I bending xy
9	1883,1	1885,38	II bending xz
10	1883,1	1885,38	II bending xy
11	2323,00	2333,88	I torsional
12	3402,60	3413,68	III bending xz
13	3402,60	3413,68	III bending xy
14	3746,10	3746,30	I axial
15	4645,60	4667,78	II torsional
18	6967,70	7002,10	III torsional
21	7475,20	7476,58	II axial

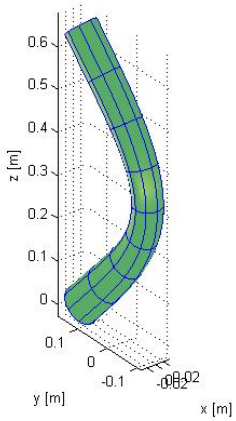
The visual comparison of the mode-shapes for the IGA and the FEM model are shown in Figure 3.7.

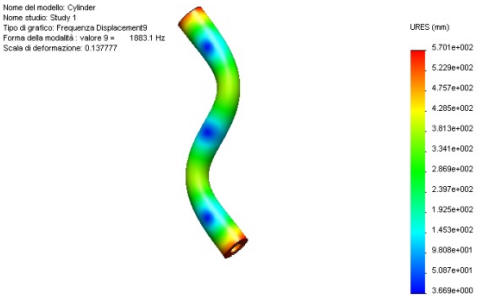


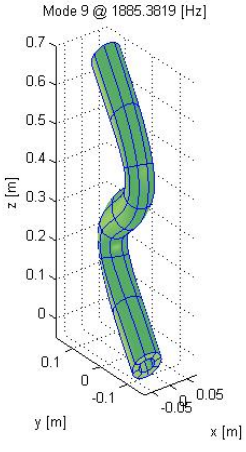


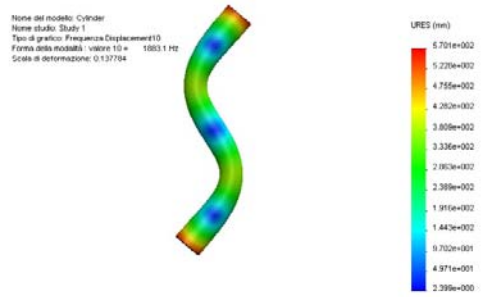


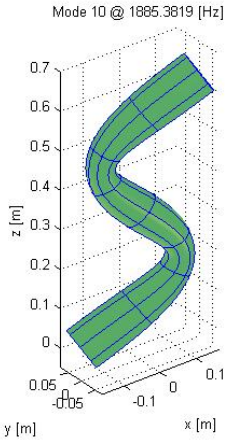


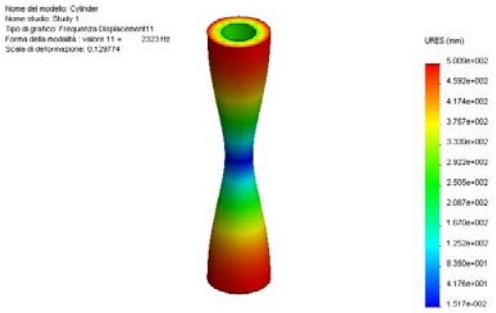


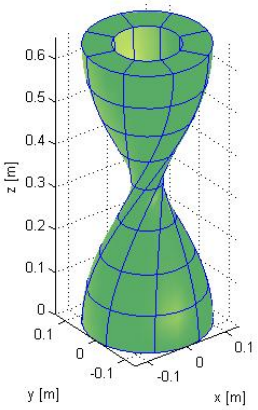


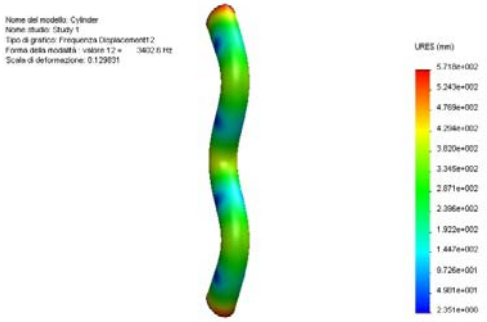


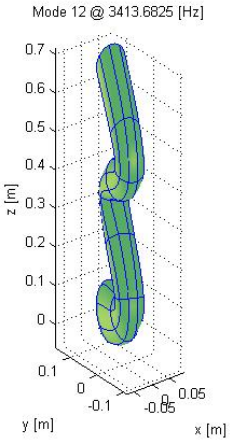

Nome del modello: Cylinder
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement10
Forma della modalità : valore 10 = 1883.1 Hz
Scala di deformazione: 0.137784
URES (mm)
5.701e+002
5.226e+002
4.755e+002
4.282e+002
3.809e+002
3.336e+002
2.863e+002
2.389e+002
1.916e+002
1.443e+002
9.702e+001
4.971e+001
2.399e+000

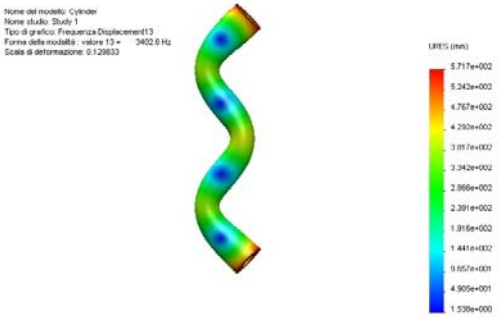

Mode 10 @ 1885.3819 [Hz]
z [m]
y [m]
x [m]

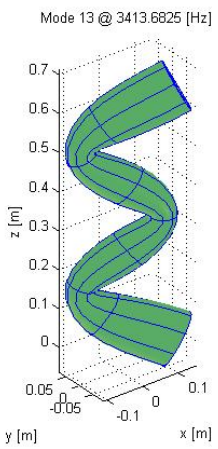

Nome del modello: Cylinder
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement11
Forma della modalità : valore 11 = 2323 Hz
Scala di deformazione: 0.129774
URES (mm)
5.009e+002
4.592e+002
4.174e+002
3.757e+002
3.339e+002
2.922e+002
2.505e+002
2.087e+002
1.670e+002
1.252e+002
8.350e+001
4.176e+001
1.517e-002


Mode 11 @ 2333.8805 [Hz]
z [m]
y [m]
x [m]


Nome del modello: Cylinder
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement12
Forma della modalità : valore 12 = 3402.6 Hz
Scala di deformazione: 0.129831
URES (mm)
5.718e+002
5.243e+002
4.769e+002
4.294e+002
3.820e+002
3.345e+002
2.871e+002
2.396e+002
1.922e+002
1.447e+002
9.726e+001
4.981e+001
2.351e+000


Mode 12 @ 3413.6825 [Hz]
z [m]
y [m]
x [m]


Nome del modello: Cylinder
Nome studio: Study 1
Tipo di grafico: Frequenza Displacement13
Forma della modalità : valore 13 = 3402.6 Hz
Scala di deformazione: 0.129833
URES (mm)
5.717e+002
5.242e+002
4.767e+002
4.292e+002
3.817e+002
3.342e+002
2.866e+002
2.391e+002
1.916e+002
1.441e+002
9.657e+001
4.905e+001
1.539e+000


Mode 13 @ 3413.6825 [Hz]
z [m]
y [m]
x [m]

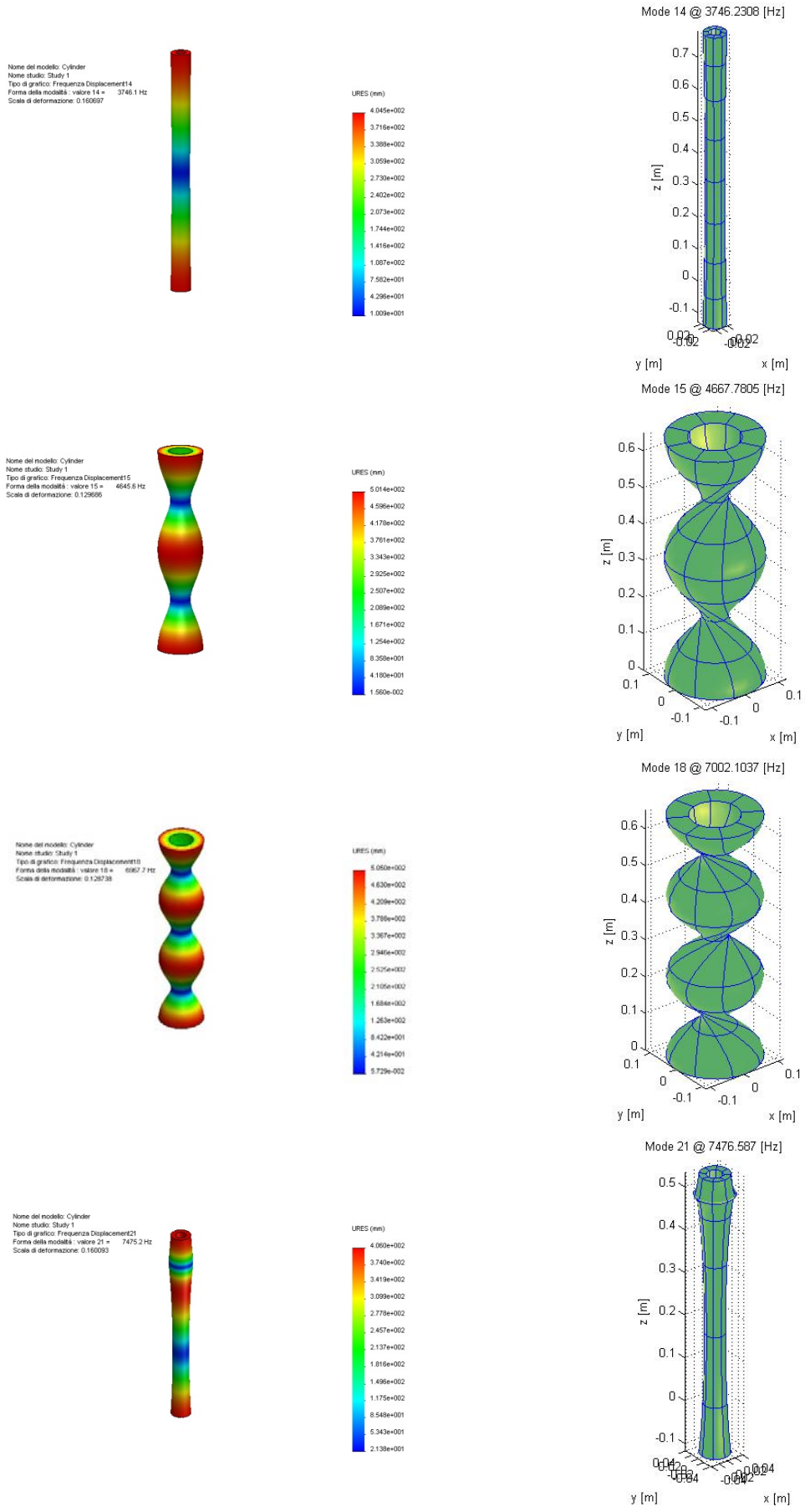


Figure 3.7 – Comparison of FEM (left) and IGA (right) mode-shapes for the pipe model.

In the analysed cases the NURBS-based model returned accurate results with lower computational costs due to the lower number of elements and degrees of freedom with respect to the standard tetrahedral model, featuring exact geometry at any stage of refinement and with the particular consideration that the 3-dimensional plot of the deformed shapes are still NURBS models and thus reusable CAD models.

4. Non-conforming multi-patch coupling and analysis

The previous section presented the isogeometric analysis method and implementation, with application on simple NURBS geometries.
But a NURBS tri-variate patch is not able to represent any kind of geometry, however complex, and cannot model cavities inside the body like holes or slots. It can be seen as a six-faced cube that, thanks to the movable position of the control points, can be deformed to represent even complex shapes, as in the pipe case where it could be seen as a cube that is pressed to obtain a plate that is bent to obtain the cylinder. Even though this stretching is possible, one drawback to take into account is that the distortion of the body will cause a distortion of the knot spans and then a decrease in the aspect ratio and in the Jacobian of the resulting computational elements, affecting the accuracy of the model.
This issue can be solved by using a *multi-patch* geometry, where different patch are modelling the body in order to reach a high geometrical complexity but maintaining a good shape of the elements inside the patches.

This possibility was originally considered even in the first paper by Hughes et Al. [1], but with some limitations: the geometry at the interface between two patches need equal parameterisation, in order of having the control points related to the two surfaces to be in the same exact position in space, with the possibility of strongly imposing the equality of the displacements of the control points involved (similarly of what happens in standard FEM when a compatible mesh is generated, collapsing the nodes at the interface of two domains); another possibility, in case of planar interface, the number of control points involved can be different but an integer multiple of one another [2].
Now the issue is how to couple together two patches where the two hypotheses do not apply, where the two surface at the interface do not feature a compatible mesh and no simple relationship among the control points can be defined.
The concept is to change the point of view (imposed by the habit of who is familiar with standard FEM) and think in terms of weak form, such as displacements and stresses on the actual geometry, and not in strong form, such as directly on the control points as if it were the nodes of standard FEM.
This concept was firstly developed by Nitsche in [24], and it was used to impose boundary conditions in weak form.
This technique is extended to domain coupling using a second domain, instead of an infinitely rigid grounding.

4.1. Nitsche's method for domain coupling theory

In Figure 4.1 it is shown a generic body divided into two separate domains, Ω_1 and Ω_2, with their interface Γ.

If a fixed constraint is applied to a portion of Ω_1 and a traction is applied to a portion of Ω_2, the conditions at the interface Γ to couple the two domains are

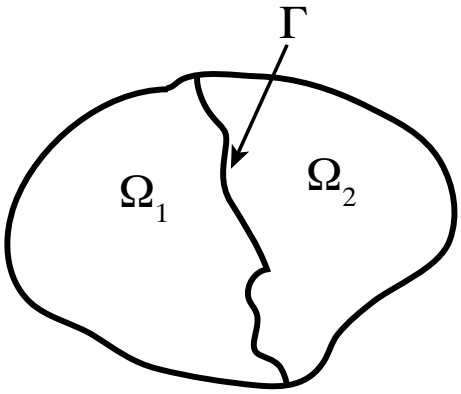


Figure 4.1 – Generic case of two adjacent bodies with a common interface.

$$\mathbf{u}^1 = \mathbf{u}^2 \tag{4.1}$$

$$\boldsymbol{\sigma}^1 \cdot \mathbf{n}^1 = -\boldsymbol{\sigma}^2 \cdot \mathbf{n}^2 \tag{4.2}$$

Considering the domain $\Omega = \Omega_1 \cup \Omega_2$, the weak form for static linear elasticity is

$$\int_\Omega \delta u_i \left(\sigma_{ij,j} + b_i\right) d\Omega = 0 \tag{4.3}$$

According to the product rule, the first term can be written as

$$\int_\Omega \delta u_i \sigma_{ij,j} d\Omega = \int_\Omega \frac{\partial}{\partial x_j}\left(\delta u_i \sigma_{ij,j}\right) d\Omega - \int_\Omega \frac{\partial \delta u_i}{\partial x_j} \sigma_{ij} d\Omega \tag{4.4}$$

and the first term of the right hand side, using the Gauss theorem, becomes

$$\int_\Omega \frac{\partial}{\partial x_j}\left(\delta u_i \sigma_{ij,j}\right) d\Omega = \int_\Gamma \delta u_i \sigma_{ij} n_j d\Gamma + \int_{\Gamma_1} \delta u_i^1 \sigma_{ij}^1 n_j^1 d\Gamma + \int_{\Gamma_2} \delta u_i^2 \sigma_{ij}^2 n_j^2 d\Gamma \tag{4.5}$$

Thanks to Neumann boundary condition ($\boldsymbol{\sigma} \cdot \mathbf{n} = \mathbf{t}$) and introducing the *jump operator* $\left[\!\left[\cdot \right]\!\right] \equiv (\cdot)^1 - (\cdot)^2$, and the *average operator* $\{\cdot\} \equiv 1/2\left[(\cdot)^1 - (\cdot)^2\right]$, it holds that

$$\int_\Omega \frac{\partial}{\partial x_j}\left(\delta u_i \sigma_{ij,j}\right) d\Omega = \int_{\Gamma_t} \delta u_i t_i d\Gamma + \int_\Gamma \left[\!\left[\delta u_i \, \sigma_{ij} \right]\!\right] n_j^1 d\Gamma \tag{4.6}$$

$$\left[\!\left[\delta u_i \sigma_{ij} \right]\!\right] = \left[\!\left[\delta u_i \right]\!\right] \{\sigma_{ij}\} + \{\delta u_i\} \left[\!\left[\sigma_{ij} \right]\!\right] \tag{4.7}$$

The second term of the right hand side is null (Equation (4.2)), then

$$\int_\Omega \frac{\partial}{\partial x_j}\left(\delta u_i \sigma_{ij,j}\right) d\Omega = \int_{\Gamma_t} \delta u_i t_i d\Gamma + \int_\Gamma \left[\!\left[\delta u_i \right]\!\right] \{\sigma_{ij}\} n_j^1 d\Gamma \tag{4.8}$$

By substituting Eq. (4.8) and Eq. (4.4) into Eq. (4.3), the weak form becomes

$$\int_\Omega \frac{\partial \delta u_i}{\partial x_j}\sigma_{ij}d\Omega = \int_\Omega \delta u_i b_i d\Omega + \int_{\Gamma_t} \delta u_i t_i d\Gamma + \int_\Gamma \left[\!\left[\delta u_i \right]\!\right]\{\sigma_{ij}\}n_j^1 d\Gamma \tag{4.9}$$

Rearranging the terms to separate known and unknown, and adding a term to preserve a symmetric method

$$\int_\Omega \frac{\partial \delta u_i}{\partial x_j}\sigma_{ij}d\Omega - \int_\Gamma \left[\!\left[\delta u_i \right]\!\right]\{\sigma_{ij}\}n_j^1 d\Gamma - \int_\Gamma \left[\!\left[u_i \right]\!\right]\{\delta\sigma_{ij}\}n_j^1 d\Gamma = \int_\Omega \delta u_i b_i d\Omega + \int_{\Gamma_t} \delta u_i t_i d\Gamma \tag{4.10}$$

And, according to Nitsche, a stabilization term is added

$$\begin{aligned}&\int_\Omega \frac{\partial \delta u_i}{\partial x_j}\sigma_{ij}d\Omega - \int_\Gamma \left[\!\left[\delta u_i \right]\!\right]\{\sigma_{ij}\}n_j^1 d\Gamma - \int_\Gamma \left[\!\left[u_i \right]\!\right]\{\delta\sigma_{ij}\}n_j^1 d\Gamma + \int_\Gamma \alpha \left[\!\left[\delta u_i \right]\!\right]\left[\!\left[u_i \right]\!\right] d\Gamma = \\ &= \int_\Omega \delta u_i b_i d\Omega + \int_{\Gamma_t} \delta u_i t_i d\Gamma\end{aligned} \tag{4.11}$$

where α is a penalty-like stabilization parameter. It is clear that Eq. (4.11) is equal to the strong form because all the $\left[\!\left[\mathbf{u} \right]\!\right] = 0$. It can be also converted in matrix form to be ready for discretisation

$$\int_\Omega \delta\varepsilon^T \sigma d\Omega - \int_\Gamma \left[\!\left[\delta\mathbf{u} \right]\!\right]^T \mathbf{n}\{\sigma\}d\Gamma - \int_\Gamma \{\delta\sigma\}^T \mathbf{n}\left[\!\left[\mathbf{u} \right]\!\right] d\Gamma + \int_\Gamma \alpha \left[\!\left[\delta\mathbf{u} \right]\!\right]^T \left[\!\left[\mathbf{u} \right]\!\right] d\Gamma = \int_{\Gamma_t} \delta\mathbf{u}^T \mathbf{t} d\Gamma \tag{4.12}$$

where

$$\sigma = \begin{Bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{xz} \end{Bmatrix},\ \varepsilon = \begin{Bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \varepsilon_{zz} \\ 2\gamma_{xy} \\ 2\gamma_{yz} \\ 2\gamma_{xz} \end{Bmatrix},\ \mathbf{n} = \begin{bmatrix} n_x & 0 & 0 & n_y & 0 & n_z \\ 0 & n_y & 0 & n_x & n_z & 0 \\ 0 & 0 & n_z & 0 & n_y & n_x \end{bmatrix} \tag{4.13}$$

and the average operator for stress generally defined as

$$\{\sigma\} = \beta\sigma^1 + (1-\beta)\sigma^2,\ 0 \le \beta \le 1 \tag{4.14}$$

4.2. Discretisation in finite elements at the interface

The two domains are now identified as Ω^k. The displacement fields are

$$
\begin{aligned}
u_i^k &= N_I^k a_{iI}^k \\
\delta u_i^k &= N_I^k \delta a_{iI}^k
\end{aligned}
\tag{4.15}
$$

and the stresses and strains are

$$
\sigma^k = \mathbf{C}^k \mathbf{B}^k \mathbf{a}^k, \;\; \varepsilon^k = \mathbf{B}^k \mathbf{a}^k \tag{4.16}
$$

which makes become Eq. (4.12)

$$
\begin{aligned}
&\int_\Omega \delta \mathbf{a}^T \mathbf{B}^T \sigma d\Omega - \int_\Gamma \left(\mathbf{N}^1 \delta \mathbf{a}^1 - \mathbf{N}^2 \delta \mathbf{a}^2 \right)^T \mathbf{n}\{\sigma\} d\Gamma - \int_\Gamma \frac{1}{2} \left(\mathbf{C}^1 \mathbf{B}^1 \delta \mathbf{a}^1 - \mathbf{C}^2 \mathbf{B}^2 \delta \mathbf{a}^2 \right)^T \mathbf{n}^T \left[\!\left[\mathbf{u} \right]\!\right] d\Gamma + \\
&+ \int_\Gamma \alpha \left(\mathbf{N}^1 \delta \mathbf{a}^1 - \mathbf{N}^2 \delta \mathbf{a}^2 \right)^T \left[\!\left[\mathbf{u} \right]\!\right] d\Gamma = \int_{\Gamma_t} \delta \mathbf{a}^T \mathbf{N}^T \mathbf{t} d\Gamma
\end{aligned}
\tag{4.17}
$$

Considering the fixed constraint (Dirichlet boundary condition) and that $\delta \mathbf{a}$ is arbitrary

$$
\begin{aligned}
\int_\Omega \mathbf{B}^T \sigma d\Omega &= \int_{\Gamma_t} \mathbf{N}^T \mathbf{t} d\Gamma \\
-\int_\Gamma \mathbf{N}^{1T} \mathbf{n}\{\sigma\} d\Gamma - \int_\Gamma \frac{1}{2} \mathbf{B}^{1T} \mathbf{C}^{1T} \mathbf{n}^T \left[\!\left[\mathbf{u} \right]\!\right] d\Gamma + \int_\Gamma \alpha \mathbf{N}^{1T} \left[\!\left[\mathbf{u} \right]\!\right] d\Gamma &= \mathbf{0} \\
\int_\Gamma \mathbf{N}^{2T} \mathbf{n}\{\sigma\} d\Gamma - \int_\Gamma \frac{1}{2} \mathbf{B}^{2T} \mathbf{C}^{2T} \mathbf{n}^T \left[\!\left[\mathbf{u} \right]\!\right] d\Gamma - \int_\Gamma \alpha \mathbf{N}^{2T} \left[\!\left[\mathbf{u} \right]\!\right] d\Gamma &= \mathbf{0}
\end{aligned}
\tag{4.18}
$$

where the first equation is the standard assembly for each domain, and the other two are the terms at the interface to perform the coupling.
Using a more compact form

$$
\left[\mathbf{K}^b + \mathbf{K}^n + \left(\mathbf{K}^n \right)^T + \mathbf{K}^s \right] \{\mathbf{x}\} = \{\mathbf{f}_{ext}\} \tag{4.19}
$$

where the first term is again the standard assembly without coupling

$$
\mathbf{K}^b = \sum_{i=1}^{2} \int_{\Omega^k} (\mathbf{B}^k)^T \mathbf{C}^k \mathbf{B}^k d\Omega \tag{4.20}
$$

and the others are the Nitsche's terms

$$
\mathbf{K}^n = \begin{bmatrix} -\int_\Gamma \mathbf{N}^{1T} \mathbf{n} \frac{1}{2} \mathbf{C}^1 \mathbf{B}^1 d\Gamma & -\int_\Gamma \mathbf{N}^{1T} \mathbf{n} \frac{1}{2} \mathbf{C}^2 \mathbf{B}^2 d\Gamma \\ \int_\Gamma \mathbf{N}^{2T} \mathbf{n} \frac{1}{2} \mathbf{C}^1 \mathbf{B}^1 d\Gamma & \int_\Gamma \mathbf{N}^{2T} \mathbf{n} \frac{1}{2} \mathbf{C}^2 \mathbf{B}^2 d\Gamma \end{bmatrix} \tag{4.21}
$$

$$
\mathbf{K}^s = \begin{bmatrix} \int_\Gamma \alpha \mathbf{N}^{1T} \mathbf{N}^1 d\Gamma & \int_\Gamma \alpha \mathbf{N}^{1T} \mathbf{N}^2 d\Gamma \\ \int_\Gamma \alpha \mathbf{N}^{2T} \mathbf{N}^1 d\Gamma & \int_\Gamma \alpha \mathbf{N}^{2T} \mathbf{N}^2 d\Gamma \end{bmatrix} \tag{4.22}
$$

The negative terms in Eq. (4.21) might cause the full stiffness matrix to be not positive defined. The stabilisation parameter in Eq. (4.22) works as a penalty-like term that does not change the nature of the coupling but it allow the matrix to be positive defined.
This parameter can be either empirically chosen using the general formula in (4.23), as studied in [36] and [37]

$$\alpha = \frac{\lambda + \mu}{2} \frac{\theta(p)}{h_e} \tag{4.23}$$

or computing the minimum value using a global [38] or local [39] eigenvalue problem. In this thesis the automatic computation is not implemented, while it is performed an evaluation at different values.

4.3. Implementation remarks

The matrices in Equation 4.21 and 4.22 must be numerically computed, thus Gauss points must be chosen.
Taking into account a general case where the two coupling surfaces do not feature compatible mesh and they are dimensionally different, the placement of the Gauss points on the two surfaces is not trivial and a procedure must be carried out.
The key point to correctly preform the coupling is that the integration points must be at the same position in order to define the equality of displacements and stresses. if one of the two surfaces is smaller than the other, this will be the surface in which the integration points are defined, in order of being sure the existence of coupling integration points on the other surface.
In the example in Figure 4.2 the case where one interface surface is smaller than the other is shown, with the position of Gauss points of both sides, while in Figure 4.3 the position of Gauss points for domain 1 and 2 separately is shown. The points are defined from domain 1.

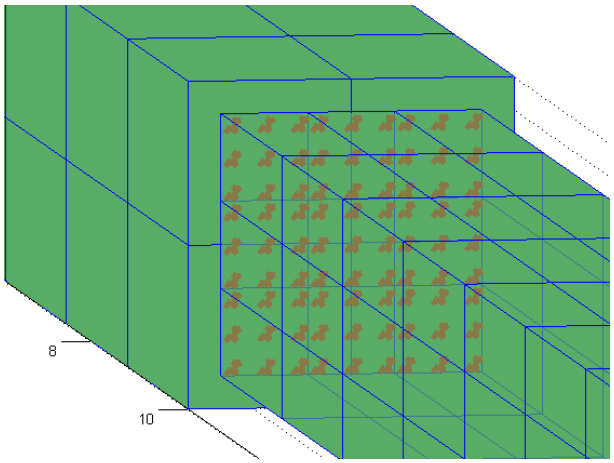


Figure 4.2 – Gauss point positions for both interface surfaces.

The translation of the points in parent element (range [0,1]) to the points in NURBS parametric coordinates (range $[\xi_i, \xi_{i+1}]$), is performed for each element of the surface of domain 1, involved in the coupling.
Once the position in parametric space are defined, the mapping to the physical Cartesian coordinates is performed, to apply an inverse mapping on the second domain to obtain the parametric coordinates on the surface on the other side, and be able to perform the integration on that side as well.

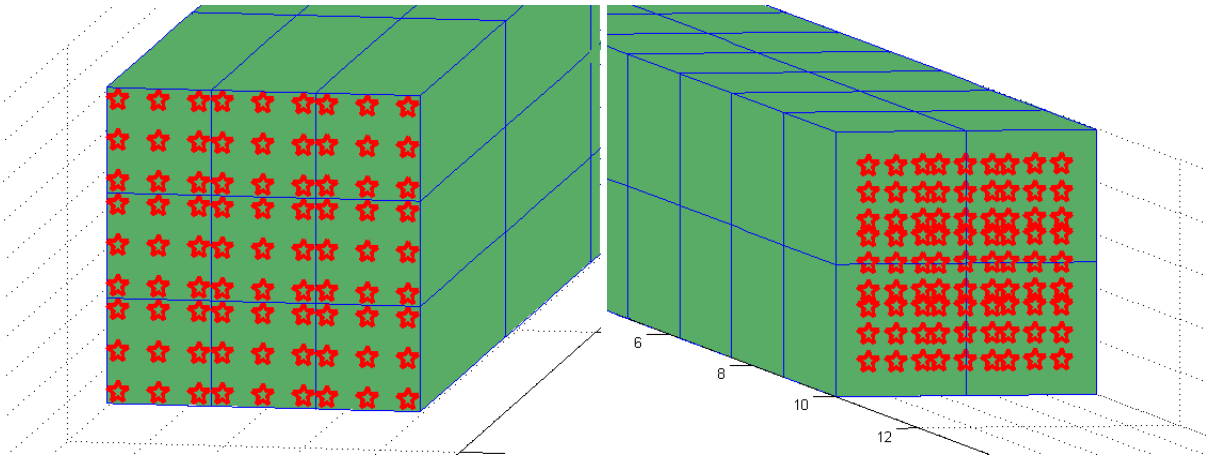


Figure 4.3 – Detail of Gauss point positions for domain 1 (left), where they are defined, and domain 2 (right).

The general algorithm for performing the coupling is then the following:

- definition of the number of GPs per direction per element;
- loop through the elements:
    - loop through Gauss points:
        - computation of basis functions and derivatives;
        - computation of global coordinates;
        - computation of tangential vectors at Gauss point;
        - computation of normal vector at Gauss point;
        - point inversion to get parametric coordinates at other side of coupling;
        - computation of Nitsche terms at Gauss point on both sides;
        - assembly of values to global stiffness matrix;

4.3.1. Simple example

Some trivial examples are used to validate the method and to prove its feasibility and reliability.
In particular the example in Figure 4.2 is represented by a beam with a squared cross section divided in two sections of different section area, one smaller than the other. The full model is shown in Figure 4.4.
The two patches are refined differently in order to ensure non-conforming mesh at the interface, obtaining the discretization in Figure 4.5.

This example is purely mathematical, then no measurement units are considered, and the boundary conditions of the test are imposed considering clamped condition on the left side and a transversal force on the right side along –z direction. The results are summarised as 3D plots in Figure 4.6, with the comparison with standard FEM code.
In Figure 4.7, the comparison of the position of non-zero values in the stiffness matrices are shown.

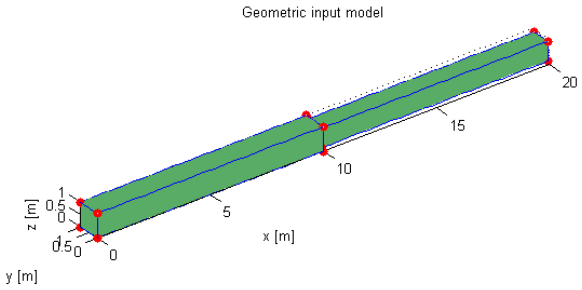


Figure 4.4 – Geometrical model of the beam with two cross-sections.

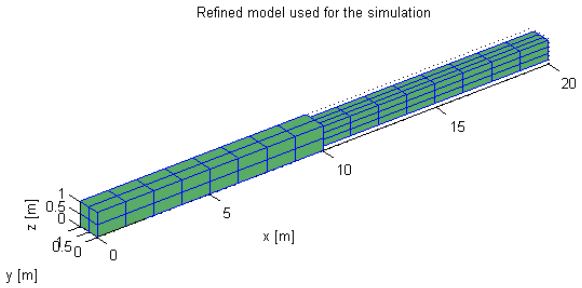


Figure 4.5 – Different discretisation of the two patches

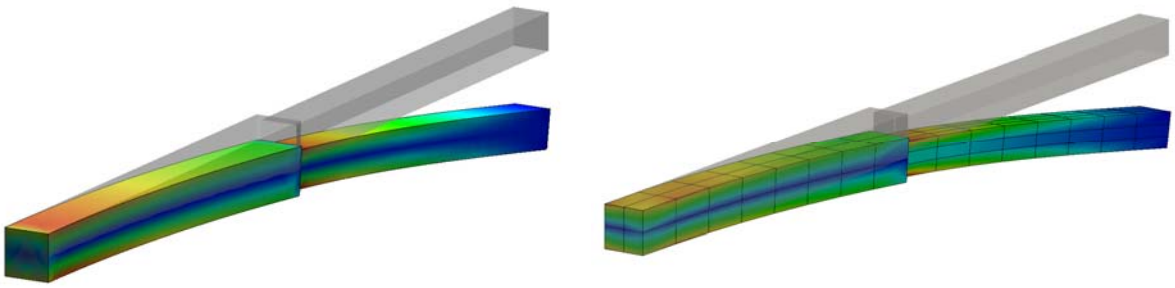


Figure 4.6 – Results of the example with standard FEM (left) and IGA (right) plots.

It is possible to note that the standard assembly is a block-diagonal matrix where each block is the assembly for each patch and the separation among the patches is clear, while in the final matrix the coupling terms that link the degrees of freedom of the first patch to the degrees of freedom of the second can be seen.

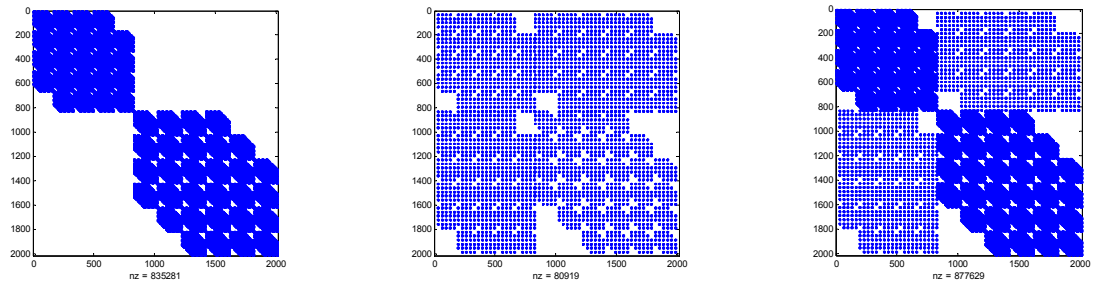


Figure 4.7 – Stiffness matrices for standard assembly (left), Nistche's terms only (middle) and the sum of the two as final matrix (right).

4.3.2. The stabilisation parameter

The value of the stabilisation parameter α is chosen empirically for all the example and application in this thesis, starting from an estimation considering the order of magnitude of the elasticity modulus and an average element dimension at the interface, and then checking the positive-definiteness of the final stiffness matrix.
For the simple example above, a parametric analysis is considered by varying the value of α and checking the results in terms of norm of the distances between the Gauss points in the undeformed and deformed configurations. The results are shown in Figure 4.8.

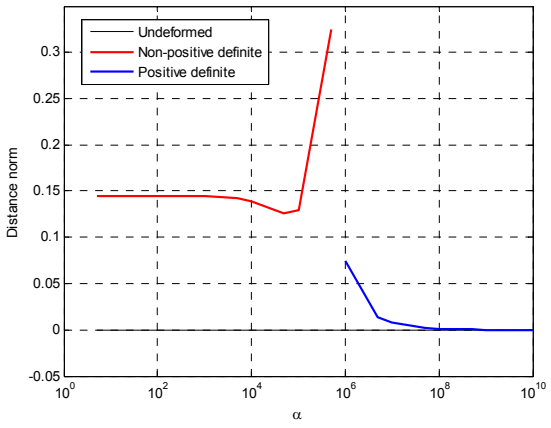


Figure 4.8 – Distance norm between coupling Gauss points vs stabilisation parameter α.

The red curve is related to the values of distance norm for parameter values that do not recover the positive definiteness of the matrix, while the blue curve is related to values where the matrix is positive definite. The spike on the red curve is due to values where the stiffness matrix is becoming close to singularity.

The Nitsche's coupling technique is valid not only for solving linear static elasticity, but also for dynamic problems like linear modal analysis.
The stiffness matrix is the same obtained with Equation 4.19 and the mass matrix is the one obtained with the standard assembly procedure.
Some of the mode-shapes for the modal analysis of the simple example are shown in Figure 4.9, with the comparison to standard FEM solution.

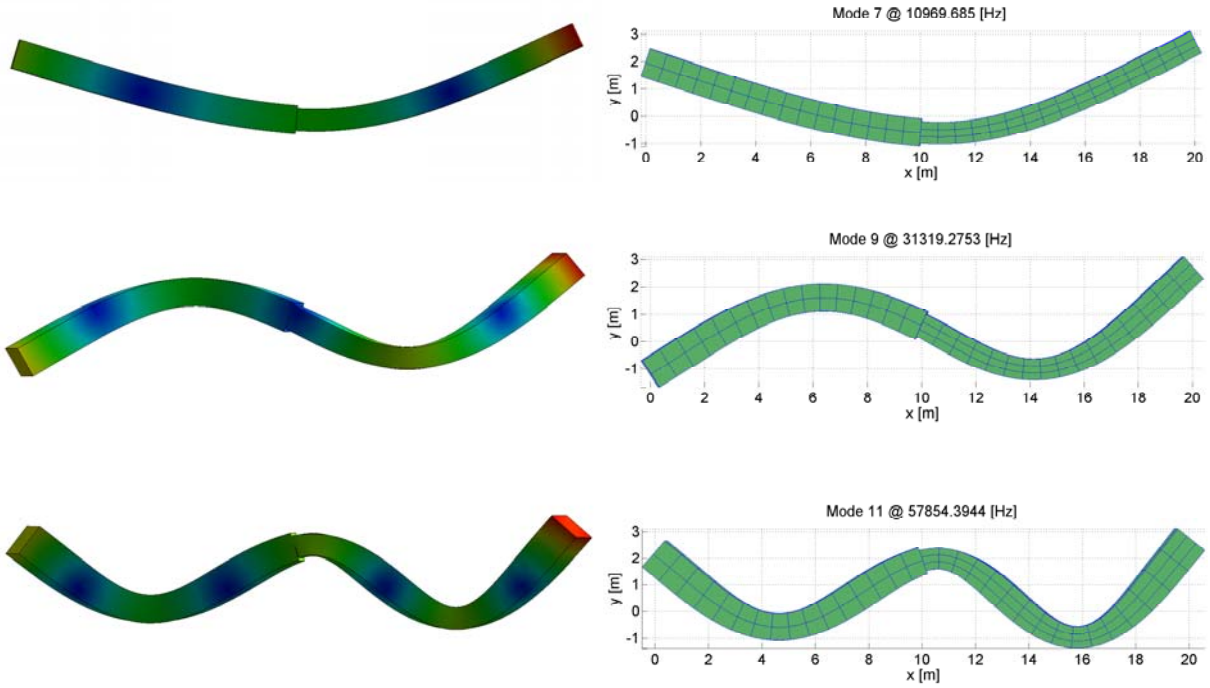


Figure 4.9 – Mode-shapes comparison between standard FEM in SolidWorks (left) and IGA with Nitsche's coupling (right).

5. Applications of tri-variate multi-patch isogeometric analysis with Nitsche's coupling

The method of multi-patch isogeometric analysis with non-conforming interface mesh joined with Nitsche's coupling is now validated considering models, geometries and conditions of increasing complexity, increasing accuracy and industrial application.
The non-conforming coupling allows the possibility of modelling geometry composed by several patches with large flexibility on the nature of the interface zones, giving more freedom to the modelling part and reaching any geometrical complexity including holes, slots, sharp edges, and interfaces with different dimensions, without the issue of severely distorted elements.
First a schematic representation of a connecting-rod is used as a mechanical example with features that require non-conforming coupling. This model is analysed in linear static elasticity.
A second model is the representation of an existing experimental test-rig to investigate dynamic properties of coupled system with a geometrical parameter-dependent value, where this geometrical dependency require a CAD-CAE integration technique for managing the change in mesh. Linear modal analysis is considered in this case and dynamic properties of *curve crossing* and *curve veering* are analysed as well.
The third case is an actual industrial geometry in the automotive field, which is a first test of application in industry. The geometry is the external layer of a truck door and two models of increasing complexity and accuracy are developed.
For the third case, the original CAD data file is in *CATIA*, and to convert it to be suitable for the required data-structure and being used in the isogeometric code, Rhinoceros is used to generate the suitable model from an IGES exported model.

5.1. 3D connecting-rod

An example that could represent a very hard geometry to intentionally model with coincident and collapsed control points to apply the coupling in strong form, would have both curved surface at the interface and different dimensions (one surface larger than the other). Even though it would be possible, to obtain the compatibility the mesh obtained might be distorted and further refinements would require additional modification to fit the compatibility.
A mechanical component that could fit these requirements and that, as a first attempt, could be schematically represented without loss on feature characteristics is an internal combustion engine's connecting-rod. An example of the actual component is shown in Figure 5.1.
This is a component of the crank-mechanism assembly that is, in general, a mechanical systems with the aim of converting a circular motion to an alternative straight motion. A drawing of the mechanism is presented in Figure 5.2.
The mechanism is suitable in both driver and user side, converting the motion coming from the crank to the piston in case of a user (for example alternative compressors) or vice-versa converting the straight motion coming from the ignition of the fluid inside the combustion chamber to the circular motion of the crank-shaft to drive vehicles.

Figure 5.1 – Picture of a connecting-rod.

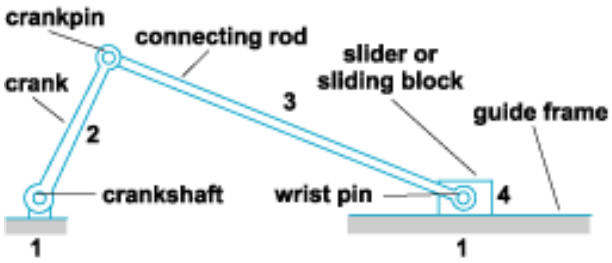


Figure 5.2 – Schematic drawing of a crank-mechanism.

The crank-pin (or *big end*) joins the connecting-rod to the crank, while the wrist-pin (or *little end*) joins it to the piston through the piston-pin.
The connecting-rod (or *conrod* in short) is one of the components of this assembly that needs more attention because of the loads that it has to face.

The example in Figure 5.3 represents a simplified schematic look for a conrod, suitable for an attempt to run a multi-patch isogeometric analysis with non-conforming meshes.

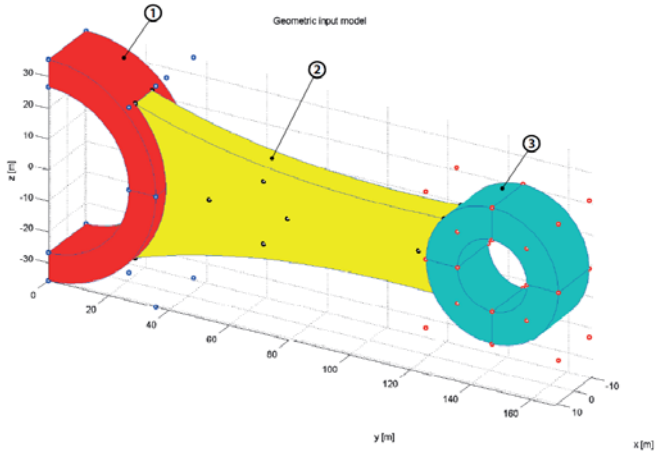


Figure 5.3 – Multi-patch NURBS for a connecting-rod.

It is composed by three patches, the first in red to represent the big end, is a half of an hollow cylinder; the third patch in cyan is an hollow cylinder; the second patch in yellow represents

the stem that connects the two ends, where the connections with the ends are arcs of circles and the sides of the stem are portions of parabola.
In Figure 5.4 a front view of the conrod model is shown, with the details of all the geometrical data and a representation of the boundary conditions (constraints and loads) imposed to the mathematical model.

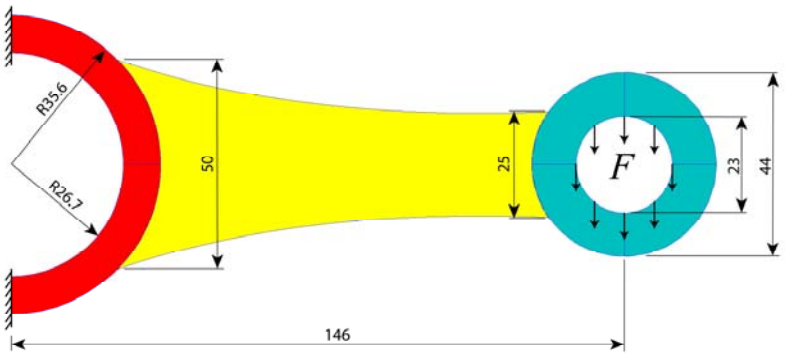


Figure 5.4 – Boundary conditions for the analysis of the connecting-rod
(all the dimensions are in mm).

For this case only MATLAB, with the help of the NURBS toolbox routines, is used to build the whole geometry of all the three patches.

For Patch #1 the first parametric direction ξ is the circumferential direction of the half hollow cylinder. The two boundary curves are created using the `nrbcirc` function, defining the two radiuses and the π angle span, obtaining the result in Figure 5.5.

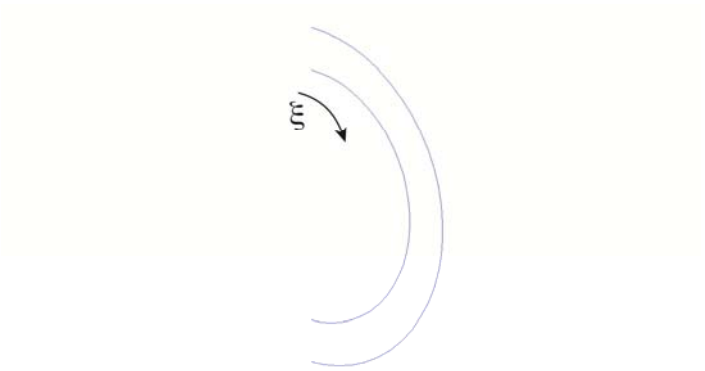


Figure 5.5 – Boundary curves in ξ direction for Patch #1.

From two boundary curves a NURBS *ruled* surface can be obtained, by linearly connect the boundary curves (see [3] for details on the ruled surface algorithm). The function `nrbruled` is used to generate this surface, with the results in Figure 5.6.
The final tri-variate patch is obtained by extruding the surface using the routine `nrbextrude`, which will be widely used in this thesis, defining the depth of the extrusion, obtaining the result in Figure 5.7. Note that the direction is concordant to the normal of the source surface.

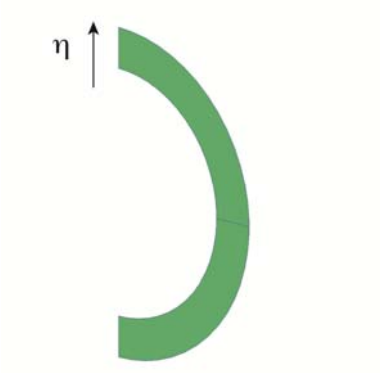


Figure 5.6 – Ruled surface for the $\xi - \eta$ bi-variate surface of Patch #1.

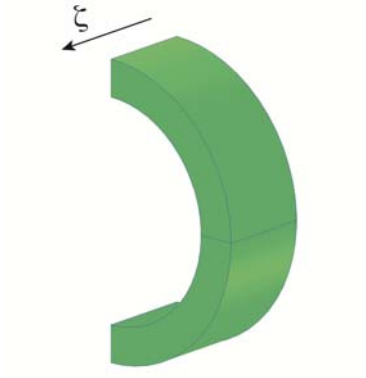


Figure 5.7 – Tri-variate extrusion for Patch #1.

The proposed modelling sequence leads to an unique numbering of the six boundary surfaces, and this numbering is shown in Figure 5.8. This numbering is important for the definition of the patch connection, described in detail further.

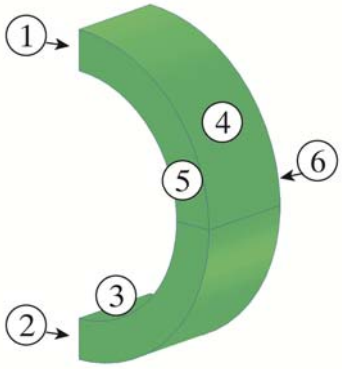


Figure 5.8 – Numbering of the boundary surfaces of Patch #1.

For Patch #2 the base surface to extrude is built by defining four boundary curves, which are the base curves to build a *Coons surface* [40]. Two circular arcs representing the interface with the crank-pin and the wrist-pin are modelled, using `nrbcirc`, taking care of the span angles, and then `nrbftorm` to translate the arcs from the (0, 0) centre origin, and `nrbreverse` to reverse the evaluation direction and obtain the result in Figure 5.9.

Figure 5.9 – The two circular arcs representing the connection of the stem (Patch #2) to the crank-pin and wrist-pin.

The other two curves are built connecting the circular arcs with a quadratic B-spline and adjusting the position of the middle control point in order of reaching the desired shape, as shown in Figure 5.10.

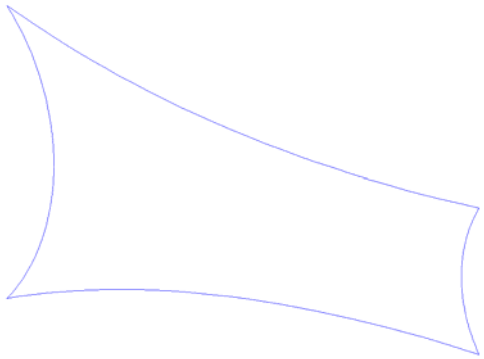


Figure 5.10 – The four boundary curves for generating the base surface of Patch #2.

The NURBS Toolbox function to generate the Coon's patch is `nrbcoons`, which requires the four boundary curves to be selected in the sequence:

- first curve in direction ξ;
- second curve in direction ξ;
- first curve in direction η;
- second curve in direction η;

The result after the function is shown in Figure 5.11.
As it is performed for Patch #1, an extrusion defining the extrusion depth is performed, obtaining the final result in Figure 5.12.
The numbering of the six boundary surfaces is defined in Figure 5.13.
The modelling strategy for Patch #3 is the same of Patch#1, with the difference of the span angle (full circles) for the circumferential direction, and for the orientation of the three parametric dimensions.

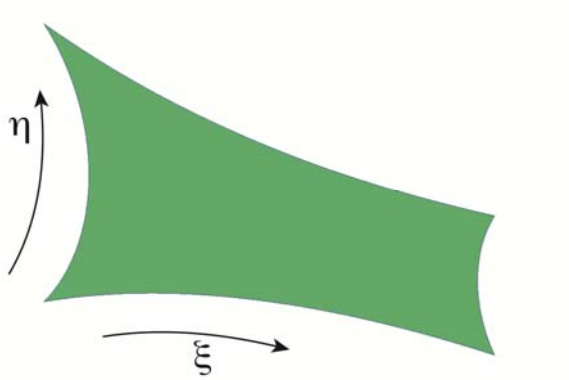


Figure 5.11 – Coon's patch for the base surface of Patch #2.

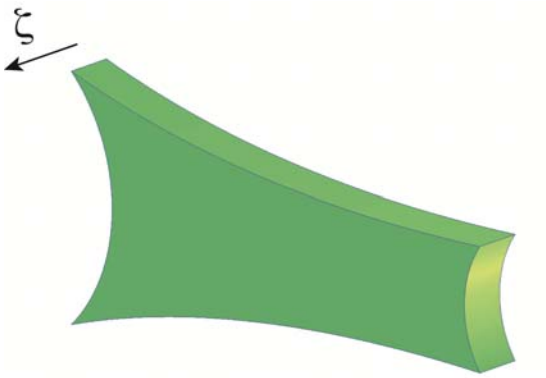


Figure 5.12 – Final result of the tri-variate Patch #2.

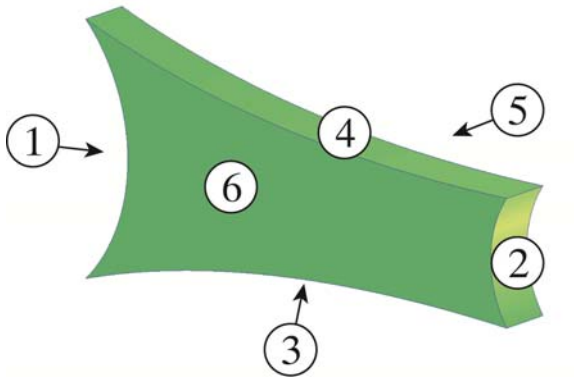


Figure 5.13 – Tri-variate Patch #2 model with label of the boundary surfaces.

A summarised Figure 5.14 shows the parametric directions, while Figure 5.15 defines the labels for the six boundary surfaces.

Note that surface #1 and #2 are coincident and then a conforming mesh coupling must applied, simply imposing the equality of the displacements of each control points of one of the surfaces (for example #2) with respect to the displacements of the other (#1).
The three patches are then assembled and positioned one another in order to follow the specifications in Figure 5.4, using `nrbtform` to roto-translate the patches, and the final result is the one in Figure 5.16.

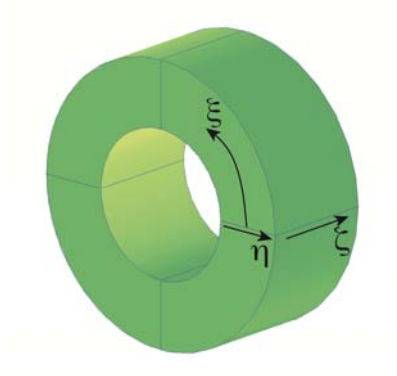


Figure 5.14 – Parametric directions for Patch #3.

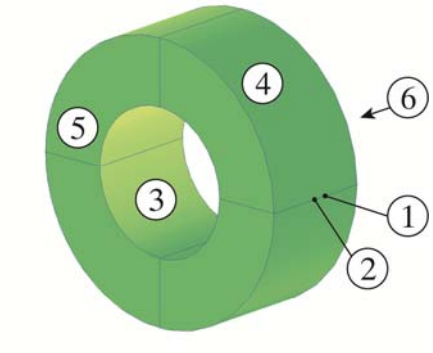


Figure 5.15 – Numbering labels for boundary surfaces of Patch #3.

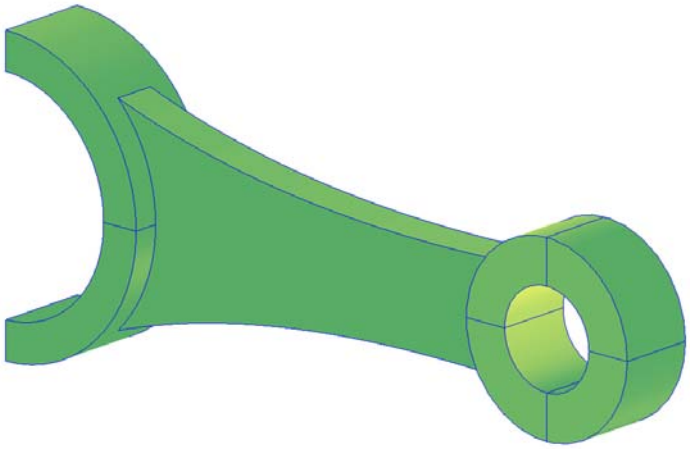


Figure 5.16 – The three patches positioned to form the full conrod model.

The model is refined to reach the desired average mesh size and to reach a good accuracy, with the intention not to prove the accuracy of the IGA method with respect to standard FEM, but to compare the results with equal computational cost in terms of stiffness matrix dimensions. In this way the refined IGA model has the same number of degrees of freedom as the FEM model used as comparison. The refinement details are summarised in Table 5.1.

Table 5.1 – Refinement details for IGA model of conrod.

Patch ID	p-refinement	h-refinement
Patch #1	Tri-cubic	32×4×8
Patch #2	Tri-cubic	24×12×4
Patch #3	Tri-cubic	64×4×8
	Nr. elements	4244
	Nr. control points	11305

The total force F applied to the internal surface of the wrist-pin (simulating the action of the piston-pin, surface #3 of Patch #3) is set to 1000 N, acting as a transversal loading, along direction -z, the boundary constraint is an ideal clamping conditions (zero-displacement) on the control points belonging to surfaces #1 and #2 of Patch #1. The material properties are summarised in Table 5.2, and the Nitsche's coupling is to be applied between surface #1 of Patch #2 and surface #4 of Patch #1, and between surface #2 of Patch #2 and surface #4 of Patch #3. The order in which the connections are defined does not have influence, while the choice inside the connection is crucial: the first face defined must be completely included in the second, in order of ensure the existence of neighbour Gauss points on the other surface. The refined IGA model is shown in Figure 5.17.

Table 5.2 – Conrod material properties

Characteristic	Property	Value
Conrod material	Young's modulus E [GPa]	200
	Poisson's ratio ν [-]	0.3

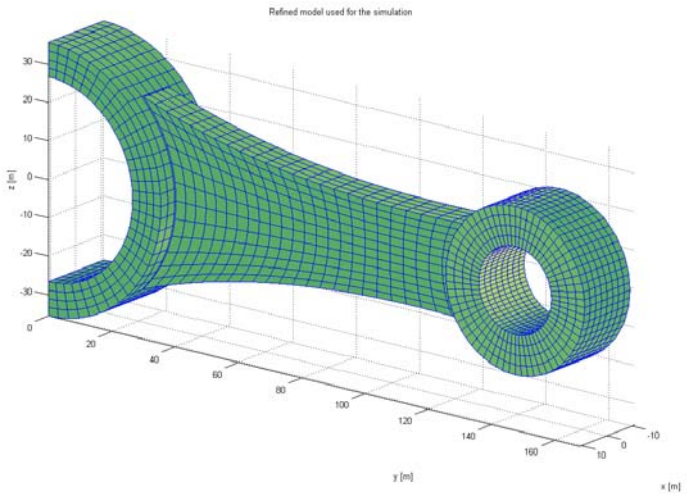


Figure 5.17 – Refined IGA model of conrod used by the computations.

The stabilisation parameter α is a global value, thus a unique parameter is defined for any interface in the model. Its choice is further described.

The results are shown in Figure 5.18 and Figure 5.19, with displacement and stress contour plots, which can be compared to the same solution of standard FEM software NX-NASTRAN. This NASTRAN model is composed by second-order tetrahedral elements, with details summarised in Table 5.3. The stress field contour plot for NASTRAN model is shown in Figure 5.20.
The IGA model result control plots are generated exporting data in ASCII file to be read by the ParaView software, which is used to produce the related figures.

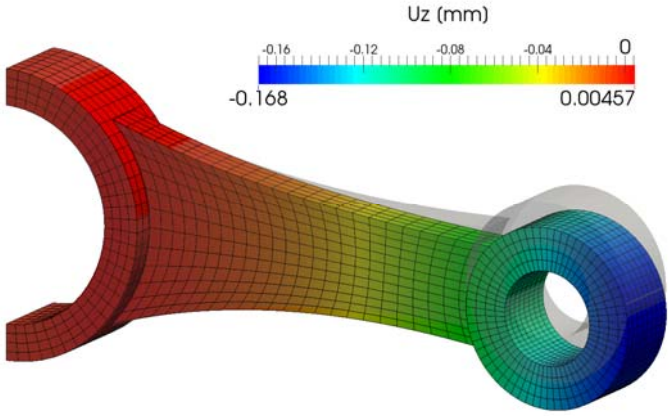


Figure 5.18 – Displacement field plot for IGA conrod model, along *z* direction.

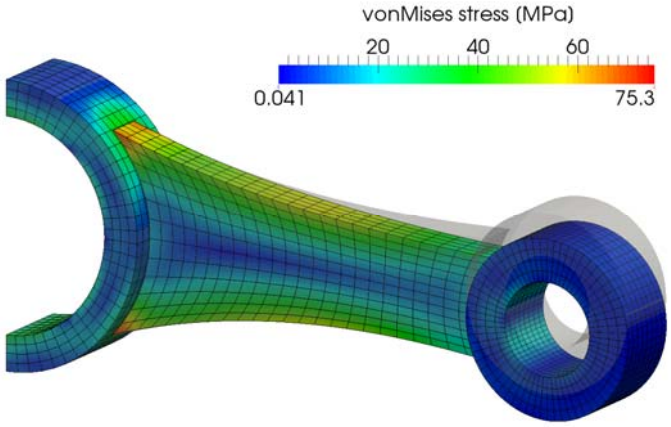


Figure 5.19 – VonMises stress field plot for IGA conrod model.

Table 5.3 – Conrod NASTRAN model details.

Characteristic	**Property**	**Value**
Conrod NASTRAN model	Number of elements.	6182
	Number of nodes.	11332

For purely investigation purposed, a parametric run is performed spanning the value of the stabilisation parameter in order to check the behaviour of the model in dependence of this parameter and for helping to empirically choice its value for further analyses.

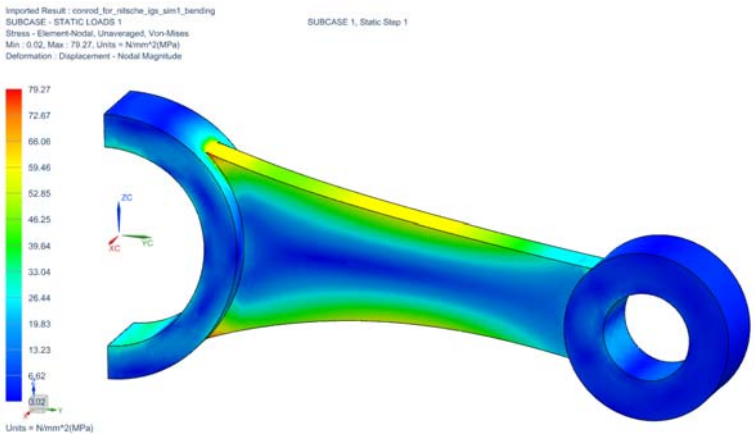


Figure 5.20 – VonMises stress plot for NASTRAN model.

Figure 5.21shows the curve representing the values of the maximum von Mises stress with respect to the stabilisation parameter, while Figure 5.22 and Figure 5.23 show the contour plot at lower and upper bounds of the investigation.

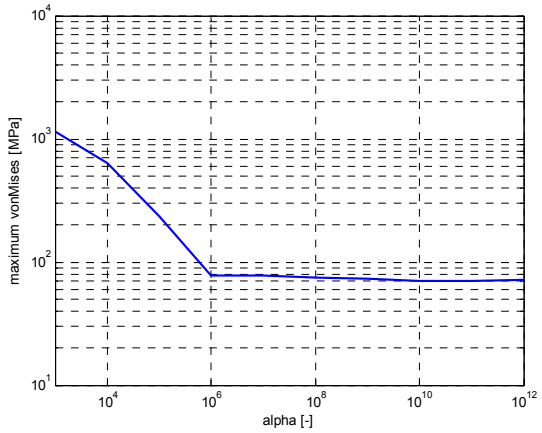


Figure 5.21 – Stress concentration vs. stabilisation parameter.

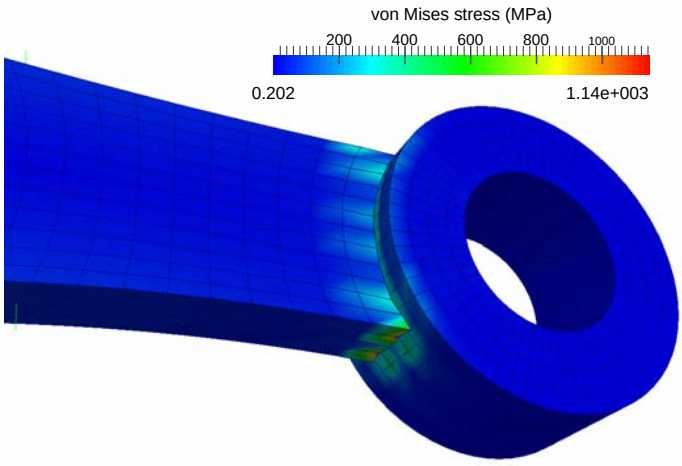


Figure 5.22 – Detail of a coupling zone for stabilisation parameter of 10^3.

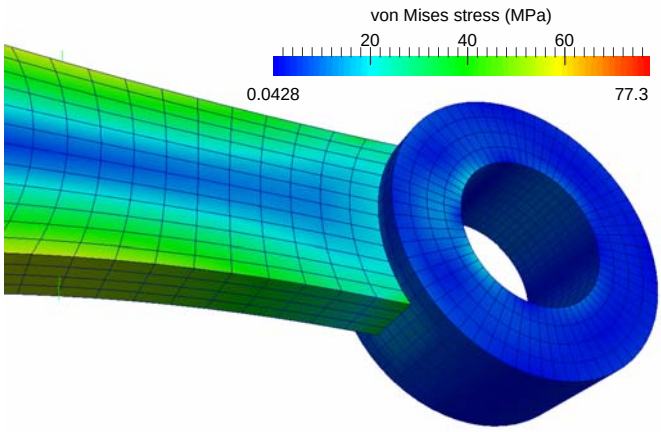


Figure 5.23 – Detail of a coupling zone for stabilisation parameter of 10^{12}

5.2. Coincident eigenvalues test-rig and parametric modal analysis

The possibility of applying multi-patch isogeometric analysis with non-conforming coupling for linear modal analysis as well, opens a widespread range of possible tests with gaining the advantages of this method.
One field where IGA and the non-conforming coupling would be an interesting solution is the *parametric modal analysis* for investigating the dynamic behaviour related to *curve crossing* and *curve veering* phenomena.
These phenomena occur when a structure features a geometrical or configuration parameter which variation determines a change in geometrical properties, in the relative positions among the components of the assembly, and those cause a dynamic coupling among vibration modes of the structure.
An experimental test-rig was developed, built and tested by the research group where the author worked with [41], and it is presented in Figure 5.24, along with its CAD model.
The test-rig is roughly composed by three flexible beams and three joints that act as added masses. The whole system is clamped to a highly rigid connection on an inertial mass in order to avoid coupling effects with the ground.
The first joint (labelled Mass #1) features a graduated rotating device which allows a relative angle between Beam #2 and the others, while the second joint (Mass #2) can be fastened or released in order of resetting the original 90° relative orientation of Beam #3 with respect of Beam #1.
The third mass is designed to intentionally lower the range of frequencies and acting as a proof mass. The configuration parameter, in this case, is then the relative angle between Beam #2 and Beam #3, with the zero-value corresponding to these two beams aligned. Figure 5.25 shows detailed pictures of the three masses, with the application of the accelerometric sensors used to run the experiments.
The variation of the angle of Beam #2 is responsible of the coupling between the different bending modes of the structure.
The parametric modal analysis could be run in SolidWorks, using a parametric tool developed by the author during his Master Thesis [42] where a C# macro was able to edit

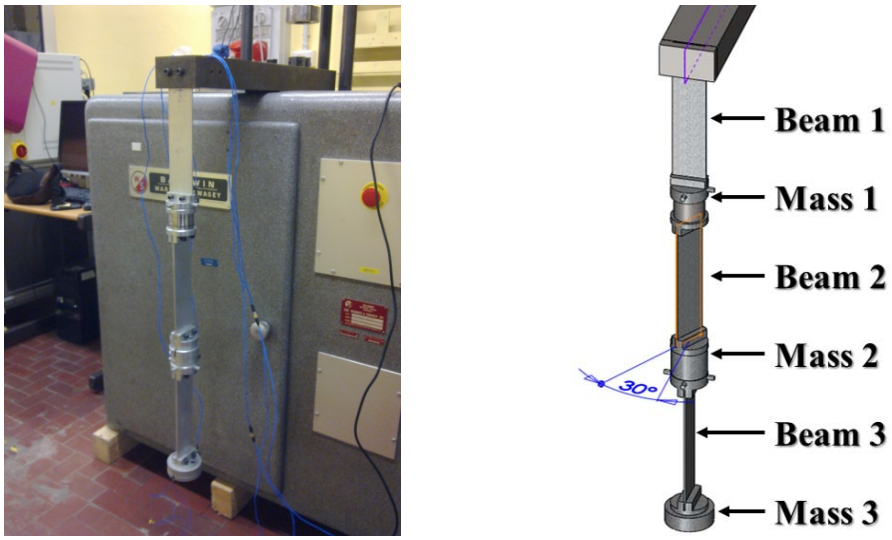


Figure 5.24 – The experimental test-rig: actual model with test hardware (left) and full CAD model with configuration parameter (right).

Figure 5.25 – Details of the test-rig: first added mass with rotating graduated device (left) second added mass with fixing device (middle) and third added mass (right).

the mating condition of Beam #2, automatically re-mesh the whole model and run a modal analysis for each configuration. This is very expensive from the computational point of view, even though the tool dramatically lowers the time involved if the operation between two configuration and the results exporting were manually performed.
Considering the very simple geometry of the test-rig, another attempt performed by the research team was to develop a 1-D Euler-Bernoulli beam-based Finite Element code (named LUPOS, Lumped Parameters Open Source) in MATLAB, with the innovative possibility to parametrically drive any desired parameter and, for this case specifically, no need of re-meshing is necessary, but only a re-orientation of the second beam is involved. The masses were modelled as thick beams, with a base area equivalent to the original. The LUPOS model is shown in Figure 5.26.
The multi-patch NURBS model can be obtained by considering a simplified condition where the beams are thin bricks and the masses are whole cylinders. The thin bricks are modelled with the three parametric NURBS dimensions aligned to the global Cartesian system, as shown in Figure 5.27, while the parametric directions for the masses are shown in Figure 5.28. These two patch are adapted for each component's specific dimensions and then assembled together using `nrbtform` to orient and position them.

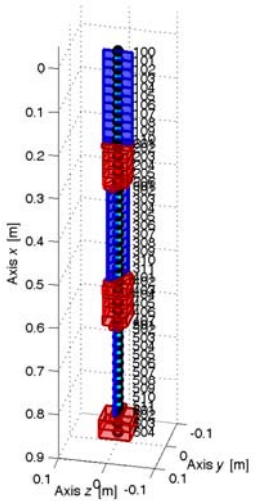


Figure 5.26 – LUPOS test-rig model with node numbering.

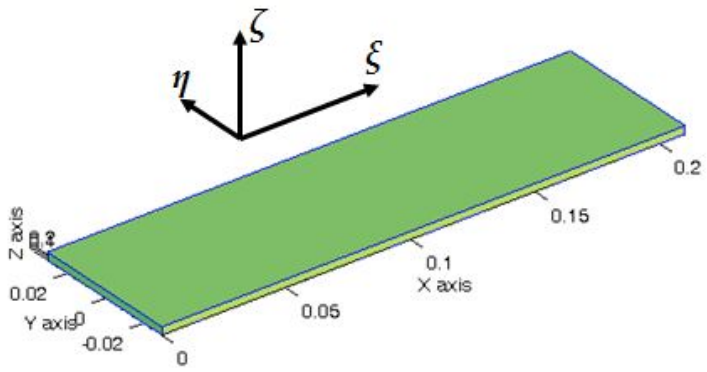


Figure 5.27 – Tri-variate patch for the beams with parametric directions.

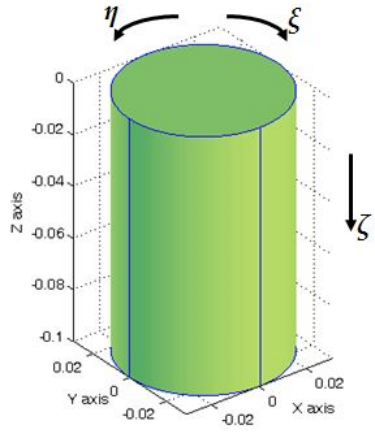


Figure 5.28 – Tri-variate patch for the masses with parametric directions.

The modelling strategy for the whole cylinder, in this case, does not imply the use of circumferential and radial direction, but it is represented by a straight prism with squared base surface, in which the ξ and η directions of the base are elevated to quadratic functions

and the middle control points of each side is stretched, along with their weight, up to reaching the circular shape. The procedure is clarified in Figure 5.29.

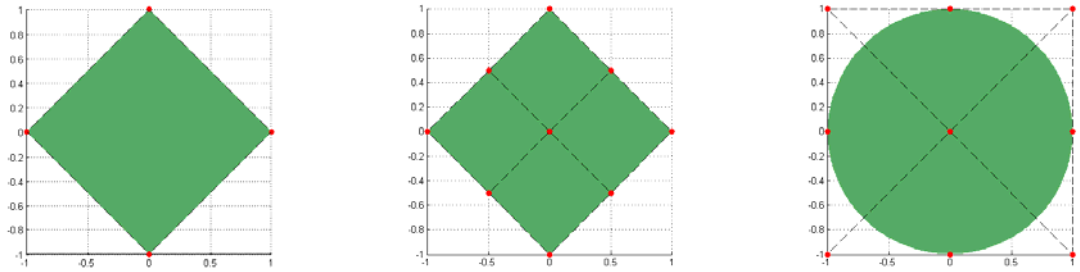


Figure 5.29 – Construction of the base surface of the cylinder starting from a square.

The resulting assembled NURBS model is shown in Figure 5.30. Even though the very schematic look, it is suitable for the analysis involved. The parametric modal analysis will investigate the first 7 global modes at low frequencies, and this is the reason why a schematic model can be consistent with the target.

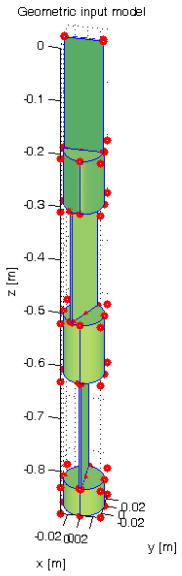


Figure 5.30 – IGA model of the test-rig.

In Figure 5.31 a model comparison of the IGA and LUPOS models with respect to the actual CAD model is shown, thus the geometrical consistency can be easily checked. The length of the masses is considered of the whole clamps, in order to give priority to the beam effective length, compensating the material in excess by computing a bulk density to keep the total mass constant and consistent with the actual components.
A refinement is then applied to the IGA model to increase the accuracy of the and avoid anti-alias issues on the mode-shapes at higher frequencies due to a too coarse mesh. The functions are increased to tri-cubic, the beams are subdivided in 5 elements in its local ξ direction, the masses in two elements for both its local ξ and η directions, and the look is shown in Figure 5.32.
Nitsche's coupling is used to join all the components together, considering that the coupling terms are applied just on the stiffness matrix, while the mass matrix remains uncoupled.

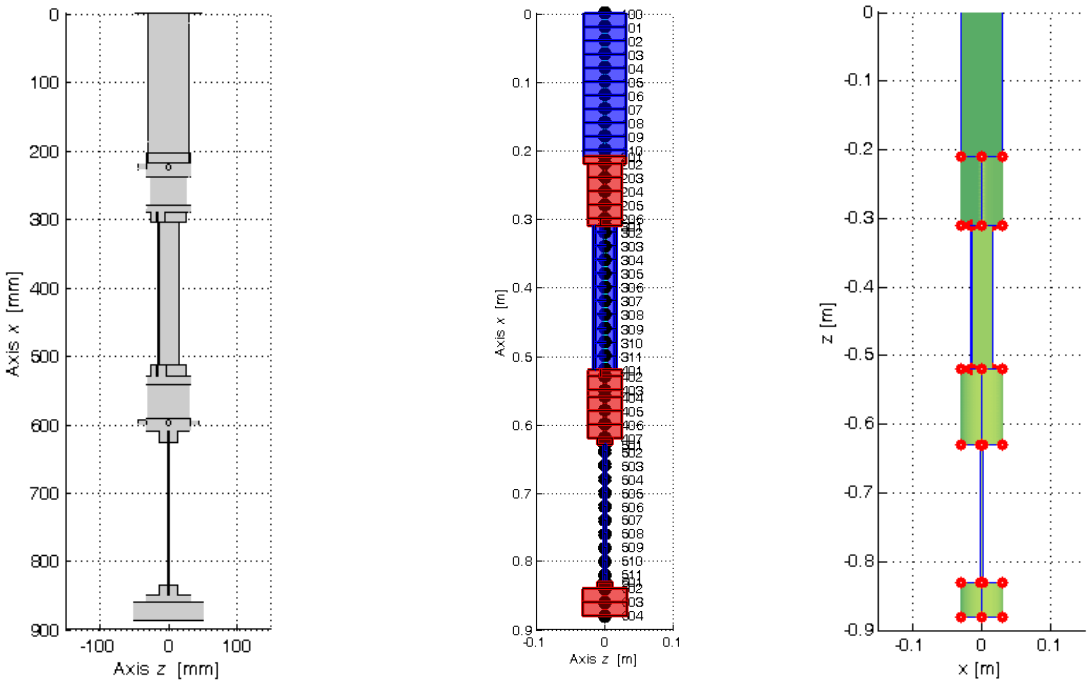


Figure 5.31 – Comparison of the models: SolidWorks CAD (left), LUPOS (middle) and IGA (right)

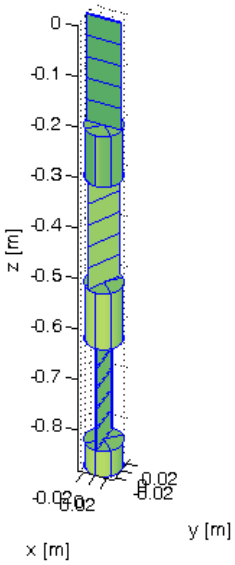


Figure 5.32 – Refined model used to perform the simulation.

The modal analysis is performed by solving the eigenvalue problem that, from the equilibrium equation of motion of the system, considered as undamped

$$[\mathbf{M}]\{\ddot{\mathbf{x}}\} + [\mathbf{K}]\{\mathbf{x}\} = \{\mathbf{0}\} \tag{5.1}$$

with [**M**] and [**K**] being the mass and stiffness matrix respectively, and {**x**}being the vector of degrees of freedom. Again, the stiffness matrix is the one considered with Nitsche's terms.

The eigenvalue problem is, then, obtained solving

$$\det\left(\left[\mathbf{K}\right]-\omega^2\left[\mathbf{M}\right]\right)=0 \tag{5.2}$$

to find the eigenvalues ω^2 and the eigenvectors matrix $[\Phi]$.

As an evidence to confirm the consistency of both LUPOS and IGA models with respect to the SolidWorks FEM one, in Figure 5.33 a comparison of the mode-shapes for modes #3 and #6 are presented, in the reference configuration angle of 30°.

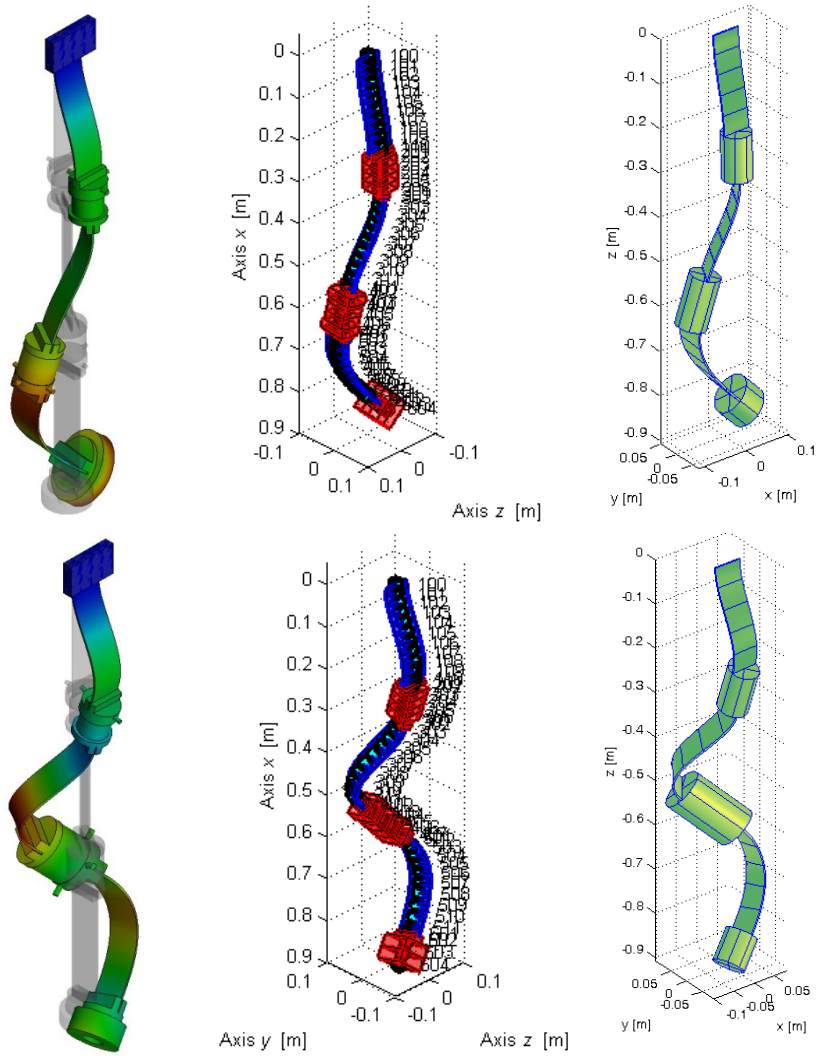


Figure 5.33 – Comparison of Mode 3 (upper) and Mode 6 (lower) among the different models:
SolidWorks (left), LUPOS (middle) and IGA (right).

The three mathematical models have different characteristics and the most interesting is the number of elements and Degrees of Freedom with respect to geometrical accuracy. These details are summarised in Table 5.4.

The shape is comparable among the three different solvers and it looks that LUPOS might be the best because of its simplicity and for the specific case of beam-based structure.

Table 5.4 – Comparison of the details of the three mathematical models.

Model	**# elements**	**# DoFs**
SolidWorks	3996	24840
LUPOS	49	294
IGA	27	2052

Two drawbacks for the LUPOS model must be taken into account: the lack of geometric details of the external shape, in case of the cylindrical components in particular, and the 1-D nature that returns the values of displacements on the neutral fibre of the beam only, thus the results in correspondence of the body's surfaces, in case of comparisons between different models and/or experimental data, are not directly obtained. In some cases the first drawback can be negligible, if the geometrical accuracy is not a point of interest, and the second drawback has a reliable workaround, that is the possibility of define nodes in correspondence of the points in which the comparisons are considered, and linking them with the actual nodes of the beams with rigid joints.
The SolidWorks model, even though is a 3D model, is based on tetrahedral elements and then also in this case the geometry is just approximated and due to the small thickness of the beams, a large number of elements are used, increasing the computational cost and time. This cost is then multiplied by the number of configuration that are analysed resulting in a larger time involved for the whole parametric analysis.
The IGA model is a compromise between the two extreme cases, because it maintains the exact geometry without the meshing step and with the coarsest mesh, good results are obtained with a low number of elements/degrees of freedom and, thanks to the Nitsche's coupling, the re-orientation of the middle beam can be performed by roto-translation of its control points, meaning that at each configuration only the matrices and the coupling terms need to be re-computed, while for the other patches they can be computed once at the beginning.
In Figure 5.34 the non-zero values of the stiffness matrix for one configuration is shown, with the comparison before and after the coupling. It can be noted that due to the in-line arrangement of the patches and of the numeration of the control points, the non-zero entries are all very close to the main diagonal, leading to very sparse matrices.

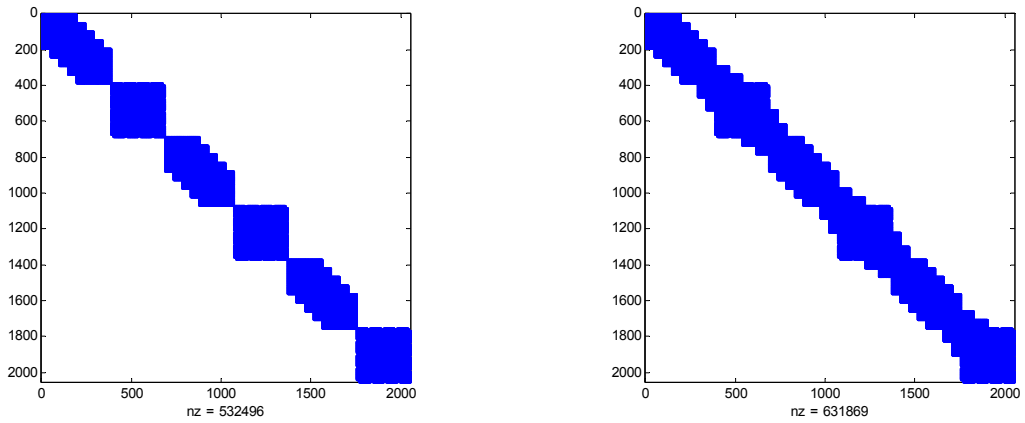


Figure 5.34 – Stiffness matrix of the IGA model before (left) and after (right) coupling.

The analysis is run parametrically, changing the value of the angular position of the middle beam to span a range between 0° and 90° (supposing a symmetric behaviour outside this range) with 1° of step size, to obtain the frequency curves that are shown in Figure 5.35. In these plots the occurrence of crossing and veering are indicated and compared among the three models.

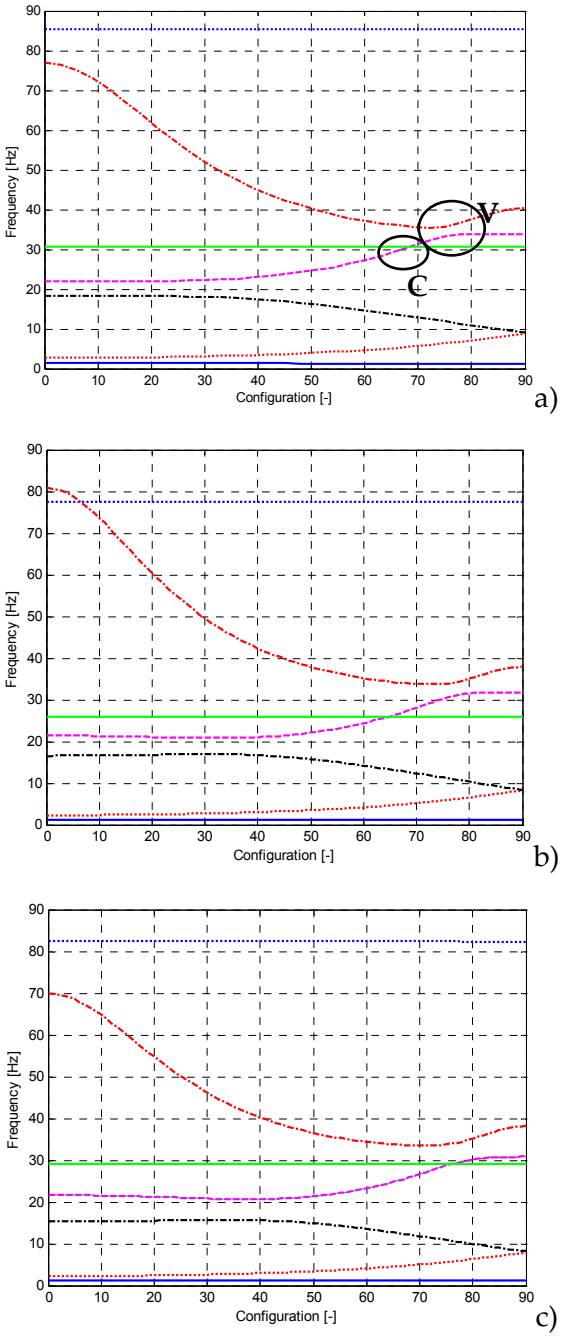


Figure 5.35 – Comparison frequency loci plots: a) SolidWorks; b) LUPOS; c) IGA.

The crossing region highlighted with *C* involves modes 4 and 5 at approximately 70°, while the veering region highlighted with *V* involves modes 5 and 6 between 72° and 76°.
The occurrence of one kind or the other depends on the mode-shapes comparison between two configurations that are across the switch zone. From a quantitative point of view, the

mode-shapes can be compared using the Modal Assurance Criterion (MAC) index, which is defined as

$$MAC_{i,j} = \frac{\left(\Phi_i^T \Phi_j\right)^2}{\left(\Phi_i^T \Phi_i\right)\left(\Phi_j^T \Phi_j\right)} = \cos^2 \alpha_{i,j} \tag{5.3}$$

and which the result value goes from 0 to 1 (or from 0% to 100%).
This index, from the mathematical point of view, takes as input two vectors (the eigenvectors) and it evaluate the parallelism between them, while from the engineering point of view, it compares the mode-shapes and thus higher values mean higher similarity.
The MACs are performed in correspondence of the crossing and veering zones to compare the mode-shapes before and after the two events occur. The obtained values are then plotted on a bar chart for an immediate evaluation. These charts are shown in Figure 5.36 and Figure 5.37.

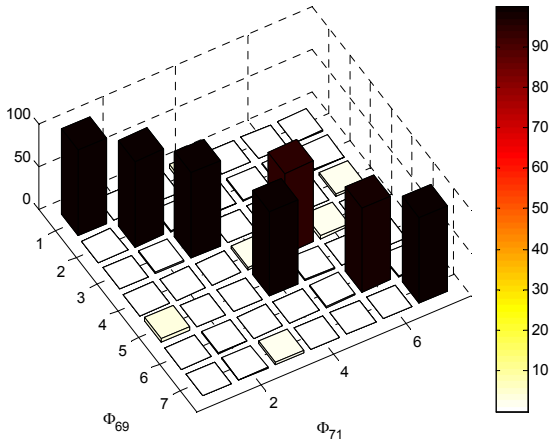


Figure 5.36 – MAC index between configurations 69° and 71°: example of crossing phenomenon.

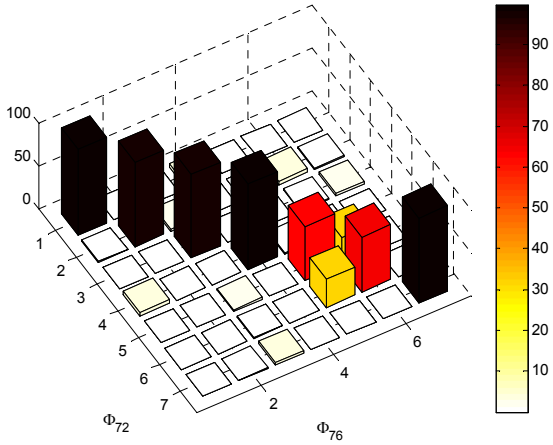


Figure 5.37 – MAC index between configurations 72° and 76°: example of veering phenomenon.

In the first case a complete switch between mode 4 and 5 is an indication that mode 4 at 69° is mode 5 at 71° and vice-versa, suggesting that no interaction between the modes occurred and then a crossing case is obtained.
In the second case, even though the span angle is higher than the first case, an interaction is shown, with the cross-comparison much higher than zero and the mode-to-mode comparison not close to 100%. This suggests that an exchange of modal information is occurring and then a veering case is obtained.
A frequency comparison with respect to the experimental data is performed by superimposing the experimental frequencies to the numerical obtained with IGA, with the aim of updating the Young's modulus of the beams to fit the computations with the experiments.
19 Experimental Modal Analysis (EMA) are performed from 0° to 90° with 5° configuration step, and the results are shown in Figure 5.38 comparing the solutions before and after an updating process to tune the actual Young's modulus of the beams.

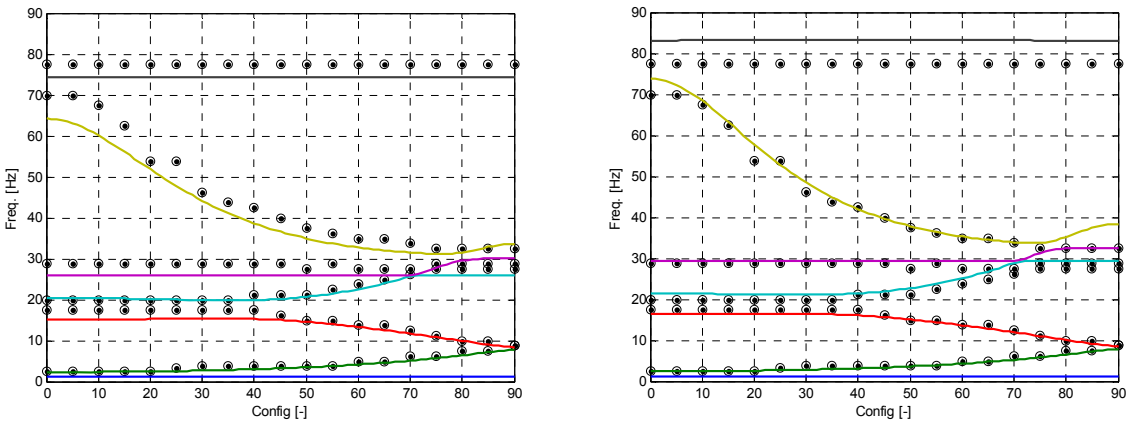


Figure 5.38 – EMA vs. IGA frequency loci before (left) and after (right) updating.

5.3. Truck door

To follow the challenge of facing industrial applications, an automotive component is chosen in collaboration of a truck maker company.
A sufficiently complex geometry that could have the possibility to be modelled with tri-variate NURBS is chosen to be the truck door in Figure 5.39.
The original CAD file was delivered as an IGES neutral file, exported from the CAD software CATIA.
To be able of converting the IGES file into control points and knot vectors to be read from MATLAB, the original information must be extracted.
The IGES was then imported into Rhinoceros, which is able to be implemented with user-defined scripts. Then a python script is written to read information from a user selected surface or set of surfaces and store the information of knot vectors, control points and weights into a text file. A MATLAB scripts then reads the ASCII text file to import the information and translate them into NURBS data structure, to be analysis-suitable.

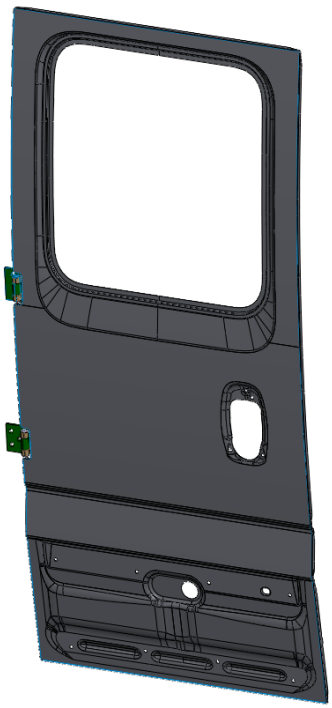


Figure 5.39 – Example of a truck door in its native CAD environment.

The IGES file imported into Rhinoceros has the look in Figure 5.40.

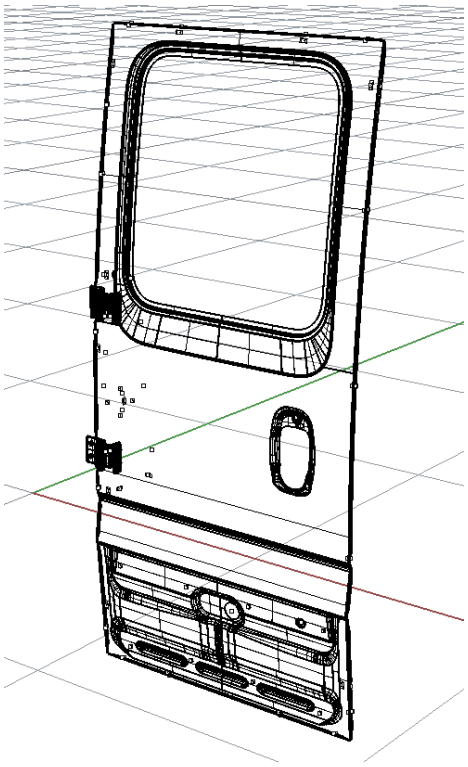


Figure 5.40 – The full door model imported into Rhinoceros.

There are about 200 surfaces related to the external shell of the door (without considering the hinges) and, as per standard in the CAD environments, they all are *trimmed surfaces*.
The concept of trimmed surfaces lay on the flexibility of defining very complex geometries with a low number of information. A simple example of a trimmed surface is shown in Figure 5.41.

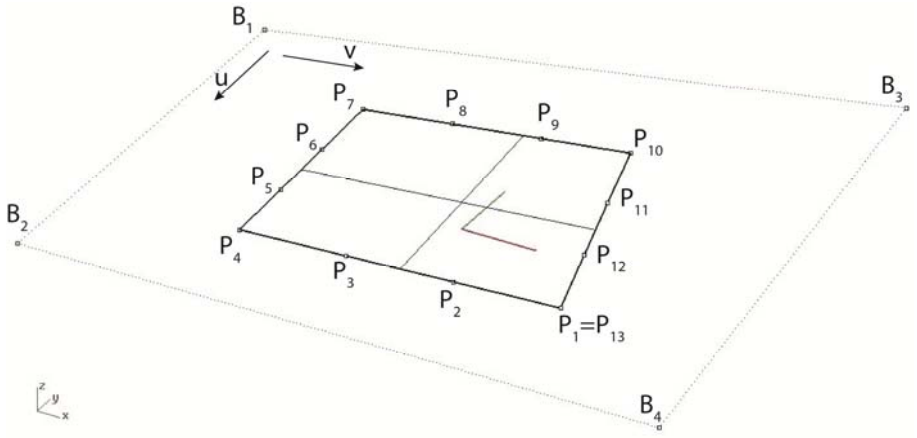


Figure 5.41 – Example of trimmed surface with its trimming curve.

In the example the trapezoidal (but planar) shape is the surface that is rendered on screen, but mathematically the actual B-spline surface is the plane with control points $\mathbf{B}_1$ to $\mathbf{B}_4$, and the borders of the rendered shape (with control points $\mathbf{P}_1$ to $\mathbf{P}_{13}$) is actually a *trimming curve*, which means is a curve that trims the original surface. This trim is virtual, because it is only necessary for the geometrical kernel to inform that what is inside the trimming curve has to be tessellated and then rendered on screen, while everything that is outside does not. The inside-outside choice is also defined by the trimming curve and in particular by its direction of evaluation.
These trimmed surfaces cannot be imported in MATLAB and used, because only the trimless surface would be considered and then in the extrusion process of the surface.
Since a trimmed-to-trimless conversion of NURBS surfaces is still an open question in the field, a first manual but effective conversion can be performed, even if it is a long process.
The general idea is to build Coon's patches of the original surfaces into Rhino building the four boundary curves from the original curves and build the Coon's patches inside the software. These patches must be cleverly built in order to minimise the number of surfaces to export and to think forward to the analysis.
The original model shows a very complex geometry, and this conversion procedure is carried out with two levels of models of increasing complexity and accuracy.
A first simplified model is built considering a planar model, without taking into account the curvatures.
The procedure is to face the model from the front view, as in Figure 5.42, and project the relevant curves and borders on the main *x*-*z* plane.
A good configuration, after many attempts, is the one shown in Figure 5.43, where the door is simplified to 18 planar patches, taking into account the two holes formed by the window and the door handle.
It is also possible to note a correction in the surface definitions because sometimes in the Coon's patch generation Rhinoceros misses some control point position and the parameterisation of the surface returned is not suitable for export.

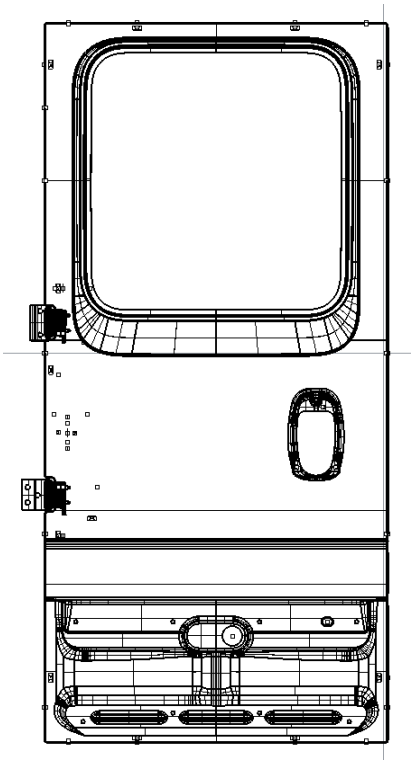


Figure 5.42 – Front view of the full door model.

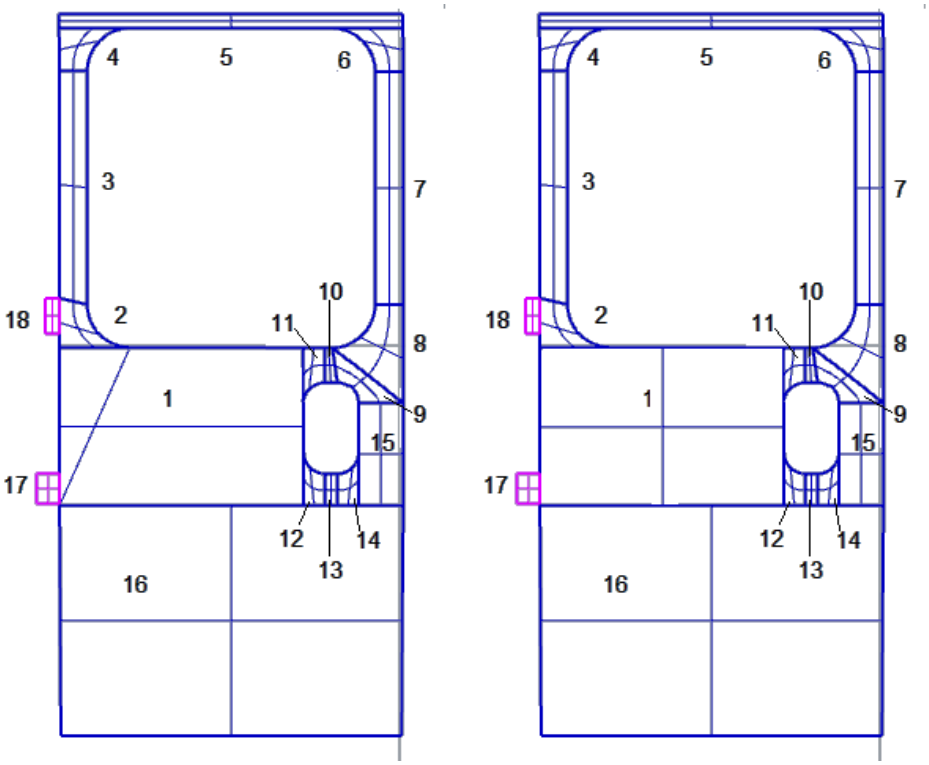


Figure 5.43 – Patch numeration for the planar model.

In this example only patch #1 has this issue, which can be corrected in Rhino directly by using its rebuild command for surfaces. Now, using the python script, the surfaces can be selected and exported.

Note that Rhino arranges the surfaces in the selection group placing the last surface selected as the first, thus the surface must be selected in the reverse order from the 18th to the 1st, for obtaining the surfaces in MATLAB with the correct order.
The imported surfaces in MATLAB must be extruded as done for the other models but two important steps must be carried out before the definition of the coupling surfaces.
These steps are the *patch surface numbering* and the *normal direction check*.
They can be performed both at once by building a routine which plots the tri-variate patch (using the `nrbkntplot` function) and then superimposing dots with different colours (blue for face #2, red for face #4, and black for face #4, see Figure 5.44) for the even surfaces which, by standard representation of tri-variate patch, are the surfaces with an outward normal vector direction.

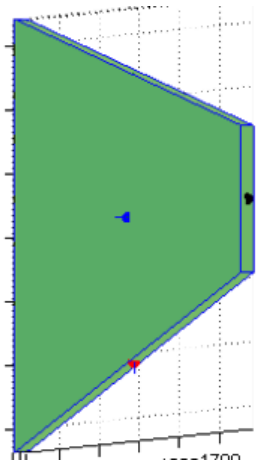


Figure 5.44 – Patch with markers for the face numbering and normal directions.

This check is important for the actual definition of the coupling (face 'x' of patch 'y', with face 'z' of patch 'w'), and because when the coupling is performed, the normal directions of the two surfaces must be discordant (as for Equation 4.2).
The geometric input model and the refined model in MATLAB are shown in Figure 5.45.
The boundary conditions are a clamps imposed on the hinges and a vertical force applied on the hole corresponding to the door's handle. The material is standard steel.
The results are validates as before, comparing them with the results of a commercial software with second-order tetrahedral discretisation (Figure 5.46).
After this confirmation on the results of the 18-patch model, a more accurate model is built considering this time also all the other curvature properties of the door, resulting in the 60-patch model with numbering scheme shown in Figure 5.47.

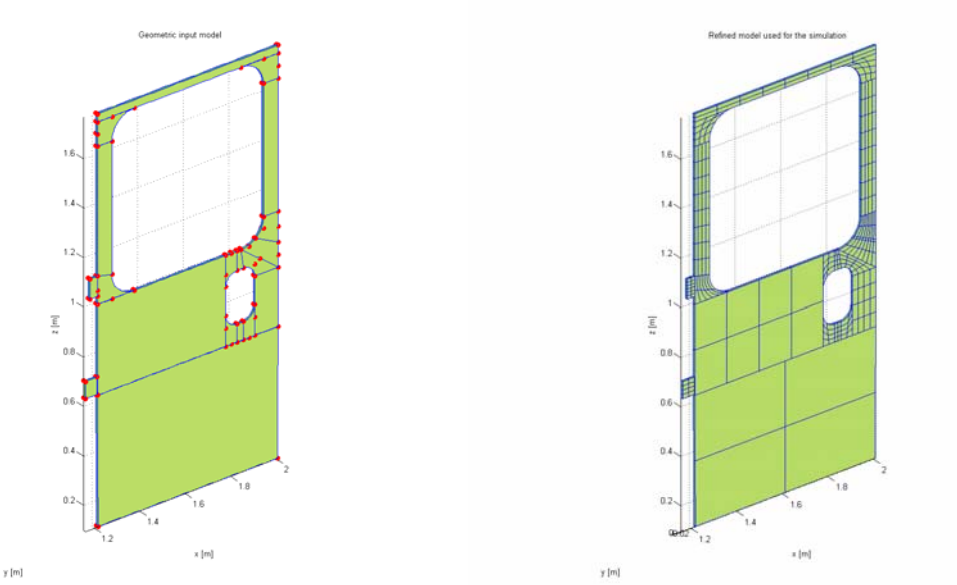


Figure 5.45 – Geometric input model with control points positions (left) and refined model on which the simulation is run (right).

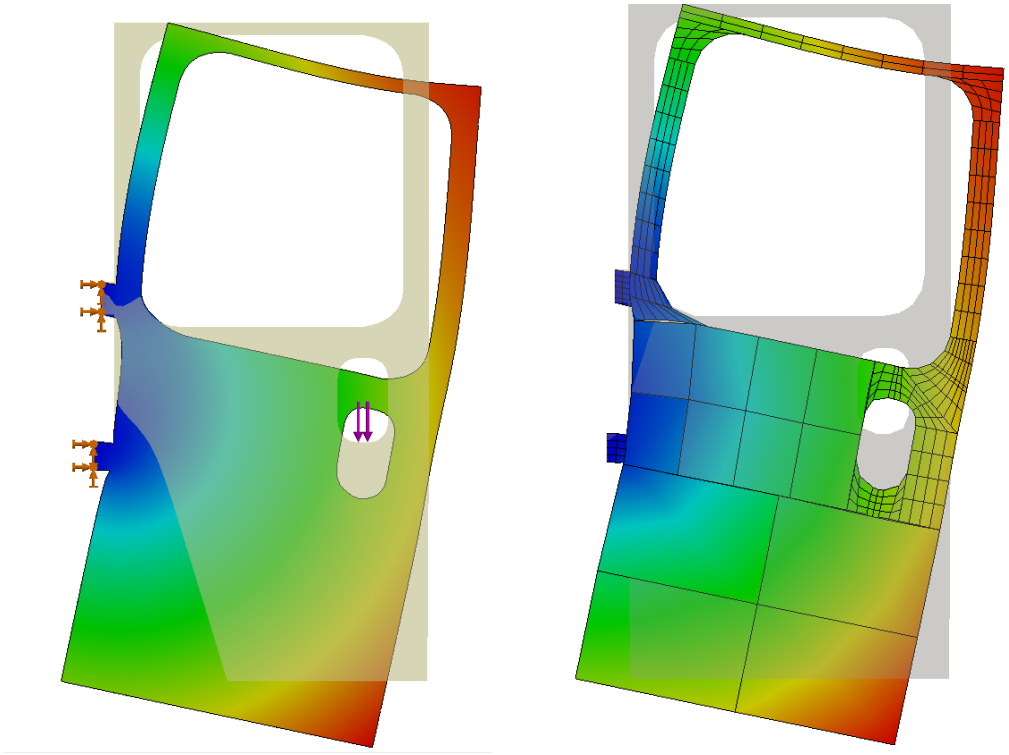


Figure 5.46 – Results of the displacement field of the SolidWorks model (left) compared to the results of the IGA solution (right).

The input model in MATLAB with the patch refinement (very important to obtain element with a better shape and aspect ratio) are shown in Figure 5.48.
The corresponding positions of the non-zero values of the stiffness matrix before and after the non-conforming multi-patch coupling are shown in Figure 5.49, while in Figure 5.50

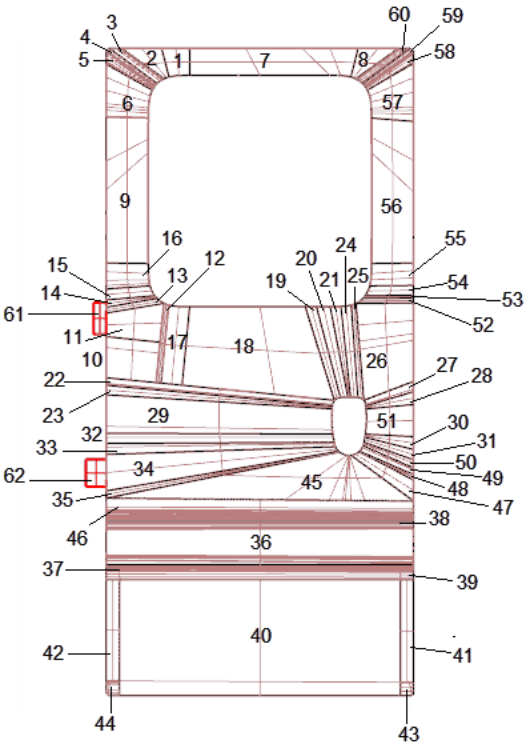


Figure 5.47 – 60-patch NURBS model numbering.

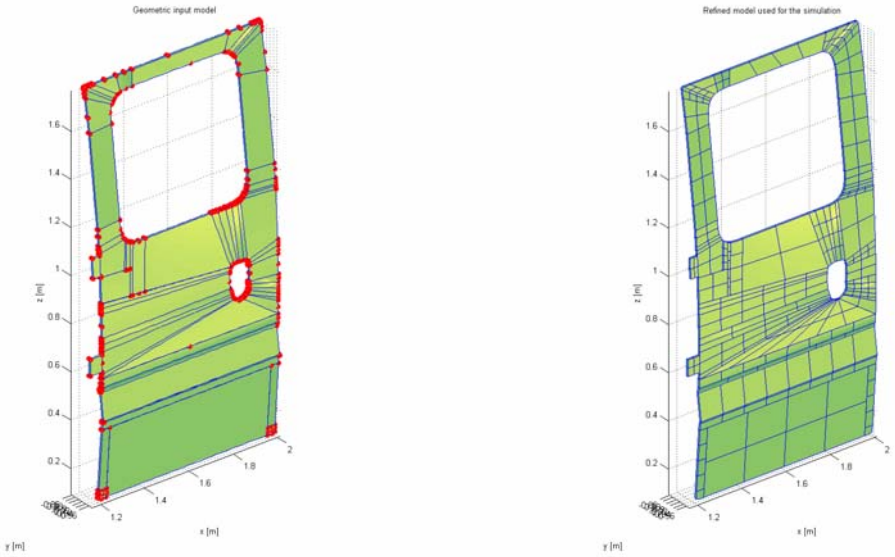


Figure 5.48 – Geometric input model with control points positions (left) and refined model on which the simulation is run (right).

and Figure 5.51 the results in terms of displacement and stress field are shown, in comparison with a commercial FEM solution.

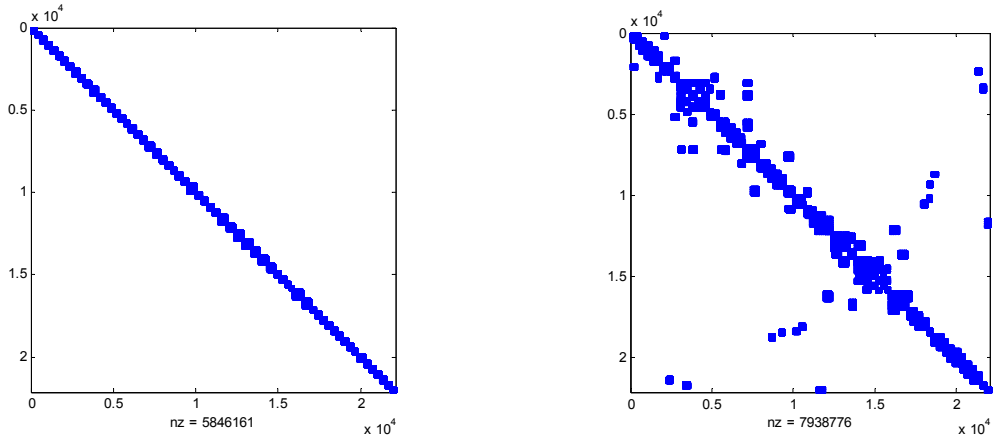


Figure 5.49 – Non-zero values in stiffness matrix before (left) and after the Nitsche's coupling terms computation (right).

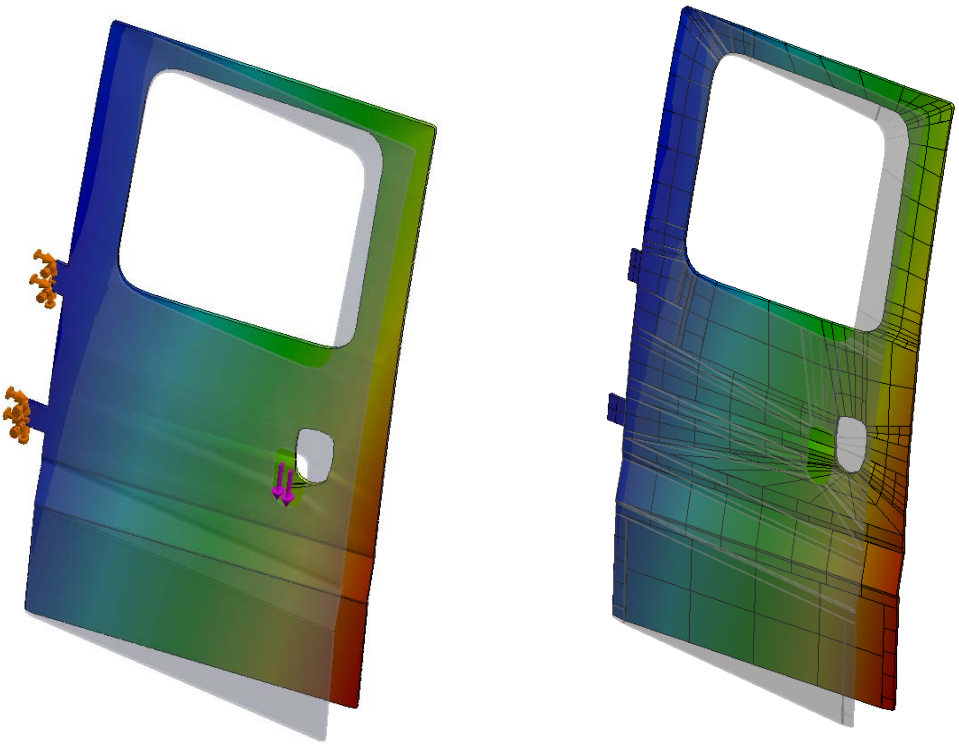


Figure 5.50 – Results of the displacement field of the SolidWorks model (left) compared to the results of the IGA solution (right).

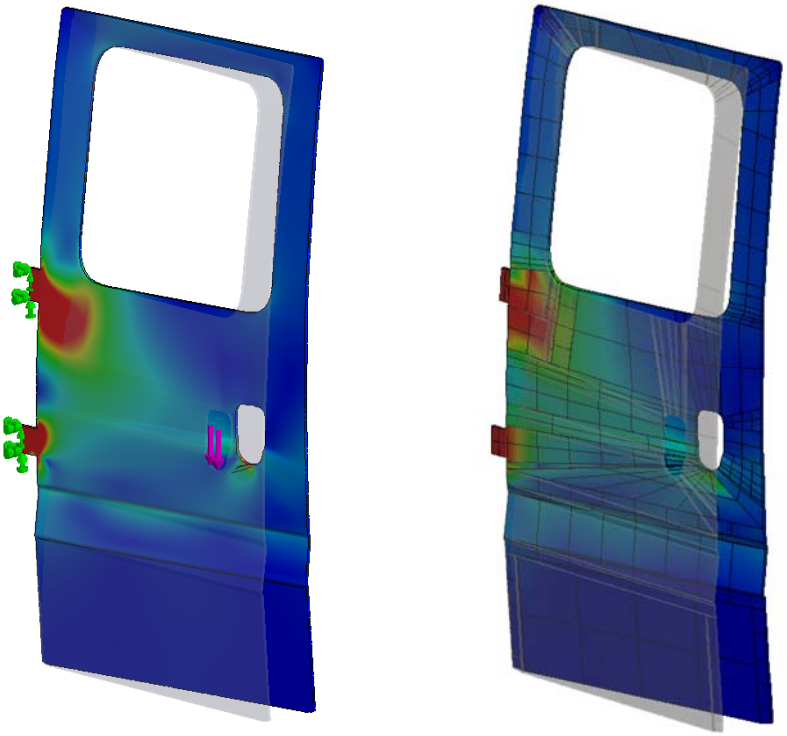


Figure 5.51 – Results of the stress field of the SolidWorks model (left) compared to the results of the IGA solution (right).

6. Additional studies

Besides the tri-variate non-conforming multi-patch coupling investigations, other smaller studies have been carried out in parallel during the Doctoral program.
They are all related to CAD-CAE integration and isogeometric analysis, performing initial investigations on possible future application or deeper study.

6.1. Topological optimisation

Another interesting advantage of the NURBS-based isogeometric analysis, in terms of tighter integration between the outcomes of a finite elements simulation and a geometrical correction on the model, is related to the topological optimisation of an object.
Some engineering application require that the designed components meet both performance specifications (in terms of mechanical resistance and reliability) and shape/mass requirements to lower the amount of material used which, in general, means lower production costs due to material savings and, for some particular applications such as automotive and aerospace, lower weight and then energy savings in vehicle fuel consumption and pollution.
In standard CAD and FEM, the furthest limit of topological optimisation process is related to software tools which integrate design and analysis for specific tasks, taking advantage of *parametric feature-based* CAD systems where some dimensions and geometrical feature are liked to parameters that are considered as *design variables* in the optimisation process.
A direct topological optimisation (in terms of direct modifications of the positions of the geometry such as point placement) is limited to small nodes modifications of the CAE model itself, with hard constraints on the elements quality, possible interpenetrations and not-feasible geometries.
With the NURBS-based computations, the design variables can be directly the control point position and, taking advantage of the lower elements distortion related to high distortion of the control mesh, lower constraints on the design parameters can be imposed. Moreover and most importantly, as a deformed configuration in static analysis or mode-shape is a CAD model itself, also the shape obtained after the final step with optimum values of the design parameters is a CAD model and then it is not necessary to perform surface fitting of the CAE nodes as it is performed now.
Different cases are carried out to investigate the possibilities of topological optimisation, and for all of them a gradient-based optimisation algorithm is used, taking advantage of the MATLAB Optimisation Toolbox and its `fmincon` function.

6.1.1. Planar 2-D beam under transversal load

One of the possible simple cases is to consider a 2-D case (in this way only a few number of control points are driven) of a clamped-free beam with a transversal load at the free-end, as shown in Figure 6.1.
The beam is modelled as a NURBS rectangular surface, single span, with cubic function in horizontal direction and linear function in the vertical.

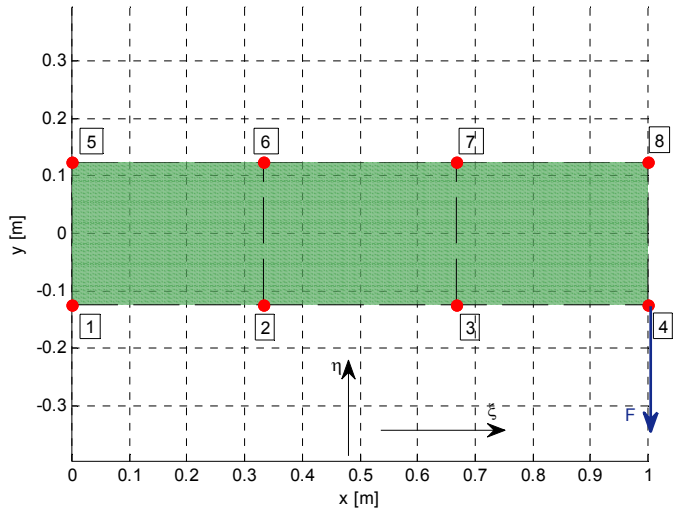


Figure 6.1 – Representation of the beam in physical domain.

The objective function in this case is to minimise the maximum vonMises stress while keeping its area and effective length constant. Another constraint is related to the lower edge of the beam, which is kept straight.
This means that the design variables are the four y-coordinate of control points 5, 6, 7, 8.
The solution of this optimisation process is presented in [11], and it is used as a reference to validate the code. The reference solution is presented in Figure 6.2.

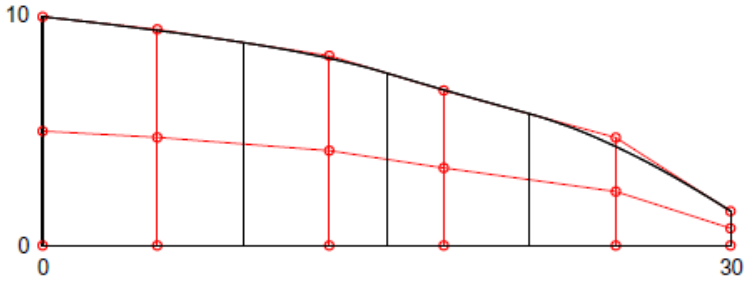


Figure 6.2 – Optimal design computed in [11].

The area constraint, due to the cubic functions, is nonlinear and then a nonlinear constraint must be defined using

$$\int_0^1 \int_0^1 J d\xi d\eta - A_0 = 0 \tag{6.1}$$

where J is the Jacobian

$$J = |\mathbf{J}| = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{vmatrix} \tag{6.2}$$

and A_0 is the reference area.
The final optimal design shape is shown in Figure 6.3, while in Figure 6.4 and Figure 6.5 the values of the design variables and the value of the objective function for each optimisation iteration is shown.

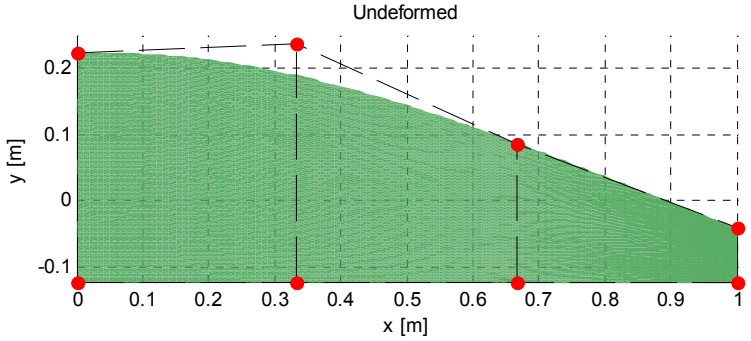


Figure 6.3 – Final design with control points position.

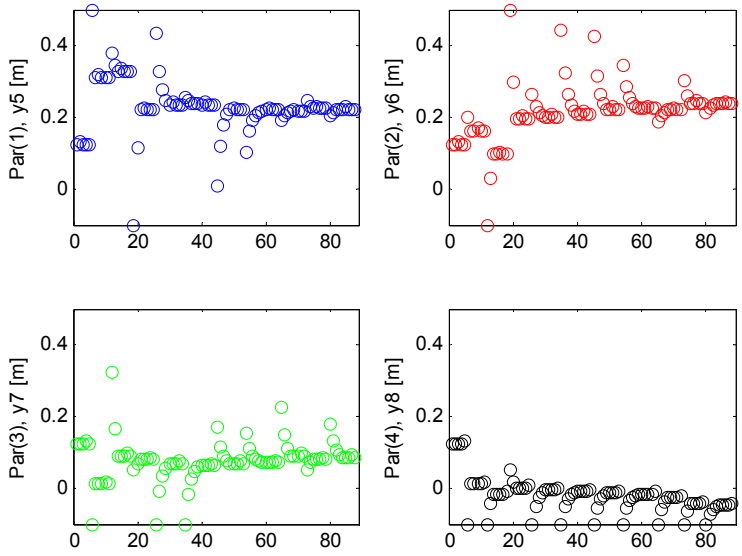


Figure 6.4 – Design variables values at each iteration.

6.1.2. Tri-variate beam under different loading conditions.

An extension of the previous example is the consideration of the third parametric direction to obtain a solid beam with rectangular constant cross section.
The objective function is the same used on the 2-D example, with constant volume and beam length, minimising the maximum equivalent vonMises stress, but in this case the y-direction values of the control points are related to the width of the beam.
The geometry is constrained to be symmetric in width and thickness, thus again for control points are driven, and all the other are consequently repositioned.
The NURBS model is cubic along the length of the beam, and linear in width and thickness.
In Figure 6.6 is represented the geometrical model.

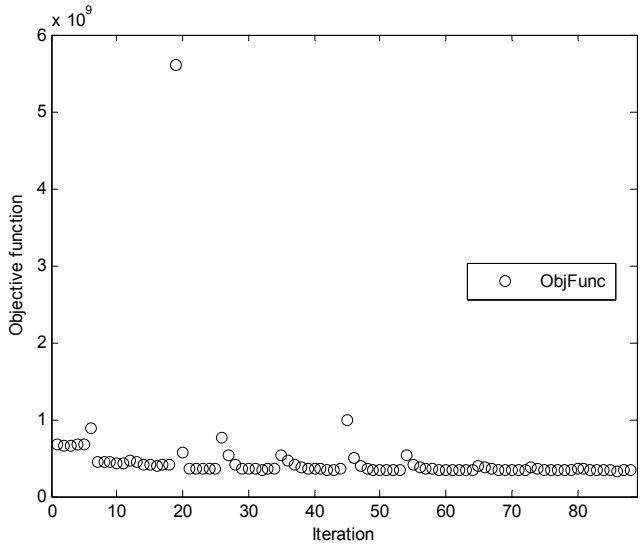


Figure 6.5 – Objective function at each iteration.

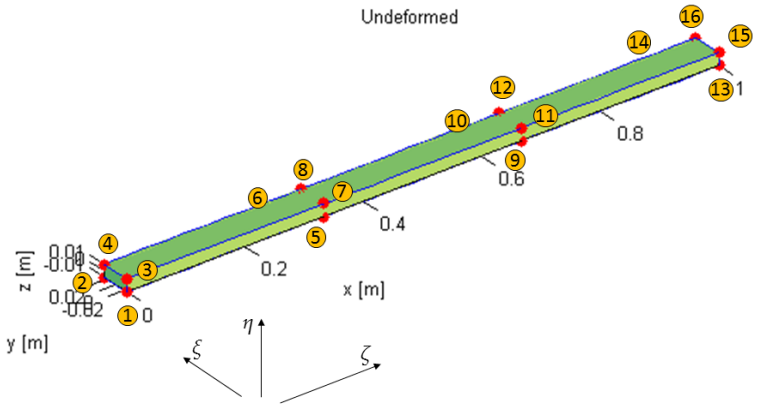


Figure 6.6 – Representation of the beam in physical domain.

The volume constraint is again nonlinear but in this case the computation of the volume is not performed on the tri-variate model but using the Divergence Theorem

$$\int_V \nabla \mathbf{F} dV = \int_S \mathbf{F} \cdot \mathbf{n} dS \tag{6.3}$$

where **F** is a vector field, **n** the unit normal pointing outwards.
The vector field must return a divergence of 1 or a constant value that is then divided by itself outside the integral. In CAD the function used to compute the volume is the coordinates of the points in Cartesian space $\{x_i \quad y_i \quad z_i\}^T$ which returns a divergence value of 3, obtaining

$$\frac{1}{3}\int_V \nabla \mathbf{x} dV = \frac{1}{3}\int_S \mathbf{x} \cdot \mathbf{n} dS \tag{6.4}$$

Using the Gauss quadrature rule for computing the integral and considering the computations of the area of the external surfaces, the previous relation becomes

$$V = \frac{1}{3}\int_0^1\int_0^1 \mathbf{x}\cdot\mathbf{n}\left|\frac{\partial S}{\partial \xi}\times\frac{\partial S}{\partial \eta}\right| d\xi d\eta \tag{6.5}$$

The optimal design is shown in Figure 6.7 along with the values of the design parameters for each iteration (Figure 6.8) and the objective function (Figure 6.9).

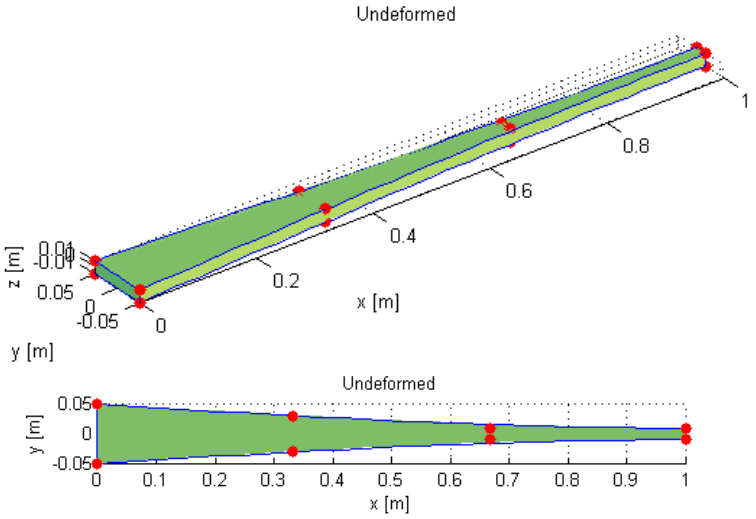


Figure 6.7 – Final design with control points position in isometric view (top) and in top view (bottom).

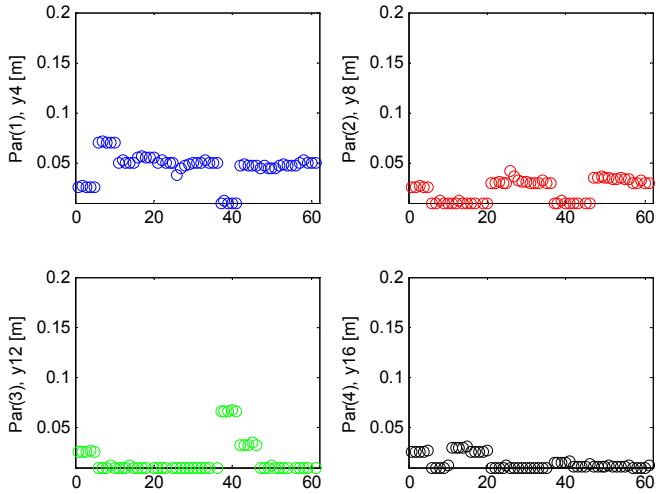


Figure 6.8 – Design variables values at each iteration.

Other two optimisations are performed and are hereby presented as added examples without detailed explanation, considering the trivial outcomes.
The model is a 3-D tri-variate beam with initial squared cross section, cubic function along the length and quadratic functions in the two parametric directions related to the cross-section.

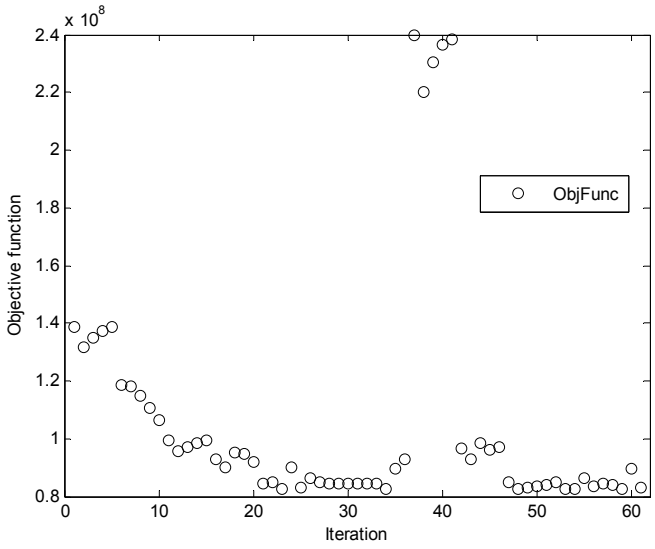


Figure 6.9 – Objective function at each iteration.

Two different loading conditions are applied: vertical transversal load and torsional load at the free-end of the beam.
The objective function is the minimisation of the cross section area and shape in order to reach a top value of stress condition with minimum mass (optimisation of the material contribution).
In both cases all the control points of the cross section are free to move in a range in which they don't intersect with each other.
The results of these two examples are shown in Figure 6.10 and Figure 6.11.

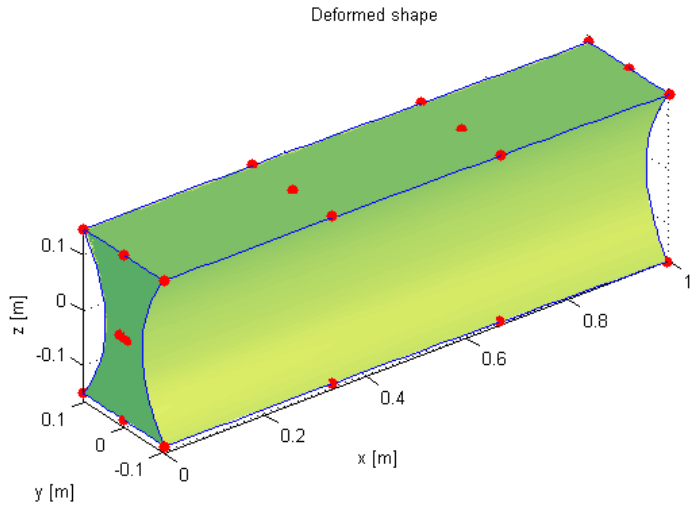


Figure 6.10 – Final design of the beam under transversal vertical load.

The two examples fit the classic shapes that can be seen in real world regarding construction beams with the standard *capital I* shape and the circular shape of rotating shafts.

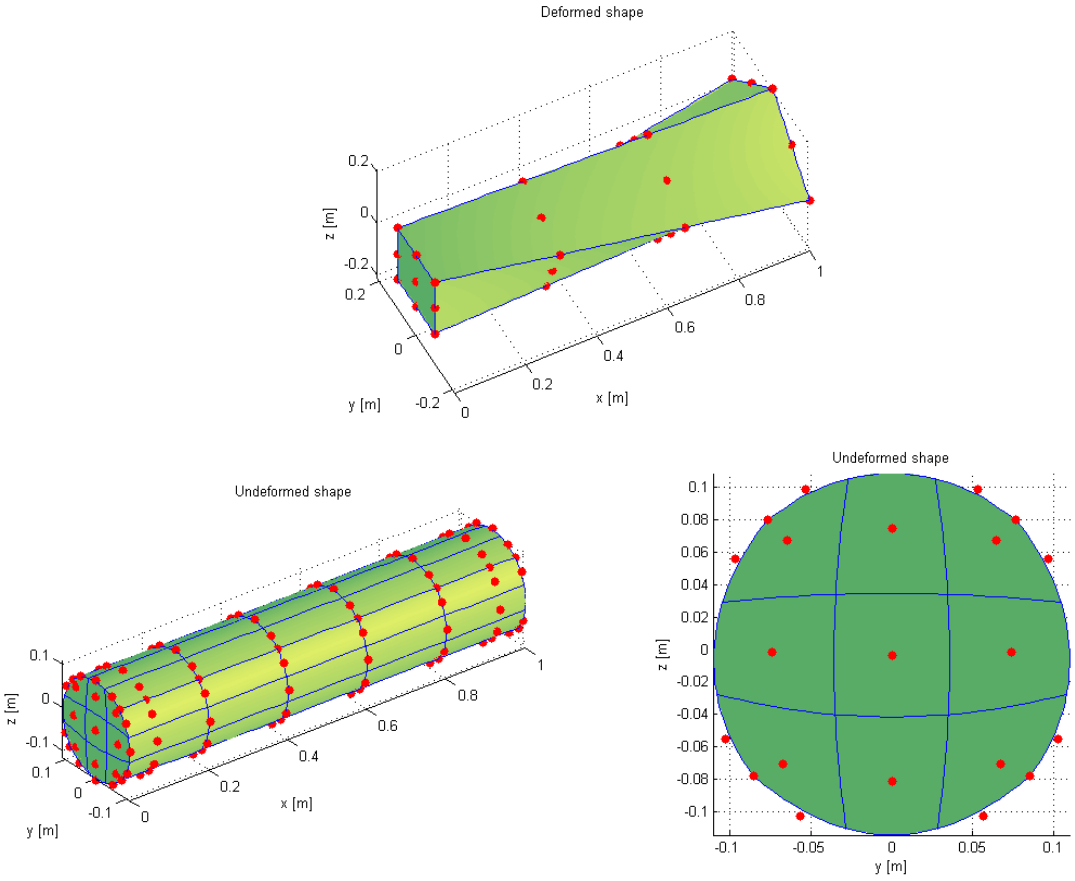


Figure 6.11 – Beam final design under torsional load:
deformed configuration of the initial model (top) and the final design
in isometric (bottom-left) and side view (bottom-right).

6.2. Surface-to-volume conversion

Another additional study carried out to be improved and deeply investigated in the future is the possibility to use the boundary representation from commercial CAD and convert it to tri-variate NURBS solids (supposing that the surfaces involved are trimless).
This work was already carried out by Xu et al. [43, 44], but it was related to B-spline and not for NURBS (no weight was considered in the studies).
The possibility to develop a step forward in the tight integration between CAD and CAE environments make this step very important for everyone who works in this field.

The basic concept behind the conversion from six separate (closed) surfaces represented by bi-variate B-spline or NURBS, to a single tri-variate patch, is the extension of the already well known and used Coon's method from a curves-to surface conversion to a surfaces-to-solid.

The conversion can be performed with these input ingredients:

- 6 B-spline surfaces;
- the opposite boundary surface must have the same parameterisation (same degree, same number of control points, same knot vectors);

- definition of n, m and l as the number of control points in ξ, η and ζ parametric directions;
- the boundary control points $\mathbf{c}_{0,j,k}$, $\mathbf{c}_{l,j,k}$, $\mathbf{c}_{i,0,k}$, $\mathbf{c}_{i,m,k}$, $\mathbf{c}_{i,j,0}$, $\mathbf{c}_{i,j,n}$ (with i=0,...,n , j=0,…,m and k=0,…,l).

The control points inside the volume are then computed using

$$
\begin{aligned}
c_{i,j,k} &= \left(1-\frac{i}{l}\right)c_{0,j,k} + \frac{i}{l}c_{l,j,k} + \left(1-\frac{j}{m}\right)c_{i,0,k} + \frac{j}{m}c_{i,m,k} + \left(1-\frac{k}{n}\right)c_{i,j,0} + \frac{k}{n}c_{i,j,n} + \ldots \\
&- \begin{bmatrix} 1-\frac{i}{l} & \frac{i}{l} \end{bmatrix} \begin{bmatrix} c_{0,0,k} & c_{0,m,k} \\ c_{l,0,k} & c_{l,m,k} \end{bmatrix} \begin{bmatrix} 1-\frac{j}{m} \\ \frac{j}{m} \end{bmatrix} - \begin{bmatrix} 1-\frac{j}{m} & \frac{j}{m} \end{bmatrix} \begin{bmatrix} c_{i,0,0} & c_{i,0,n} \\ c_{i,m,0} & c_{i,m,n} \end{bmatrix} \begin{bmatrix} 1-\frac{k}{n} \\ \frac{k}{n} \end{bmatrix} + \ldots \\
&- \begin{bmatrix} 1-\frac{k}{n} & \frac{k}{n} \end{bmatrix} \begin{bmatrix} c_{0,j,0} & c_{l,j,0} \\ c_{0,j,n} & c_{l,j,n} \end{bmatrix} \begin{bmatrix} 1-\frac{i}{l} \\ \frac{i}{l} \end{bmatrix} + \left(1-\frac{k}{n}\right) \left(\begin{bmatrix} 1-\frac{i}{l} & \frac{i}{l} \end{bmatrix} \begin{bmatrix} c_{0,0,0} & c_{0,m,0} \\ c_{l,0,0} & c_{l,m,0} \end{bmatrix} \begin{bmatrix} 1-\frac{j}{m} \\ \frac{j}{m} \end{bmatrix} \right) + \ldots \\
&+ \frac{k}{n} \left(\begin{bmatrix} 1-\frac{i}{l} & \frac{i}{l} \end{bmatrix} \begin{bmatrix} c_{0,0,n} & c_{0,m,n} \\ c_{l,0,n} & c_{l,m,n} \end{bmatrix} \begin{bmatrix} 1-\frac{j}{m} \\ \frac{j}{m} \end{bmatrix} \right)
\end{aligned}
\tag{6.6}
$$

which, thanks to its matrix nature, can be easily implemented in a computer routine.
The same equation can be considered extending it the weights of the control points

$$
\begin{aligned}
w_{i,j,k} &= \left(1-\frac{i}{l}\right)w_{0,j,k} + \frac{i}{l}w_{l,j,k} + \left(1-\frac{j}{m}\right)w_{i,0,k} + \frac{j}{m}w_{i,m,k} + \left(1-\frac{k}{n}\right)w_{i,j,0} + \frac{k}{n}w_{i,j,n} + \ldots \\
&- \begin{bmatrix} 1-\frac{i}{l} & \frac{i}{l} \end{bmatrix} \begin{bmatrix} w_{0,0,k} & w_{0,m,k} \\ w_{l,0,k} & w_{l,m,k} \end{bmatrix} \begin{bmatrix} 1-\frac{j}{m} \\ \frac{j}{m} \end{bmatrix} - \begin{bmatrix} 1-\frac{j}{m} & \frac{j}{m} \end{bmatrix} \begin{bmatrix} w_{i,0,0} & w_{i,0,n} \\ w_{i,m,0} & w_{i,m,n} \end{bmatrix} \begin{bmatrix} 1-\frac{k}{n} \\ \frac{k}{n} \end{bmatrix} + \ldots \\
&- \begin{bmatrix} 1-\frac{k}{n} & \frac{k}{n} \end{bmatrix} \begin{bmatrix} w_{0,j,0} & w_{l,j,0} \\ w_{0,j,n} & w_{l,j,n} \end{bmatrix} \begin{bmatrix} 1-\frac{i}{l} \\ \frac{i}{l} \end{bmatrix} + \left(1-\frac{k}{n}\right) \left(\begin{bmatrix} 1-\frac{i}{l} & \frac{i}{l} \end{bmatrix} \begin{bmatrix} w_{0,0,0} & w_{0,m,0} \\ w_{l,0,0} & w_{l,m,0} \end{bmatrix} \begin{bmatrix} 1-\frac{j}{m} \\ \frac{j}{m} \end{bmatrix} \right) + \ldots \\
&+ \frac{k}{n} \left(\begin{bmatrix} 1-\frac{i}{l} & \frac{i}{l} \end{bmatrix} \begin{bmatrix} w_{0,0,n} & w_{0,m,n} \\ w_{l,0,n} & w_{l,m,n} \end{bmatrix} \begin{bmatrix} 1-\frac{j}{m} \\ \frac{j}{m} \end{bmatrix} \right)
\end{aligned}
\tag{6.7}
$$

The formulas are applied to simple example to check the reliability of the method.
First, a known example is chosen, considering a straight brick composed by cubic functions in each directions of the surfaces. The result is known because it should give the same values of a tri-cubic brick. In Figure 6.12 the original Rhino model is used as input geometry.

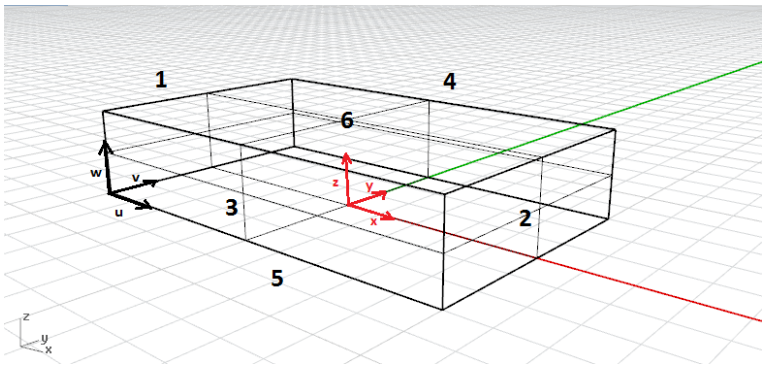


Figure 6.12 – Rhino brick model with surface numbering.

As input request, the six boundary surfaces need to fit those requirements:

- 1: the surfaces with normal n so that $\mathbf{n}\cdot\mathbf{u} < 0$;
- 2: the surfaces with normal n so that $\mathbf{n}\cdot\mathbf{u} > 0$;
- 3: the surfaces with normal n so that $\mathbf{n}\cdot\mathbf{v} < 0$;
- 4: the surfaces with normal n so that $\mathbf{n}\cdot\mathbf{v} > 0$;
- 5: the surfaces with normal n so that $\mathbf{n}\cdot\mathbf{w} < 0$;
- 6: the surfaces with normal n so that $\mathbf{n}\cdot\mathbf{w} > 0$.

The final result with the inner control points is presented in Figure 6.13

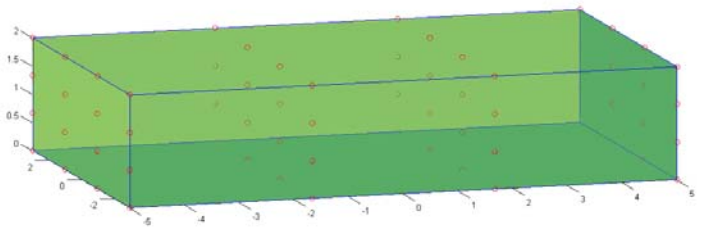


Figure 6.13 – Resulting tri-variate brick with inner control points.

and the position of the resulting inner control points is verified to be the same of a direct tri-variate brick.
The brick is then stretches in all its boundary surfaces to obtain a model with complicated geometry. This stretched model is shown in Figure 6.14, while the result with the inner control points is shown in Figure 6.15.

In order to check the suitability of the resulting model to be used in isogeometric code, a static simulation clamping one face and applying a transversal loading to the opposite face is used to check the suitability of the result, with the solution in Figure 6.16.
The weights for the inner control points are verified considering the model of half a hollow cylinder, for known results. The Rhinoceros input model is shown in Figure 6.17.

The result of the tri-variate patch is shown in Figure 6.18, with the inner control points.

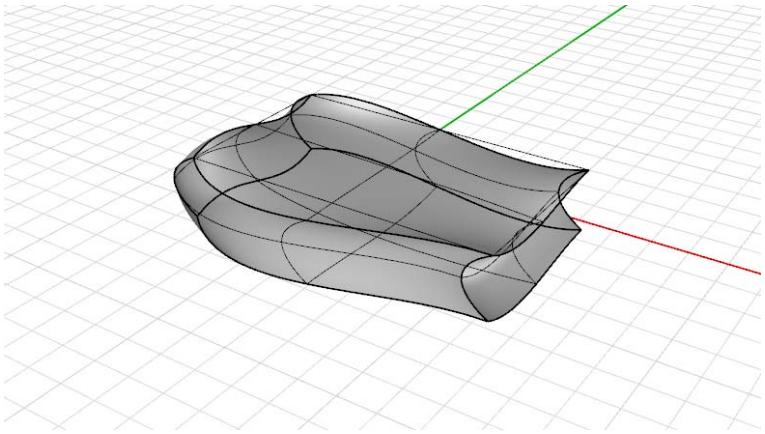


Figure 6.14 – Brick with stretched faces.

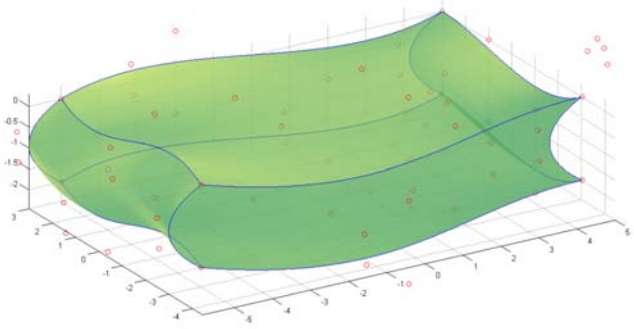


Figure 6.15 – Solution of the inner control points.

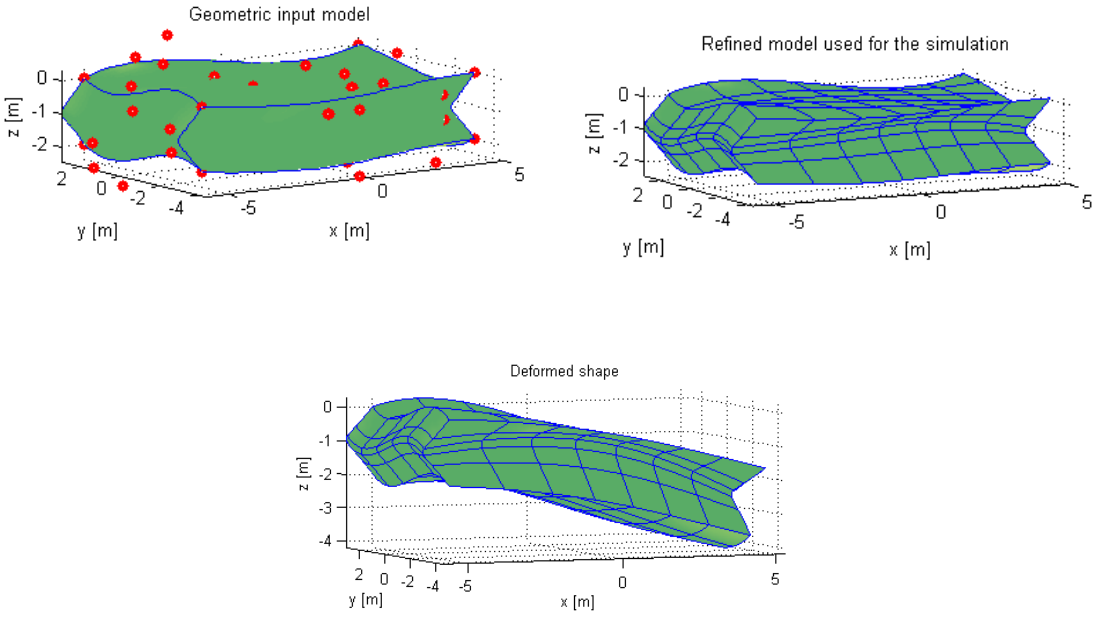


Figure 6.16 – Input geometry (top-left), refined model (top-right) and solution (bottom) of the stretched brick model.

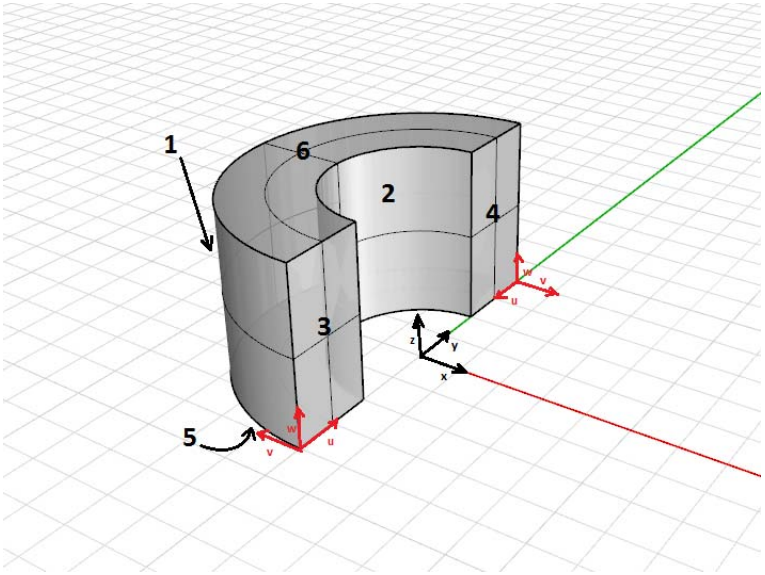


Figure 6.17 – Boundary surfaces for the half hollow cylinder model with surface numbering.

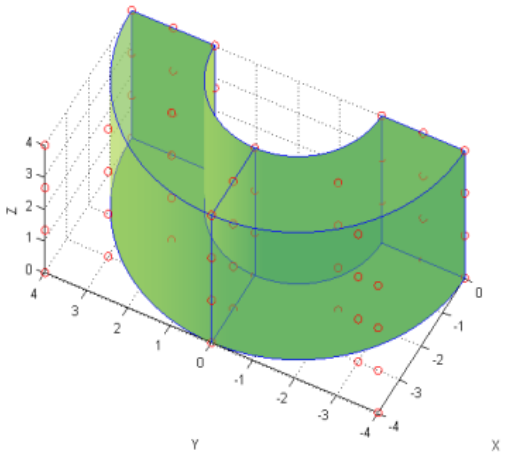


Figure 6.18 – Result with inner control points.

This model is also verified for suitability running a static analysis with clamp constraint on face #3 and a transversal vertical applied load on face #4.
The IGA solution is shown in Figure 6.19.
In these examples the inner control points were placed in the optimal position due to their simple nature, but for complex geometries the solution could lead to not optimal elements, featuring distorted or badly shaped elements.
An optimisation process to minimise the energy function can be performed to finally obtain geometry with optimal shape of the inner elements.

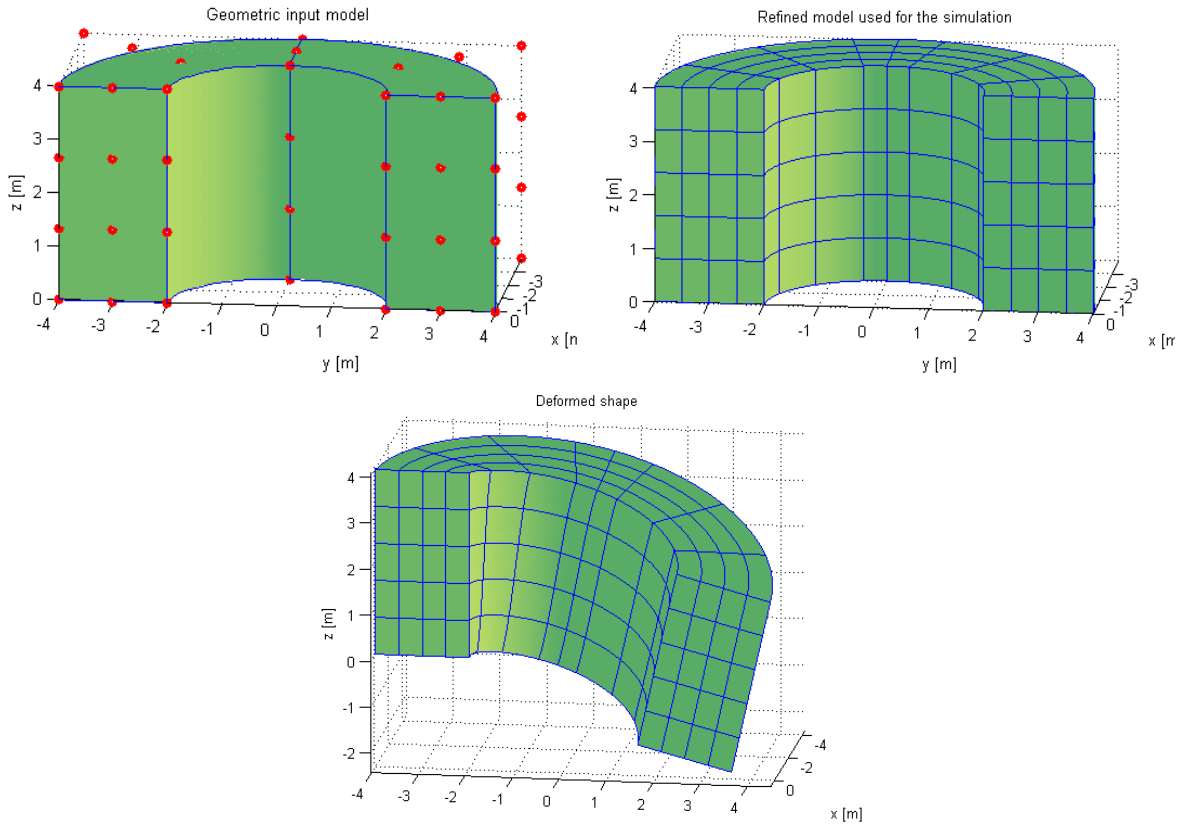


Figure 6.19 – Input geometry (top-left), refined model (top-right) and solution (bottom) of the half hollow cylinder model.

Conclusions and possible further investigations

In this dissertation some CAD-CAE integration methodologies are evaluated and developed, focusing on the new isogeometric analysis computational method, which under some future improvement could be the possible ultimate solution to have design and analysis under the same framework.

At the date of January 2012, starting moment of the author's three-year Doctoral program which led to this final dissertation, neither the author nor other members of the Virtual Product Development Team (VPDT), the research team in which the author is working, knew about isogeometric analysis and its potential. Furthermore, apart from LUPOS, which is beam-based, no standard finite element code for solid elements was previously developed by the team.

Nevertheless, the isogeometric analysis features were deeply studied and analysed, bringing to the step-by-step development that is the structure of this dissertation, starting from how CAD models work, what B-spline and NURBS are and to build a simple isogeometric code in MATLAB capable of single-patch static and modal analysis in linear elasticity.

The team believed in this challenge and the possibility to give original and new contributions to the method, allowing it to get closer to its final goal of CAD-CAE integration.

Nitsche's method for coupling non-matching and non-conforming meshes and its application to multi-patch isogeometric analysis gives a wider range of possibility to face problem with high complexity without forcing the modelling of the single patches to feature matching mesh with the adjacent patches, leading to applications in both experimental and industrial fields.

The examples of tri-variate multi-patch presented and described in Section 5, proved the reliability of the non-conforming coupling technique, thanks to the development of the mathematical definition in Section 4, obtained during the author's collaboration period at the Institute of Mechanics and Advanced Materials of Cardiff University (United Kingdom).
In particular the simple examples in Section 4 and the connecting-rod in Section 5.1, were included in the international journal paper [45] and the international conference proceeding [46], while the coincident eigenvalues test-rig in Section 5.3 were published with the IMAC proceeding [47].

The truck-door example proved the reliability of the method for increasing geometrical and patch interface complexity, and allowing a closer possibility of implementation of the method for complex industrial applications still proving the advantages of isogeometric analysis with respect to standard finite elements, such as good accuracy of the results with fewer elements and degrees of freedom, very sparse and narrow-banded matrices, inter-element higher continuity and the feature of representing the exact geometry even in the coarsest mesh configuration.

The additional studies in Section 6 had the aim of investigating different other possibilities and advantages of the isogeometric analysis in order to face them in deeper analysis and

development for future work, in particular the surface-to-volume conversion for helping a possible automatic process of obtaining the analysis-suitable models from commercial CAD software reducing at a minimum the operator contribution for steps that can be automatized.

Possible future developments are:

- for Nitsche's method, the implementation of an automatic estimation of the stabilisation parameter;
- the application to other geometries where rounds, curvatures and other features typically cause an increase in time consumed for defeaturing and generation of the mathematical model;
- further development of the surface-to-volume conversion of the generation of the patches of a multi-patch geometry like the truck-door;
- the development of a general and effective trimmed-to-trimless CAD surfaces for allowing a direct use of the original parameterisation.

Author's publications list

1. Bonisoli E., Marcuccio G., Brino M., "*Parametric CAD/CAE tool in modal analysis: numerical results and experimental outcomes*", TMT 2011, 15th International Research/Expert Conference "Trends in the Development of Machinery and Associated Technology", 2011, September 12-18, Prague, Czech Republic, pp. 609-612, pp. 21-28.
2. Bonisoli E., Brino M. "*Modelli virtuali 3D per le nocciole delle Langhe*". Il Progettista Industriale, ISSN 0392-4823, vol. XXXI n. 8, 2011, pp. 35-38.
3. Tornincasa S., Bonisoli E., Brino M., "*3D models for Langhe hazelnuts*", International Journal Of Mechanics And Control, ISSN 1590-8844, vol. 12(2), 2011, pp. 21-28.
4. Bonisoli E., Marcuccio G., Brino M., "*Modelli CAD/CAE parametrici nella progettazione e simulazione di componenti motore*", Il Progettista Industriale, ISSN 0392-4823, vol. XXXII n. 2, 2012, pp. 55-57.
5. Tornincasa S., Nosenzo V., Bonisoli E., Brino M., "*Open questions on Product Lifecycle Management (PLM) with CAD / CAE integration*", Proceedings of the ADM Virtual Concept workshop, 2012, September 19-21, Anacapri, Italy, pp1-11 Tornincasa S., Nosenzo V., Bonisoli E., Brino M., "*Open questions on Product Lifecycle Management (PLM) with CAD / CAE integration*", International Journal on Interactive Design and Manufacturing, ISSN 1955-2513, 2013, pp1-17
6. Nguyen V.P., Kerfriden P., Brino M., Bordas S.P.A., Bonisoli E., "*Nitsche's method for two and three dimensional NURBS patch coupling*", Computational Mechanics, vol. 56(3), ISSN 0178-7675, 2013, pp1163-1182.
7. Tornincasa S., Bonisoli E., Kerfriden P., Brino M., "*Investigation of crossing and veering phenomena in an isogeometric analysis framework*", Proceedings of IMAC XXXII, 2014, February 3-6, Orlando, Florida, USA, pp1-15.
8. Tornincasa S., Bonisoli E., Brino M., *"NURBS patch coupling with Nitsche's method for isogeometric analysis"*, IDMME Conference, 2014, June 18-20, Toulouse, France.
9. Bonisoli E., Brino M., *"Robust updating of operational boundary conditions for a grinding machine"*, Proceedings of IMAC XXXIII, 2015, February 2-5, Orlando, Florida, USA

Ringraziamenti

Questo lavoro di tesi è il risultato del triennio di dottorato in Sistemi di Produzione e Design Industriale presso il Politecnico di Torino.

Un ruolo chiave, non solo in questo periodo, l'ha sicuramente avuto Elvio Bonisoli (titolo di studio volutamente omesso, renderebbe troppo impersonale la frase...) che oltre ad essere il mio relatore, tutore, collega e capo, è sempre più un amico, mi conosce sempre meglio e insieme formiamo una squadra tosta. Il suo supporto sia sul lato tecnico scientifico sia sul lato motivazionale ha portato a questo punto. Qualsiasi grazie non è abbastanza, ma se non ci fosse stato lui, questo ringraziamento non avrei potuto scriverlo.
Alcuni sviluppi contenuti in questo lavoro sono stati oggetto di tesi di Laurea, in particolare ringrazio Vincenzo Commisso, Antonio Russo e Alessandro Matera, i quali hanno dato un grosso contributo sia operativo che di idee.
Un partner molto importante per conoscere concretamente le problematiche aziendali è stato Vladi Nosenzo, che oltre ad aver contribuito nello sviluppo del caso industriale crede fortemente nel metodo e nel progetto, vedendone chiaramente tutte le potenzialità.
Insieme a loro i miei ex colleghi dottorandi Francesco e Gabriele, e gli attuali Nicolò e Matteo, mi hanno accompagnato in questo percorso, senza dimenticare Denis, Alessandra e Libero, con i quali ho percorso un pezzo di strada. E a cappello di tutti il Prof. Stefano Tornincasa, con cui questo grosso capitolo della mia vita partì, nel Febbraio 2008.
Per tutte le ore passate assieme, in condivisione di ufficio, ringrazio anche i colleghi Gianluca, Paola, Federica, Giovanni, Giulio e Gianfranco, oltre a Manuela e gli acustici Louena, Lorenzo, Giusi ed Elena con cui ho condiviso corsi e non solo.

Durante questo periodo sono stato ospitato dal gruppo di ricerca dell'Institute of Mechanics and Advanced Materials della Cardiff University, School of Engineering, di cui il Prof. Stéphane Bordas è il leader nonché ispiratore e grazie al quale mi è stato possibile far parte del suo gruppo. Un periodo molto importante di rapida crescita sia tecnica che personale, grazie al lavoro svolto insieme a Phu, Pierre, Andrés, Chang-Kye, Daniel, Iulia, Olivier, Danas, Claire, Ahmad, Chi, Haojie e Xuan.

Un pensiero particolare anche al Prof. Alessandro Reali dell'Università di Pavia, il quale da pioniere della materia mi ha dato un grosso contributo soprattutto nelle fasi iniziali, trovando sempre il tempo per aiutarmi e suggerirmi.

Il mondo al di fuori del Politecnico non è stato meno importante e le amicizie sono la miglior fonte di energia e supporto. I miei sempre più fratelli acquisiti Andrea, Riccardo e Lorenzo hanno seguito tutto il mio percorso universitario e di superiori fino a qui, mentre Dario anche elementari e medie... Con loro Erina, Alessandra, Giacomo, Giulia, Roberta e Diego sono sempre stati presenti.

Dal lato più sociale e spirituale ringrazio chi, in particolare durante la stesura della tesi, mi ha incitato, sostenuto e dato la carica come un maratoneta agli ultimi metri, Fausto, Maria e Carlo, e assieme a loro Alberto, Cecilia, Eleonora, Valentina, Sergio, Oreste, Simone, Daniele, Marco e Francesca, mentre dal lato ludico Maurizio, Paolo e Francesco.

Tutti questi nomi sento, in un modo o nell'altro, aver dato un contibuto a questo traguardo e perciò desidero ringraziarli caldamente.

References

[1] Hughes T.J.R., Cottrell J.A., Bazilevs Y., "Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement". *Computer Methods in Applied Mechanics and Engineering*, 194(39-41), pp. 4135-4195, 2005.

[2] Cottrell J.A., Hughes T.J.R., Bazilevs Y., "Isogeometric analysis: toward integration of CAD and FEA", Wiley, 2009.

[3] Piegl L.A., Tiller W., "The NURBS book", *Springer*, 1996.

[4] Rogers D.F., "An Introduction to NURBS with Historical Perspective", *Academic Press*, 2001.

[5] Cottrell J.A., Hughes T.J.R., Reali A., "Studies of refinement and continuity in isogeometric structural analysis". *Computer Methods in Applied Mechanics and Engineering*, 196(41-44), pp. 4160-4183, 2007.

[6] Temizer I., Wriggers P., Hughes T.J.R., "Contact treatment in isogeometric analysis with NURBS". *Computer Methods in Applied Mechanics and Engineering*, 200(9-12), pp. 1100-1112, 2011.

[7] Jia L., "Isogeometric contact analysis: Geometric basis and formulation for frictionless contact". *Computer Methods in Applied Mechanics and Engineering*, 200(5-8), pp. 726-741, 2011.

[8] Temizer I., Wriggers P., Hughes T.J.R., "Three-dimensional mortar-based frictional contact treatment in isogeometric analysis with NURBS". *Computer Methods in Applied Mechanics and Engineering*, 209-212, pp. 115-128, 2012

[9] De Lorenzis L., Temizer I., Wriggers P., Zavarise G., "A large deformation frictional contact formulation using NURBS-based isogeometric analysis". *International Journal for Numerical Methods in Engineering*, 87(13), pp. 1278-1300, 2011.

[10] Matzen M.E., Cichosz T., Bischoff M., "A point to segment contact formulation for isogeometric, NURBS-based finite elements". *Computer Methods in Applied Mechanics and Engineering*, 255, pp. 27-39, 2013

[11] Wall W.A., Frenzel M.A., Cyron C., "Isogeometric structural shape optimization". *Computer Methods in Applied Mechanics and Engineering*, 197(33-40), pp. 2976-2988, 2008.

[12] Manh N.D., Evgrafov A., Gersborg A.R., Gravesen J., "Isogeometric shape optimization of vibrating membranes". *Computer Methods in Applied Mechanics and Engineering*, 200(13-16), pp. 1343-1353, 2011.

[13] Qian X., Sigmund O., "Isogeometric shape optimization of photonic crystals via Coons patches". *Computer Methods in Applied Mechanics and Engineering*, 200(25-28), pp. 2237-2255, 2011

[14] Qian X., "Full analytical sensitivities in NURBS based isogeometric shape optimization". *Computer Methods in Applied Mechanics and Engineering*, 199(29-32), pp. 2059-2071, 2010.

[15] Cottrell J.A., Reali A., Bazilevs Y, Hughes T.J.R., "Isogeometric analysis of structural vibrations", *Computer Methods in Applied Mechanics and Engineering*, 195(41-43), pp. 5257-5296, 2006.

[16] Hughes T.J.R., Reali A., Sangalli G., "Duality and unified analysis of discrete approximations in structural dynamics and wave propagation: Comparison of p-method finite elements with k-methods NURBS". *Computer Methods in Applied Mechanics and Engineering*, 197(49-50), pp. 4104-4124, 2008.

[17] Wang D., Liu W., Zhang H., "Novel high order mass matrices for isogeometric structural vibrations analysis", *Computer Methods in Applied Mechanics and Engineering*, 260, pp. 92-108, 2013.
[18] Evans J.A., Bazilevs Y., Babuska I., Hughes T.J.R., "n-Widths, sup-infs, and optimality ratios for the k-version of the isogeometric finite element method", *Computer Methods in Applied Mechanics and Engineering*, 198(21-26), pp. 1726-1741, 2009.
[19] Bazilevs Y., Calo V.M., Cottrell J.A., Evans J.A., Hughes T.J.R., Lipton S., Scott M.A., Sederberg T.W., "Isogeometric analysis using T-splines". *Computer Methods in Applied Mechanics and Engineering*, 199(5-8), pp. 229-263, 2010.
[20] Scott M.A., Li X., Sederberg T.W., Hughes T.J.R., "Local refinement of analysis-suitable T-splines". *Computer Methods in Applied Mechanics and Engineering*, 213-216, pp. 206-222, 2012.
[21] Takacs T., Juttler B., "Existence of stiffness matrix integrals for singularly parameterized domains in isogeometric analysis", *Computer Methods in Applied Mechanics and Engineering*, 200(49-52), pp. 3568-3582, 2011.
[22] Xu G., Mourrain B., Duvigneau R., Galligo A., "Parameterisation of computational domain in isogeometric analysis: Methods and comparison", *Computer Methods in Applied Mechanics and Engineering*, 200(23-24), pp. 2021-2031, 2011.
[23] Cohen E., Martin T., Kirby R.M., Lyche T., Riesenfeld R.F., "Analysis-aware modeling: Understanding quality considerations in modelling for isogeometric analysis", *Computer Methods in Applied Mechanics and Engineering*, 199(5-8), pp. 334-356, 2010.
[24] J. Nitsche. "Uber ein variationsprinzip zur losung von dirichlet-problemen bei verwendung von teilraumen, die keinen randbedingungen unterworfen sind", *Abhandlungen aus dem Mathematischen Seminar der Universitat Hamburg*, 36, pp. 9-15, 1971.
[25] Leissa W., "On a curve veering aberration", *Journal of Applied Mathematics and Physics (ZAMP)*, 25, pp. 99-111, 1974.
[26] Chen X., Kareem A., Matsumoto M "Multimode coupled flutter and buffeting analysis of long span bridges". *Journal of Wind Engineering*, 89, pp. 649-664, 2001.
[27] Chan Y, Inman D.J., "Management of the variability of vibration response levels in mistuned bladed discs using robust design concepts. Part 1 Parameter design". *Mechanical Systems Signal Processing*, 24, pp. 2777-2791, 2010.
[28] Bae J.S., Inman D.J., Lee I, "Effects of structural nonlinearity on subsonic aeroelastic characteristics of an aircraft wing with control surface". *Journal of Fluids and Structures*, 19, pp. 747-763, 2004.
[29] Khodaparast H., Mottershead J.E., Badcock K., "Propagation of structural uncertainty to lineat aeroelastic stability". *Journal of fluids and structures*, 88, pp. 223-236, 2010.
[30] Tang D, Dowell E., "Aeroelastic Response of Aircraft with freeplay structural nonlinearity". *Proceedings of 2nd Aircraft Structural Design Conference*, London (UK), 1, 2010.
[31] De Falco C., Reali A., Vazquez R., "GeoPDEs: a research tool for Isogeometric Analysis of PDEs", *Advances in Engineering Software*, 42(12), pp. 1020-1034, 2011.
[32] https://sourceforge.net/projects/cmcodes/
[33] http://www.mathworks.com/matlabcentral/fileexchange/26390-nurbs-toolbox-by-d-m-spink
[34] Zeid I., "Mastering CAD/CAM". McGraw-Hill, 2004.

[35] Borden M.J., Scott M.A., Evans J.A., Hughes T.J.R., "Isogeometric finite element data structure based on Bézier extraction of NURBS", *International Journal for Numerical Methods in Engineering*, 87(15), pp. 15-47, 2011.
[36] Fritz A., Hüeber S., Wohlmuth B.I., "A comparison of mortar and Nitsche techniques for linear elasticity". *CALCOLO*, 41(3), pp. 115-137, 2004.
[37] Bazilevs Y., Hughes T.J.R., "Weak imposition of Dirichlet boundary conditions in fluid mechanics". *Computers & Fluids*, 36(1), pp. 12-16, 2007.
[38] Griebel M., Schweitzer M.A., "A Particle-Partition of Unity Method – Part V: Boundary Conditions". *Geometric Analysis and Nonlinear Partial Differential Equations*, pp. 519-542, Springer Berlin, 2002.
[39] Embar A., Dolbow J. Harari I., "Imposing Dirichlet boundary conditions with Nitsche's method and spline-based finite elements". *International Journal for Numerical Methods in Engineering*, 83(7), pp. 877-898, 2010.
[40] Coons S.A., "Surface patches and B-spline curves". *Computer Aided Geometric Design*, Academic Press, pp. 1-16, 1974.
[41] Bonisoli E., Delprete C., Esposito M., Mottershead J.E., "Structural dynamics with coincident eigenvalues: modelling and testing". *Proceedings of the 29th IMAC*, Detroit, Michigan, USA, 3(6), pp. 325-337, 2011.
[42] Brino M., "Parametric CAD and dynamic CAE analysis on test structures". *MSc thesis*, I Facoltà di Ingegneria, Politecnico di Torino, 2011.
[43] Xu G., Mourrain B., Duvigneau R, Galligo A., "Optimal analysis aware parameterization of computational domain in 3D isogeometric analysis". *Computer-Aided Design*, 45(4), pp. 812-821, 2013.
[44] Xu G., Mourrain B., Duvigneau R, Galligo A., "Analysis-suitable volume parameterization of multi-block computational domain in isogeometric applications". *Computer-Aided Design*, 45(2), pp. 395-404, 2012.
[45] Nguyen V.P., Kerfriden P., Brino M., Bordas S.P.A., Bonisoli E., "Nitsche's method for two and three dimensional NURBS patch coupling". *Computational Mechanics*, 56(3), pp. 1163-1182, 2013.
[46] Tornincasa S., Bonisoli E., Brino M., "NURBS patch coupling with Nitsche's Method for isogeometric analysis". *IDMME Conference*, 2014, June 18-10, Toulouse, France.
[47] Tornincasa S., Bonisoli E., Kerfriden P., Brino M., "Investigation of crossing and veering phenomena in an isogeometric analysis framework". *Proceedings of IMAC XXXII*, 2014, February 3-6, Orlando, Florida, pp. 1-15.