



**HAL**  
open science

# Shape Operators and Mechanical Criteria in the Preparation of Components for Engineering Analysis

Rosalinda Ferrandes

► **To cite this version:**

Rosalinda Ferrandes. Shape Operators and Mechanical Criteria in the Preparation of Components for Engineering Analysis. Engineering Sciences [physics]. Institut National Polytechnique de Grenoble - INPG, 2008. English. NNT: . tel-00580733

**HAL Id: tel-00580733**

**<https://theses.hal.science/tel-00580733>**

Submitted on 29 Mar 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.







# Table of Contents

<b>Table of Contents .....</b>	<b>I</b>
<b>Introduction .....</b>	<b>7</b>
<b>Chapter 1 Behaviour simulations in product development processes .....</b>	<b>11</b>
1.1 Different Product Views in the Product Development Process.....	11
1.2 Shape models in a PDP.....	15
1.2.1 Basic shape entities of digital shape models .....	15
1.2.2 Constructive Solid Geometry .....	18
1.2.3 Boundary Representation .....	19
1.2.4 Polyhedral representation as a robust shape model.....	22
1.2.5 Wider shape diversity through non-manifold models .....	23
1.2.6 Meshes as an example of shape representation based on the cellular decomposition .....	25
1.3 Reference Models for Design PV .....	26
1.4 Simulation models for behaviour simulation PV .....	27
1.4.1 Objectives of behaviour analysis.....	28
1.4.2 Existing analysis methods .....	31
1.4.3 Domain discretization for FEA .....	32
1.4.4 Boundary Conditions.....	33
1.5 Process of simulation model preparation.....	37
1.5.1 The interface between Design and Behaviour Simulation PVs: historical development .....	37
1.5.2 Issues in the FE model preparation process .....	38
1.5.3 Formulation of the problem hypotheses.....	39
1.5.4 Generation of the domain shape of the mechanical model .....	41
1.5.5 Generation of the simulation model .....	43
1.6 Existing approaches for the generation of a simulation model .....	45
1.6.1 Shape adaptation during or after FE mesh generation .....	46
1.6.2 Shape adaptation prior to FE mesh generation.....	47
1.6.2.1 Form Feature approaches as an “hard” approaches .....	47
1.6.2.2 Virtual topology approaches as a “soft” approach.....	50
1.6.2.3 Layered approach.....	51
1.6.2.4 The polyhedral approach.....	52

1.7	Inaccuracies in the FE model preparation .....	52
1.8	Criteria supporting the process of simulation model preparation.....	54
1.8.1	A priori and a posteriori criteria .....	56
1.8.2	Criteria monitoring shape simplifications .....	60
1.9	Conclusions .....	62
<b>Chapter 2</b>	<b>From a CAD model to the simulation model for FEA .....</b>	<b>65</b>
2.1	Generation of an intermediate model .....	65
2.2	Use of a polyhedral model.....	69
2.3	The mixed shape representation .....	70
2.3.1	HLT data structure .....	71
2.3.2	How to generate the mixed shape representation.....	74
2.3.3	Adding semantics to the mixed shape representation .....	80
2.4	Adopted criteria supporting the shape adaptation process .....	83
2.4.1	Nature and usage of criteria: Constraints and Indicators .....	83
2.4.2	A priori mechanical criteria driving the simplification process.....	85
2.4.3	Use of an a posteriori criterion .....	86
2.5	Definition of simplification details.....	87
2.6	FE mesh preparation.....	88
<b>Chapter 3</b>	<b>Shape simplification operators.....</b>	<b>93</b>
3.1	Operators acting on the polyhedral model.....	93
3.1.1	Skin detail removal operator .....	94
3.1.2	Topological detail removal operator .....	99
3.2	Operators modifying the boundary decomposition .....	102
3.2.1	HLT-Vertex invalidation operator .....	103
3.2.2	Polyedge invalidation operator.....	104
3.3	Exploiting the two shape representations .....	110
3.3.1	Simplification process constrained by boundary representations ....	110
3.3.2	Localised simplification process .....	112
3.4	Identification and removal of simplification details.....	112
3.4.1	Interactive identification of simplification details .....	113
3.4.2	Automatic identification of shape features as simplification details	115
3.4.2.1	Hole form features .....	115
3.4.2.2	Fillet form features.....	120

3.5	Storing data for later reuse.....	127
3.5.1	Storing operations based on the polyhedral representation.....	127
3.5.2	Storing operations exploiting the mixed shape representation.....	128
3.5.3	Storing data on removed sub-domains .....	129
<b>Chapter 4</b>	<b>Toward a FE mesh generator: Approach and operators .....</b>	<b>133</b>
4.1	Integration of a surface FE mesh generator in the software environment based on the mixed shape representation .....	133
4.2	Generation of FE meshing constraints based on the shape domain of the mechanical model.....	136
4.3	Mesh enrichment operator .....	137
4.4	Operators for FE mesh optimization .....	138
4.4.1	Smoothing operator .....	139
4.4.2	Swapping operator.....	143
4.5	Combination of operators for preparing a surface FE mesh.....	145
<b>Chapter 5</b>	<b>A posteriori criterion characterizing the influence of shape variations on FE results .....</b>	<b>149</b>
5.1	Advantages of using an a posteriori criterion .....	149
5.2	Principles of the a posteriori mechanical criterion .....	150
5.2.1	Error on the FE solution due to sub-domain removals.....	151
5.2.2	Error approximation by means of a FE local problem.....	154
5.3	Generation of the FE local problems .....	156
5.3.1	Obtaining closed sub-domains for the application of the a posteriori criterion .....	158
5.3.2	Generation of FE local problems .....	159
5.4	Validation of the a posteriori mechanical criterion .....	160
5.4.1	Effectiveness of the a posteriori mechanical criterion .....	160
5.4.2	Influence of the discretization error on the criterion accuracy.....	163
5.4.3	Influence of the mesh non-conformity .....	164
5.5	Adaptive modelling process .....	166
5.5.1	Redefinition of the geometric domain of the mechanical model .....	166
5.5.2	Shape operators for the simulation model re-adaptation.....	169
5.6	Conclusions .....	171
<b>Chapter 6</b>	<b>Application of the a posteriori mechanical criterion to different scenarios of PDP .....</b>	<b>173</b>

6.1	Towards a stronger integration of the engineering analysis in a PDP.....	173
6.2	Use of the a posteriori criterion during PDP .....	174
6.3	The behavioural modeller paradigm.....	175
6.4	Consistency of mechanical models when designing component variants	184
6.5	Impact of BCs modification on the shape of a mechanical model .....	189
6.6	Conclusions .....	192
<b>Conclusions and Perspectives.....</b>		<b>195</b>
<b>Bibliography .....</b>		<b>199</b>







## Introduction

Recently, worldwide competition and technological progress brought to product time-to-market reduction and specialisation of competencies in the Product Development Process (PDP). PDP is a very complex process, covering all successive states, e.g. design, manufacturing, assembly, that the product goes across prior to its distribution. It requires specific expertises according to the considered phase of the process, and therefore involves several individuals - stakeholders - each one expert in a specific field of knowledge.

Each stakeholder, who focuses on a particular process of the PDP, considers a specific description of the product, which normally relies on a core shape model, and deals with complementary data, which are application-specific and may include logical, alphanumeric or semantic information. The combination of stakeholder, activities, and shape model specific to a PDP process corresponds to a Product View (PV).

PV stakeholders mutually interact by exchanging information, product data and instructions. Information specific to a PV can be essential for the specification of the product description related to another PV. However, moving between different PVs is not trivial. Indeed, different PVs may require not only changes of the model used for describing the product, but also modifications of the shape product itself.

Technical advances and market pressure have led to the development of Concurrent Engineering approaches, based on procedures aiming at the parallelisation of tasks originally executed sequentially. In a concurrent engineering context, the integration of the various activities of PDP is essential to provide rapid evaluation of the impact of shape modifications occurring during a particular activity on the other ones. Thus, the need for an effective product shapes processing emerges.

Here, we focus on the integration of two particular PVs, design and behaviour analysis. Some of the issues emerging could be anyhow generalized to the integration of other PDP activities. Typically, models designed for engineering purposes are simulated and verified in order to confirm, prior to the product manufacturing, that design requirements are satisfied. Following the classical approach, during the engineering analysis process, two different models are created together with their related information: the design model, whose shape meets the product specifications defined by the design office, and the simulation model, used to perform engineering analyses in the engineering office.

The generation of a simulation model starting from a design model is a complex process. It is based on a top-down approach, which starts from CAD data corresponding to the design model but does not simply consist in the transfer and use of these data. Indeed, it is also supported by several criteria that drive modelling choices. These criteria are deduced from the objectives and hypotheses

characterizing the specific analysis problem. They can constrain or orient the shape adaptations performed when creating the shape model for the behaviour simulation. Moreover, they can act either a priori, i.e. before performing the engineering analysis, or a posteriori, i.e. aiming at validating the shape modifications performed by means of behaviour analysis results. Therefore, the shape of the simulation model is the result of the hypotheses and objectives that are the basis of the considered criteria.

The explicit characterization of the hypotheses and objectives on the simulation model is a difficult task. Indeed, they are difficult to be discriminated from the parameters related to the method and the software chosen for the analysis. In this context, the use of an intermediate model can be of significant help. It allows a more explicit formalization of the mechanical problem, since it is the outcome of all the hypotheses about the mechanical behaviour of the object and can be accessed when needed for traceability and update.

However, the use of an intermediate model for the interface between the design and engineering analysis activities is not sufficient to obtain an explicit characterization of some hypotheses. Indeed, even if the shape of the intermediate model results from the hypotheses made, no straightforward correspondence between the hypotheses and the related shape transformations is derivable on the intermediate model.

The main motivation of the present work arises from the above considerations. Here, we propose a general framework for the definition of an intermediate model for realizing the interface between the design and engineering analysis PVs. We introduce a shape representation -the mixed shape representation- that allows the user to define explicitly on the shape of the intermediate model some hypotheses driving his/her modelling choices. This is achieved by associating the problem hypotheses to constraints that must be satisfied during the generation of the intermediate model, or later during the generation of the simulation model. In this way, it is possible to establish a correspondence between the shape modifications occurring on the design model and the mechanical hypotheses related to the analysis problem.

The manuscript structure is as follows:

- **Chapter 1** investigates the interface between two PVs, with particular emphasis on the transition from design to behaviour simulation. Changes in terms of shape representations and descriptions are analysed, together with the need for transferring and exploiting complementary information involved in the targeted activities. The different approaches already discussed in the literature and used in the industrial practice are presented. On the basis of the performed analysis, the complexity of the process and the need for an intermediate modelling stage, relying on an appropriate shape representation and driven by effective criteria, emerge;
- **Chapter 2** proposes possible solutions to the needs identified in terms of efficient processing of models for the interface between the design and

behaviour simulation PVs. The concepts of mixed shape representation and multiple topological layers are introduced. They allow the user to exploit a polyhedral model as reference, and simultaneously obtain descriptions of the model conforming to his/her objectives and driving the process of generation of the simulation model. Moreover, a general formulation of mechanical criteria supporting the shape transformations is provided and the link between the application of a criterion and the removal of a shape sub-domain is highlighted;

- **Chapter 3** provides details about the shape transformation operators developed by exploiting the mixed shape representation. These operators allow the user to characterize constraints on the shape model, which drive the shape adaptations at the interface between the design and the behaviour simulation activities. In particular, new methods for the identification, simplification and storage of shape detail sub-domains are described;
- **Chapter 4** describes the operators developed in order to integrate a surface FE mesh generator in the software environment based on the mixed shape representation, starting from the polyhedral model obtained with the shape adaptation process, and benefiting by the possibility of expressing explicitly FE meshing constraints;
- **Chapter 5** introduces the use of a specific a posteriori criterion that evaluates the influence of shape simplifications, performed when generating the shape support of a simulation model, on the obtained behaviour simulation results. Therefore, the impact of the a priori choices performed during the preparation of a simulation model can be verified. In the case where some shape adaptations are considered as exceedingly influencing the obtained analysis results, the shape of the simulation model needs to be adapted in order to provide more accurate results.
- **Chapter 6** provides an overview of additional PDP scenarios where the proposed a posteriori criterion could be of significant help. Indeed, the criterion provides information about the mechanical behaviour of a component without performing a brand new behaviour simulation whenever a shape modification occurs. Therefore, it allows one to save the time usually dedicated to the adaptation of the shape related to a simulation model.



# Chapter 1

## Behaviour simulations in product development processes

*The present chapter describes the context of the manuscript, investigating the role of behaviour simulations in the framework of a product development process and focussing on their relation with the design stage. The need for different models at different stages of the product development process is discussed and the concept of Product View is introduced. A description of existing models for design and behaviour simulations is also provided and followed by a detailed analysis of the interface between these two activities. Moreover, the different approaches existing in the literature and used in the industrial practice are presented, which tackle the transfer between the design and the analysis stages and the control of the process reliability. The complexity of the above process emerges together with the needs for appropriate shape representations and effective control criteria.*

---

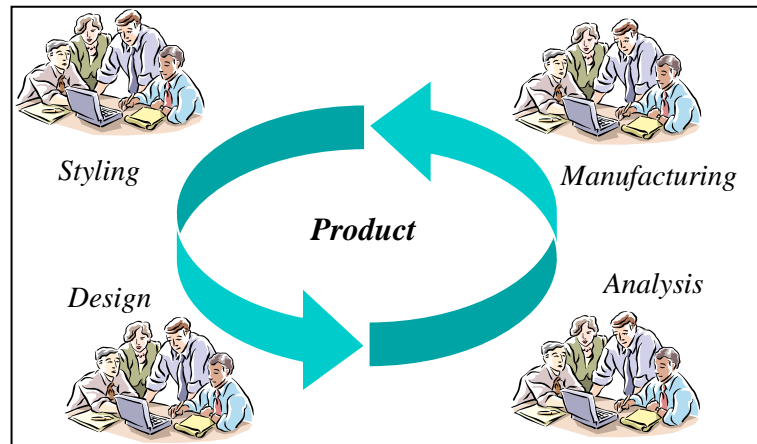
### 1.1 Different Product Views in the Product Development Process

The **Product Development Process (PDP)** deals with all the aspects concerning the realisation of an artefact [Lee99; Suh90]. It covers all the successive states (design, manufacturing, assembly, etc) which the product goes across prior to its distribution. In the case the entire lifecycle is taken into account, the stages following the product commercialisation (product use, destruction and recycling) are also considered.

In the industrial context, **Product Design** can be defined in different ways, according to the domain, organization, structure and history of a specific company. Here, we will consider Product Design as the first phase of PDP, including activities which range from the identification of customers' requirements to a detailed description of the product [BK94].

The PDP involves several experts, usually called **stakeholders**, which address specific aspects of the product characterized by a particular field of knowledge. Figure 1.1 shows an example of rough decomposition of the main PDP's activities.

Each stakeholder, concentrating on a specific process of the PDP, creates and operates on a specific description of the product that reflects the important issues to be considered in the considered process. In this context, the concept of **Product View (PV)** emerges [Che06].



**Figure 1.1:** Rough decomposition of product design activities according to the field of knowledge of the stakeholder.

Each PV is associated to a task or a process performed by a stakeholder, and relies on a core shape model. This shape model consists of an abstraction and a formalization of the reality that must meet the specific application's requirements and allow the stakeholder to perform the evaluations and modifications needed during the specific activity. At this purpose, we introduce the concept of **PV Reference Model** [HLG\*06]. A PV Reference Model is formed by a shape model plus some complementary data that are application-specific and may include logical, alphanumerical or semantic information. As an example, if we consider a behaviour simulation activity, the reference model should include a Finite Element (FE) model together with all the data required for the mechanical simulation. In contrast, in the case where the objective consists of a milling process, the appropriate reference model, in addition to the component shape, should also contain information about tool path trajectories, clamping devices, tool description, etc.

Multiple views of the product, corresponding to different models and descriptions, are necessary to meet the specific purposes and skills of the stakeholders and therefore are required during the PDP [BN03; RG98]. The need for efficiently handling multiple PVs and representations of the product is particularly important in the context of a **Concurrent Engineering (CE)** approach. Indeed, the Concurrent Engineering approach aims at integrating the different points of view occurring during the product development. It cannot be considered as an entirely formalised design method, but includes procedures to parallelise some tasks that originally were executed sequentially. This parallelism enables to take into account as soon as possible the various constraints and parameters that would be managed later in a PDP organised sequentially [Alt93; JPS93; Kus93]. In the CE framework, the PDP is:

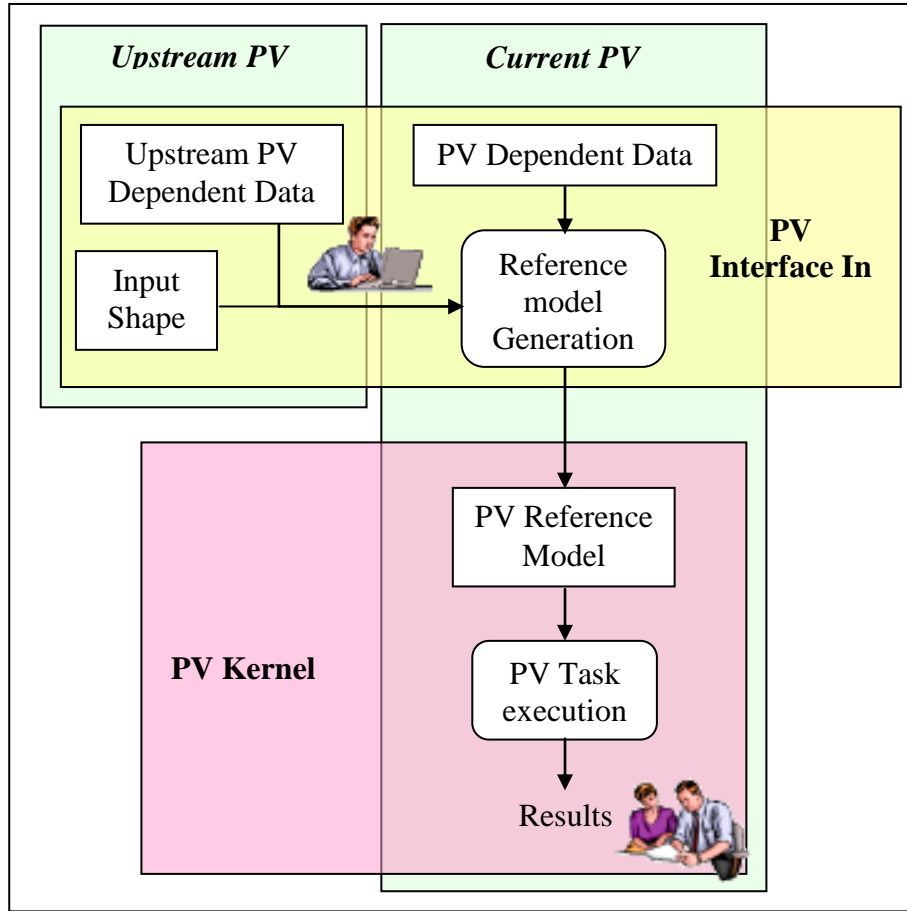
- **Collaborative**, since it is based on the collaboration of different stakeholders having complementary competences and viewpoints;
- As **parallel** as possible, since the stakeholders work simultaneously on the definition of the same product;



- **Distributed**, since the stakeholders need to communicate, collaborate and exchange information even in the case where they are physically distant.

This notion of integration aims at modifying the prescriptive behaviours of PDPs organised sequentially and reorganising the decision mechanisms during the PDP, in order to develop and manage more efficiently the cooperation among all the stakeholders involved. This makes crucial the efficiency of the exchange of information and the transfer process between different PVs, so that modifications occurring when moving across PVs can be rapidly evaluated. In addition, it should be noticed that PVs are not unrelated. Indeed, information specific to a PV can be crucial for the specification of another PV, being explicitly transferred or providing means for deducing essential data. For instance, if we consider assembly models, we could infer Boundary Conditions (BCs) (see section 1.4.4), which are part of a behaviour simulation PV, by looking at contacts among assembly components in the design PV. Moreover, in a CE environment it is important that the various PV reference models are integrated to provide capabilities to rapidly evaluate modifications occurring in a PV with respect to all the other affected views.

Moving from one PV to another one, not only the PV reference model changes, but also the shape of the product. Therefore, the need for processing the product shape during the activities of the PDP emerges. As an example, let us consider the design and behaviour simulation PVs. The reference model of a design PV is based on a B-Rep NURBS model (see section 1.2.3), generally supported by CAD software, while the behaviour simulation PV reference model is based on a FE mesh (see section 1.4.3). Besides the change of reference model, shape modifications occur when moving from design to simulation PV, e.g. through holes can be removed and thin flat volumes can be transformed into bounded planar surfaces of a plate model. As an additional example, even if the design stage based on CAD software produces a shape close to the manufactured one, the shape of the manufacturing view may contain holes represented by their axis and radius only, rather than their real representation based on cylindrical surfaces, cones, etc. Often, it is sufficient to represent drilled holes by their axis due to the fact that this information alone is sufficient to characterize the drill trajectory.



**Figure 1.2:** Integration process of two PVs.

The passage between two different PVs (referred to as *Upstream PV* and *Current PV*) relies on the concept of *PV Interface In*, illustrated in Figure 1.2, according to the approach suggested in [HLG\*06]. In the *PV Interface In*, the *Upstream PV* delivers **Input shape data**, which are strictly related to shape, and **Upstream PV dependent data**, which are complementary data depending on the task objectives. In the *Current PV*, the stakeholder supplies additional information, the **PV dependent data**. As an example, if the *Current PV* is devoted to a dynamic simulation and the data provided by the *Upstream PV* include the CAD model only, *PV dependent data* contain information about materials and velocities. These data cannot be inferred from the shape information provided by the input CAD model, and must be necessarily added. Next, the **Reference Model generation** process, which exploits all the input information and performs possible shape changes, takes place. New shape models can be created to achieve better performances of the integration process between PVs. The stakeholder of the *Current PV* deals with a **PV Reference Model** i.e. the kernel shape representation [Dri06] adapted to the **PV Task**, and a set of interface data, used for communicating data with others stakeholders.

In the following sections, we present the shape models commonly used in the definition of reference models (see section 1.2). Then, we will concentrate on two specific PVs and their related reference models. More precisely, section 1.3 will

describe the PV related to the actual design process, while section 1.4 the one related to the activity of behaviour simulation. Next, we will analyse the process of interface between the above PVs and examine the changes in terms of shape models, the transfer, exploitation and setting of the needed complementary information. The analysis of the issues which originate from this process constitutes most part of the present work.

## 1.2 Shape models in a PDP

Throughout a PDP, different shape models of the product are required to support the various PVs. According to [Req80], a **shape model** is a computational structure that captures the spatial aspects of the objects of interest for an application.

### 1.2.1 Basic shape entities of digital shape models

Here, we are interested in digital shape models described in the 3D Euclidean space ( $E^3$ ) of analytic geometry.

To describe a shape model, two types of entities are used, which are anyway interconnected:

- **Topological entities**, which are useful for describing how geometric entities are connected within this shape model;
- **Geometrical entities**, which define the geometric nature of the topological ones. Indeed, a topological entity can have several different geometric representations.

To define the different kinds of entities required for the description of 3D models, we need to introduce the concept of **manifold**:

#### Definition 1.1

A **manifold** having dimension  $d$  is a  $d$ -dimensional topological space  $M$  in which each point has a neighbourhood that is topologically equivalent to an open disk of a Euclidean space of dimension  $d$ .

Then, the entities able to describe 3D models can be classified according to the dimension  $d$  of the geometric manifold they define [Fin01]:

- **Punctual entities**: topological entities (usually referred to as *vertex*) defining a geometric manifold  $d = 0$ . These elements are used in defining the connections of entities with  $d > 0$ , i.e. curves or surfaces. They can also designate particular loci, as the location of a force or an attribute. The corresponding geometric entities are usually referred to as *point*;

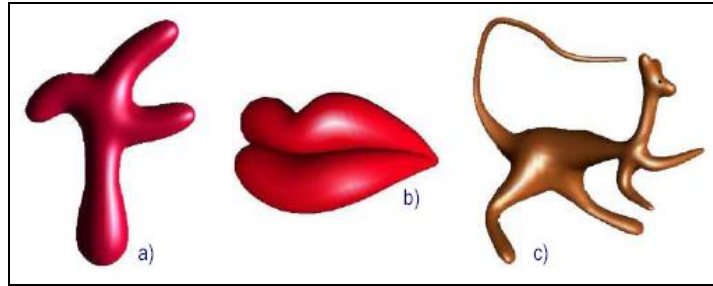
- **Linear entities:** topological entities (usually referred to as *edges*) defining a geometric manifold  $d = 1$ . This class of elements can be used for sketching the model at a preliminary stage of design, or for the description of reference elements of a FE model. As in the previous case, they are used in defining the connections of entities having higher manifold dimension. The corresponding geometric entities are referred to as *curves*;
- **Surface entities:** topological or geometrical entities (usually referred to as *faces*) defining a geometric manifold  $d = 2$ . Faces may be used either for defining the boundary of volume objects or describing open surfaces. In addition, they can correspond to reference or construction elements, e. g. symmetry plans. The corresponding geometric entities are usually referred to as *surfaces*;
- **Volume entities:** objects having a geometric manifold  $d = 3$ . They imply the notion of interior and exterior [Mor85] and are characterised by:
  - closed domains,
  - orientation of the boundary surfaces,
  - non self-intersection of the boundary surfaces.

Two categories of description for curves and surfaces can be considered:

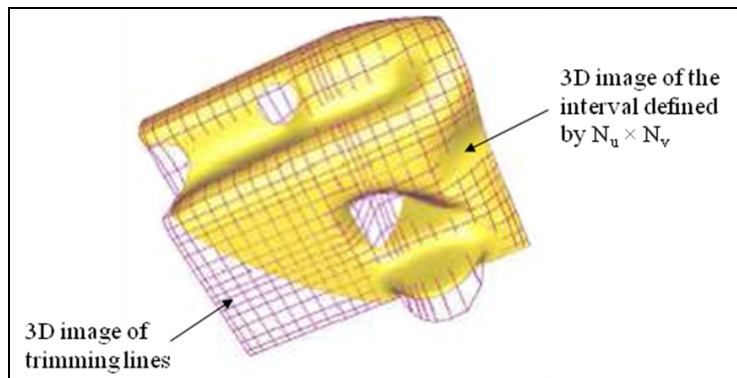
- **Implicit descriptions**, used for representing shapes by an implicit formulation,  $f(x, y, z) = 0$ . In engineering design, the functions are simple and correspond to the definition of simple primitives, e.g. planes, cylinder, spheres. In computer animation applications, more complex functions are used and combined together to represent complex shapes (Figure 1.3);
- **Parametric descriptions**, extensively used in CAD systems to represent both simple primitives and free-form surfaces [Leo91]. Several descriptions fall in this category, but the most commonly used are tensor product surfaces, like the Bézier, B-Spline and NURBS representations. A parametric surface is defined in terms of functions of two variables,  $u$  and  $v$ , with  $(u, v) \in N_u \times N_v$ :

$$\vec{P}(u, v) = \frac{\sum_{i=0}^m \sum_{j=0}^n h_{ij} N_{ip}(u, v) N_{jq}(u, v) \cdot \vec{s}_{ij}}{\sum_{k=0}^m \sum_{z=0}^n h_{kz} N_{kp}(u, v) N_{zq}(u, v)}, \quad (1.1)$$

where the control points  $\vec{s}_{ij}$ , the weights  $h_{ij}$  and the knot sequences  $N_u, N_v$  are the parameters used to modify the surface shape. The concept of trimming lines is then introduced, that enables a restriction of the definition domain (Figure 1.4).



**Figure 1.3:** Examples of implicit surfaces [HAC03]



**Figure 1.4:** A trimmed B-Spline surface with its control polyhedron highlighted [Che06].

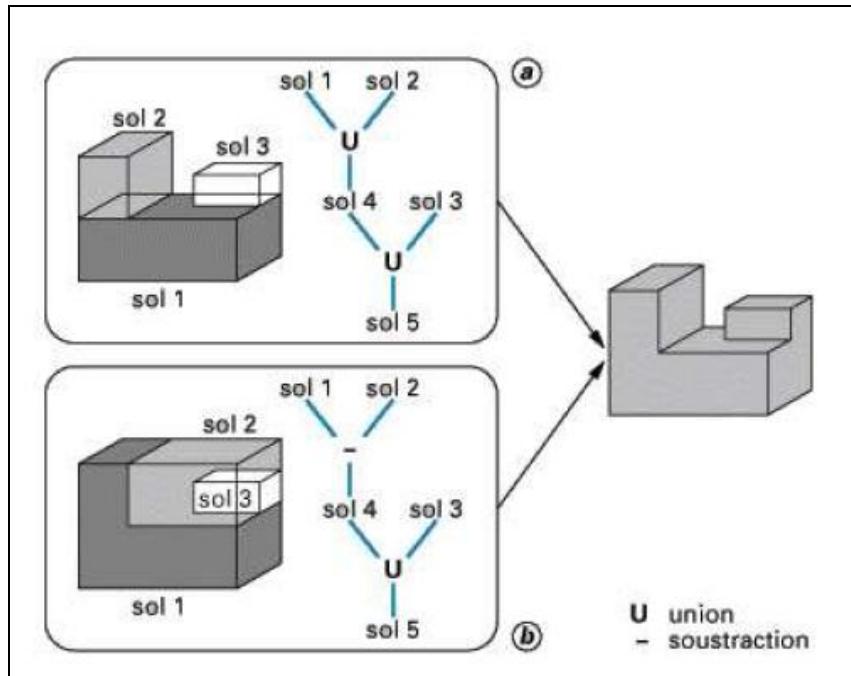
Most of the physical objects that are issue of a PDP are solids. However, it can happen, according to the type of product and/or the hypotheses considered, that they can be modelled as surfaces or curves. For example, in stress analysis, thin shells can be analyzed as if their thickness was effectively zero. Next, we often need to model not only solid objects but also operations on them. For example, fabrication processes such as machining or welding are important in CAD/CAM. The geometrical aspects of machining may be modelled as the difference between the initial state of the work piece and the volume swept by the cutter in its motion. Welding and other additive processes may be modelled by set union.

Volume objects are frequently described through Constructive Solid Geometry and Boundary Representation approaches. These representations decompose space into three partitions, i.e. interior, exterior and boundary. More details about this subject will be given in sections 1.2.2 and 1.2.3. Another common representation for volume objects, especially used for visualization applications, is the cellular decomposition, where the shape is partitioned into a set of elementary primitives, named cells or voxels. The shape is completely defined by the aggregation of the cells. A mesh (see section 1.2.6) can be seen as a shape representation based on the cellular decomposition.

## 1.2.2 Constructive Solid Geometry

Constructive Solid Geometry (CSG) has been the first representation adopted in CAD systems [Rec80]. It provides a concise and implicit representation of the volume object as a Boolean composition of parameterized volumetric primitives (e.g. cylinders, spheres, extruded surfaces) or more complex features (e.g. slots or counter-bored holes), suitable for a particular application domain. The primitives may be instantiated multiple times (possibly with different parameter values, positions and orientations) and grouped hierarchically. Primitive instances and groups may be transformed through rigid body motions (which combine rotations and translations) or scaling operations. Regularised set-theoretic union, intersection and difference are the boolean operations used in CSG to guarantee the model validity, i.e. to preserve the concept of volume.

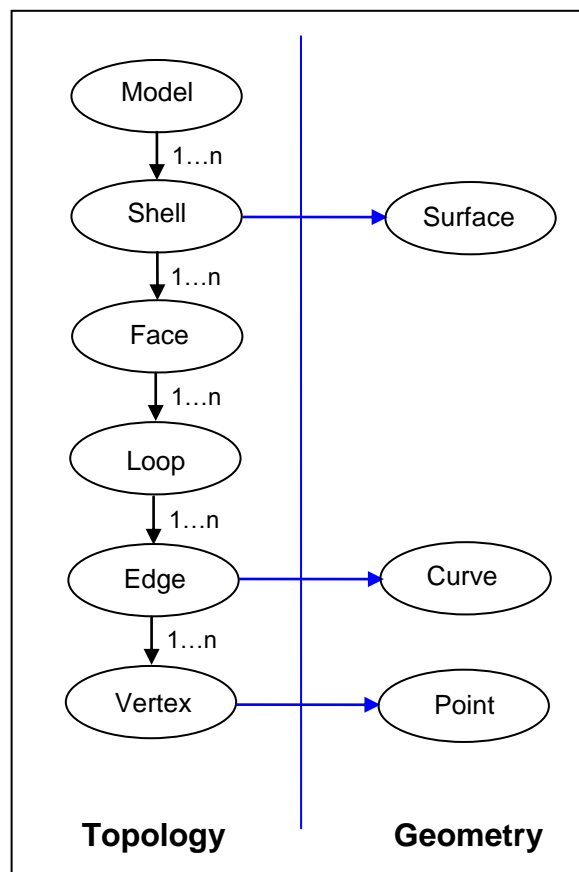
A CSG is a volume decomposition represented as a tree structure, which provides a hierarchical representation. Figure 1.5 shows an example of CSG tree structure. Although this modelling approach has the advantage of enabling an easy solid construction, the resulting representation is too simple and not flexible enough. Indeed, it is limited to the basic primitives accessible to the modeller and it is also impossible to insert free-form surfaces in the solid. Additionally, CGS is syntactic and does not provide an explicit evaluation of the object entities (faces, edges, vertices). That is the reason why CSG modellers have evolved to be combined with boundary modellers.



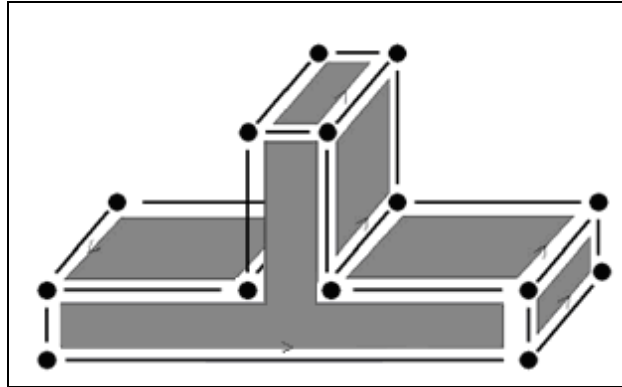
**Figure 1.5:** Example of shape model described by a CSG representation.

### 1.2.3 Boundary Representation

Boundary schemes are the most widely used representations for volume objects. In Boundary Representation (B-Rep), volume objects are described in terms of the surfaces representing their boundaries. It is based on the principle that a solid object divides the space in two disjoint partitions: one closed, bounded and oriented (the object itself), the other one infinite (the outer space). The object boundary is described by a topological structure. Faces are the main topological entities, properly connected and consistently oriented. Each face is bounded by one or more loops and embedded in one surface. Each loop is a sequence of edges, properly connected to form a simple oriented closed curve. Each edge is an open or closed curve, embedded in a curve and bounded by two vertices, possibly coincident. In addition, the collection of faces whose union is a connected 2-manifold without boundary forms the so-called shell. Figure 1.6 shows the diagram of such a hierarchical structure. Adjacency and incidence relations among the above topological entities are referred to as topology or topological structure, and their underlying geometries as geometry or geometric representation.



**Figure 1.6:** Entities and relationships in Boundary representations.



**Figure 1.7:** Example of B-Rep representation.

The subdivision in faces of a B-Rep model is performed so that the shape of each face has a compact mathematical representation, i.e. it lies on a single analytical or parametric surface [Man88]. As a result, the surface modelling capabilities increase, and it is possible the representation of free-form shapes. Figure 1.7 shows an example of B-Rep representation.

To combine the advantages of both B-Rep and CSG approaches, the development of hybrid solid modellers [Man88] has been investigated. By using a hybrid modeller, it is possible to pick the most suitable representation and operators for each task. However, these systems combine also some disadvantages of these representations. Conversions have to be made constantly from one representation into the other one, and continuous checking of topological consistency is also required. Moreover, the coexistence of the two models induces an even higher complexity of the data structure.

However, the current industrial solid modellers, as ACIS [Spa93] and PARASOLID [Eds95], use B-Rep as kernel representation. Then, the CSG operators can be as well used, which are built on the basis of B-Rep ones.

However some conditions must hold for assuring the validity of a B-Rep model. A B-Rep model is valid if it defines the boundary of a “reasonable” solid object. Specifically, a B-Rep model is valid if the following conditions hold [Man88]:

1. Faces may intersect only at common edges;
2. Each edge is shared by exactly two faces;
3. Faces around each vertex can be arranged in a cyclical sequence such that each consecutive pair shares an edge incident to this vertex.

The first and second conditions exclude self-intersecting objects. The third condition disallows “open” objects, and ensures that the surface forms a 2-manifold.

The above conditions for validity address *topological integrity*, meaning that it is possible to assign to each topological entity a geometric instance such that the overall geometry is a 2-manifold without boundary, and *geometric integrity*,



meaning that the geometric representation conforms to all topological relations used in the topological structure. The geometric integrity of a model is computationally expensive to check, especially if the faces are defined by curvilinear surfaces. In order to avoid checking validity after the B-Rep model has been built, most geometric modelling systems attempt to embed the required validity conditions in the algorithms used to construct the representations.

Any valid manifold B-Rep can be constructed by a sequence of Euler operations [Man88], which ensure that some necessary conditions for validity are satisfied. Euler operators allow one to model the B-Rep object while satisfying the Euler-Poincare theorem,

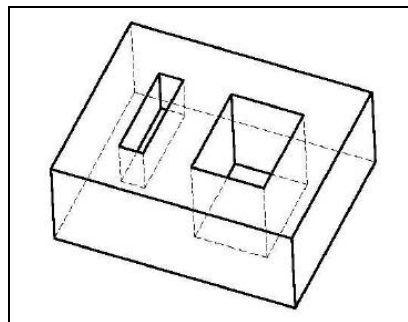
$$v - e + f = 2 \times (s - h) + r, \quad (1.2)$$

where:

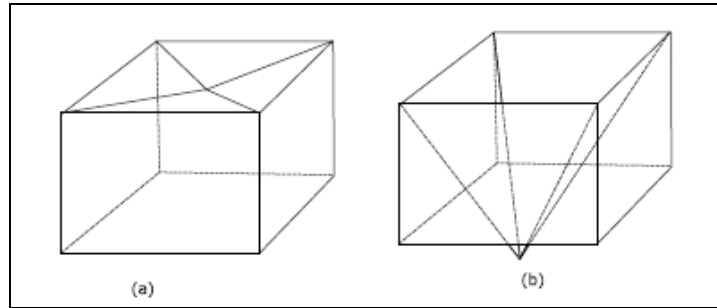
- $v, e, f$  and  $s$  are respectively the number of vertices, edges, faces, shells;
- $r$  (ring) is the number of interior loops in the faces and  $h$  that one of through holes or handles in the solid.

An example of B-Rep model satisfying the Euler-Poincare theorem is showed in Figure 1.8. In the example,  $s=1, f=15, e=36, v=24, r=3, h=1$ , and  $24 - 36 + 15 = 2 \times (1 - 1) + 3$ .

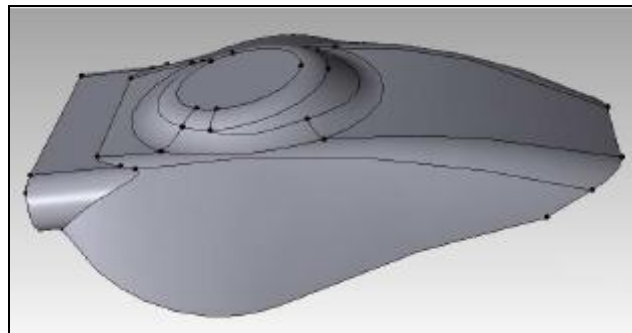
However, the use of Euler operators does not ensure that the B-Rep represents a valid 2-manifold object. Indeed, the resulting B-Rep model may or may not be valid, depending on the metric data associated with it [SSP00; EWW96]. Let us assume that the dimpled cube shown in Figure 1.9 was built through Euler operations. By simply assigning different coordinates of one vertex we obtain the set of self-intersecting faces shown on the right, which does not correspond to a valid B-Rep model. Therefore, a representation constructed with Euler operators may or may not be valid, depending on the values assigned to the geometry of faces, edges and vertices.



**Figure 1.8:** Example of B-Rep model satisfying the Euler-Poincare theorem.



**Figure 1.9:** B-Rep models modelled by using Euler operators. The coordinates of a vertex of the 2-manifold object depicted in (a) change and the B-Rep model depicted in (b) is no more valid.



**Figure 1.10:** Example of object decomposition not reflecting adequately the object shape.

Most of the commercial modellers describe manifold B-Rep models according to the above described topological properties. Figure 1.10 shows an example, where the shape of the object is decomposed into a set of surfaces. The object decomposition used does not necessarily reflect the shape features of the object, but rather the volume generation process. Therefore, the object decomposition prescribed by commercial modellers often does not reflect adequately the shape of the object and is not adapted to represent the intrinsic information attached to it.

Besides the described representation for volume models, two additional categories of models need to be described in the context of this work. In fact, they may intervene all along the PDP and not only at the proper design stage.

#### 1.2.4 Polyhedral representation as a robust shape model

Polyhedral models are a recurrent representation for volume models, and can be considered as a particular category of B-Rep model. A polyhedral model describes arbitrary shapes through a piecewise linear approximation of their external surface. A polyhedral model is robust compared to a B-Rep NURBS one. Indeed, the connection between the basic entities of a polyhedral model is purely based on topology, whereas B-Rep NURBS models rely on trimmed patches and therefore require geometric connections through tolerances.

When the polygons are all of the same type, e.g. all quadrangles or all triangles, the B-Rep is called **tessellation**. Tessellations with triangular faces are called **triangulations**. Therefore, a triangulation is a specific category of B-Rep representation, where the faces are triangles and the edges are line segments. The validity conditions for a triangulation are the following [Req99]:

1. Each face must have exactly 3 edges; otherwise it cannot be a triangle;
2. Each edge must have exactly two vertices; otherwise it cannot be a line segment;
3. The edges associated to a face must form a loop or closed circuit, to ensure that they enclose a 2-D area. This condition is satisfied if and only if each vertex in a face belongs exactly to two of the face edges;
4. The faces must form one or more closed surfaces or shells, to ensure that they enclose a 3-D volume. This condition is satisfied if and only if each edge belongs to an even number of faces. If we restrict the domain to manifold polyhedrons, an edge has to belong exactly to two faces;
5. Each 3-tuple of coordinates must correspond to a distinct point in 3-space;
6. Edges must either be disjoint or intersect at a common vertex; otherwise there would be missing vertices in the representation;
7. Similarly, faces must either be disjoint or intersect at a common edge or vertex.

These conditions are easy to establish intuitively, and can be derived mathematically. Conditions 1-4 are *combinatorial*. They are easy to check algorithmically by counting nodes or links in the boundary graph. Instead, conditions 5-7 are *metric*, i.e. they involve coordinates of vertices and equations of lines and planes. They are computationally expensive to check, because they require intersection tests.

Polyhedral models are often used for visualization purposes, since graphic libraries and hardware graphic accelerators can render such simple shape primitives very fast. In what follows, we will discuss how a polyhedral representation can be very useful for behaviour simulation purposes.

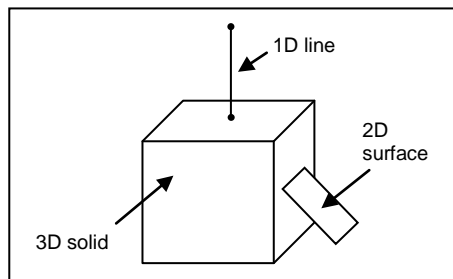
### 1.2.5 Wider shape diversity through non-manifold models

A non-manifold model allows the coexistence of entities having different manifold dimensions. Indeed, an object can be formed by sub-domains of different dimensionality, i.e. wireframes, surfaces and volumes. Therefore, models need to be able to handle such configurations (see Figure 1.11). To this end, non-manifold models relax some of the topological correctness criteria applied to 2-manifold B-Rep models, in particular those requiring that:

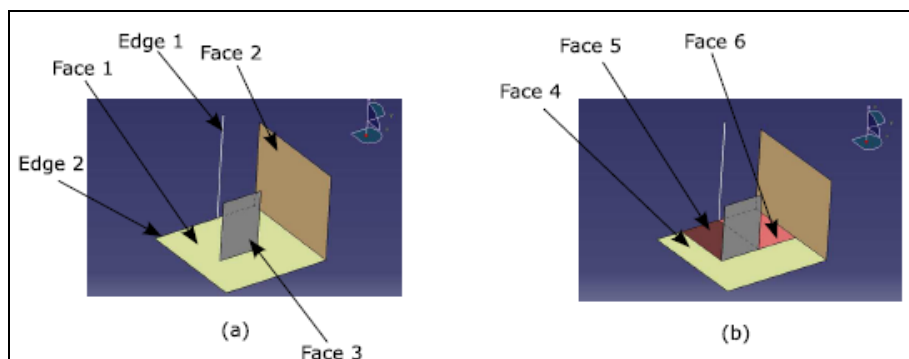
- All the edges separate exactly two faces,
- All the vertices are surrounded by a single sequence of adjacent faces.

Non-manifold models are constructed using the same basic topological elements than traditional boundary representations, i.e. faces, edges and vertices, whereas the connection elements expressing the adjacencies among them have their data structures modified to deal with more general configurations. An example of a non-manifold B-Rep is given in [Wei88], where the entity-use has been added to indicate the occurrence of the entity into a higher dimensional element.

The capacity of a modeller to represent both manifold and non-manifold configurations is necessary for describing FE simulation models where different mechanical behaviours take place over the object and the dimensional reduction of some parts may be mandatory either to match mechanical hypotheses or to reduce the solution computation time. Actually, differences occur between the required component representation and the component actually constructed by the geometric modeller. Figure 1.12 shows an example. To describe the idealized model in Figure 1.12(a), which is formed by manifolds having different manifold dimensions, face 3 should be connected to face 1, and edge 1 should be connected to edge 2. Figure 1.12(b) shows the configuration required by the geometric modeller, where face 1 must be split into three faces (face 4, face 5, face 6) in order to match the arising non-manifold configuration. Similarly, edge 2 is split into two edges to enable the connection at the common vertex.



**Figure 1.11:** Object formed by sub-domains having different manifold dimensions.



**Figure 1.12:** (a) Non-manifold required object representation; (b) Representation actually constructed by the geometric modeller.

However, many geometric modellers describe these representations according to a specific extension of the topological properties. For example, certain geometric modellers provide capabilities to reduce internal loops to degenerated configurations as either vertices or open loops [CL01]. Configurations reduced to vertices can be used to describe isolated vertices on faces. In this case, the geometric entity associated with this loop is a point. Similarly, configurations reduced to open loops are used to describe isolated edges on faces; the associated geometric entity is an open curve. Actually, this extended object description is possible by using the concept of attribute, i.e. the degenerated status is assigned to loops to distinguish them from others. Therefore, there is no explicit description of the corresponding configurations and their properties, and processing these attributes becomes mandatory to distinguish faces containing isolated vertices, or any other non-manifold configuration, from faces containing just regular internal loops.

### 1.2.6 Meshes as an example of shape representation based on the cellular decomposition

As introduced at section 1.2.1, shape objects can be partitioned into a set of elementary entities, whose aggregation is able to entirely describe the object shape.

In the following, we will describe in more details a specific category of shape representation based on the cellular decomposition, which is particularly useful in engineering applications (see section 1.4.3): a **mesh**.

Here, some definitions are given [Geo91]:

#### Definition 1.2

A decomposition of a domain  $\Omega$  containing  $V$  elements is a **mesh**  $M$  if:

- $M = \bigcup_{V \in M} V$ ,
- no element  $V$  has an empty interior,
- the intersection of the interior of two elements  $V$  is empty.

In particular:

#### Definition 1.3

A mesh  $M$  satisfying the conditions defined in Def. 1.1 is **conform** if the intersection of two elements  $V$  is reduced to either an empty set or a common part of their respective boundary (either a node or an edge or, if the manifold dimension is equal to 3, a face).

A mesh can be also considered as a non-manifold object, according to what introduced in section 1.2.5. Moreover, a mesh can have different manifold

dimensions. A linear mesh is formed by a finite number of segments, a surface mesh by 2D manifold elements, e.g. triangles or quadrangles, a volume mesh by 3D manifold elements, e.g. tetrahedrons, hexahedrons, prisms. Moreover, a mesh can combine element having either different geometric nature (mixed mesh) or different manifold dimension (hybrid mesh).

The elements forming a mesh must generally verify some additional constraints according to the product view in which they are used. In the context of the present work, we will use meshes as reference models for the behaviour analysis PV (see section 1.4.3).

### **1.3 Reference Models for Design PV**

As introduced at section 1.1, a PV reference model is formed not only by a shape model, but also by complementary data specific to the considered PV. Therefore, the PV reference model may include the geometrical, technological and functional description of a product in a specific PV of the PDP.

The objective of the design PV is to provide a detailed shape satisfying the specification of the characteristics of the product to be developed. Therefore, in a design PV the shape model assumes a crucial role. However, we could consider as part of a design PV reference model the functional information associated to the product to be designed.

A typical example of information associated to the shape model of the product to be designed is constituted by functional data related to the assembly. Indeed, often the product is not a single component, but is formed by an assembly, i.e. a group of components having material or functional interactions. Therefore, it is necessary to define the data related to:

- The geometrical constraints, i.e. localisation and orientation, existing among the geometrical entities of the various components. In this way, the exact location of components into the assembly can be calculated;
- The kinematical constraints defining the authorized relative movements, starting from the knowledge about the kinematical constraints among the geometrical entities of the various components.

A different kind example of design approach allowing a designer to integrate complementary data in the design PV is the design by features. A generic definition of feature can be found in [Sha88], where the author defines the feature concept as an abstract entity that has several significances depending on the context. In [Sha88], four requirements that a form feature should fulfil at the minimum are identified. A feature:

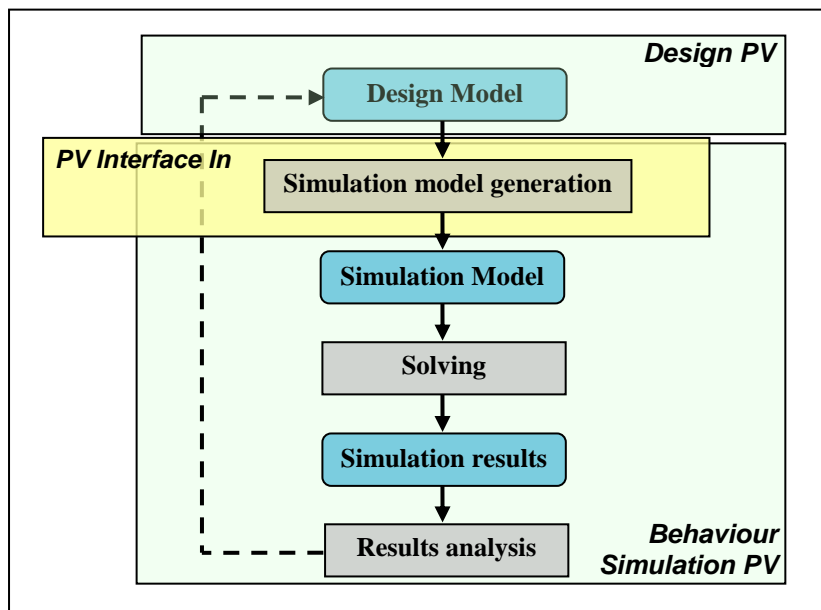
1. has to be a physical constituent of a part (component),
2. should be mappable to a generic shape,

3. should have engineering relevance,
4. must have predictable properties.

Initially, the feature concept was used for the enhancement of shape semantics, in particular when correlating product geometry to the product's manufacturing significance [JC88]. Later on, different applications of the feature concept were developed according to the information content they provide. They include functional features [CFL01], assembly features [Den98], meshing features [CM99; RB03], analysis features [ADS06].

#### 1.4 Simulation models for behaviour simulation PV

Models designed for engineering purposes are typically simulated and verified, in order to confirm, prior to the product commercialisation, that design requirements are satisfied [ABS91]. This simulation, which predicts the physical behaviour of an engineering component, is commonly termed "engineering analysis". In a classical approach, three different entities are formalised during the engineering analysis process (Figure 1.13): the design model, i.e. the reference model for the design PV, the simulation model, i.e. the reference model for the behaviour simulation PV, and the analysis results. This is the minimum level of information required to perform a behaviour analysis.



**Figure 1.13:** Process flow related to an engineering analysis process.

For a given design model, many simulation models may be defined, according to the type and physics of analysis desired, but also to other factors, e.g. the level of accuracy required.

A simulation model is generally composed of:

1. a shape domain of study,
2. some BCs providing information about forces and displacements,
3. other analysis parameters about the structure, e. g. defining the behaviour law of the material, or the analysis procedure, e.g. taking in account a non-linear behaviour, etc.

#### **1.4.1 Objectives of behaviour analysis**

The starting point to perform an engineering analysis consists in acquiring geometrical, functional and technological data related to the design model, as introduced in section 1.2.5. The amount and the proportion of these data depend on the progress of the design process. Consequently, a behaviour simulation can be performed at different stages of the design process, providing different kinds of answer. Three different scenarios where a behaviour analysis can bring support to the design process have been identified by Troussier in [Tro99]:

##### **1. Analysis supporting decision making**

A behaviour analysis can help in evaluating the performances of different design variants, therefore supporting the decision making process. What is interesting is not the quantitative evaluation of the performance, but the evaluation of the capability of a design alternative to embed a specific functionality, in comparison with the other design choices. This kind of analysis contributes to a qualitative evaluation of the mechanical behaviour and a classification of the possible design options for each key function that the product has to fulfil. The designer can select a design alternative according to its global performance when looking at the entire set of functionalities. Figure 1.14 shows an example where this type of analysis supports the selection of design alternatives. The behaviour analysis is performed before choosing the design option.

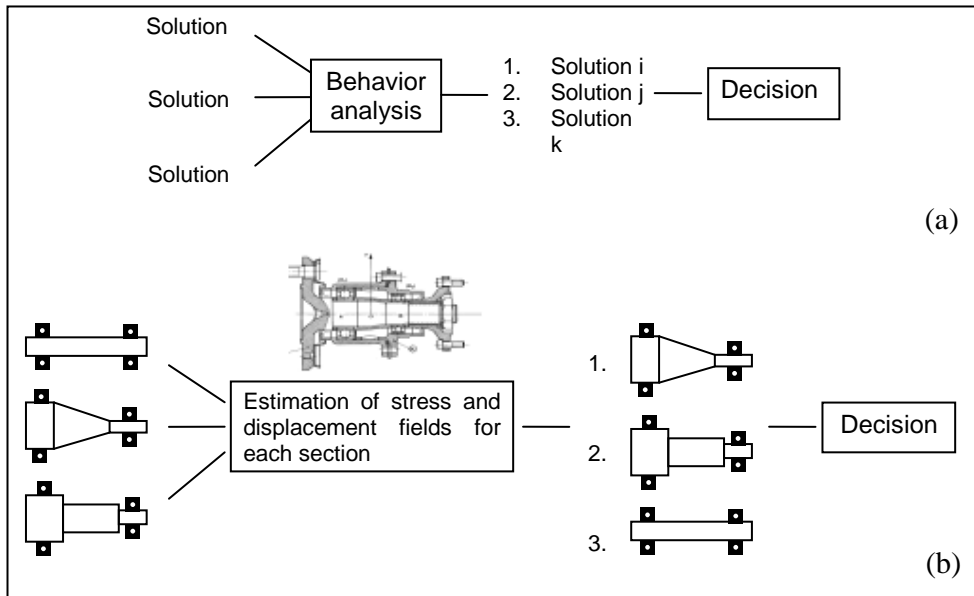
##### **2. Analysis for product validation**

This kind of analysis is the most commonly performed during a PDP. It takes place when early design decisions have been already taken. The product design is at a more advanced state and the designer needs to verify whether the product satisfies functional requirements or not. By means of this category of analysis, the product performances related to the required functionalities are evaluated quantitatively. Figure 1.15 situates this kind of analysis in the PDP with respect to the design choices that have been already taken.

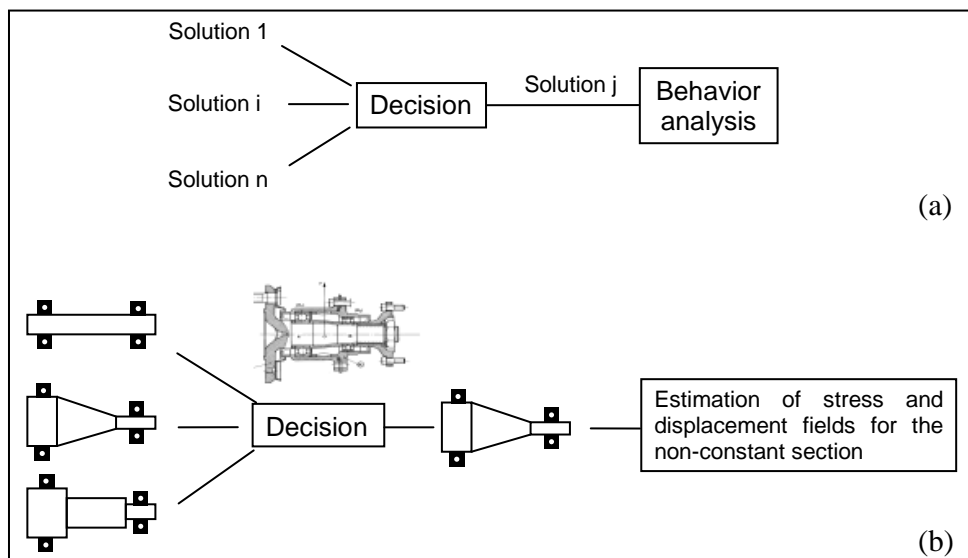


### 3. Post design analysis for knowledge capitalization

This category of analysis is performed in order to gain a better knowledge about an already designed product. In fact, it can happen sometimes that tests contradict the experts' expectations, and a behaviour analysis may help understanding how the product reacts to different usage conditions. Results provided by this category of analysis help supporting the improvement of the product design.



**Figure 1.14:** Behaviour analysis supporting decision making: (a) General process flow; (b) Example of an analysis performed on a drive shaft.



**Figure 1.15:** Behaviour analysis for product validation: (a) General process flow; (b) Example of an analysis performed on a drive shaft.

Whatever is the context in which the behaviour analysis is performed, the analysis process can be decomposed in three macro-stages, as depicted in Figure 1.13:

**a. Formulation of a simulation model**

The formulation of a simulation model starting from a design model is a crucial stage and a typical example of passage between two different PVs. This is the main subject of the present work and will be deepened in more details all along this manuscript.

To define a simulation model for an analysis, several objectives have to be taken into account, among which:

- The type of analysis to perform. Several disciplines of analysis exist, which focus on different mechanical aspects during the PDP. An analysis may be structural, thermal, electromagnetic, etc;
- The stage of the PDP at which the analysis is performed, as described above;
- The mechanical object under observation and its usage conditions;
- The mechanical quantities of interest, which could be even defined implicitly, e.g. if a study about a seal is required, the stress field of the component needs to be examined;
- The required accuracy and time and/or cost constraints, e.g. time schedule, required competences, materials to use.

**b. Solving of the simulation model**

Once the reference model of the behaviour simulation PV is available, the actual behaviour analysis can be performed. As highlighted above, a behaviour analysis can be of great benefit at different stages of the PDP.

**c. Result analysis**

Results coming from the behaviour analysis process are exploited and discussed. The conclusions coming from the analysis of the results depend on the objectives defined during the formulation of the simulation model and are strictly linked to the stakeholder in charge of the analysis. The interpretation process of the analysis results can be considered as formed by two different steps:

- The evaluation of the reliability and the accuracy of the analysis performed, through the use of specific quantities (this concept will be detailed later in sections 1.7 and 1.8);

- The interpretation of the analysis results. This stage takes place only if the values provided by the estimations performed in the previous step consider the results analysis as accurate enough.

During this stage, a strong integration with the stakeholders of the design PV is required. The analysis results need to be communicated to the stakeholders of the design process and some negotiations may be necessary between the stakeholders belonging to the two PVs in order to find a compromise.

## 1.4.2 Existing analysis methods

### a. Analytical methods

These methods were for a long time the tools used to evaluate the mechanical behaviour of design models. They follow the mechanics of materials (also known as strength of materials) or the elasticity theory approaches, which have been expounded or initiated by Timoshenko [Tim30]. They make use of analytical formulations that apply mostly to simple linear elastic models, and therefore, they can be utilised only if the model geometry is simple enough to be reduced to simple shapes, like beams, plates or shells.

Analytical methods are useful for an analysis made at early design stages, as support to the decision making process. Indeed, this category of analysis allows different technological solutions to be quickly validated, leading to a pre-dimensioning of the structure.

### b. Numerical methods

The development of information technologies has introduced numerical methods, which have enabled the behaviour analysis to be extended to more complex problems than those addressed by analytical ones.

At the present time, the **Finite Element Method (FEM)** is certainly the prominent technique in the industrial field [Zie77; Imb79, DT84]. It allows the resolution of a wide range of problems, e.g. structural, thermal, acoustic, electromagnetic analysis. A mechanical problem can be solved by formulating a system of partial differential equations which express, at a local scale, the equilibrium on an infinitesimal element of the structure. The FEM allows one to solve these systems of differential equations related to a continuum problem through the definition of an equivalent integral formulation over a discrete problem. The mesh (see section 1.2.6) discretizing the continuum is called FE mesh (see section 1.4.3) and is obtained by decomposing the geometric domain of the continuum into small elements having simple and arbitrary sizes (the finite elements). This formulation approximates at each point (node) of the mesh the value of a solution field, e.g. displacement, caused by the application of an input one,

e.g. force. The size of the model (proportional to the number of elements discretizing the geometry) has of course a direct influence over the solving time. Therefore, it is preferable that the domain where applying the FEM is less detailed than the initial domain defining the object studied. The process of adaptation of the initial geometric domain will be discussed in detail in section 1.5, and is one of the main subjects of this work.

The FEM is a well-known and standardised method in the industry. In the following, the term behaviour analysis will always implicitly refer to a **Finite Element Analysis (FEA)**. Anyway, other numerical approaches exist and are here cited for sake of completeness.

The **Finite Difference Method** is a technique similar to FEM, which still needs a mesh to discretize the continuum. This method provides an approximated solution of the differential formulation of the mechanical problem. However, this method is less used than FEA for structural analyses.

The **Boundary Element Method** is a method that, contrary to a FEM, requires only the discretization of the boundary of the domain to study. However, this method requires the use of models to describe the phenomena occurring in the internal domain. The implementation complexity has so undermined the spreading of this method.

### 1.4.3 Domain discretization for FEA

The shape domain of a simulation model needs to be discretized in order to perform a FEA. A mesh is used at this purpose, which in the context of a FEA is called FE mesh.

As introduced at section 1.2.6, a mesh needs to respect some additional constraints according to the considered PV, which in our case is the behaviour simulation. A criterion stating the validity of a FE mesh is the conformity (see Def. 1.3). Indeed, for most of the FE resolution methods, the mechanical problem can be solved only when having a conform mesh. In practice, the non-conformity of a mesh may consist in:

- lack of some element,
- overlapping of some elements,
- supernumerary connections.

Moreover, the quality of a FE mesh can be also evaluated according to the shape of its elements. The aim is to obtain a shape as regular as possible. To this end, some geometrical criteria, such as equilaterality and size constraints, can be used. This subject will be deepened at section 1.8.

Some hypotheses on the mechanical behaviour of the component may lead to the idealization of some areas, e.g. shell, beam, plate, solid. The domain of the resulting shape model is then defined as a set of sub-domains  $\Omega_i$ , where  $\Omega = \bigcup_i \Omega_i$ , having different manifold dimensions according to their prescribed mechanical behaviour and to their geometry. Therefore, the domain  $\Omega$  to be meshed is a non-manifold model, and its topology may include geometric entities that are connected or not together. This will result in a hybrid FE mesh.

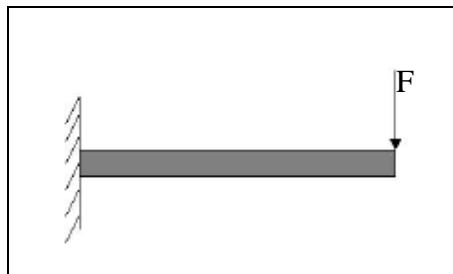
In the present work, models under focus will be formed either by triangles when considering surface domains, or by tetrahedrons when considering volume domains.

#### 1.4.4 Boundary Conditions

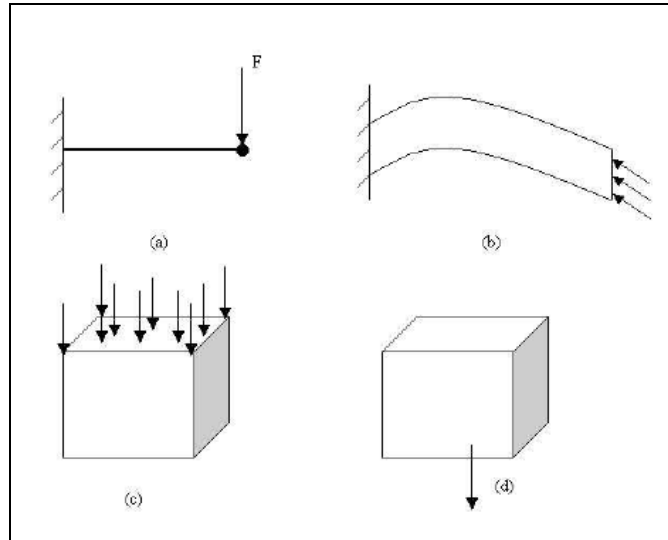
As introduced at the beginning of section 1.4, a simulation model, as reference model of the behaviour analysis PV, includes complementary information besides the shape model of the domain  $\Omega$  of study, e.g. the materials parameters, the analysis procedure and the hypotheses about the object behaviour.

Most of this complementary information is represented by **Boundary Conditions (BCs)**. BCs are a set of additional restraints that contribute to set the FE problem together with the system of partial differential equations expressing the equilibrium of the FEs discretizing the domain  $\Omega$ . The solution of a FE problem is a solution to the system of differential equations that also satisfies the BCs. From a mechanical point of view, BCs express and model the interactions between:

- The sub-domains  $\Omega_i$  forming the complete domain  $\Omega$ , which can be either a mechanical component or an assembly. Therefore, in the case of assemblies, i.e. when the model is formed by a set of mechanical components, the functional links between components can be considered as BCs, whose specification requires specific algorithms for identifying the contact areas between each component and its neighbourhood;
- Sub-domains  $\Omega_i$  and the external environment.



**Figure 1.16:** Cantilevered beam clamped in a wall and subjected to a force F.



**Figure 1.17:** Possible forms of loads, according to the type of the geometric entity where they are applied.

Three main categories of BCs for structural analyses can be considered:

### 1. Displacements:

Displacements are set of kinematic constraints applied as translations and rotations at the boundary of sub-domains  $\Omega_i$ , i.e.  $\partial \Omega_i$ . It often happens that some subsets of the object boundary are fixed in space or have restricted movements in certain directions. To insert such BCs, constraints on the degrees of freedom of the FE nodes are prescribed over the model. Let consider as an example the cantilevered beam in Figure 1.16. Figure shows that one end of the beam is clamped into a wall. This condition can be simulated by considering as fixed the anchored section, i.e. constraining all its displacements and rotations to 0.

### 2. Loads:

Usually, FE models are associated to loading configurations, whose nature (mechanical, electrical, thermal, etc.) depends on the problem to be studied. Often, the effects of several loads have to be simultaneously taken into account during the FE problem solving. To this aim, loads are often grouped together to form a load case. There are four possible forms of loads, according to the type of the geometric entity where they are applied (see Figure 1.17):

- point loads,
- curvilinear loads,
- area loads,
- volume loads.

### 3. Mixed Boundary Conditions:

In the case of mixed BCs, both displacement and load act at  $\partial\Omega_i$ , the boundary of a sub-domain  $\Omega_i$ . If we consider the orthogonal reference frame at a point of  $\partial\Omega_i$ , the displacement and load usually do not have both a non zero component along each Cartesian direction. On the contrary, if this is the case, the displacement and load are mutually dependent.

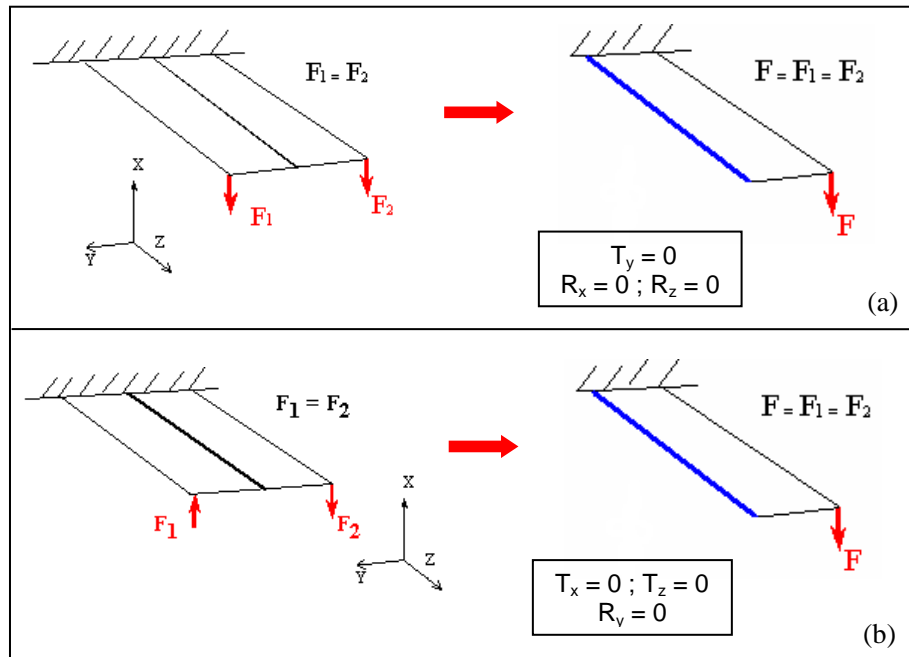
**Symmetries** are a typical example of mixed BCs. In this case, we cannot have both displacement and load with non zero components along each Cartesian direction of the orthogonal reference frame at a point of  $\partial\Omega_i$ . They are due to combined symmetries of the domain shape and the loading configuration. To take advantage of the symmetrical modelling technique, the following conditions for symmetry (or antisymmetry) must exist:

- the geometry and the material properties are symmetric;
- the loading is symmetric or antisymmetric.

Taking advantage of symmetry, we can analyze a structure or a system by modelling only a portion of it (half, quarter, eighth, etc). This allows the reduction of the FE mesh's size, and therefore the analysis run time and CPU required.

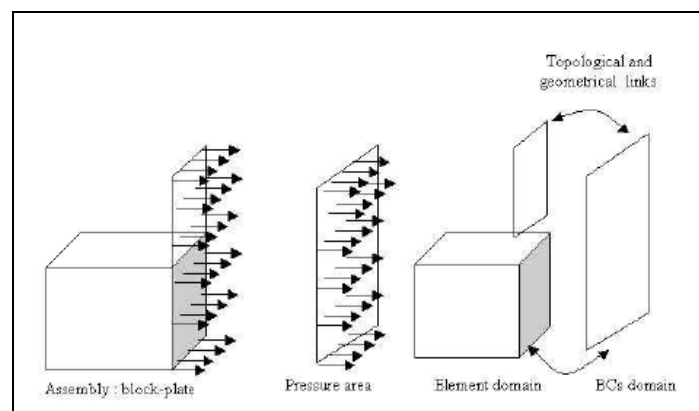
Appropriate BCs have to be applied depending on whether the loading is symmetric or antisymmetric:

- **Symmetric Load:** the loading of the model is identical on both sides of a dividing line or plane (see Figure 1.18(a)). Along the line or plane of symmetry, BCs must be applied to represent the symmetrical part as follows:
  - Out-of-plane displacement = 0
  - The two in-plane rotations = 0
- **Antisymmetric Load:** the loading of the model is oppositely balanced on the two sides of a dividing line or plane (see Figure 1.18 (b)). BCs must be applied along the line of symmetry as follows:
  - Out-of-plane rotation = 0
  - The two in-plane displacements = 0



**Figure 1.18:** Example of mixed BCs derived from symmetry: (a) The model presents symmetry both of the geometry and of the loading conditions; (b) The model presents symmetry of the geometry and antisymmetry of the loading conditions.

Each BC is applied over a geometric domain that does not always correspond to a single sub-domain  $\Omega_i$ . In fact, the location of each BC can be spread over several areas, like for example in the case showed in Figure 1.19, where the geometric domain corresponding to the BC is connected to two sub-domains  $\Omega_j$  and  $\Omega_k$ . Therefore, it could happen that the topology of BCs is of non-manifold type, as described at section 1.2.5. If we want to preserve the intrinsic meaning of the BCs, a separation between the geometric domain  $\Omega$  and the one of the BCs proves to be necessary. Therefore, to represent correctly the geometric domain of BCs, a specific geometric and topological representation is required to distinguish the shape of BCs from the component geometric domain  $\Omega$ .



**Figure 1.19:** Example of geometric domain of BCs connected to two sub-domains having different manifold dimension.



## 1.5 Process of simulation model preparation

In the previous sections, the design and the engineering analysis have been presented as two distinct PVs. As any PV, each one of them has its own reference model, i.e. design model and simulation model, formed by a suited geometric support plus some PV's specific information.

The creation of a simulation model starting from a design model is a complex process (identified as *simulation model formulation* in Figure 1.13), based on a top-down approach, which does not consist simply in the transfer and the utilization of CAD data from the design model. In what follows, we will describe in details the main steps of the preparation process of a simulation model, which also constitutes the main subject of the whole manuscript.

### 1.5.1 The interface between Design and Behaviour Simulation PVs: historical development

Until the eighties, analysts used a bottom-up process to define the analysis domain, since no CAD packages existed and therefore no shape model was available for the behaviour analysis PV. Thus, simplifications and idealisations for analysis were naturally incorporated in the simulation model during its creation. Only more recently, when CAD packages started to mature, the engineering drawings were replaced with more complex computer models capable of supporting the description of a variety of mechanical components and assemblies, the computation of their volume properties (mass, volume, surface area), simulation of mechanisms, numerically controlled machining and interference detection.

Anyway, design and analysis disciplines were still considered as independent. CAD and CAE applications were developed to serve different communities and little interfacing between the two PVs was provided for a long time, leading to a considerable gap between them. That resulted in various geometrical and topological incompatibilities [Noe94; BS96; JPB95; SBC\*00] when early trials of interoperability were made. Therefore, analysts were often forced to rebuild the model shapes in the behaviour simulation PV in order to be able to analyze their performances. As a consequence, the time required for the analysis was excessively long. In addition, the analysts intervened often late in the PDP, and therefore the only action that they were finally able to perform was to validate the design rather than contributing to the product conception.

The impressive hardware and software advances occurred in more recent years have provided significant capabilities to engineers and the integration of CAD and FE analysis systems is now improving. However, there is still a need to improve the link between CAD and FEA, making them technically closer together. Future CAE applications will have to be flexible enough for:

- Handling CAD models no matter how much complex and with different representation schemes;

- Being applicable to multi-disciplinary scenarios;
- Obtaining the simulation model in an automated and transparent way.

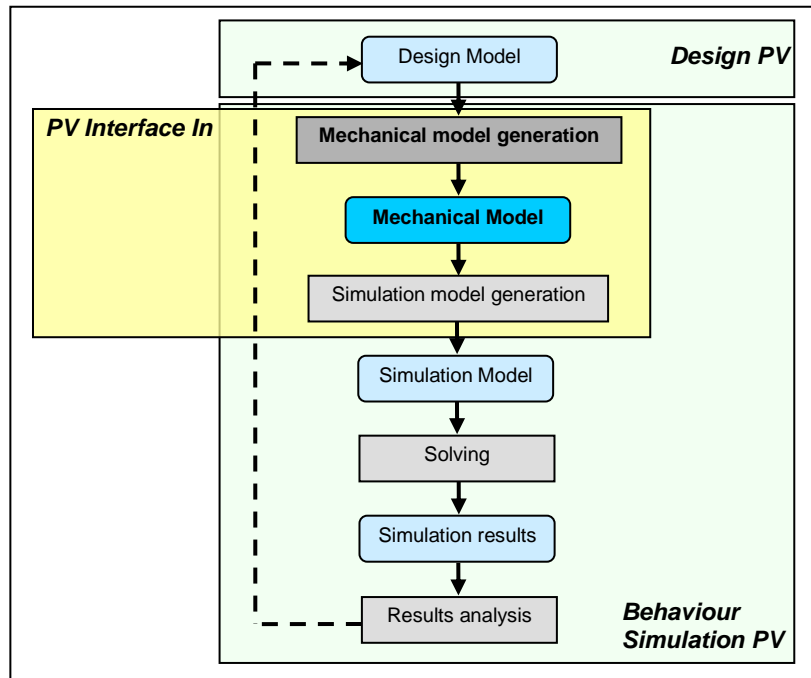
### 1.5.2 Issues in the FE model preparation process

Starting from the information contained in a design model, the engineer in charge of the analysis builds the simulation model. It will be the outcome of all the hypotheses and the objectives related to the targeted mechanical problem.

As showed in Figure 1.13, in a classical scenario of engineering analysis that follows a top-down approach, three levels of information are usually formalized: the *design model*, the *simulation model* and the *simulation results*. However, these entities do not allow an effective interpretation of the behaviour analysis object of study. Indeed, the hypotheses of the mechanical behaviour are difficult to express explicitly over the simulation model, because they are hard to distinguish from the parameters related to the software chosen for the analysis. This leads also to difficulties in characterizing and identifying improper and inconsistent hypotheses.

Then, the need of enabling the user to analyse the hypotheses performed during the generation of the simulation model emerges. To this end, the use of an intermediate model can be of great support for a better formalization of the mechanical problem object of study. In [Tro99], this intermediate model is named **Mechanical Model** (see Figure 1.20). The definition of the mechanical model consists in characterizing the shape domain of study, driven by the hypotheses that are related to the mechanical problem of analysis. In the mechanical model, the analyst incorporates most of the knowledge related to the analysis to perform. Therefore, the hypotheses about the mechanical behaviour of the model are made explicit, and can be well distinguished from those related to the method and the software used for the behaviour analysis. In addition, the mechanical model can be accessed when necessary, and constitutes a clear trace of the hypotheses that have been effectuated.

The process of generation of an intermediate mechanical model contributes to the generation of the reference model for the behaviour simulation PV, together with the actual process of simulation model preparation, i.e. FE mesh generation. It occurs at the interface between the design and behaviour simulation PVs and, according to the framework presented in Figure 1.2, it can be considered as part of the behaviour simulation PV. In the chapter 2, we will describe in more details how a proper shape model should be chosen for generating the shape domain of the mechanical model, so that it can be easily handled by the stakeholder of the behaviour simulation PV and a good integration with the successive stage of simulation model preparation is guaranteed.



**Figure 1.20:** Process of engineering analysis. At the interface between the design and behaviour simulation PV, an intermediate mechanical model is explicitly defined.

### 1.5.3 Formulation of the problem hypotheses

To perform a mechanical analysis, problem hypotheses need to be stated. The hypotheses performed are strictly linked to the mechanical behaviour of the problem of analysis, and drive the passage from the design to the simulation model. The problem hypotheses can be either traduced in the explicit formulation of the mechanical model, as stated in the previous section, or implicitly taken into account during the preparation of the simulation model but not stored.

Defining the hypotheses related to the mechanical problem to solve is a crucial and sensitive step, which influences the choices made in terms of problem modelling. The hypotheses strongly depend on the context and therefore on the objectives of the simulation, as introduced at section 1.4.1. Also the user's experience plays an important role at this crucial stage. Indeed, the user makes use of knowledge that is hard to formalize, since it is really difficult to describe the process of problem definition. This is why an explicit mechanical model may contribute to a better definition of the user's objectives [Mer98] and to a better characterization of the hypotheses. The generation of the mechanical model corresponds makes explicit the mechanical problem to solve and is therefore linked to the hypotheses made.

In [Tro99], the author identifies three main steps occurring during the generation of the mechanical model and associates to each step some categories of hypotheses that need to be formulated. Therefore, the user is provided with a process methodology that allows him/her to better manage the problem hypotheses

(see Figure 1.21). Hereafter, the process steps identified are listed, together with the detail of the associated hypotheses:

### **1. Qualitative estimation of the behaviour of the mechanical model.**

At first, the focuses on the areas considered as critical according to the simulation objectives, mentally shapes the model and “visualizes” the critical areas, the model behaviour, etc. At this stage, the definition of the important areas, i.e. the specification of the domain to be studied, is the first hypothesis made. It is based on the usage conditions stated in the objectives of the simulation (see section 1.4.1). Then, a hypothesis about the material behaviour needs to be made, which consists in specifying properties of the behaviour law of the material, e.g. homogeneity, elasticity, isotropy. Afterwards, a qualitative estimation of the interaction between the mechanical model and its environment is conducted, leading to the definition of BCs (see section 1.4.4).

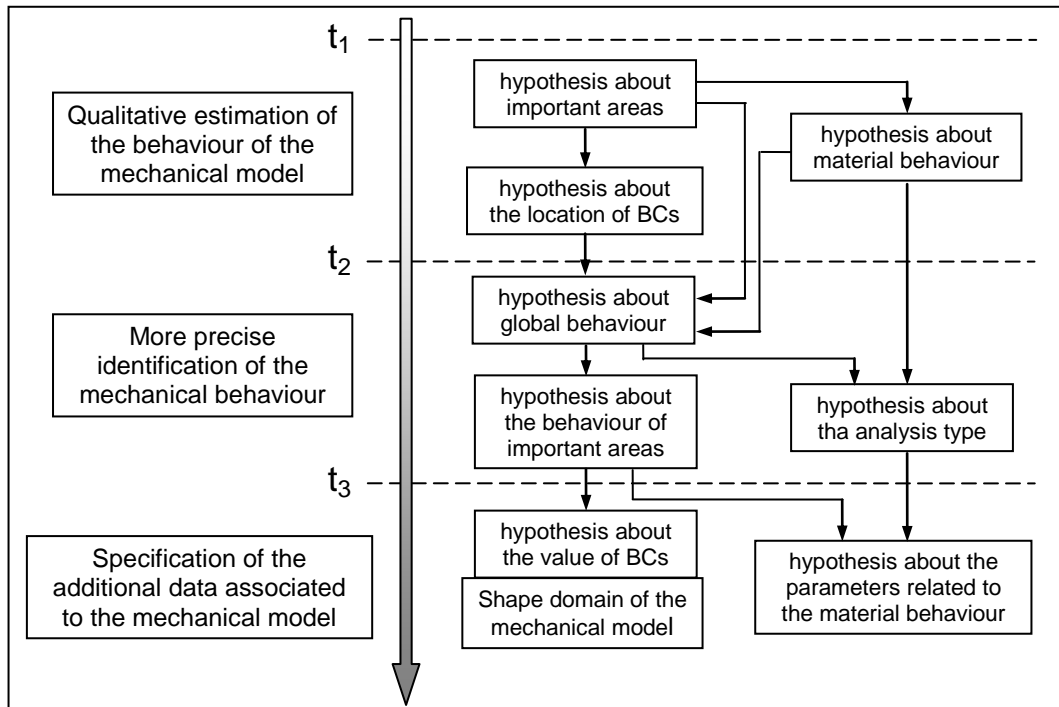
### **2. Specification of the mechanical behaviour.**

During this stage, other hypotheses are performed. The global mechanical behaviour of the component is estimated, which depends on the mechanical system under observation and on the response of the system linked to the hypothesis on the material behaviour. The hypotheses on the global mechanical behaviour influence both the hypotheses about the mechanical behaviour of the important areas of the mechanical model and the type of analysis to perform.

### **3. Specification of the additional data associated to the mechanical model.**

A hypothesis on the type and on the value of the BCs and on the corresponding geometric domain is made, and the parameters related to the material behaviour are specified. Then, according to the specific BCs defined, it is possible to define the shape domain of the mechanical model, which is a simplified and idealized version of the initial one (see section 1.5.4).

The approach above described is “process-oriented”. It provides the user with a useful methodological guide that constitutes an outline that he/she has to follow during the mechanical model preparation. Making explicit the hypotheses formulated during the construction of the mechanical model allows the user to better master the modelling choices performed and identify the sources of possible errors.

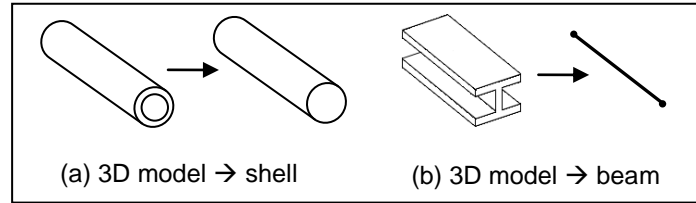


**Figure 1.21:** Process of formulation of the hypotheses driving the construction of the mechanical model, following the approach suggested in [Tro99].

A complementary approach would be a “model-oriented” one, where the different hypotheses formulated are clearly identifiable onto the mechanical model to be generated. By using this kind of approach, the problem hypotheses can be translated in constraints to be respected during the generation of the mechanical model. An example of mechanical hypotheses defined directly on the model is the map of FE sizes [Fin01]. Indeed, a map of FE sizes acts as a geometric representation of the mechanical behaviour of the component. Other examples of mechanical hypotheses embedded into the definition of shape constraints that have to be respected when moving from design to behaviour simulation PV will be given in section 1.8.2.

#### 1.5.4 Generation of the domain shape of the mechanical model

The hypotheses about the behaviour of the studied product/object are usually made explicit during the generation of the mechanical model (see section 1.5.2). From a shape point of view, a mechanical model is a simplified version of its input design model. In fact, the shape of the design model often needs to be adapted in order to obtain a simplified model where building the FE mesh. Even in the case where no explicit mechanical model is used, the FE mesh will be anyway built using a simplified shape. In this last case, the simplification of the initial design model is totally incorporated into the process of FE mesh generation. Details about the different existing approaches followed during the preparation of a FE analysis model will be given in section 1.6.



**Figure 1.22:** Examples of idealization: (a) A 3D volume model is idealized in a shell; (b) A 3D volume model is idealized in a beam.

The simplified shape has to be consistent and compatible with the subsequent FE meshing process, which will generate the simulation model, and the software chosen for solving the mechanical problem. The objectives and the rules driving the construction of the mechanical model's shape differ from those used during the design stage, even if the same type of geometrical representation (see section 1.2) may be used. The simplifications performed can be driven by two kinds of objectives:

- To express the hypotheses made about the model behaviour, related to the physics of the problem studied;
- To obtain a simpler shape that is compatible with the subsequent FE mesh generation process.

A typical kind of operation performed at this stage is the so-called idealization. **Idealization** (or dimensional reduction) of a geometric design model consists in reducing locally the degree of spatial analysis. This may involve reducing a 3D shape model to a 2D or a 1D one. Armstrong et al. [ABD\*98] defined various topological modification operations for interactive and automatic CAD model idealization. Figure 1.22 shows examples of idealizations, where 3D volume models have been idealized to 2D (Figure 1.22(a)) or 1D (Figure 1.22(b)) analysis models, which are a shell and a beam respectively.

Other kinds of shape adaptations can be performed on the initial design model. They allow the simplification of the object shape by suppressing some details, e.g. small holes, fillets, chamfers, located far away from areas where the stress are concentrating, and which can be therefore removed without influencing the model behaviour.

Moreover, often, the CAD model that is the input design model has some inconsistencies and therefore requires a set of modifications. These inconsistencies may be originated from various reasons. Most of the time, they are the consequence of model exchange via neutral formats such as IGES, STEP or VDA. The tolerances (tolerances of connection between faces of B-Rep models) and the parametrization depend on the modeller used. Therefore, the resulting models may not reflect the initial shape, and the subsequent meshing process may turn to be difficult. Another source of inconsistencies is due to bad choices made during the generation of entities, which lead to self-intersecting surfaces. These inconsistencies are local and difficult to detect. In [BS96], an analysis of the possible sources of these errors in an industrial context is carried out, and the necessity of an interactive tool to correct

CAD models is highlighted. The operations correcting model's inconsistencies are difficult to automate. Often, the inconsistencies need to be interpreted by the user. Moreover, the modification of a geometric entity, which is performed to correct some inconsistency, may lead to the redefinition of a large subset of the input shape. Indeed, the preservation of geometric constraints attached to the model (e.g. position, tangency, curvature) can turn out to be impossible without a modification of this model.

### 1.5.5 Generation of the simulation model

The simulation model, based on a FE mesh, is generated on an adapted shape of the initial design model. As described above, if an explicit mechanical model exists, the FE mesh is built on the domain shape of the mechanical model. The processes of mechanical model and simulation model generation will both occur in the PV Interface In between the design and the behaviour simulation PVs (see Figure 1.20). If no mechanical model has been generated, the shape adaptation is hidden to the user and is totally incorporated into the process occurring in the PV Interface In, which can be described with the Figure 1.13.

Several methods exist for discretizing the domain during the generation of a FE mesh. Their aim is to provide the users with a tool as automatic as possible, in order to limit user's interactions during the discretization phase, while assuring a good quality mesh. The most frequently methods used for meshing enumerate:

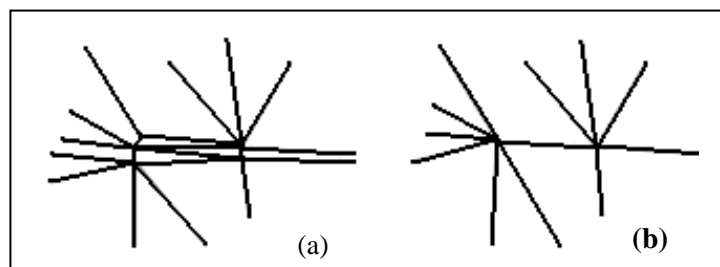
- **Spatial decomposition methods.** They consist in decomposing the shape domain in a set of cells that will be discretized for generating the FE mesh [SY84];
- **Delaunay-Voronoi methods.** The FE mesh is generated incrementally according to the Delaunay criterion. First the boundary of the shape domain is meshed, and then the internal points of the mesh are added. In addition, some optimisation operations are necessary so that the Delaunay triangulation of the boundary respects the initial boundary discretization [Geo91];
- **Frontal methods.** The meshing process begins laying FE elements on the domain boundary, and then they are propagated towards the domain interior. The various existing methods differ from the rules governing the mesh progression [Fra98; Cui98].

The accuracy of the FE analysis results strongly depends on the quality of the FE mesh. The smaller the size of the FE elements, the more accurate and reliable the results are. In fact, a compromise needs to be found between the result accuracy and the time required for the problem resolution. This is the reason why the input shape is adapted (see section 1.4.3). Small shape details are removed, which would require the use of small FE elements, while they are not necessary from a mechanical point of view.

Moreover, adaptations can be performed directly on the FE mesh. Basically, adaptations are performed by refining the size of elements in some predefined areas and using elements having a larger size in the other ones. The specification of FE sizes can be formalized by using a map of FE sizes,  $H(x, y, z)$ , which specifies the desired size of the elements over the shape domain  $\Omega$ . In the first place, the map of FE sizes is specified a priori, thanks to the user's experience, based on considerations about the domain shape, problem's BCs, desired accuracy, behaviour laws and material properties. Then, automatic procedures of mesh adaptations can be set up when FE results are available. Procedures used to monitor the FE mesh generation and to adapt the FE mesh will be discussed in more details in section 1.8.

During the generation of the FE mesh, the shape of the input domain needs to be preserved. This may lead to two main problems:

- a. The algorithms for meshing the domain will try to respect the shape of the model. Therefore, all the shape details (e.g. sharp edges, fillets, small holes) whose dimension is not compatible with the FE map of sizes become a constraint for the mesh generation process and make difficult the respect of quality criteria. Some approaches suggest the use of operators for locally modifying the mesh in order to suppress harmful configurations due to the underlying geometric representation. In [DSG97], the authors developed some operators suited for the identification and the removal of configurations harmful for the mesh quality, e.g. small edges, skewed elements. Figure 1.23 shows an example of such configuration, where mesh transformations are performed by means of subdivision and merging of entities, i.e. edges and nodes;
- b. During the design of a new product, each update of the shape model in the design PV may imply the construction of a new FE mesh for the behaviour simulation PV, a mesh whose discretization has to be compatible with the input shape and the mesh's quality requirements. At this purpose, François [Fra98] suggested some methods of automatic and remeshing processes, which allow the local redefinition of a FE mesh in the case where its underlying shape is modified. This approach reduces the time dedicated to the generation of the mesh associated to the new shape and improves the process of passage between the design and the behaviour simulation PVs, since it is not necessary to restart the meshing process each time that the model shape changes.



**Figure 1.23:** (a) Detail of a mesh with bad elements; (b) Operations are performed on the mesh in order to improve the quality of its elements.



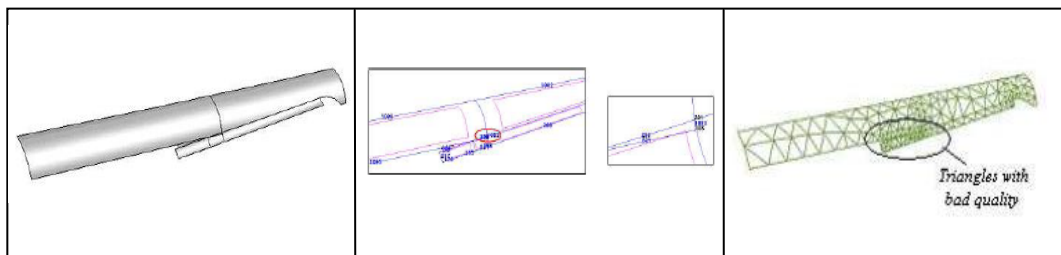
## 1.6 Existing approaches for the generation of a simulation model

We showed in the previous section that the reference model generation occurring at the interface between the design and simulation PVs is a complex process, which can be performed by using different approaches.

The first distinction that emerges is between bottom-up and top-down approaches. Section 1.5.1 showed that early techniques of behaviour analysis followed a bottom-up approach to define the domain of study, since no CAD packages existed and then no digital shape was available. Then, even when CAD packages started to mature, the lack of an efficient interface between the design and the simulation PVs led the analyst to recreate the related analysis shape from scratch in the behaviour simulation PV [ABS91].

In contrast, the approaches currently used follow a top-down approach for the definition of the simulation model, whose shape model is a FE mesh. The upstream PV is the design PV and the input model is generated in a CAD environment.

However, problems of interoperability may still subsist in the industrial field. Integrated software environments, where the CAD model and the FE mesh are created into the same environment, are currently able to satisfy a limited number of requirements of the PDP, and model exchanges are often needed. If models are exchanged between different software, the approach is efficient only when simple shapes are handled. In fact, as stated before, some problems of inconsistent topology still occur when data are exchanged between different software systems or when some shape modifications are required during the preparation of the mechanical model for the FE analysis. Figure 1.24 shows an example of typical difficulties encountered. The design model initially imported results disconnected (see Figure 1.24(a)), and it turns out to be difficult to obtain a connected component when the healing process is performed on it (see Figure 1.24 (b)). Therefore, the FE meshing process can be difficult to perform and the resulting FE mesh may have a poor quality (see Figure 1.24(c)).



**Figure 1.24:** Example of problems due to the model exchange between different software: (a) Design model initially imported, which is disconnected; (b) Difficulties in performing an efficient healing process; (c) FE mesh obtained, whose elements have a poor quality.

In the following, existing methods conform to a top-down approach will be categorized as:

- a. Methods either adapting the shape of the initial design model simultaneously with the FE mesh generation process or performing transforming directly on the FE mesh;
- b. Methods performing the domain adaptation prior to the FE mesh generation. Among them, some approaches make an explicit use of an intermediate modelling stage at the interface between the design and the behaviour simulation PVs.

### **1.6.1 Shape adaptation during or after FE mesh generation**

Several approaches aim at directly adapting a FE mesh by removing shape details after it has been generated based on the input shape model. These methods perform some topological transformation in order to improve the FE mesh quality.

In [DSG97], all small edges creating small angles are searched over the model, and if their collapse does not invalidate the mesh or reduce the dimensionality of the model, the collapse operation is allowed. Also, triangular faces or tetrahedral elements having extremely large dihedral angles are searched and collapsed over the model. In [SBO98], the authors propose a list of operations enabling to simplify the topology of a mesh model:

- The edge collapse operation collapses an edge into one of its end vertices;
- The degenerate face collapse operation collapses a face bounded by two edges into one of those edges;
- The single-edge face collapse operation reduces a face bounded by a single edge to a vertex;
- The degenerate region collapse operation reduces a volume defined by two faces into a single face;
- The single face region collapse to vertex operation collapses a volume bounded by a single face to a vertex.

Through the iterative use of these operations, small features can be removed on the mesh.

Although the process of eliminating elements from a mesh using primitive operations is well understood, these approaches have some strong limitations:

- The shape details characterising the initial shape model can severely complicate the process of FE mesh generation, which can be both time and memory intensive;

- In the cases where the desired analysis accuracy is not high, these approaches seem quite wasteful;
- They are not able to suppress details which modify the genus of the model shape, e.g. through holes. In this case, they need to be coupled with a through hole removal operator acting before the FE mesh generation;
- The modifications performed on the initial FE mesh break the link between the FE mesh and the design model, since the entities of the simplified FE mesh will no longer correspond to faces and edges of the design model. As a consequence, the information attached to this model, e.g. the type of underlying geometry, the BCs, the form features, is lost, and therefore no real integration between the models in the design and behaviour simulation PVs views is achieved.

## **1.6.2 Shape adaptation prior to FE mesh generation**

To overcome the limits identified in the previous section, some approaches adapt the input shape before the FE mesh generation. Often, the shape adaptation is performed by working in the same software environment where the design model is generated, i.e. a CAD system. Currently, two main approaches address shape editing procedures that take place before meshing. They are referred to as “hard” and “soft” approaches.

### **1.6.2.1 Form Feature approaches as an “hard” approaches**

The “hard” geometric approach is generally adopted within the industry, where tools and procedures were designed to provide the analyst with the necessary shape editing through direct shape adjustments. Thus, the underlying definition of surfaces and edges is modified to accomplish the required changes. This approach is generally based on form feature identification and suppression.

As mentioned at section 1.3, CAD systems exist that allow a user to design by features. Design-by-features systems are based on generic feature definitions that are used as templates for creating individual features [SM95; DFG98]. Then, feature instances are related to through a suitable data structure encoding the required feature relationships, e.g. a graph.

In present CAD modellers, some efforts have been made to define important features for the manufacturing PV. However, the features considered as important may change according to the considered PV, or moving from a PV to another one simplified and idealized representations of features could be required. Mapping features from one PV to another stay a difficult task, and therefore the resulting feature-based models cannot be used easily [Sha88]. Despite the limitations illustrated above, if the simplification of the initial design shape domain is performed in a CAD system, approaches exist that simplify a shape exploiting the feature information attached to the design model during its creation. They modify

the shape model through transformations of the feature tree available in the CAD modeller [VS02; JD03; LGT01]. In this way for example, blends or holes can be removed. These approaches face some problems, like:

- The lack of formalisation of simulation objectives and hypotheses. The user simplifies the design model according to his/her own experience, since only geometric information about features is provided;
- Knowledge about the CAD system is mandatory for the user, who often is the analyst and therefore has not thorough enough skills in both CAD and FEA;
- Only features explicitly created by the feature-based modelling system can be processed;
- The approach acts on the feature tree of the CAD model. Therefore, it is subjected to the corresponding dependency constraints. The modification of a feature may lead to undesired and complicated modifications over the whole design model;
- In a common industrial context, the design model containing information about form features often needs to be transferred to a downstream analysis system. However, the standard file formats currently available do not allow the information about features to be transferred, which are therefore lost;
- In the case where the analysis requires more accuracy in a given area than initially believed, changes made in the CAD system may be difficult to reverse.

When the model has not been created using a feature-based modelling system or when the construction tree, which could be useful for retrieving features, is no more available, feature identification can be performed. Various feature detection and removal techniques are found in the literature for CAE downstream applications [JC88; Kim92; SG90; TJ03; VSK01; LL02; ZM02]. Moreover, commercial software packages, as SolidWorks and Catia [Sol, Cat], allow the identification and the suppression of form features in a CAD environment, in the case where no feature list is provided. By using these approaches, volumetric features are identified via various geometric and topological tests and subsequently the feature volume is modified through extending or contracting the neighbouring faces.

In [LAP\*05], the authors propose a technique of form feature identification and removal that analyses the features tree of the CAD model and generates a new feature tree, where most of feature suppression and reinsertion operations are independent from the tree hierarchy. In this way, the features can be reinstated onto the model apart from their suppression order.

In [BBB00], the authors attempt to use a feature-based processing relying on a morphological analysis of the volume model. Then, they propose to suppress form

features corresponding to shape details and to idealize model sub-domains ascribable to plate/shell/beam areas.

As already mentioned above, relevant *features* may change according to the considered PV. For example, shape details that make difficult the FE mesh generation process are features important in the behaviour simulation PV. In this regard, in [BWS03] the authors make a distinction between *intended features* and *artefact features*. Intended features are those defined into feature-based modelling systems and hence can be manipulated only inside their original modelling system environment. In contrast, artefact features are those requiring an actual identification process, since they are either created or identified in a “not feature-based” system.

Into the category of artefact features, we can consider features introduced to maintain validity and integrity of the model, which are small with respect to the model size and can result harmful for the FE mesh generation. According to [BS96] and [JPB95], they are called *slivers*. Native models contain far fewer slivers than exchanged models. Indeed, slivers are often created by healing or repair algorithms used after a model exchange process to remove gaps and overlaps and resolve tangency conditions. Modellers and healing algorithms may also introduce a large number of faces into the model to ensure that the model is valid. This often occurs with blends and chamfers, and in areas of similar surface curvature. These additional faces may over-constrain the mesh generation. A solution to this problem is to combine faces into a single larger logical face. Another approach devoted to model healing resulting from data exchange is presented in [RBO02]. These approaches have had significant success, since most of the standard CAD exchange formats used do not contain sufficient information or may have incorrect tolerances. Geometric and topology interactive editing such as entity collapse, replacement and regeneration is useful. The results from these previous research projects have been successfully implemented in commercial products, like Cadfix [Cad] and Cimne [Cim].

Mobley et al [MCC98] propose an object-oriented approach for an automatic defeaturing of CAD models. The developed algorithm uses criteria about length, angles and curvatures to identify form features incorporated into the shape model. This approach has gained popularity and is currently used by commercial CAD systems like ProEngineer or Ideas [Pro; Ide]. However, the shape adaptations focus on entities having small dimensions, and aim at facilitating the subsequent FE meshing process, rather than adapting the model to simulation objectives and hypotheses.

Several authors [ARM\*95; PSB95, DMB\*96, SAR96] used a method based on the medial axis transform (MAT) applied to the input model as a tool to identify small features for removal. The property making the medial axis attractive to feature detection is that it has lower dimensionality than the object itself. Intuitively, the method consists in determining the “spine” of the object running along its “geometric middle”. The “spine” is an alternative representation of an object that is capable of providing all the information contained in the object. For a 2D object the medial object consists of a union of curves, while for a 3D object it is a set of

surfaces. In this latter case, we speak more precisely of Medial Surface Transform (MST) [Arm94].

MAT techniques can be useful for retrieving features such as constrictions, holes and concavities, and are also used for performing idealizations on a model by means of dimensional reductions [RT95; CS04]. In particular in [Rez96], the authors propose a mid-surface abstraction approach which is more suited to idealization for FE purposes.

Anyway, despite the existence of MST techniques [Arm94], this approach stays more adapt to be used on 2D models.

However, all the approaches based on form features above described are able to manage a limited number of predefined features, and the interactions among features remains a major issue. Even identified features are often difficult to suppress without impact and modifications over the whole geometric domain.

Moreover, as introduced above, these approaches do not take into account mechanical hypotheses and objectives. Feature suppression operators are solely based on geometric evaluations, and besides do not guarantee that the object's boundary decomposition can be directly used for FE meshing.

### **1.6.2.2 Virtual topology approaches as a “soft” approach**

Hard geometric approaches generally require the comprehensive understanding and definition of the geometries to edit, and therefore are generally computationally expensive. To overcome the computational expense and lack of generality, soft geometric editing approaches have been introduced, generally referred to virtual topology approaches. Instead of directly manipulating the mathematical definition of the shape geometry, the focus is shifted to the modification of the model topology in order to create a simpler topological model for analysis, while preserving its original geometry. Therefore, virtual topology operators act on the curvilinear space representing the object boundary.

Works based on the virtual topology have been proposed by [SBC\*00; She01; IIS\*01; Fou07]. They implement *split* and *merge* operators, based on the Euler operators described in [Man88] aimed at clustering adjacent faces into regions having similar normal vectors or matching other geometric criteria. The user is allowed to create as much as possible geometric domains free of small edges, sharp angles, highly curved areas, so generating a new topology that makes the FE mesh generation easier and more efficient.

In [Tau01], the author classified different types of features by their removal methods: the direct detail removal is performed using a virtual topology approach, while the blend removal uses also a model geometry transformation.

The virtual topology approaches are gaining support from major software vendors and are now adopted by the pre-processors of some commercial CAE software packages, such as Abaqus/CAE or Gambit [Aba; Gam].

Anyway, virtual topology approaches do not make use of topological data-structure and operators general enough for the generation of FE-models. Indeed, the B-Rep topology does not entirely satisfy mesh generation requirements: FEA often requires the representation of edges and vertices isolated in a face domain in order to model interior BCs, highly stressed features, and sharp shape features. To overcome this limitation, in [Fou07], the author aims at providing a topological data-structure (Meshing Constraint Topology, MCT) more suited to an explicit representation satisfying FE meshing requirements and better handling the diversity of topological configurations required for simulation models.

However, the approaches based on a virtual topology do not perform modifications of the model geometry. This leads to several drawbacks:

- Only the original geometry is available and if access to the geometry is required after the topology modifications, the approximations performed may compromise the operators' robustness;
- No explicit representation of the adapted model is provided, i.e. no explicit mechanical model (see section 1.5.2) exists. Therefore, a clear characterization of the hypotheses related to FE problem is not available, and it is not possible to highlight the differences between the initial and the adapted model;
- No operators have been dedicated to the suppression of topological details.

### **1.6.2.3 Layered approach**

One solution discussed is to use multi-level or layered design models. In this approach, a component is designed in progressively higher-resolution layers. Modern CAD systems like Pro/Engineer [PRO] and SolidWorks [SOL] have built-in support for a layering design. The details are removed by suppressing one or more layers. An advantage of this approach is that it reflects how component design is usually performed, starting with an abstract concept and moving toward a detailed design.

However, there are also disadvantages when these models are used in analysis:

- If the desired resolution does not closely match that of a pre-defined layer, changes of the shape model are still required;
- The layered approach does not work when the desired analysis resolution varies across the model; in this case, the fully detailed layer must be used over the entire model, again requiring the removal of details at other locations;

- Often, small details in the model are not the result of small features in feature-based modellers, but rather the outcome of the interaction between larger features. Such details cannot be removed by suppressing high-resolution layers of the CAD model;
- Often, a component is not designed from scratch; rather, the models to work with are constructed and modified over the years. These models are often difficult to convert into layered models.

#### **1.6.2.4 The polyhedral approach**

Another category of approaches exists that does not perform modifications on the CAD models, but applies the adaptation of the initial shape domain on a polyhedral model [RHG01; Fin01]. Therefore, an intermediate modelling stage is used at the interface between the design and behaviour simulation PVs.

A polyhedral representation is robust [BWS03; HC03; BS96] and allows one to easily handle and modify the shape [Fin01], by supporting local geometry transformations, topological changes and idealization processes. In addition, it is an adequate model for FE mesh generation. The operators acting on a polyhedral model can be exploited even for direct adaptations of a FE mesh, following the approach introduced in section 1.6.1 [JH02; LF05].

The polyhedral approach is based on conform models, therefore some healing or conformity set up processes [Rez96; BDK98; Ham06] could be required prior to the shape adaptation process. At this purpose, it is useful to maintain a topological link between the initial design model and its tessellated model. Such topology information could be also useful to monitor the shape changes during the simplification process, thus maintaining the consistency between the initial design model and its polyhedral representation. Besides topological information, it would be useful also to exploit the geometrical information related to the initial design model, generated in a CAD system, in order to perform high-level reasoning.

In this thesis work, we chose to profit by advantages of a polyhedral representation, extending the polyhedral operators described in [Fin01]. More details about this subject will be given in chapters 2 and 3.

### **1.7 Inaccuracies in the FE model preparation**

During all the stages of the process of FE model preparation, some inaccuracies are more or less incidentally introduced, which lead to distances between the results provided by the FE solution and the real mechanical behaviour. Therefore, it is necessary to understand the effect of the different sources of inaccuracies on the solution obtained and to provide procedures for reducing the distances between the estimated solution and the real one to a level acceptable for the analysis objectives.

In [Aia98], a distinction is made between uncertainty and error:



- **Uncertainty** is defined as a potential deficiency in any phase or activity of the modelling process that is due to the lack of knowledge;
- **Error** is defined as a recognizable deficiency in any phase or activity of modelling and simulation that is not due to lack of knowledge.

The key phrase differentiating these two definitions is the lack of knowledge. This implies that for errors the deficiency under consideration is identifiable upon examination.

Anyhow, the definition of error presented here differs from that an experimentalist may use, that is "the difference between the measured value and the exact value". In the case of behaviour simulation analyses, the exact value is indeed typically not known.

Errors can also be classified as acknowledged or unacknowledged:

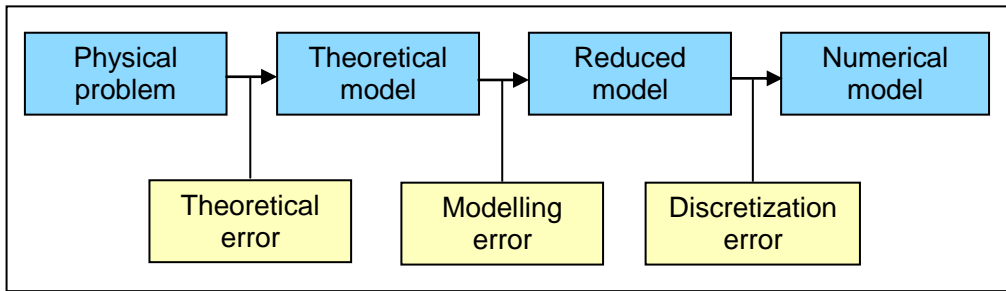
- **Acknowledged errors** (examples include errors due to bad shape simplification or discretization error) have procedures for identifying and possibly removing them.
- **Unacknowledged errors** (examples include computer programming errors) have no set procedures for finding them and may continue within the code or simulation.

Procedures exist for controlling the acknowledged errors. Several authors propose a classification of errors occurring during the process of FE model preparation and a consequent systematic approach for their control [MV97; VMB98; Sza96; KS97].

In particular in [Sza96], the process of FE model preparation is considered as decomposed in three main stages with associated some hypotheses (see Figure 1.25):

- Problem formulation, where, starting from the physical problem of analysis, a theoretical model is generated. The physical object needs to be isolated from its environment and the corresponding interactions have to be identified. The resulting theoretical model depends on the hypotheses made on the mechanical behaviour;
- Construction of a reduced model starting from the mathematical model, in order to simplify the behaviour of the object assumed in the theoretical model. A set of attributes related to the physical object are specified on the reduced model. The reduced model corresponds to the mechanical model introduced in section 1.5.2;
- Construction of a numerical model from the reduced one, and then solving of this numerical model through the resolution method. In the case where of

FE analysis, the numerical model is the simulation model, i.e. the FE mesh and additional information about BCs, materials, etc.



**Figure 1.25:** Process of simulation model preparation and its errors [Sza96].

When moving from a stage to another one different sources of errors can be introduced:

- **Theoretical errors**, linked to bad assumptions about the mechanical behaviour of the physical object, e.g. bad definition of BCs;
- **Modelling errors**, caused by the adaptations and idealizations performed during the construction of the reduced model;
- **Discretization errors** associated to the representation of the continuum model in a discrete domain of space, depending on the resolution method used (finite-difference, finite-volume, finite-element).

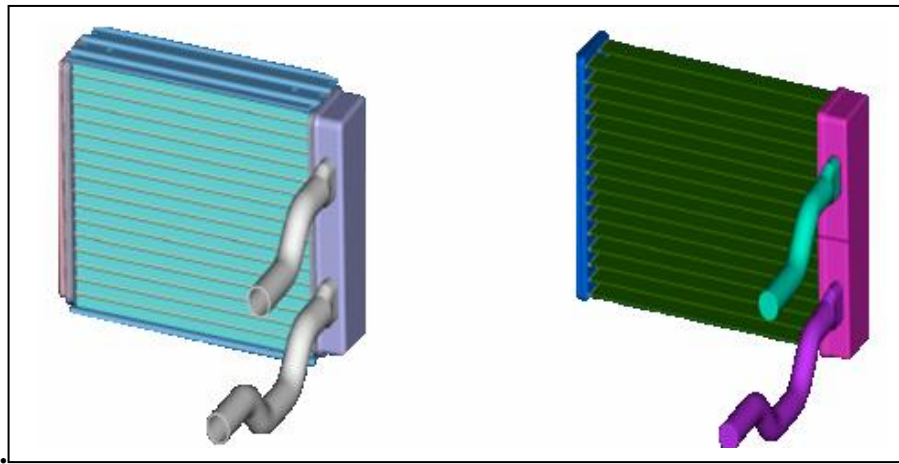
The discretization error can be defined as the difference between the exact problem solution and that one obtained by using the FE method. The discretization error vanishes as the number of FEs increases and their size decreases.

Moreover, we can distinguish between local and global errors. **Local errors** refer to errors at a grid point or cell (in the case where of FE analysis errors at a mesh node or element), whereas **global errors** refer to errors over the entire shape domain.

In the following of the manuscript, we will concentrate on modelling errors produced during the construction of the mechanical model, due to bad shape simplifications performed on the shape of the initial design model. Although methods for the control of these kinds of errors exist, as described in the next section, they are often exclusively based on expert’s knowledge. In particular in chapter 5, we will introduce the use of an a posteriori mechanical criterion able to estimate the influence of shape simplifications over global FE results.

## 1.8 Criteria supporting the process of simulation model preparation

Some criteria are used during the simulation model preparation process in order to avoid errors that may occur or at least to bind them. These criteria can help in defining the appropriate shape and FE mesh to be employed, as function of the analysis to be performed and of the desired accuracy. An optimal simulation model does not exist, since it depends on the analysis to be performed. In fact, different analysis types require different instances of the domain's shape to capture the physical behaviour of the object of study. For example, if a dynamic structural response analysis and a Computational Fluid Dynamics (CFD) analysis have to be performed on the same object, the former will require the solid geometry of the object, while the latter will need to know the geometry of the cavities through which the fluid flows. This concept is showed in Figure 1.26.



**Figure 1.26:** Example (from [BWS03]) of different shape domains related to the same design model needed for performing different kinds of FE analysis.

### 1.8.1 A priori and a posteriori criteria

Two main types of criteria exist (see Figure 1.27):

- a. **a priori** criteria, acting before solving the mechanical problem,
- b. **a posteriori** criteria, evaluated on the basis of the FE results and usually embedded in a mesh adaptation process.

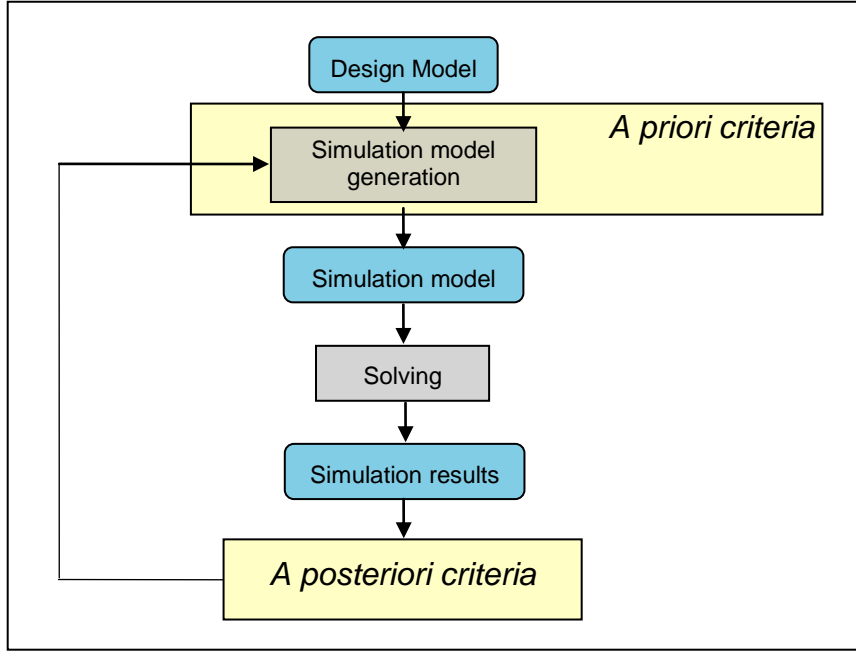
Existing a priori criteria are mainly based on the estimation of the regularity of the FEs' shape. These criteria allow a software application to qualify:

- The shape of each element. The triangles should be as much equilateral as possible, the tetrahedrons as much regular as possible. At this purpose, we could evaluate quantities as the ratio between the circle inscribed in a triangle element and the maximum length of its sides, the minimum angle defined by elements' nodes, etc;
- The size distribution of the elements in the shape domain. This distribution must involve low gradients in order to improve the shape of the elements and ensure the convergence of FE computation;
- The discretization error,  $\delta$ , evaluated as the maximal distance between a FE mesh element and the shape domain  $\Omega$ . In addition, if we consider the a priori FE map of sizes  $H(x, y, z)$  related to the domain  $\Omega$ , we can evaluate a relative discretization error by estimating the ratio between  $\delta$  and  $H(x, y, z)$ . Anyway, the definition of the map of FE sizes is left to the user's expertise. To overcome this drawback, some works have concentrated on heuristic and mechanical criteria leading to an a priori definition of the map of FE sizes [CM99; Fin01].

However, most of studies devoted to the estimation of the discretization errors, initiated in '70s, have lead to the development of a posteriori criteria. Nowadays, a posteriori criteria analyzing the discretization error are currently applied to improve the definition of a FE mesh and the quality of FE results. However, they still lack of being completely integrated into commercial software for FE analysis.

In general, the use of a posteriori criteria allows one:

- To evaluate a posteriori an approximation error made at the resolution step;
- To define an adapted map of FE sizes in order to build a FE mesh more adequate with the problem to be solved;
- To define a new degree of interpolation for each element.



**Figure 1.27:** Role of a priori and a posteriori criteria in the process flow related to an engineering analysis.

As a general remark, the posteriori criteria help in identifying a FE mesh that minimizes the difference between the FE computation results obtained and the exact solution of the initial partial derivative equations.

Let us set an example. The FE solution  $(U_h, \sigma_h)$  is an approximation of the true solution  $(U, \sigma)$ . Indeed, it satisfies the kinematical constraints and the elastic constitutive relation but the equilibrium equations are only satisfied in a weak sense. Then, the local discretization error on each point over the stress field can be assessed as:

$$e_h = \sigma - \sigma_h. \quad (1.3)$$

The local definitions of errors are in practice seldom used. Their applications are devoted to the convergence study for elasticity problems [ZZ92; BZ97], but, as regards more complex problems or problems having some singularities, these definitions are not sufficient [ZT00] and it is necessary to express a global error by using a convenient norm. The choice of the norm for expressing the global error  $\mathbf{e}_h$  depends on the problem studied and on the quantity under observation. With respect to structural analysis problems, the error is generally expressed by using an energy norm, since it is particularly suited to the estimation of the error in terms of strain energy in the structure or of forces applied to it:

$$\mathbf{e}_h = \|e_h\|_{\Omega} = \|\sigma - \sigma_h\|_{\Omega}, \quad (1.4)$$

with

$$\| \cdot \|_{\Omega} = \left[ \int_{\Omega} \cdot^T K^{-1} d\Omega \right]^{\frac{1}{2}}, \quad (1.5)$$

where  $K$  is the elasticity tensor.

In general, the exact solution is not available and is evaluated through the estimation of an error  $\mathbf{e}$ . The basic requirement for an a posteriori error estimator is that it should produce reliable estimation of the true error, because the error estimator is not only used to assess the quality of the approximate solutions, but also in an adaptive analysis procedure to guide the mesh refinement process. Moreover, the computation of the error estimator is also required to be inexpensive. The reliability of the error estimator is measured by the effectiveness index  $\gamma$ , which is the ratio of the estimated error  $\mathbf{e}$  and the true error  $\mathbf{e}_h$  in some norm:

$$\gamma = \frac{\mathbf{e}}{\mathbf{e}_h}. \quad (1.6)$$

The error estimator is considered as reliable if  $\gamma$  is close to 1, and anyway it is preferable to have a conservative estimation, i.e.  $\mathbf{e}$  should always over-estimate the true error:

$$\gamma \geq 1. \quad (1.7)$$

Another method to assess discretization errors is based on the concept of error in the constitutive relation [Lad75]. Let suppose that  $\hat{U}$  is a kinematically admissible displacement field, i.e. a displacement field verifying the kinematical constraints, and  $\hat{\sigma}$  is a statically admissible stress field, i.e. a stress field verifying the equilibrium equations. Then, the quantity:

$$\hat{e} = \hat{\sigma} - K\varepsilon(\hat{U}) \quad (1.8)$$

is called error in the constitutive relation associated to the pair  $(\hat{U}, \hat{\sigma})$ . If  $\hat{e}$  vanishes, the pair  $(\hat{U}, \hat{\sigma})$  is the solution of the mechanical problem. Otherwise,  $\hat{e}$  allows us to estimate  $(\hat{U}, \hat{\sigma})$  as an approximate solution of the problem.

To measure the global error  $\hat{e}$ , still standard energy norm over the whole structure  $\Omega$  is used:

$$\mathbf{e} = \|\hat{e}\|_{\Omega} = \hat{e} = \hat{\sigma} - K\varepsilon(\hat{U}). \quad (1.9)$$

To apply this process, a post-processing of the FE solution  $(U_h, \sigma_h)$  must be carried out in order to build an admissible displacement stress pair  $(\hat{U}, \hat{\sigma})$  from the solution  $(U_h, \sigma_h)$ . Within the framework of FE method, the displacement field  $U_h$  is kinematically admissible,

$$\hat{U} = U_h. \quad (1.10)$$

In contrast, the calculated stress  $\sigma_h$  is not statically admissible. Therefore, it is necessary to build an admissible stress field, i.e. a stress field that verifies the equilibrium equations.

The use of complementary energy formulations for obtaining a posteriori error estimates was put forward by Fraejes Veubeke in [Fra65]. However, the method failed to gain popularity, since it attempted to look for the equilibrium of the system, and was based on two different FE computations. Interest in a posteriori error estimation for FE methods really began with the pioneering work of Babuska and Rheinboldt [BR78], who introduced the **residual type of error estimator**, based on the computation of the residual of equilibrium equations as error indicator. The element residual method was then studied by many others, among whom [BR82, GKZ\*83, Kel84]. An extensive study of error residual methods is reported in the paper by Oden et al. [ODR\*89]. Later, in [AO93], Ainsworth and Oden produced extensions of the element residual method in conjunction with equilibrated boundary data. This was extended to elliptic boundary value problems, elliptic systems, variational inequalities and indefinite problems such as the Stokes problem and steady Navier-Stokes equations with small data.

A different kind of technique assessing the error estimator is based on the concept of **residual error on the constitutive relation**. As already introduced (see Eqs. (1.8-1.10)), it builds dual admissible fields  $(U_h, \sigma_h)$  to obtain error estimates. This technique was advanced by Ladeveze [Lad75] to solve linear elastic FE problems, and then extended to thermal, plastic, viscoplastic problems [LL83; LP84; CLP\*92; CLP95; GLM\*92]. Evolutions in the technique recovering statically admissible stress field can be found in [LR97].

Another category of approaches providing an a posteriori error was introduced by Zienkiewicz and Zhu [ZZ87], namely the **recovery type of error estimator**. This type of error estimator is computed by first using some recovery techniques to achieve a more accurate solution starting from the FE approximation. The recovered solution is then used in place of the exact solution to compute the error. This type of error estimator is very easy to implement and is computationally more efficient. As the recovered solution is always computed in the post-processing stage of the FE computation, there is no additional computation required in obtaining the recovered solution. Obviously, the accuracy of the error estimator depends on the accuracy of the recovered solution. Although the formation of this type of error estimator is much simpler than the residual type of error estimators, the mathematical proof of its convergence seems to be more difficult. Unlike the theory involved in the residual type of error estimator which often only provides an upper bound for the true error, the assessment of the convergence of the recovery type of error estimator requires more precise analysis, which often needs the theory of super-convergence. The theoretical work related to this type of error estimator can be found in the works of [ZZ92; WA93, BB94; Rod94; ZZ95; ZZ98]. The robustness of this type of error estimator is also dependent of the regularity of the problem and the mesh used in the approximation.

Several authors carried out research and proposed improvements about each one of these approaches. Most of the error estimates developed prior to the mid-90s

pertained to global bounds in energy norms. The late 90s saw significant advances in extending the theory to local estimates, aiming at providing error estimates and error bounds for local quantities of interest that are crucial in applications. The emphasis then shifted from the development of new techniques to the study of robustness of existing estimators and identifying limits on their performance. Particularly noteworthy in this respect is the work of Babuska et al. [BSU\*94] who conducted an extensive study of the performance and robustness of the main error estimation techniques applied to first order FE approximation. The literature on a posteriori error estimation for FE approximation is vast. For a detailed description of the existing approaches, see [AO97; Ver99; GB04].

### **1.8.2 Criteria monitoring shape simplifications**

During the last years, much work has been devoted to study the discretization error and the associated adaptive shape refinement. The criteria developed, as showed in the previous section, mainly aim at controlling and reducing this type of error. On the contrary, not much work has been dedicated to the development of criteria controlling the shape simplifications performed on a component that, as discussed in section 1.7, are another important element influencing the accuracy of a FE computation. The suppression of critical shape features on the model can lead to:

- Retain non-critical details, needlessly increasing the complexity of the FE meshing and simulation;
- Delete erroneously critical features, leading to incorrect results. This may go unnoticed during the product development since the FE mesh corresponding to the original design model is never generated or analyzed in practice.

As explained in section 1.5.4, engineers are somehow forced to simplify the shape of the initial design model prior to behaviour analysis, since building the FE mesh and then solving a simulation model that contains small geometric features would significantly increase the computational cost and reduce the efficiency of the FEA. Since the shape simplification needs to be performed during the process of simulation model generation, i.e. when the FE analysis on the simulation model has not been performed yet, the main approaches for controlling and driving the simplification process which have been developed on this topic were centred on a priori criteria. A priori criteria are typically geometric ones or are based on the user's know-how.

In [FL05], the a priori criterion used to monitor shape changes is subjective and based on the user's expertise. The user's know-how is embedded in a map of FE sizes which reflects, over the entire model, the targeted size of FE elements modelling the stress distribution or some other mechanical field. This FE map of sizes acts as a geometric representation of the mechanical behaviour of the component and can be generated by the user. He/she uses it to characterize the gradients of mechanical parameters, e.g. localising small FE where stress concentrations take place and large FE in areas where the stress values stay constant.



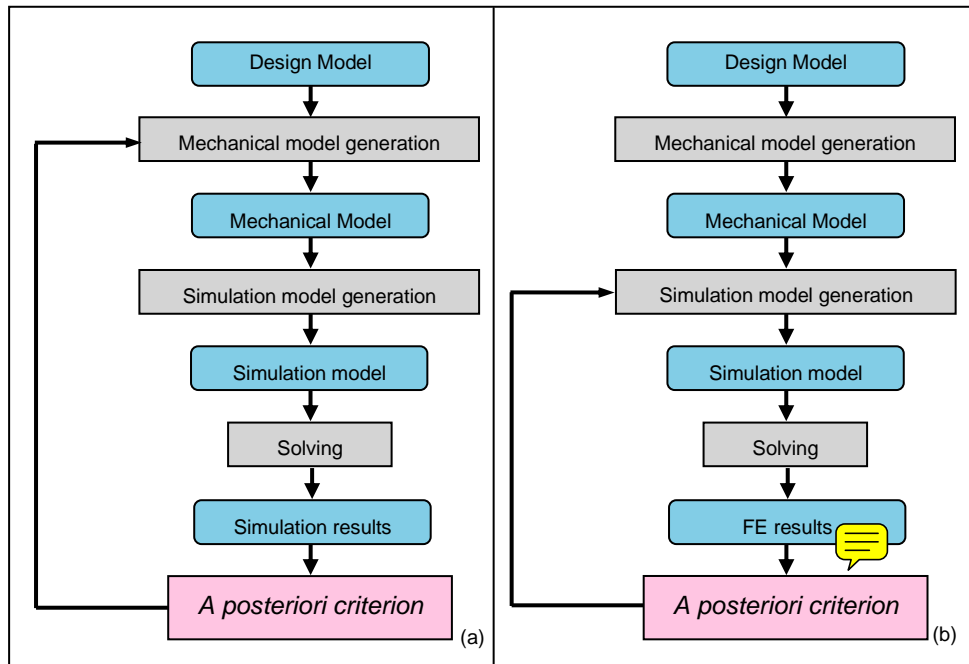
In [HLG\*04], the concept of map of FE sizes has been exploited to support the definition of simplification features. A simplification feature is defined as a form feature whose removal does not affect simulation results of the targeted mechanical problem. Here, the concept of FE map of sizes allows the user to specify whether the shape sub-domain represented by a form feature can be removed during the simplification process or not.

In [FML04], a priori objective geometry-based criteria are proposed, e.g. volume, area, centre of inertia and of gravity variations, providing an evaluation of the mechanical influence of shape simplifications performed on a shape model. In this case too, the criteria are related to volume sub-domains obtained when recovering variations between the models obtained before and after a shape simplification process.

Nevertheless, a priori criteria cannot refer to quantities obtained from the simulation results, like displacement and stress fields, and then are not able to quantify and define objectively the real influence of a shape simplification on some output parameters of the FE simulation. For example, the already mentioned FE map of sizes is a strictly subjective criterion. In the case of complex problems, it is difficult to determine areas having strong gradients, and it may be even impossible to quantify a priori the gradient's magnitude. In addition, the gradient of some mechanical parameter is only one of the factors influencing the effect of shape simplifications, i.e. some areas may have a null gradient but a large stress value, which influences FE analysis results.

Therefore, a mechanical criterion needs an a posteriori approach to be more precise and objective, where the criterion used focuses on some objective parameters, like stresses, strains or strain energy. The available simulation results can be useful to set up a criterion estimating the influence of the shape simplification performed, thus predicting the behaviour of the original design model without an explicit analysis.

The evaluation provided by an a posteriori criterion can help in readapting the simulation model. The adaptation of the model can be performed either at the level of the simulation model, i.e. directly on the FE mesh, or on the shape of the simplified model, in the case where the mechanical model is explicitly defined (see Figure 1.28).



**Figure 1.28:** Adaptation of simulation model due to an a posteriori analysis: (a) Adaptation performed at the level of mechanical model's shape domain; (b) Adaptation performed directly on the FE mesh.

Although some a posteriori approaches exist for assessing the influence of shape detail removal [FRL00, GS07], not much work has been dedicated to this topic yet. In [GS07], an interesting method was proposed for estimating errors occurring during the defeaturing of a design model. The authors developed a confidence interval for the behaviour of the initial design model so that the user is sure that the quantities of interest are within a prescribed range. Therefore in [TGS07], a direct estimation of the various quantities of interest is proposed. The methodology extends topological sensitivity method [SZ99; NFT\*03], which allows to quantify the sensitivity of a given cost function when the problem domain is perturbed by introducing a hole, applying the method to small internal and boundary features of arbitrary shapes. In other words, specific algorithms are provided to compute changes in quantities of interests when a cluster of small geometric features are deleted from a domain. However, at the moment the experiments provided are limited to 2D problems.

In the following of this manuscript, we will concentrate on the process of construction of the mechanical model and, to monitor the shape changes occurring on the design model, we will use an approach combining both a priori and a posteriori criteria [FML\*06].

## 1.9 Conclusions

The necessity of processing product shapes emerges during the activities occurring in a PDP. The concept of PV has been introduced. It is related to a task or a process of a PDP, implies the involvement of a stakeholder and relies on a PV Reference

Model, i.e. a core shape model together with some PV dependent data. The information exchange and the shape transfer process between different PVs is crucial. Dependent data related to a PV could be explicitly transferred or could help in deducing PV data and in specifying the Reference Model related to a different PV.

In this chapter, attention has been paid in particular to the information transfer and the shape modifications occurring at the interface between the design and behavior simulation PVs. Changes in terms of shape models and descriptions have been analysed, together with the need for transferring and exploiting complementary information involved in these PVs. From the performed analysis, the need for generating an intermediate model, namely the mechanical model, emerged. However, the existing approaches making use of a mechanical model, do not allow an explicit characterization of the mechanical hypotheses leading to the shape of the simulation model.

In the next chapters, we provide a possible solution to the above requirements. We describe a general framework for the generation of a mechanical model, at the interface between the design and behaviour analysis PVs. In particular, we show that the proposed framework allows the user to explicitly highlight on the shape of the mechanical model some hypotheses related to the targeted mechanical problem



## Chapter 2

### From a CAD model to the simulation model for FEA

*The need for an intermediate modelling phase, at the interface between design and behaviour simulation PVs, emerged in the analysis detailed in the first chapter. Here, we describe the key features required at this intermediate modelling stage, and, consequently, the choices we made in terms of models. The reference shape model is a polyhedral one but it is enriched with a higher-level description. The resulting mixed shape representation allows one to obtain a shape description that is closer to the user's perception. Another issue remains the capability to formalize the additional information involved and therefore efficiently process models in the interface process. Then, we propose a general framework for translating some of the problem hypotheses and objectives in terms of constraints that the shape of the mechanical model, and later in the process, of the simulation model, must satisfy. In this context, the concept of multiple topological layers is introduced, which allows us to associate additional data to the shape models. Moreover, a general formulation of mechanical criteria supporting the shape transformations is provided and the relation between the application of a criterion and the removal of a shape sub-domain is highlighted.*

---

#### 2.1 Generation of an intermediate model

In the previous chapter, we provided evidences of the benefits in using an intermediate model at the interface between the design and the behaviour simulation PVs. This helps in tracing the process of conversion of the initial design model into the behaviour simulation one. Then, a strong link and integration between the two PVs can be obtained. At the same time, the intermediate model enables to get independence both from the initial design model and the behaviour simulation one, where the latter depends on the chosen analysis tool. In addition, the intermediate model, i.e. the Mechanical Model (see Figure 1.20), is the result of all the hypotheses made about the mechanical behaviour of the product being developed.

The analysis of the preparation process of a simulation model presented in section 1.5, together with the study of the existing approaches in the literature and industry carried out at section 1.6, have led to the conclusion that a crucial issue is not only the use of an intermediate model, but also the shape transformation process

leading to its generation. Indeed, attention must be paid to the transformations performed on the intermediate model in order to guarantee a robust interface.

Initially, the intermediate model is the translation of the design model in any shape model suited for performing the shape adaptation process. The link between these two models can be addressed and exploited along the two directions:

1. Link design model-intermediate model:

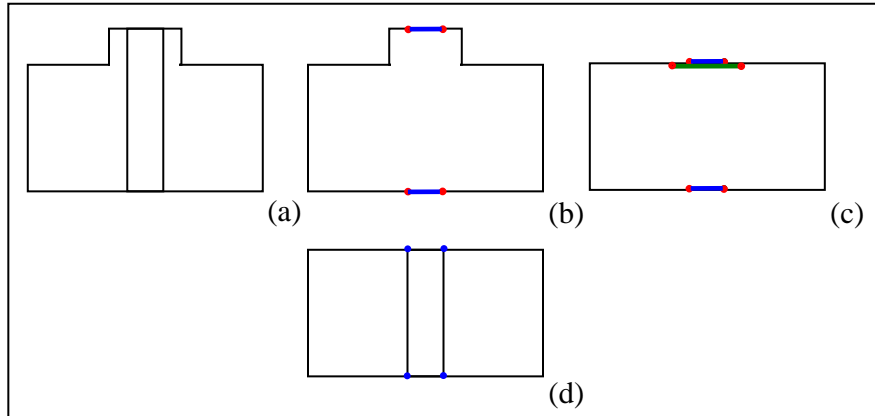
Data defined over the initial design model can be transferred and, if necessary, exploited in the shape adaptation process generating an intermediate model.

2. Link intermediate model-design model:

Data defined on the adapted domain, e.g. the one resulting from the shape modifications, could be propagated backward to the initial shape of the intermediate model, and then associated to the design model.

In this work, we will mainly concentrate on the former type of information propagation. Indeed, we are interested in the interface mechanism between the design and behaviour simulation PVs, which allows us to obtain a proper simulation model starting from a design one. Therefore, the corresponding adaptations made would be at the level either of the mechanical model or of the simulation one (see Figure 1.28).

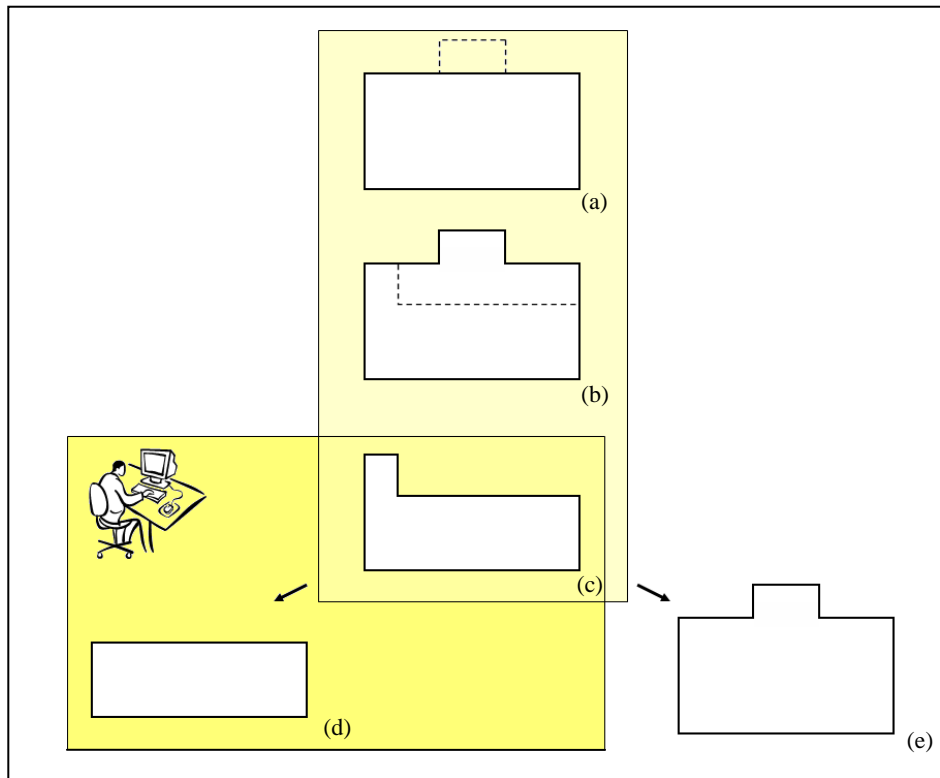
An efficient information propagation during the progressive shape modification process performed when generating of a mechanical model would be useful in many circumstances. As an example, when one wants to evaluate an alternative shape simplification from the one created, possibly at a different time, the trace of shape modifications performed over the shape domain of the mechanical model could be helpful for the model re-adaptation. A simple example is shown in Figure 2.1. Figure 2.1(a) depicts the 2D section of the initial design model. The model exhibits a protrusion and a through hole. During the shape adaptation stage, the user chooses to remove first the through hole (Figure 2.1(b)), and then the protrusion (Figure 2.1(c)). Traces of the shape modifications performed are kept. If the user needs to reinsert the through hole on the simplified domain (Figure 2.1(d)), the exact location of the hole can be retrieved thanks to the “imprint” left on the simplified model.



**Figure 2.1:** A simple example of possible information propagation during the shape modification process: (a) Initial configuration; (b) Through hole removal, maintaining its trace on the simplified model; (c) Removal of the protrusion and propagation of the information about the simplifications performed; (d) Reinsertion of the through hole based on the trace left on the simplified model.

Tracing the simplifications performed could be also useful for retrieving shape sub-domains corresponding to shape differences between different stages of the simplification process. We will demonstrate later on in the manuscript the importance of recovering shape sub-domains suppressed during a shape simplification process. Moreover, another key characteristic required for an efficient shape simplification process is the flexibility about the modification capabilities.

Current CAD software systems are not suitable to answer such needs. First, they do not offer the possibility of tracing shape modifications and recovering shape sub-domains related to differences between two stages of design modelling. The evolutions of a shape model in a CAD environment correspond to modifications of the construction history tree. At each modelling step performed by the user, a new design primitive is created and the history tree is accordingly updated. A design primitive corresponds to a modelling choice, which is selected among a finite set of categories made available to the user by the CAD system. The insertion of a design primitive on the shape model corresponds to the variation of a volume sub-domain (we only consider volume models in our analysis). The shape information about the addition or the removal of a shape sub-domain could be retrieved; but, in the current CAD systems, this task is not transparent for the user.



**Figure 2.2:** Simple 2D example of limitations of shape modifications based on the history tree: (a)(b)(c) Steps of design modelling; (d) Adapted shape domain required by the user; (e) Shape model obtained by using undo operations based on the history tree.

Next, modifications on a CAD model are feasible only when a whole design primitive is modified. As introduced in section 1.6.2.1, these design primitives can be modified through the direct access to the history tree of the object or, when it is not available, using techniques of feature identification and removal. However, possible modifications of the shape of the design model are intrinsically limited by its chronology. Indeed, the shape entities related to a modelling operation could be the reference for others and their removal could break the consistency of the overall model. Moreover, undo operations on the history tree could lead to geometrical configurations that do not correspond to the removal of a shape sub-domain that is meaningful for the simulation expert, but just come back to a previous modelling step. As an example, let us consider Figure 2.2. Modelling operations registered in the construction history tree of the CAD system are shown: first, a protrusion is added on the top of the model (see Figure 2.2(b)), and then a “cut” operation is performed (see Figure 2.2(c)). Now, let us suppose that the user’s aim is to simplify the model by removing the protrusion that is on the top-left of the shape model and thus to obtain the model shown in Figure 2.2(d). However, if he/she performs an undo operation on the construction tree, he/she will obtain the shape model depicted in Figure 2.2(e), i.e. the only configurations he/she is allowed to obtain are those related to old modelling steps.



## 2.2 Use of a polyhedral model

In the approach proposed in the present work, the intermediate modelling stage at the interface between the design and the behaviour simulation PVs takes place in a software environment using polyhedral models as reference ones.

The use of polyhedral models allows one to meet several needs that are not covered by B-Rep models commonly used in CAD systems:

- It enables to consider as input not only CAD models, which can be converted into polyhedral ones, but also digitized models or even pre-existing FE meshes. Indeed, when the input model is obtained through optical scanning or tomography devices, it consists in a point cloud and the construction of the correspondent CAD model is not trivial. The use of a polyhedral model as intermediate shape representation avoids constructing a CAD model. Data coming from the digitalisation process can be directly used and processed in order to generate a conform triangulation;
- It is a model easy to understand for all the stakeholders involved. Indeed, while simulation engineers are quite used to handle polyhedral models, they have no deep enough knowledge about CAD models processing. An extension of analysts' know-how would be therefore required if CAD models were used as intermediate models. Besides, a polyhedral model has a representation very close to surface FE meshes that the analysts manipulate during the simulation process and, if conform, can be directly used for generating a FE mesh;
- It is a model based on simple geometric entities, i.e. planar faces, whose connection is robust, i.e. it is a straight-line segment exactly defined through its two extreme vertices. Oppositely, NURBS models require complex modelling operations, and require approximations for processing and connecting patches. All these operations reduce the robustness of models, which is important to preserve when it has to be exchanged among different PVs;
- It supports local geometric and topological transformations. As highlighted at the previous point, polyhedral models have a simple mathematical definition of surfaces. Therefore, their shape can be easily modified everywhere over the component, through simple operation such as vertex removal or edge collapse. This issue has a particular importance since the shape simplification process generating a model for the FE analysis often needs to perform local modifications on the input shape.

As highlighted at the previous section, when the user deals with a CAD model together with its related representation, this task is not trivial. Indeed, local modification of the NURBS surfaces is limited according to the surface degree, and requires a certain knowledge concerning the effects of the changes of the control points on the surface. Moreover, the user is often not

the person who designed the model, and the construction tree, even if available, could not reflect his/her product understanding.

Despite all advantages listed above, a polyhedral model does not enable high-level reasoning. The operators acting on a polyhedron are simple and robust, but the basic entities and data structures of a polyhedral model are not able to embed the shape perception of the user. Moreover, if a simple polyhedral model is used as reference model [Fin01], when the input is a CAD model, no dependency/consistency is assured between the initial B-Rep NURBS coming from the design PV and its polyhedral approximation. Therefore, a polyhedral model is not sufficient to get an efficient process of simulation model preparation.

### 2.3 The mixed shape representation

To overcome the limits and needs identified in the previous section, we propose to achieve the integration process occurring in the *behaviour simulation PV Interface In* by means of the concept of ***mixed shape representation*** [LFG08]. The mixed shape representation allows one to represent simultaneously two different shape models information:

- The tessellated model, also called polyhedral model, is the master model of the mixed shape representation. In this way, we are able to accept a wider range of input models, and therefore also handle PV Interfaces In where the Upstream PV is not the design one. This lets more freedom in the organisation of a PDP and improves its efficiency by reducing the prescriptive effect of CAD-centric architecture;
- When the input model comes from a CAD system, we are able to represent the B-Rep NURBS model simultaneously to the polyhedral one. However, the CAD model is slave with regard to the polyhedral one.

Both models co-exist and are consistently maintained depending on the shape processing operators applied. The shape transformation operators act on the polyhedral model in order to be more generic and robust. In this way, some limits of CAD models, which have been identified in the first chapter and in section 2.1 are overcome, while at the same time the link with the input B-Rep NURBS model allows the exploitation of geometric and topological information of higher level than in the polyhedral model itself. Therefore, the efficiency of the shape processing operators is improved and the complexity of the shape detail identification tasks needed during the preparation of structural analysis models is reduced.

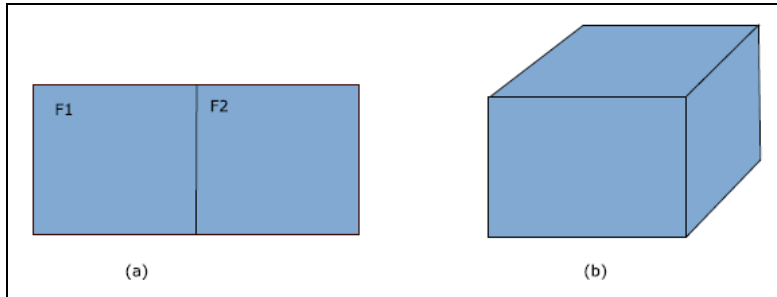
In the mixed shape representation, the link with the B-Rep NURBS model is maintained through the use of a specific data structure, introduced by Hamri in [Ham06], called **High Level Topology** (HLT). When loading a model from a CAD system, the faces and edges of its B-Rep NURBS model will coincide with a connected set of faces and edges in the HLT data structure.

### 2.3.1 HLT data structure

The HLT allows the description of general topological configurations that may be required during the process of simulation model generation. It satisfies several requirements:

- It describes the object’s topology at the level required by the user, enabling him/her to obtain any boundary decomposition that can fit the shape perception he/she has in mind;
- When the input shape model comes from a CAD system, it is able to describe the topology of the initial B-Rep NURBS model and provides a direct link to its geometric entities;
- It supports non-manifold models that can be useful during adaptation and the idealization processes (see section 1.2.5) and currently are not describable in CAD systems;
- It allows representing boundary decompositions useful for the explicit description of mechanical attributes, e.g. BCs, materials;
- It allows the description of the topology required to specify the FE meshing constraints;
- It is independent of any geometric modeller and can be linked to any CAD/CAE software without modifying the existing tools for shape representations;
- It contributes to the polyhedron conformity set up process.

In manifold B-Rep solid modelling, a component is represented by a collection of faces, edges and vertices (see section 1.2.3), while in non-manifold geometric modelling, a component (called HLT-Component in our data structure) is a cell-complex represented by a collection of volumes (3-cells), faces (2-cells), edges (1-cells), and vertices (0-cells). In our representation, we will refer to a face, an edge and a vertex as HLT-Face, HLT-Edge, HLT-Vertex, respectively. Additional conceptual entities have also been included to logically group the above elements (HLT-Loops, HLT-Regions, HLT-Shells) or define the specific element occurrences in the non-manifold configuration (HLT-CoFace, HLT-CoEdge, HLT-CoVertex).



**Figure 2.3:** Examples of HLT-shells. (a) An open HLT-Shell defined by two 2-cells connected along a common HLT-Edge; (b) a closed HLT-Shell defined by 2-cells connected along the common HLT-Edges.

For the implementation of the HLT data structure, [Ham06] used an object oriented data structure, where all the entities are derived from the abstract entity called HLT-Entity. Hereafter, some details about HLT entities are given. For more details, see [Ham06].

**HLT-Component:** It is the highest level of topological entity. It is composed of one or more HLT-Bodies.

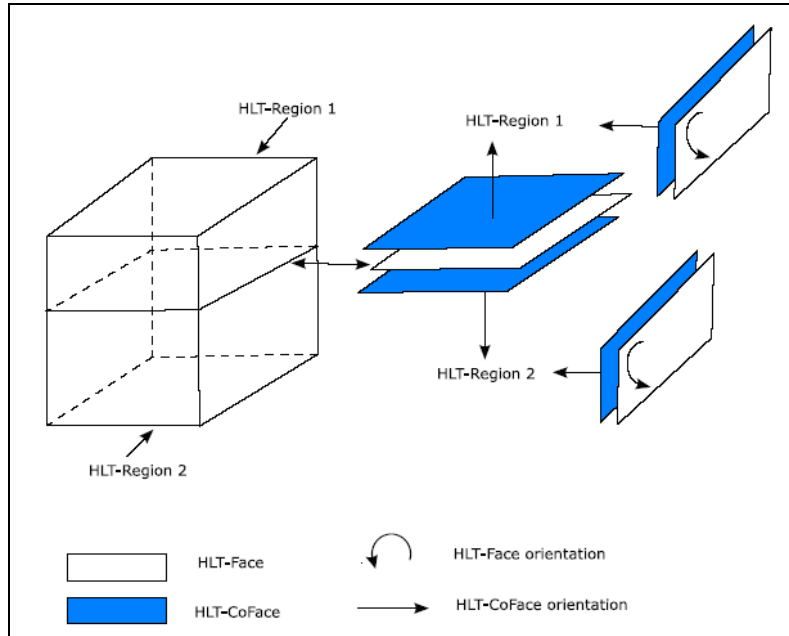
**HLT-Body:** It is composed of a set of connected  $n$ -cells, with  $0 \leq n \leq 3$ . If a HLT-Component is formed by more than one HLT-Body, the bodies must be disconnected. In this case the HLT-Component can be considered as an assembly, whose entities are HLT-Bodies. A HLT-Body is composed of one or more HLT-Regions.

**HLT-Region:** It depends on the manifold dimension of the  $n$ -cells composing the cell-complex and the Euclidean space dimension where they are immersed.

**HLT-Shell:** It is a set of 2-cells connected along edges only, and defining a 2-manifold sub-domain (see Figure 2.3) that can be either open or closed.

**HLT-Face:** It is a two dimensional manifold (2-cell). It is a bounded element of a HLT-Shell and can be geometrically represented by a surface or, if a polyhedral model is represented, by a set of connected polyhedron faces. A HLT-Face is bounded by one or more HLT-Loops.

**HLT-CoFace:** It defines the use of an HLT-Face by an HLT-Shell, thus characterizing the common area between two regions (see Figure 2.4). Indeed, a HLT-CoFace links a HLT-Face to one or two HLT-Shells, providing adjacency information for the HLT-Face and specifying the orientation of the face with respect to the HLT-Shell. For a manifold surface, one HLT-CoFace is attached to a HLT-Face, while to model a non-manifold configuration two HLT-CoFace are attached to a HLT-Face.



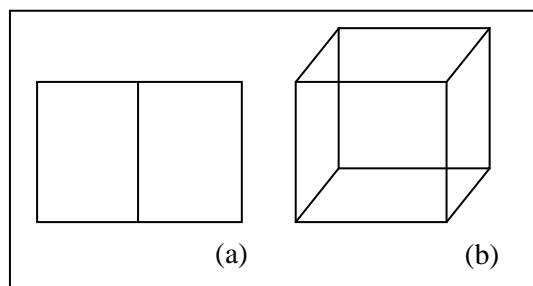
**Figure 2.4:** The HLT-CoFace concept, useful to characterize non-manifold configurations between 3-cells.

**HLT-Wire:** a set of 1-cells connected along 0-cells only to define a 1-manifold sub-domain (see Figure 2.5) that can be either open or closed.

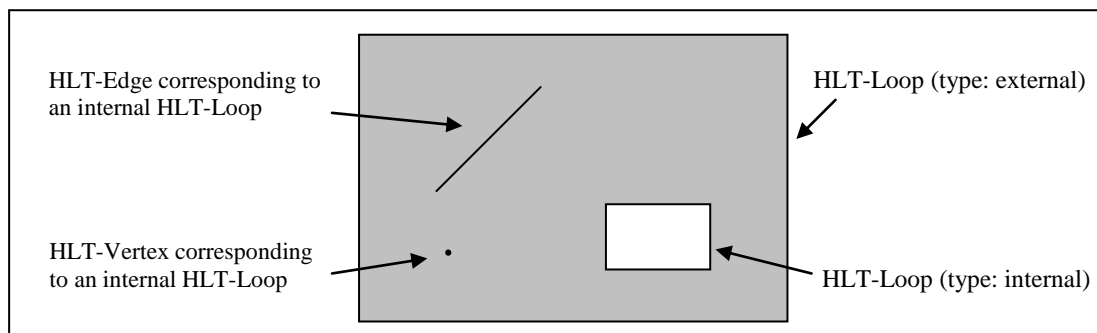
**HLT-Loop:** It is a boundary of a HLT-Face and can be either closed or open. A HLT-Face is bounded by a single external HLT-Loop, and may have multiple interior HLT-Loop. External and internal boundaries of a HLT-Face can be distinguished through attributes (see Figure 2.6). A HLT-Loop is oriented according to the orientation of the associated HLT-Face. This is equivalent to define a specific side of the HLT-Face.

**HLT-Edge:** A HLT-Edge represents the topology associated with a curve or a segment or a sequence of edges of the underlying polyhedron model. A HLT-Edge is associated to exactly one 3D curve and has a link with its geometric description. The geometric orientation of a HLT-Edge is defined as the direction from its starting point to its ending point, according to the corresponding 3D curve description. If a configuration needs to be described where an edge is immersed in a HLT-Face, the HLT-Edge will correspond to an open HLT-Loop (see Figure 2.6).

**HLT-Coedge:** A HLT-CoEdge is attached to a HLT-Edge and characterizes its use in a HLT-Loop. Indeed, a closed HLT-Loop consists of a connected sequence of HLT-CoEdges in a complete circuit, i.e. the starting vertex of the first HLT-CoEdge coincides with the ending vertex of the last HLT-CoEdge. HLT-CoEdge orientations are coded using a binary flag to determine how the HLT-Loop related to the HLT-CoEdge accesses the HLT-Edge. If the HLT-Edge is running in the same direction as the surface does, a binary flag is set to true, otherwise to false.



**Figure 2.5:** Examples of HLT-Wires: (a) a open wire, (b) a closed wire.



**Figure 2.6:** Examples of HLT-Loops related to a HLT-Face.

**HLT-Vertex:** It is a zero dimensional manifold and has a link with the geometric description of a 3D point. If the HLT-Vertex is located in the interior of a HLT-Face, it corresponds to an open HLT-Loop (see Figure 2.6), similarly to what happens with HLT-Edges. This status is mandatory to create non-manifold configurations where a vertex is isolated inside a HLT-Face. It is used to force a point to be located at that specific geometric position, e.g. as a reference for applying a force. Based on this representation, it is possible to obtain an explicit representation that is intrinsic to each of these requirements.

**HLT-Covertex:** A HLT-Covertex is attached to a HLT-Vertex and characterizes its use in a HLT-Edge and hence, depending on the configurations considered, in HLT-Faces or HLT-Wires.

### 2.3.2 How to generate the mixed shape representation

The use of the mixed shape representation enables to maintain simultaneously the shape models on which it is based, in order to take advantage of their respective benefits. A link between the two models needs to be created and appropriately updated all along the shape adaptation process.

Some preliminary operations are mandatory to get the initial version of the intermediate model, based on the mixed shape representation. The topology of the input B-Rep NURBS model is transferred to the software environment based on the mixed shape representation in order to generate the corresponding HLT data structure. The model transfer is reached by using STEP standard [STEP]. In

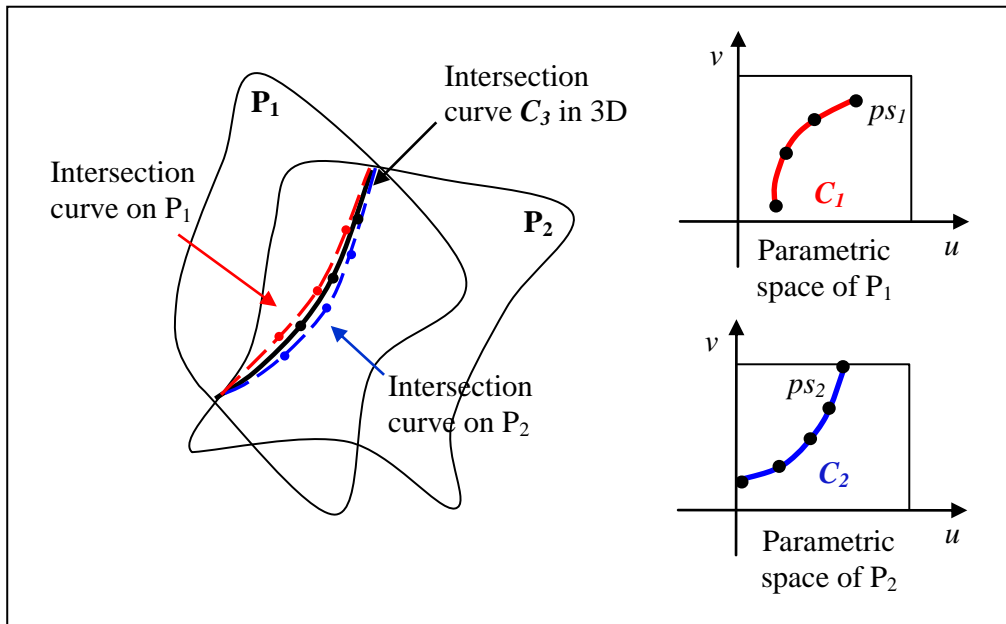
comparison with other standards, STEP incorporates the model tolerance in the exchanged file, although tests performed with current commercial software systems have revealed that these tolerances are not taken into account when a shape model is generated based on the input STEP file. Moreover, the STEP standard part 41 contains the geometric information about 3D curves generated by the intersection of patches. The advantages provided by this characteristic will be described later in this section.

### **Tessellation process**

To generate the mixed shape representation, we need to obtain the reference polyhedral model related to the input B-Rep NURBS model. The tessellation process has been included into the software environment and is independent of any CAD modeller, so that the used control parameters are suited to the process of simulation model preparation. It uses Ruppert's algorithm [Rup95; She96] and adopts an edge length criterion while avoiding degenerated triangles. However, the tessellation process needs to be controlled by the mechanical engineer in charge of the behaviour simulation to ensure that the discretization of the input CAD model is somehow compatible with the FE sizes that will be required for generating a FE mesh. To this purpose, some tessellation parameters are given as input by the user: the max edge length and the deflection between the NURBS model and the polyhedral one.

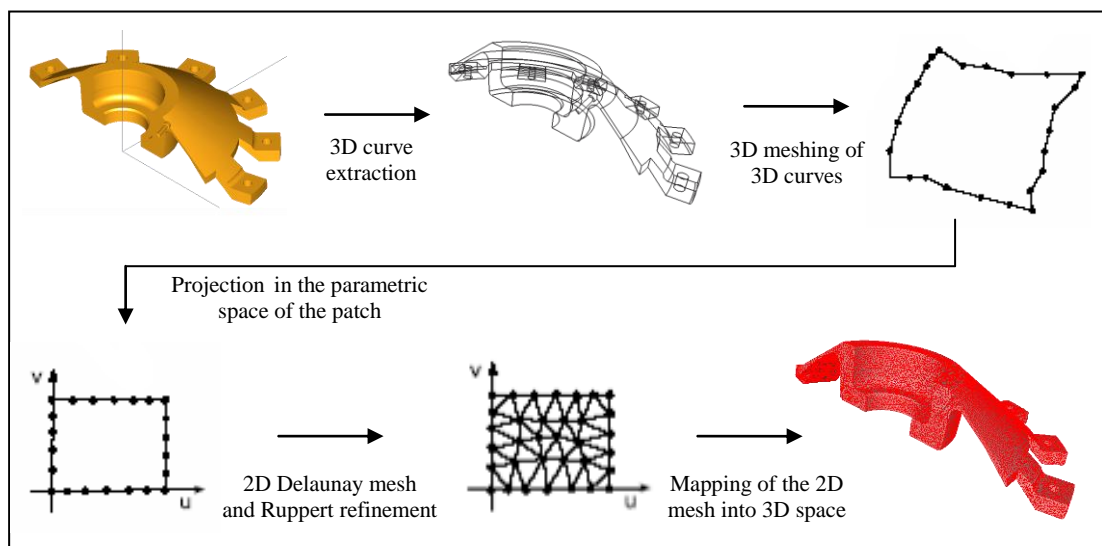
The tessellation process is performed on a patch-by-patch basis. In arbitrary patch configurations, numerical methods allow to determine a set of points  $ps$ , lying close to the theoretic 3D intersection curve between two patches  $P_1$  and  $P_2$ . Based on  $ps$ , an approximation of the intersection curve  $C$  between patches in 3D can be generated. Usually, the trimming curves  $C_i$  related to the two patches  $P_i$  differ from  $C$ , since they are estimated first by approximating  $ps$  to a point set  $ps_i$  lying in the parametric space of the patch  $P_i$ , and then by constructing the curve  $C_i$  based on the obtained point set  $ps_i$ . This leads to surface discontinuities, i.e. gaps or overlaps, across the patch boundaries. In the case of a STEP file, the information about the curve  $C$  is available, and the tessellation process is performed on the 3D boundary curve corresponding to the curve  $C$ . In this way, inaccuracies due to the initial CAD modeller are avoided. Figure 2.7 shows an example of the trimming process of two patches  $P_1$  and  $P_2$ . For more details about this topic, see [Ham06].

As a result, the boundary curves, which are shared by several faces, are meshed at first. Then, for each HLT-Face, the tessellation process is constructed on the mesh of its boundary curves. The 3D discretized contours of each face are projected into the parametric space of the face. A first constrained Delaunay triangulation [Geo91] uses as input the resulting polygonal contours. Since nodes are located on the face contours only, the resulting triangulation does not satisfy the edge length criterion yet. Hence, internal points are added until all edge lengths become smaller than the prescribed length. The triangulation of faces is performed in the parametric space of each face by using a 2D mesh refinement algorithm proposed by [Ham06], which is based on the Ruppert's algorithm. The use of the 2D mesh refinement algorithm is justified by the need for controlling the tessellation process to ensure a good quality of triangulation and reach the target edge length.



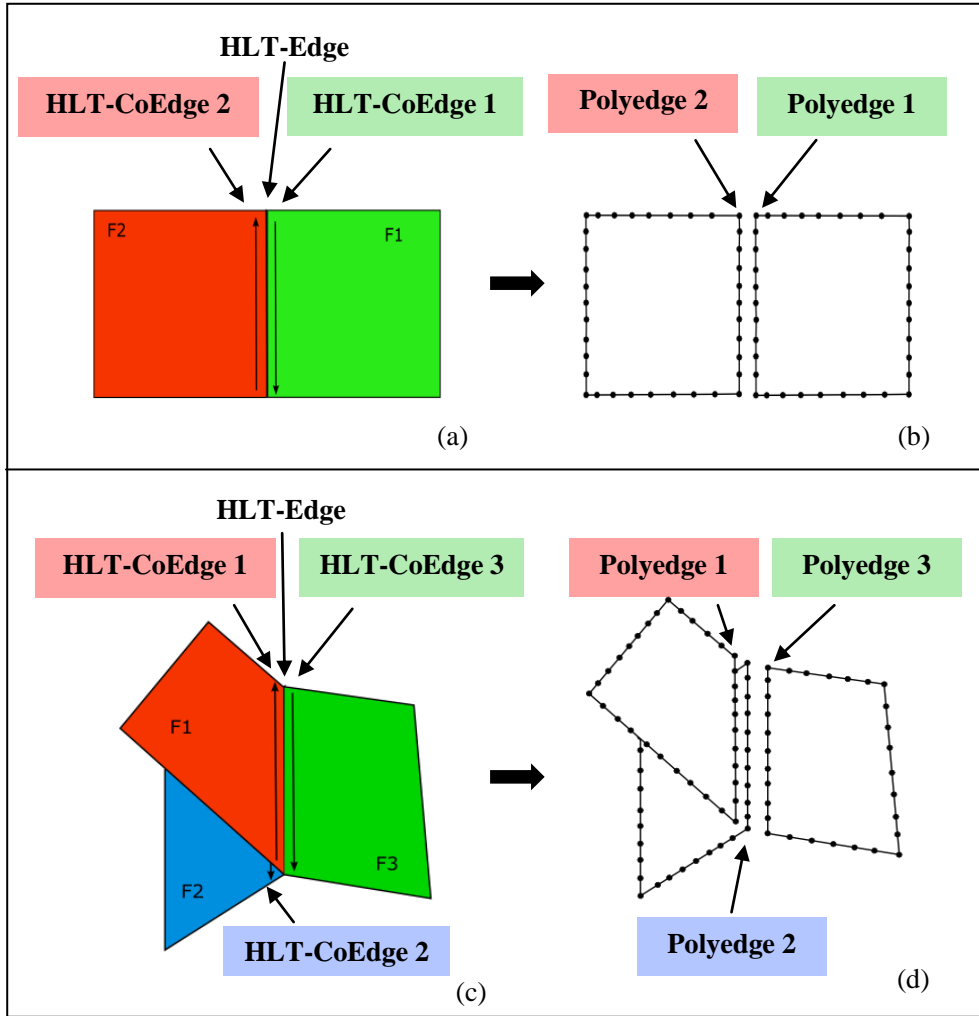
**Figure 2.7:** Trimmed patches  $P_1$  and  $P_2$  and corresponding approximations.

The tessellated 2D domain obtained with the modified Ruppert's algorithm needs then to be mapped in 3D, to produce the polyhedral model of each B-Rep face of the input STEP model. This mapping will not create degenerated triangles as long as the distortion between the parametric space and the 3D Euclidean space is not too large and there is no singularity in the mapping between the parametric and 3D Euclidean spaces. The resulting polyhedral model will be characterized by a set of disconnected partitions, each of them matching exactly one face of the B-Rep model taken as input in STEP format. An overview of the described process flow is given in Figure 2.8.



**Figure 2.8:** Mains steps of the tessellation process for a given case of study.





**Figure 2.9:** Polyedge concept: (a) Manifold configuration. The HLT-Shell consists of two adjacent HLT-Faces sharing the same HLT-Edge; (b) Two polyedges are derived from the HLT-CoEdges (1 and 2) of the manifold configuration showed in (a); (c) Non-manifold configuration. Three HLT-Faces shares the same HLT-Edge; (d) Three polyedges are derived from the HLT-CoEdges (1, 2 and 3) of the non-manifold configuration showed in (c).

To transfer the 2D triangulation from the parametric space of the face to the 3D space, we make use of two concepts, which have been proposed by [Ham06]. They are essential to maintain the link between the HLT data structure of a B-Rep NURBS model and the polyhedral model.

#### Definition 4.1

A **polyedge** is the set of edges of the polyhedron initially corresponding to the tessellation of the 3D curve representing an edge of the B-Rep NURBS model.

A polyedge is generated from its corresponding HLT-CoEdge, where a HLT-CoEdge is associated to one and only one HLT-Edge (the 3D curve). Two configurations are distinguished and shown in Figure 2.9:

- In a manifold configuration, each HLT-Edge is associated to two HLT-CoEdges, i.e. the HLT-Edge is adjacent to no more than two HLT-Faces (see Figure 2.9(a)). As a result, when we describe a volume, for each HLT-Edge exactly two polyedges are derived from their corresponding HLT-CoEdges. The polyedge is composed by a set of nodes and edges ordered according to the orientation of its reference HLT-CoEdge (see Figure 2.9(b));
- In a non-manifold configuration, a HLT-Edge can be associated to many HLT-CoEdges. In this case, for each HLT-Edge many polyedges can be derived from their corresponding HLT-CoEdges (see Figure 2.9(c)(d)).

The polyedge is linked both to the 2D tessellation, through 2D tessellation nodes described in the parametric space of the HLT-Face, and to the 3D tessellation, through 3D nodes and edges.

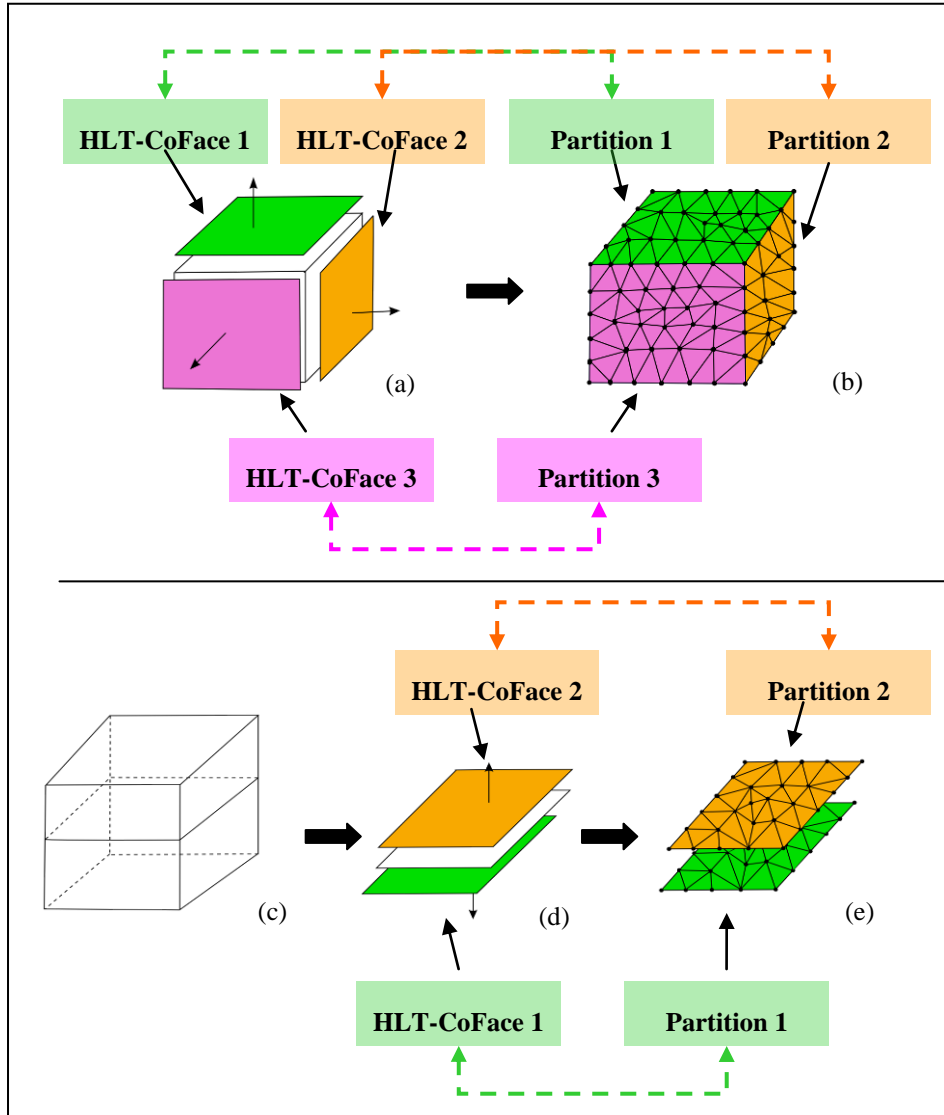
#### **Definition 4.2**

A **partition** is the set of triangles of the polyhedron initially corresponding to the discretization of a face of the B-Rep NURBS model.

A partition is generated from its corresponding HLT-CoFace, where a HLT-CoFace is associated to one and only one HLT-Face.

Two configurations are distinguished and illustrated in Figure 2.10:

- In a manifold configuration, each HLT-Face is associated to one HLT-CoFace (see Figure 2.10(a)). As a result, exactly one partition is derived from its corresponding HLT-CoFace;
- In a non-manifold configuration, a HLT-Face can be associated to no more than two HLT-CoFaces. In this case, a partition is derived from each corresponding HLT-CoFace (see Figure 2.10(b),(d)).



**Figure 2.10:** Partition concept: (a) Manifold configuration, where each HLT-Face is associated to only one HLT-CoFace; (b) Partitions derived from HLT-CoFaces; (c) non-manifold configuration, non-manifold HLT-Body; (d) a common HLT-Face associated to two HLT-CoFaces; (e) two partitions derived from two HLT-CoFaces.

The partition is linked both to the 2D triangulation, through 2D tessellation nodes described in the parametric space of the HLT-Face, and to the 3D triangulation, through 3D nodes and triangular faces. These links allow the 2D triangulation to be mapped into 3D space.

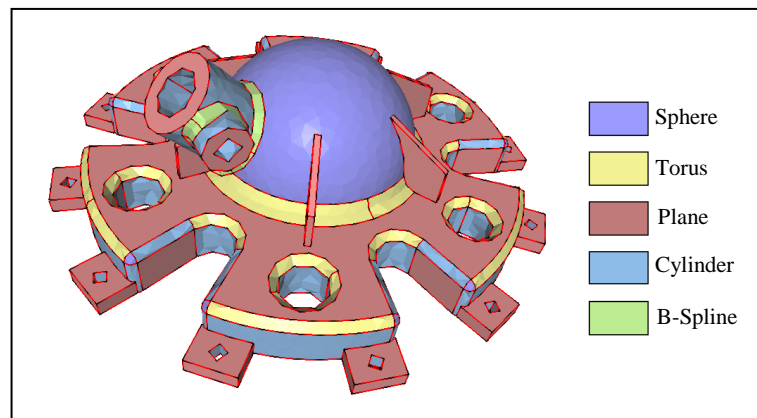
The polyhedron resulting from the tessellation process is not conform, since its tessellation has been generated patch by patch. Anyway, for each HLT-Edge, the vertices and edges located along the corresponding polyedges coincide exactly, since they are generated from the same curve  $C$ . Therefore, the polyhedron conformity can be easily reached by applying an operator that merges duplicated vertices and edges. First, we merge polyhedron vertices coinciding with HLT-Vertices, and then polyhedron vertices internal to polyedges together with their

adjacent edges. The topology of the resulting polyhedron is identical to the topology of the input B-Rep CAD model, i.e. the polyhedron and the object input produce the same genus either through the vertices, edges, triangles of the polyhedron model or through the topological entities of the B-Rep CAD model. In conclusion, the conformity set up process can be performed automatically and robustly since it is based on topological information provided by the link between the initial B-Rep topology and the polyhedron.

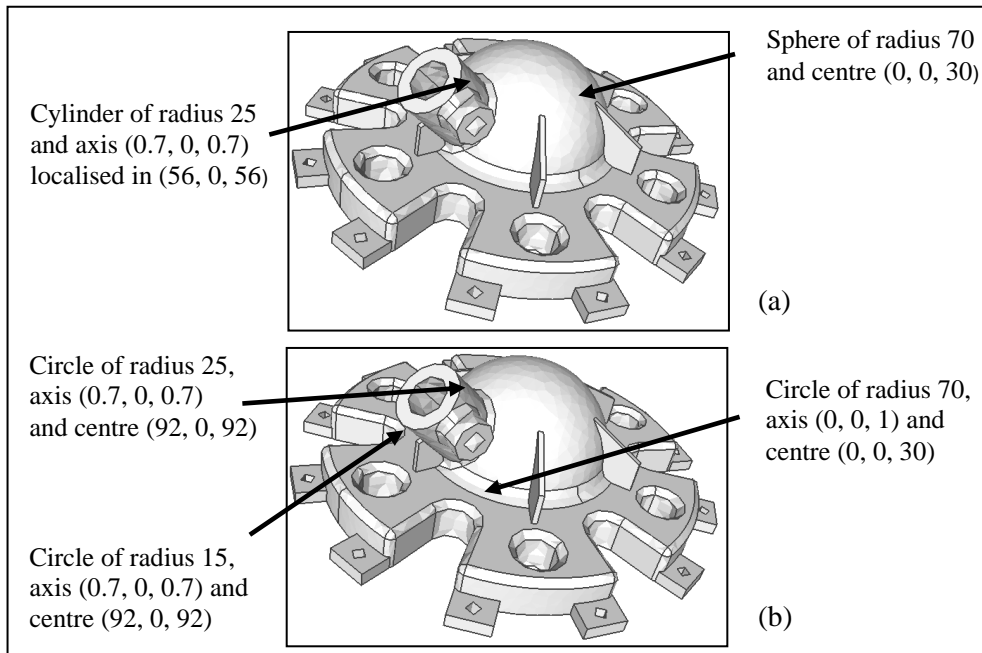
### 2.3.3 Adding semantics to the mixed shape representation

Attributes can be attached to polyedges and partitions, thus enriching the mixed shape representation with semantic data. They are either inherited by the input STEP model or inferred into the software environment based on the mixed shape representation.

In particular, data available in the STEP file about the geometric entities of the initial B-Rep NURBS model, i.e. surfaces, curves, points, are imported and directly linked to their corresponding HLT entities. Therefore, each partition keeps information about the geometric type of its corresponding HLT-Face and the related geometric parameters. At the same way, geometric data can be attached to polyedges. Figure 2.11 and Figure 2.12 show some examples. In Figure 2.11, all partitions having the same geometric type are depicted in the same colour. In Figure 2.12(a) and (b), geometric parameters, respectively associated to some partitions and to some polyedges of the model, are indicated.

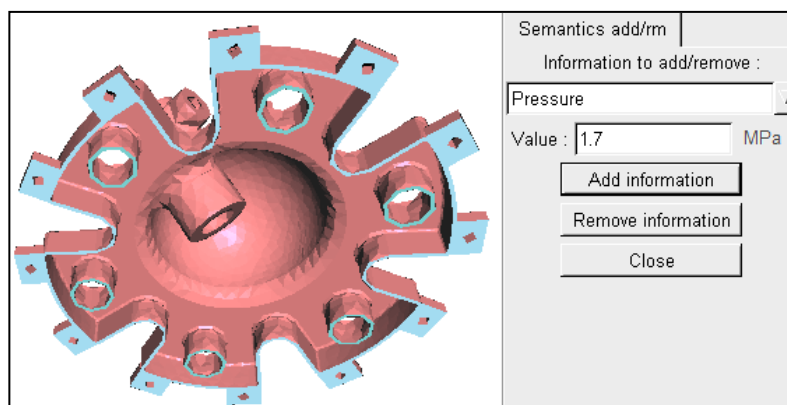


**Figure 2.11:** Geometric type of HLT-Faces associated to partitions.

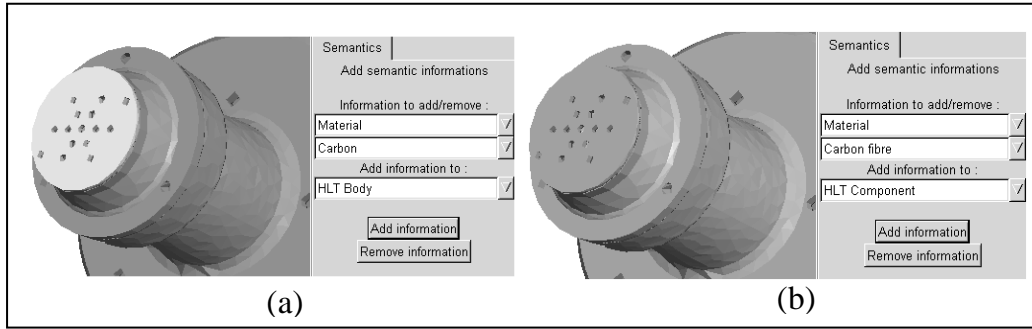


**Figure 2.12:** Example of CAD geometric information semantically enriching the shape model: (a) Geometric parameters associated to some partitions; (b) Geometric parameters associated to some polyedges.

Initially, the shape of the input CAD model matches exactly the polyhedral one. Later in the shape adaptation process, the shape of the polyhedral model will evolve in accordance with the shape transformations performed in the 3D Euclidean space, possibly not preserving the meaning of the semantic annotations. Some semantic information can be removed if no longer relevant, or on the contrary maintained where it acts as a constraint for the simplification process and/or the FE meshing process. Shape changes can be performed inside the area of a partition or across several ones. It is then important that shape modifications preserve the consistency of the model and information is transferred and propagated correctly during the successive model adaptations.



**Figure 2.13:** Interface for the insertion of pressure areas information.



**Figure 2.14:** Attaching information about material to: (a) a single HLT-Body; (b) the whole HLT Component.

Besides the semantic information directly derived from the CAD model, data enriching semantics of the mixed shape representation may be added all along the shape adaptation process. Additional information can be inferred in two ways:

- By performing some specific reasoning that exploit both levels of the mixed shape representation. Examples are the automatic identification of specific areas of the model, e.g. holes or fillets form features. This topic will be detailed in the next chapter;
- In an interactive way. This is particularly useful in the case where the information is not contained in the STEP file, and not identifiable through algorithms. According to what has been stated at section 1.1, this is typically the case of PV dependent data related to a Current PV. For example, in the behaviour simulation PV, it is possible to add data about BCs, e.g. pressure areas (see Figure 2.13) or material (see Figure 2.14), related to a specific mechanical problem.

In the software environment based on the mixed shape representation, PV dependent data, which can be either inherited by the Upstream PV or inferred in the Current PV, can be associated to shapes using the concept of **multiple topological layers** [LFC09]. Indeed, the use of the HLT data structure allows us to obtain any decomposition of the object boundary, even an arbitrary non-manifold one. Then, for each kind of information to be attached to the model, a specific boundary decomposition can be obtained and associated to a dedicated topological layer.

The first topological layer is the one initially coinciding with the boundary decomposition of the B-Rep NURBS model. Then, successive topological layers can be placed on top of the previous one in order to characterize the properties of the desired concepts, e.g. geometric support for the BCs or FE meshing constraints. These layers should be connected to each other as well as to the first one. The purpose of each topological layer is to provide an arbitrary boundary decomposition that meets the specific user's needs. This is achieved by performing transformations in the *curvilinear space* of the object, i.e. in the curvilinear domain defined by its boundary, which do not have a direct impact on the shape of the object. The operators performing transformations in the object's curvilinear space will be detailed in the next chapter.

## 2.4 Adopted criteria supporting the shape adaptation process

As introduced in the previous section, in our approach the process of shape simplification, which allows one to obtain the shape domain of the Mechanical Model, is performed in the software environment based on the mixed shape representation.

The mixed shape representation gives access to complementary information about the object shape, both geometrical and topological, which is richer than the polyhedral one and is closer to the user's perception. In summary, according to his/her needs, the user can:

- Profit by the complementary geometric information where it is useful, e.g. identifying areas of the model that have particular meaning (see section 3.4) or providing constraints for the FE meshing process (see chapter 4);
- Adapt the topology initially imported from the input B-Rep NURBS model, by generating several topological layers on the model boundary. Each topological layer allows the user to achieve an arbitrary topological description of the boundary suited to a specific objective, e.g. description of BCs or FE meshing constraints, and may become a constraint driving the process of generation of the simulation model.

Taking advantage of the potentials provided by the mixed shape representation, we have developed some operators for the adaptation of the shape domain of a component in view of a behaviour analysis.

However, the aim of a shape adaptation process is to define the model best suited for the considered mechanical analysis, and the operators based on the mixed shape representation are only partially able to take into account the mechanical problem considered. Then, as introduced in section 1.8, some additional mechanical criteria, which act either a priori or a posteriori, should be used to drive or to check the shape simplifications performed (see section 1.8.2). A good mechanical criterion must be able to correlate the geometric definition of the model and the desired accuracy of the FE analysis with regard to a set of user-defined mechanical parameters.

### 2.4.1 Nature and usage of criteria: Constraints and Indicators

In this section, we will analyse and structure different types of criteria that can be used during a shape adaptation process occurring either at the interface between the design and behaviour simulation PVs or within the behaviour simulation PV. In particular, we focus on mechanical criteria, both a priori and a posteriori ones. More generally, the definitions provided are applicable to each reference model generation process occurring in a PDP at the interface between two PVs, and to the reference model processing inside a Current PV.

A criterion used during the shape adaptation process can have two different objectives:

1. to *drive* shape adaptations. In this case, the criterion is designated as a **constraint** because it acts as a mandatory condition. A constraint may either prescribe the value of the quantity of interest, i.e. it is an equality constraint, or provide a threshold for its value, i.e. it is an inequality constraint;
2. to *check* the result of shape adaptations. In this case, we designate the criterion as an **influence indicator**. The response of an influence indicator supports the stakeholder of the considered current PV in his/her decision making process. In opposition to the point 1, the decision making process purely relies on a human being.

Two main categories of constraints can be considered:

- a. Constraints directly related to entities defining the shape model. This kind of constraints aims at distinguishing the behaviour of some model entities when an operator is applied, and therefore establishing specific treatments for them. For example, if a vertex removal operator is applied to the polyhedral model, we can specify the status of some vertices as ‘not removable’. This category of constraints is straightforwardly expressed by one or more attributes;
- b. Constraints correlating internal parameters of a shape model  $(\alpha_0, \dots, \alpha_n)$ , which are not explicitly provided with the definition of the shape model, with external parameters  $(a_0, \dots, a_n)$  regarded as meaningful for a stakeholder. Therefore, to make the constraint explicit, we need to define a function  $f$  such that:

$$(a_0, \dots, a_n) = f(\alpha_0, \dots, \alpha_n), \quad (2.1)$$

with  $a_i \leq A_i$  or  $a_i \geq A_i$ , where  $A_i$  are the threshold values specified by a stakeholder.

In this case, the aim is to find a new configuration of the initial shape model that satisfies the function  $f$  considered. Hereafter, examples of constraints belonging to this category are given, where  $Sh_i$  indicates the initial configuration of the shape model and  $Sh_j$  the new one:

- $Sh_j$  must not deviate from  $Sh_i$  more than a user-defined chordal distance  $\varepsilon$  [FL05; DKK\*05];
- $Sh_j$  must be compatible with the user-prescribed map of FE sizes to correctly describe the targeted mechanical phenomenon, i.e. the FE mesh that will be generated over  $Sh_j$  must be consistent with the user-defined map [FL05; FCF\*08];



- $Sh_j$  must not modify the global strain energy in the structure of more than  $\alpha\%$  compared to  $Sh_i$  [MF05];
- the mass of  $Sh_j$  must not differ from that of  $Sh_i$  by more than  $\beta\%$  [FML04].

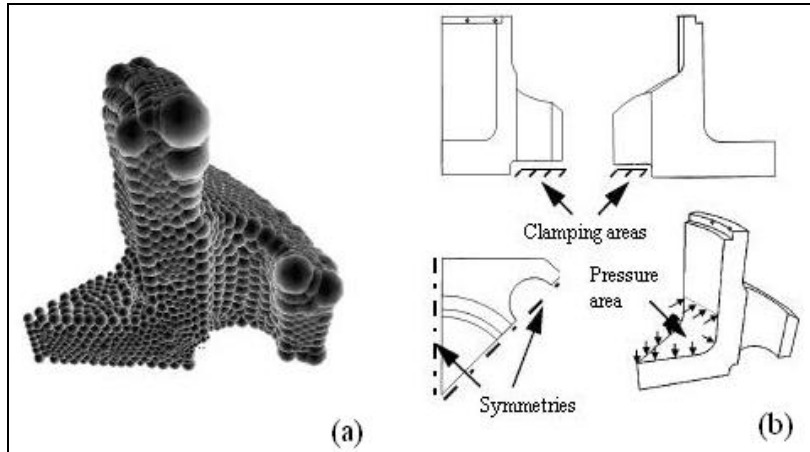
Functions  $f$  correlating the quantities of interest and the intrinsic parameters of the model can be also used for expressing an influence indicator.

## 2.4.2 A priori mechanical criteria driving the simplification process

A priori criteria support the user in making preliminary decisions about the shape simplifications to perform, i.e. these criteria are applied prior to the behaviour simulation process. As introduced at section 1.8.2, criteria used in a first stage of simplification, when no precise information about mechanical parameters is available, are typically based on the user's know-how, and are therefore subjective. If simulation objectives are clear enough and if the user has enough knowledge about the mechanical problem to be solved, the simplification process can be guided by the user's experience. As an example, suppressing shape details he/she considers as irrelevant with respect to the targeted FE analysis.

However, geometry-based criteria can be also set-up (see [FML04]). In this case, the quantities involved in the definition of the function representing the constraint are of geometric type, and therefore the evaluation provided, although if not embedding a direct mechanical meaning, is more objective than an information purely based on the user's experience.

In our approach, we have chosen to incorporate the user's knowledge about the mechanical problem to study using of a map of FE sizes, which the user aims at relying on to generate a FE mesh [FL05]. The FE map of sizes can be considered as a geometric representation characterizing the mechanical behaviour of the component and can be easily generated. This map reflects, for each area over the model, the desired size of finite elements a priori modelling some mechanical parameters. e.g. stress or displacement fields for a given FEA. It enables the user to characterize the gradients of mechanical parameters, e.g. localising small FE where stress concentrations take place, and large FE in areas where the stress values stay constant. We could also interpret the map of FE sizes as the minimum discretization level of the object geometry required for a FE analysis.



**Figure 2.15:** An example of FE map of sizes (a) related to a polyhedron to constrain the a priori simplifications with regard to the mechanical problem of interest; (b) Each local FE size is represented by a sphere.

The map of FE sizes is expressed by means of a discrete function  $H(x, y, z)$ , defined in all polyhedral vertices of the shape model based on the mixed shape representation, and therefore it acts as a constraint on the discretization of the polyhedral model during the shape adaptation process. The comparison between the target FE sizes and the size of the edges defining the input polyhedral model allows the shape transformation operators to remove more vertices in areas where a shape change is considered as not affecting analysis results. For example, where the user estimates that a stress concentration takes place, the size of the FE map will be small and shape simplifications will be hardly allowed, since they could refrain from obtaining the desired accuracy of the FE simulation.

We use the concept of map of FE sizes to generate a geometric envelope around the component. Indeed, defining the minimum level of discretization in an area of the model corresponds to establishing the maximum acceptable deviation between the initial and the simplified geometry. The total deviations all along the model represent an envelope defined around the initial geometric model. Therefore, all shape simplifications generating a polyhedron that stays inside this envelope respect the discretization constraints embedded into the map of FE sizes. In 3D, we assign a discrete representation to the envelope by using a set of spheres whose size reflects the map of FE sizes. Figure 2.15(a) shows an example of 3D discrete envelope, which is related to the mechanical problem defined in Figure 2.15(b).

### 2.4.3 Use of an a posteriori criterion

As already mentioned, the geometric envelope constraining the shape simplifications is a subjective criterion. Indeed, it cannot refer to quantities obtained from the simulation results, e.g. displacement and stress fields, and is not able to quantify and define objectively the real influence of a shape simplification on some output parameters of the FE simulation. Therefore, we cannot guarantee the desired accuracy of FE results only using the a priori map of FE sizes. In the case of complex problems, it is difficult to determine areas having strong gradients, and it is

even impossible to quantify a priori the gradient's magnitude. In addition, the gradient of some mechanical parameter is only one of the factors influencing the effect of shape simplifications, i.e. some areas may have a null gradient but a large stress value, which influences FE analysis results.

Then, the need of using an a posteriori criterion emerges. It can be applied after the behaviour simulation process, when some FE results are available. Therefore, it is based on parameters that are more objective, since directly related to the mechanical behaviour of the analysis model. In this work, we will introduce the use of an a posteriori mechanical criterion, which is able to evaluate the influence of shape simplifications on the FE results obtained in the case where of problems of linear static FE analysis or FE thermal problems for stationary linear conduction. This criterion can be applied to volume sub-domains that have been removed during a shape simplification process with the support of some a priori mechanical criteria, and whose geometric models have been stored, and evaluates the influence of their removal on FE results. It can both constraint the shape adaptation process and act just as an influence indicator. Details about the shape sub-domains storage will be given at section 3.5, while a full description of the a posteriori criterion will take place at chapter 5.

In our approach, the possible reinsertion of shape sub-domains, which the a posteriori criterion estimates as influencing the accuracy of FE results, is performed on the mechanical model, i.e. in the software environment based on the mixed shape representation.

## 2.5 Definition of simplification details

The application of a mechanical criterion is always associated to the shape variation of a volume model. Therefore, we can derive a relationship between a 3D shape variation and the application of a mechanical criterion, either a priori or a posteriori. This 3D shape variation can be characterized through the concept of shape sub-domain. Consequently, operators to identify and generate these shape sub-domains in connection with the simulation specific information become a key element of FE simulation model preparation and evaluation.

In this context, we introduced the concept of **simplification detail** [FML\*06b]: a simplification detail is a shape sub-domain that can be suppressed without influencing the mechanical behaviour of the associated model. The removal of a simplification detail stays consistent with the hypotheses related to the targeted mechanical problem and reduces the time dedicated to the meshing and solving processes, without modifying significantly FE analysis results. Therefore, the concept of simplification detail incorporates:

- A **shape meaning**, i.e. a simplification detail is a shape sub-domain.

In our approach, the shape sub-domain can be open or closed, depending on the stage of the removal process we are considering.

During the a priori shape simplification process, we will consider shape sub-domains and therefore simplification details that are open, since they correspond to sets of polyhedral faces belonging to the surface of the model and issue of the simplification.

In contrast, during the a posteriori analysis, what we need to retrieve is the differences between the initial and the simplified shape model. Therefore, at this stage of the process, simplification details will be formed by volume sub-domains corresponding to the differences between the initial model and the simplified one. When open sub-domains, simplified during the a priori shape simplification process, are available for the application of the a posteriori criterion, they need to be closed, in order to generate a volume. This concept will be detailed at section 5.3.1;

- A **mechanical meaning**, since the removal a simplification detail is not supposed to influence simulation results.

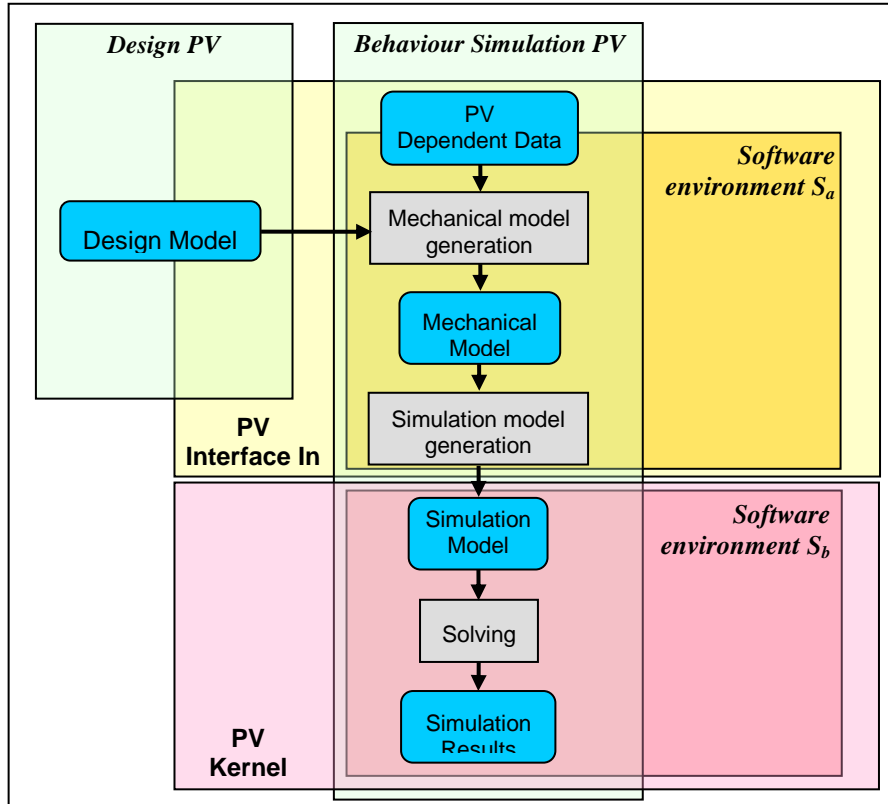
The mechanical meaning is strictly linked to the hypotheses about the mechanical behaviour of the analysis problem, and can be expressed through a mechanical criterion.

In the proposed approach, the use of a priori criteria is considered as not sufficient to get accurate FE results. Therefore, in the a priori shape adaptation we will deal with *a priori simplification details* suppressed with the help of an a priori criterion, which is generally the FE map of sizes. If the criterion used is the FE map of sizes and the shape sub-domain removed corresponds to a form feature, e.g. a hole or a fillet, the concept of a priori simplification detail is equivalent to the concept of simplification feature introduced by [HLG\*04] and described at section 1.8.2.

Then, during the a posteriori validation, the removal of a priori simplification details should be checked and therefore *actual simplification details* should be characterized. Oppositely, if the a posteriori criterion states that the shape sub-domain corresponding to an a priori simplification detail influences the FE results, it cannot be considered as a simplification detail and it has to be reinserted into the simulation model.

## 2.6 FE mesh preparation

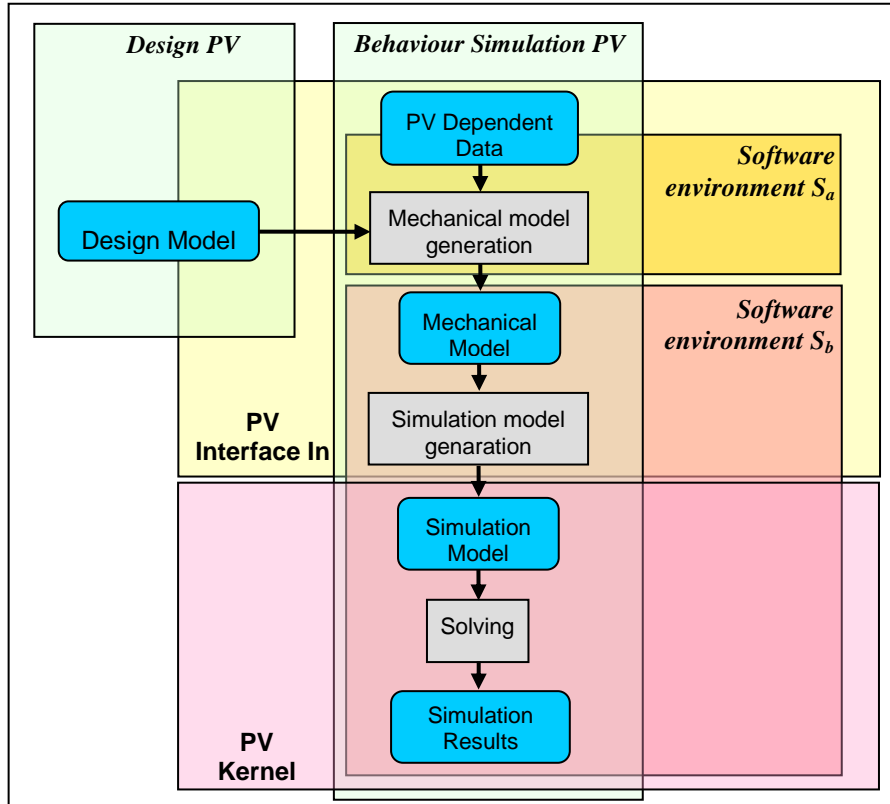
As it has been already described in section 1.5.5, once obtained the mechanical model, we need to generate the simulation model, which is based on a FE mesh. In our approach, the FE mesh generation is a step distinct from the creation of the shape model supporting the FE analysis, although both steps are part of the process of reference model generation for the behaviour simulation PV.



**Figure 2.16:** Ideal situation in model processing from a design to a behaviour simulation PV, where the change of software environment corresponds to the model transfer from the PV Interface in to the PV kernel.

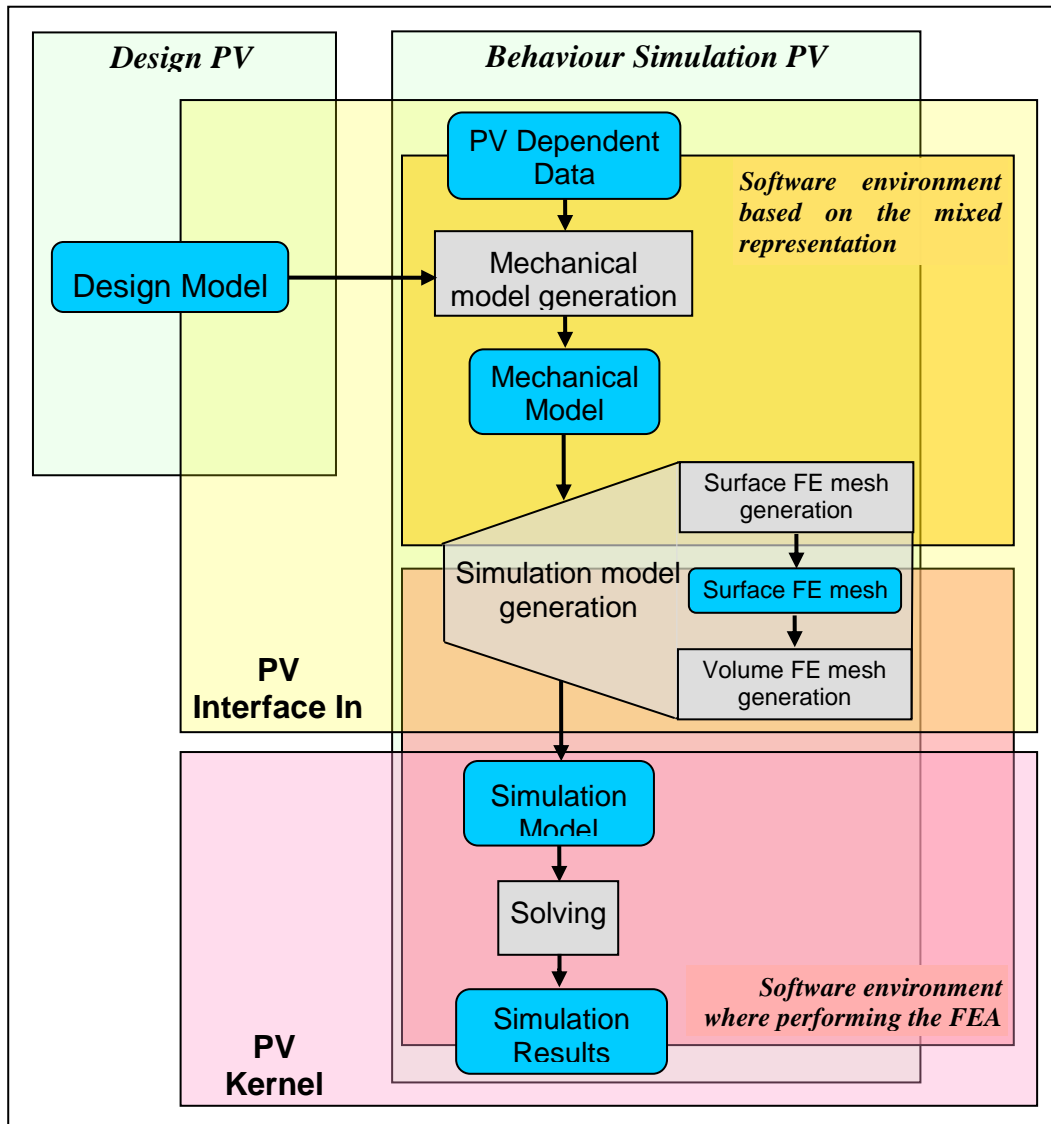
According to Figure 1.2, the whole process of reference model generation should occur within the PV Interface In. Then, in the PV kernel, we should only execute the PV task, whose input is the PV reference model. An ideal software organization should follow this scheme. Therefore, in the case of behaviour simulation PV, the FE mesh generation process should be as much as possible integrated into the behaviour simulation PV Interface In. In this way, the software environment related to the behaviour simulation PV kernel would accept as input the simulation model and would only dedicate to solve the mechanical problem with a FEA technique (see Figure 2.16).

However, in the common practice, the FE model generation process is not possible in the software environment where the mechanical model is generated. Therefore, the model has to be transferred between two different software environments at a stage of the process where the simulation model has not been defined yet. As a result, the FE mesh generation occurs in the same software environment where the FEA, which is the behaviour simulation PV task, will be performed (see Figure 2.17). The software change leads to an inevitable loss of information that could have been useful for the generation of a proper simulation model.



**Figure 2.17:** Current practice in model processing from the design to the behaviour simulation PV, where a change of software environment occurs before achieving the generation of the kernel model.

Due to the above observations, work has been performed in this direction, aiming at integrate the generation of the mechanical and simulation models into the same software environment. A process of FE surface mesh generation has been integrated into the software environment based on the mixed shape representation (see Figure 2.18). In the case where of volumetric domains, a shift between software environments is still needed for generating the 3D mesh. Anyhow, we can benefit by the possibilities offered by the software environment based on the mixed shape representation to explicitly define a topological layer defining FE meshing constraints. Therefore, a proper FE surface mesh can be generated, which the 3D mesh generation process will profit by. This subject will be further detailed in chapter 4.



**Figure 2.18:** Proposed approach, where the FE surface mesh generation is integrated into the software environment based on the mixed shape representation.





## Chapter 3

### Shape simplification operators

*The aim of this chapter is to describe the operators acting on the two levels of the mixed shape representation. Operators working on the polyhedral model modify the shape of the model in the 3D Euclidean space where the component is embedded, while operators acting in the curvilinear space of the object boundary reorganise the object domain boundary to better meet the user's needs. To improve the efficiency of the shape transformation process, these two categories of operators are combined, taking advantage of the mixed shape representation. In particular, new methods for the identification, simplification and storage of shape detail sub-domains are described, which exploit the possibilities offered by the mixed shape representation.*

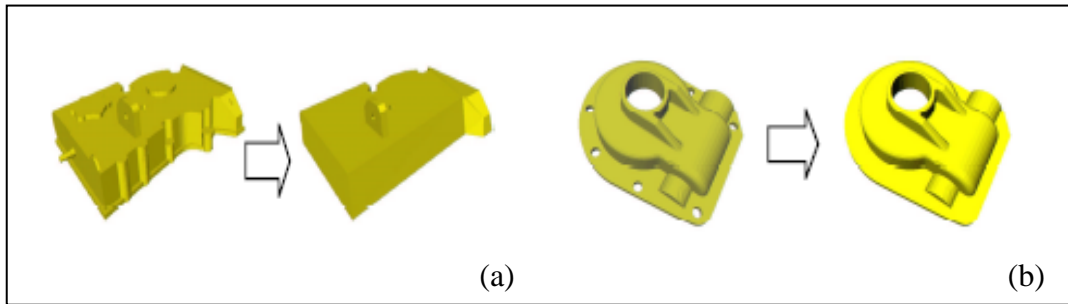
---

#### 3.1 Operators acting on the polyhedral model

As introduced in the previous chapter, the mixed shape representation consents to represent simultaneously two different shape models, i.e. polyhedral and B-Rep NURBS ones. This allows one to take advantage of both shape representations during the shape adaptation process.

To obtain robust and generic shape changes, we use simplification operators that act on the polyhedral representation [Fin01]. These operators act on the shape of a component, in the 3D Euclidean space where the component is immersed, thus modifying its volume. Two main categories of operators, illustrated in Figure 3.1, are used to transform a volume-based polyhedral model:

- **Skin detail removal operator** (Figure 3.1(a)), which is a kind of continuous deformation that changes the shape of a component without modifying its topology. This simplification operator is based on an iterative vertex removal and a local remeshing process;
- **Topological detail removal operator** (Figure 3.1(b)), which changes the topology of a component while preserving the dimension of its geometric manifold.



**Figure 3.1:** Examples of shape simplifications performed with the operators acting in 3D Euclidean space: (a) Simplifications obtained with the skin operator; (b) Simplifications obtained with the topological operator.

### 3.1.1 Skin detail removal operator

The skin detail removal operator is based on three complementary steps:

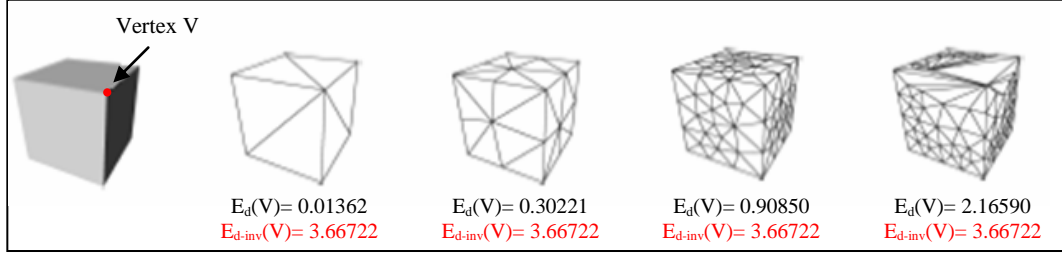
1. Selection of the vertices to remove;
2. Remeshing of the polyhedron after the vertex removal;
3. Validation of the vertex removal.

In the following, a description of these steps is provided.

#### Selection of the candidate vertex for removal

Vertices to be removed can be sorted by using different criteria, according to the targeted shape transformations. The default criterion is based on the absolute discrete curvature of vertices,  $E_d(V)$  [Fin01; LLV05], and on their corresponding invariant quantities,  $E_{d-inv}(V)$ . It allows to take into account the different local shapes that occur on the model.

Invariant quantities are preferred due to the fact they do not depend on the triangulation type, e.g. edge sizes or face angles, since the same value of curvature is provided for equivalent triangulations around a vertex. Two triangulations are equivalent at a vertex if they describe the same shape even if number, form and size of the faces around that vertex are different [GB97] (see Figure 3.2).



**Figure 3.2:** Equivalent triangulations around a vertex  $V$ . Comparison of values of the absolute discrete curvature  $E_d(V)$  and their corresponding invariant quantities  $E_{d-inv}(V)$ .

The value of the invariant discrete absolute curvature  $E_{d-inv}(V)$  of the polyhedron depends on the invariant values of the discrete Gaussian curvature,  $K_{d-inv}(V)$ , and on the discrete mean curvature,  $H_{d-inv}(V)$ , and is calculated as:

$$E_{d-inv}(V) = 4H_{d-inv}^2(V) - 2K_{d-inv}(V). \quad (3.1)$$

$K_{d-inv}(V)$  is the invariant discrete Gaussian curvature:

$$K_{d-inv}(V) = \frac{K_d(V)}{Mod(V)}, \quad (3.2)$$

where  $K_d(V)$  is the discrete Gaussian curvature. It is calculated by Eq. (3.3):

$$K_d(V) = 2\pi - \sum_i \alpha_i, \quad (3.3)$$

with  $i \in \{1, \dots, n\}$ , where  $n$  is the number of faces adjacent to  $V$  and  $\alpha_i$  is the angle of the face  $F_i$  at the vertex  $V$ .

The “module”  $Mod(V)$  is the term needed to get independence from the kind of triangulation around the vertex. It defines a corrected version of the area of the polyhedron star around the vertex  $V$  and is given by [Boi95]:

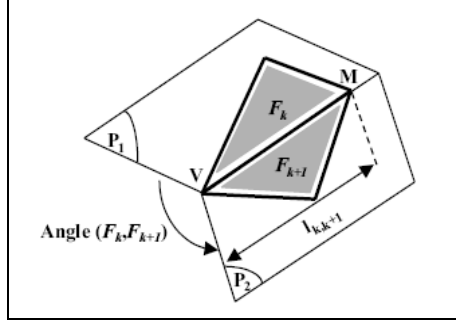
$$3Mod(V) = \frac{1}{8} \left( \sum_i (4a_i - l_i^2 \cot \alpha_i) \right). \quad (3.4)$$

Here,  $a_i$  is the area of the face  $F_i$  and  $l_i$  is the length of the edge belonging to the face  $F_i$  and opposed to  $V$ .

The value of the invariant discrete mean curvature is given by:

$$H_{d-inv}(V) = \frac{3H_d(V)}{Mod(V)}. \quad (3.5)$$

Note also in this case the use of  $Mod(V)$  to get independence from the geometric configuration of the polyhedron star around  $V$ .



**Figure 3.3:** Estimation of mean curvature  $H_{k,k+1}$  of a polyhedron edge  $VM$ .

The value of the discrete mean curvature  $H_d$  at a polyhedron vertex  $V$  is given by:

$$H_{d-inv}(V) = \frac{1}{2} \sum_k H_{k,k+1}. \quad (3.6)$$

It depends on the mean curvature  $H_{k,k+1}$  of each edge  $VM$  concurrent in  $V$ , which is given by [Bou97]:

$$H_{k,k+1} = \frac{1}{2} (\pi - \text{angle}(F_k, F_{k+1})) \cdot l_{k,k+1}, \quad (3.7)$$

$H_{k,k+1}$  is defined by the product of the angle formed by the polyhedron faces  $F_k$  and  $F_{k+1}$  that are adjacent to the edge  $VM$ , and the length  $l_{k,k+1}$  of the edge  $VM$  (see Figure 3.3).

Sorting the vertices according to a decreasing value of absolute discrete curvature produces a shape smoothing, i.e. the vertex removal rounds the corners and the sharp edges of the polyhedron. This is the classical approach used for removing shape details during the preparation of a simplified shape support where generating the FE mesh. In contrast, reverse ordering for sorting the vertices, according to an increasing value of the absolute discrete curvature, better preserves the initial shape. This is the approach classically used for visualization applications.

The vertex removal operator can be subjected to further geometric constraints, like for example:

- **Edge length:** The vertex  $V$  is removed only if the new edges created when remeshing the polyhedral star around  $V$  are shorter than a target edge length threshold value,  $l_{max}$ , which is specified by the user in the map of FE sizes. In the case where  $V$  has  $m$  adjacent polyhedron vertices, the remeshing scheme is formed by the edges  $E_i$ , and the constraints are:

$$l_i \leq l_{max}, \quad (3.8)$$

where  $i \in \{1, \dots, (m - 3)\}$ ;

- **Minimum angle value:** The vertex  $V$  is removed only if the angles associated to the created faces are larger than a user prescribed threshold value,  $\alpha_{min}$ , which targets the desired regularity of triangular elements. If  $V$  has  $m$  adjacent polyhedron vertices, the remeshing scheme is formed by the faces  $F_j$ , with internal angles  $\alpha_k^j$  that satisfy:

$$\alpha_k^j \leq \alpha_{min} , \quad (3.9)$$

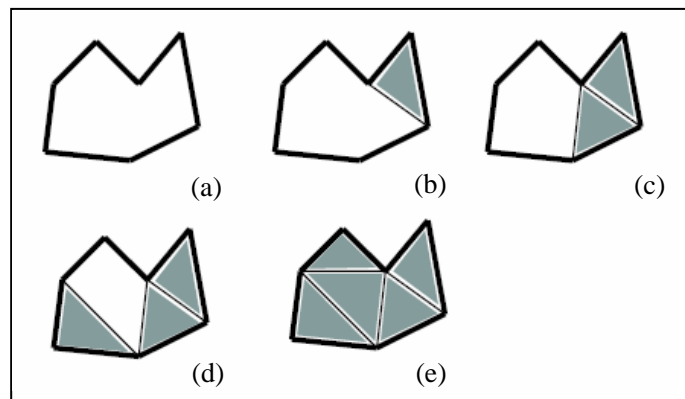
with  $k \in \{1, 2, 3\}$  and  $j \in \{1, \dots, (m - 2)\}$ .

Introducing constraints for the form and size of polyhedron entities is a common practice when the shape adaptation and the FE mesh generation are performed in the same software environment, as it will be described in chapter 4.

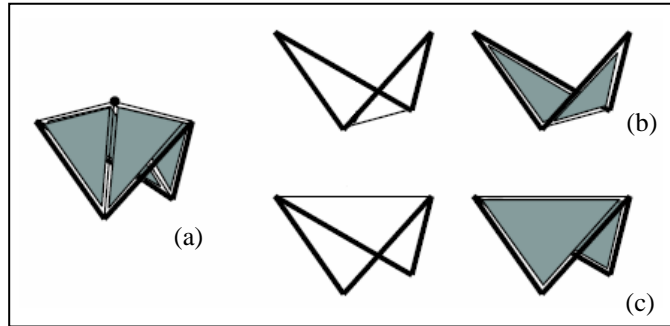
### Polyhedron remeshing

Two main approaches are used for remeshing the opening in the polyhedron created by the removal of a vertex (see [VL98] for more details):

- **Equilaterality criterion** (see Figure 3.4), aiming at generating a triangulation as equilateral as possible, i.e. a Delaunay triangulation. It enables the generation of a good quality triangulation, which unfortunately cannot respect the initial geometry, e.g. model sharp edges, in the case where only polyhedron information is available;
- **Curvature criterion** (see Figure 3.5), which attempts to minimize the difference between the initial and the simplified geometry, therefore preserving as much as possible the object shape.



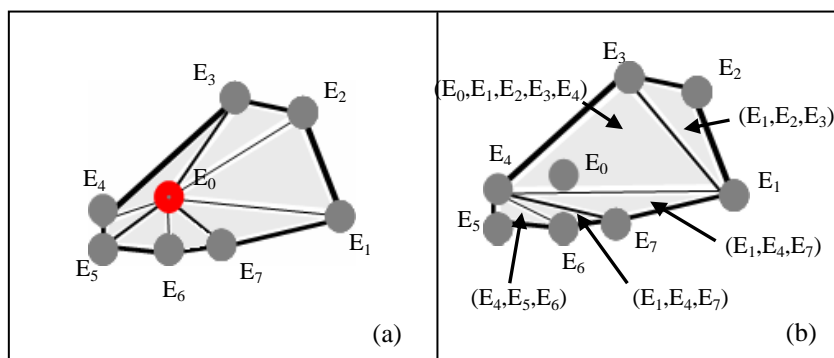
**Figure 3.4:** Example of remeshing process steps based on the equilaterality criterion.



**Figure 3.5:** Example of remeshing based on the curvature criterion: (a) Initial configuration; (b) a possible remeshing; (c) remeshing solution obtained when the initial shape is optimally preserved.

### Vertex removal validation

To validate the removal of the candidate vertex from the behaviour simulation perspective, the operator exploits the information associated to the map of FE sizes, which constitutes the a priori mechanical criterion adopted in our approach to shape adaptation and described in section 2.4.2. At each step of the simplification process, the spheres of the discrete envelope, related to the faces surrounding the vertex to be removed, are reassigned to the faces produced by the remeshing scheme. This inheritance process allows them to be kept as active entities of the discrete envelope all along the entire decimation process. The reassignment process is based on an intersection between the spheres of the discrete envelope (also the inherited ones) and the newly created polyhedron entities. Figure 3.6 shows a 2D example. If one of the spheres does not intersect the entities of the remeshing, this means that the decimated polyhedron stays outside the discrete envelope, and therefore the vertex is not allowed to be removed.




**Figure 3.6:** Example of the inheritance process of the FE map of sizes: (a) Initial polyhedron with its associated spheres of the discrete envelope; (b) Final triangulation and lists of spheres associated to the faces of the remeshed area.

### 3.1.2 Topological detail removal operator

The removal of a topological detail changes the component topology, i.e. it modifies its genus. The topological details considered in our approach are through holes in volume domains. Then, they will be formed by a connected set of polyhedron faces and bounded by two loops of polyhedron edges.

First, the number of through holes in the object can be computed. For a closed and oriented surface, the number  $h$  of topological features in the object (including both through holes and handles) can be determined by using Eq. (1.2), where:

- The number of shells,  $s$ , is always considered equal to one, since we only deal with connected volumetric domains without voids;
- The number of rings,  $r$ , is always considered equal to one, since we deal with manifold triangulations. 

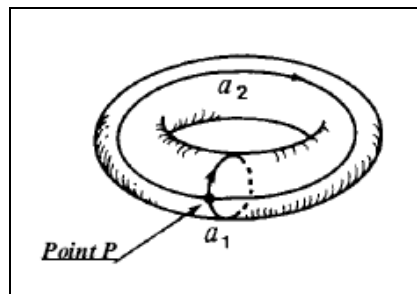
The above topological analysis is a global approach, which only determines the number of through holes existing in the model. Hence, means to localize through holes over the polyhedron are needed, which should be able to identify the hole entities, i.e. the set of faces and the two boundaries characterising the hole.

The algorithm which identifies topological details is based on four steps:

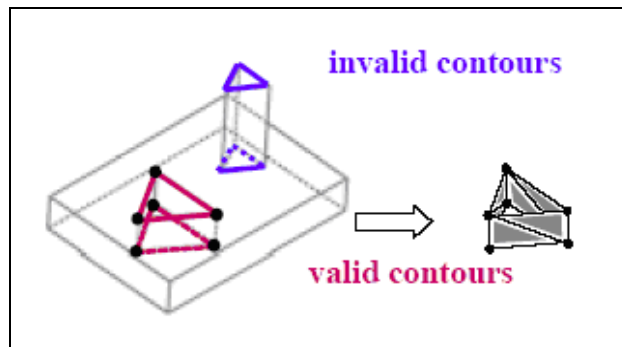
1. Selection of polyhedron edges possibly being part of through hole boundaries. In order to detect contours of different kinds, a broad selection criterion is used: edges having a negative discrete Gaussian curvature at least at one vertex are included in the list. The value of discrete Gaussian curvature is computed with Eq. (3.2), which returns a value  $K_{d-inv}(V)$  depending only on the angles of the polyhedron faces adjacent to  $V$ ;
2. Concatenation of the selected edges into closed contours;
3. Selection of the closed contours that are actually boundary of holes and subsequent identification of the set of faces defining each through hole. To this end, properties of the fundamental groups of curves over a surface are exploited as it follows [FG82]. For each through hole, two independent families of curves can be defined, which either pass through the hole (curves  $a_1$  in Figure 3.7) or match the boundaries of the hole (curves  $a_2$  in Figure 3.7). Two curves can be considered as independent if it is not possible to deform in a continuously way one to obtain the other. Then, a front propagation process is initialised starting from an edge belonging to one of the identified closed contours. A face adjacent to the chosen edge is arbitrary selected. Next, the propagation process is performed until crossing the starting hole boundary. When the front propagation process cannot continue, in the case where the front obtained contains all the faces of the current polyhedron, the selected contour is indeed a hole boundary, otherwise it is an invalid contour (see Figure 3.8). Valid contours of hole boundaries partition

the polyhedron in two domains. The domain having the smallest area is the one forming the actual through hole.

4. Among all valid edge contours that have proven to respect the properties of fundamental groups, equivalent contours, i.e. belonging to the same fundamental group of curves, need to be detected, since they will be part of the same through hole. The through hole identification algorithm chooses by default the maximal hole. Indeed, the determination of the entities belonging to a hole is not unique. Figure 3.9 shows possible solutions for simplification of a through hole. Then, additional criteria can be considered which help in choosing the simplification to be performed. These criteria could be driven by different kinds of considerations, e.g. geometric and/or technological. In our context, only through holes corresponding to a priori simplification details need to be treated. Therefore, we will validate a through hole removal based on additional mechanical criteria.

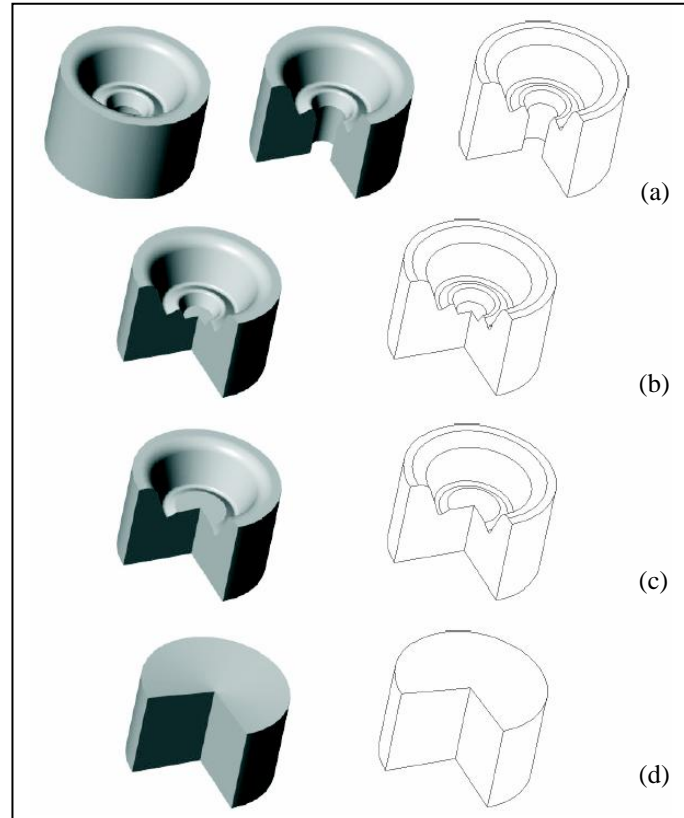


**Figure 3.7:** Fundamental groups of curves for a torus: the curves either pass through the hole (fundamental group of curves  $a_1$ ) or match the boundaries of the hole (fundamental group of curves  $a_2$ ).



**Figure 3.8:** Identification of valid contours to define the group of faces belonging to a through hole.





**Figure 3.9:** Examples from [Fin01] of different interpretations of holes suppression: (a) Initial model with a through hole; (b)(c)(d) Possible solutions of hole simplification. Additional criteria can be used to define hole boundaries and then choice the simplification to be performed.

When an a priori mechanical criterion, which in our case is the geometric envelope generated by using the concept of map of FE sizes (see section 2.4.2), validates the removal of the identified through hole, the topological operator removes all its related polyhedral entities and adds new entities to the polyhedral model. This is needed in order to close the gaps formed by edge loops corresponding to the boundaries of the removed hole.





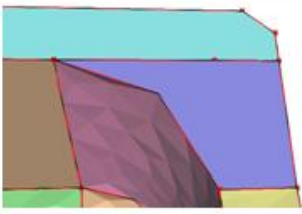
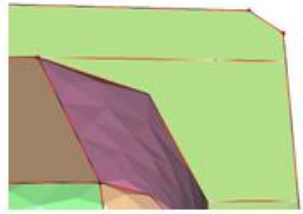
The hole identification process described above is mainly based on topological reasoning on the polyhedral representation. However, some geometrical properties need to be also taken into account at the step 1 of the identification algorithms, i.e. when possible hole boundaries are selected by estimating discrete curvatures at polyhedron vertices. The geometrical properties strongly depend on numerical approximations occurred when obtaining the initial tessellation of the shape model. These approximations are subjected to an inevitable numerical sensitivity that may lead the through hole identification algorithm to the failure. Obviously, additional criteria could be used in order to select edges candidates as hole boundaries. Unfortunately, in this case the computational cost of the algorithm would substantially grow and they could anyhow fail due to their geometrical nature. An alternative approach is that proposed by [FL05], where through holes are identified whose boundary are formed by three edges. In this case, the retrieval of the

boundary would be more efficient, but a previous simplification of the polyhedral model leading to boundaries of through holes formed only by three edges should be performed.

In view of a more general identification of holes (which will be provided at section 3.4.2.1), higher-level information, which is obtained thanks to the mixed shape representation, can be exploited. Moreover, reasoning on the mixed shape representation also allows one to identify blind holes, which do not belong to the category of topological details explored in this section, i.e. their removal does not changes the genus of the polyhedron.

### 3.2 Operators modifying the boundary decomposition

A B-Rep representation is merely the surface boundary representation of the volume model of a component and consists of a patch decomposition. The semantics of the initial patch decomposition produced by the CAD system during the design stage reflects the initial decomposition in HLT-Faces and their corresponding partitions, as described in the previous chapter. Therefore, at the beginning, every partition matches exactly a face of the input B-Rep NURBS model. However, the use of the HLT data structure allows the user to modify the initial boundary decomposition and obtain additional topological layers that better meet his/her needs and characterize the properties of some desired concepts. To this purpose, aside polyhedral operators described in the previous section, which work in the 3D Euclidean space, new operators have been developed, which work in the curvilinear space of the object i.e. the space defined by its boundary. These operators do not modify the object shape, changing instead the organization of the object boundary into partitions. They allow one reaching any arbitrary decomposition of the object boundary that meets specific user's needs, e.g. recovering shape features like sharp edges or defining partitions having an application dependent meaning, such as BCs in the case of FE models.

	Valid	Invalid
HLT-Vertex		
Polyedge		
		

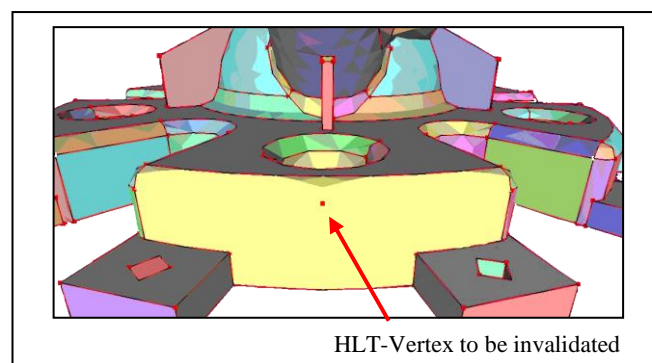
**Figure 3.10:** Graphical visualisation of valid and invalid HLT-vertices and polyedges.

Modifications of the boundary decomposition are performed through a process of invalidation of the mixed shape representation entities. Indeed, the entity is not suppressed but simply invalidated, so that it could be declared valid again in a successive step of modelling or through undo operations. Figure 3.10 illustrates the graphical visualization of valid and invalid HLT-vertices and polyedges.

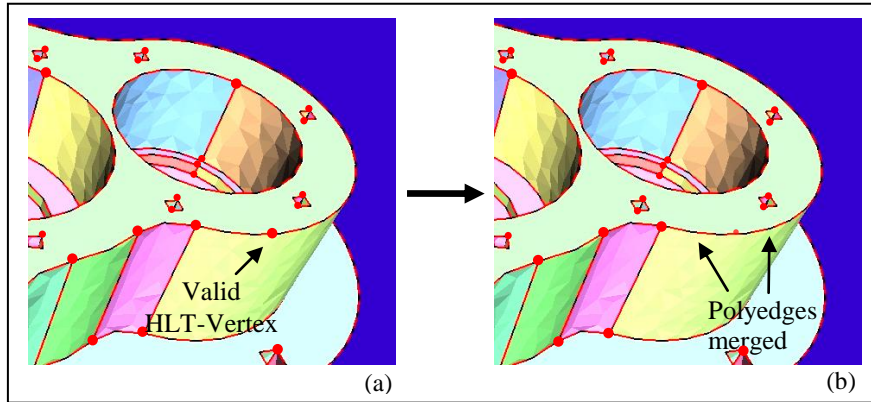
### 3.2.1 HLT-Vertex invalidation operator

Different configurations may occur when a HLT-Vertex needs to be invalidated:

- a. The HLT-Vertex is isolated in a partition, i.e. it corresponds to an internal HLT-Loop. This situation usually occurs following previous modifications of the boundary decomposition, since it corresponds to a generic non-manifold configuration, which generally cannot be found in the initial design model. An example is shown in Figure 3.11. In this case, besides the HLT-Vertex invalidation, no other modifications are necessary to update the boundary decomposition;
- b. The HLT-Vertex is shared by exactly two valid polyedges (an example is shown in Figure 3.12). In this case, besides the invalidation of the HLT-Vertex, a merging of the two polyedges is required. The resulting polyedge is formed by the two initial HLT-Edges. As introduced at section 2.3.3, geometric information available in a STEP file about the initial CAD model is attached to HLT entities. Then, if the HLT-Edges have common geometric parameters, this information is propagated on the new polyedge;
- c. The HLT-Vertex is the extreme of only one valid polyedge or more than two valid polyedges. In these cases, it is not allowed to be invalidated, unless a degenerated configuration occurs, where the HLT-Vertex represents both the extremes of a polyedge.

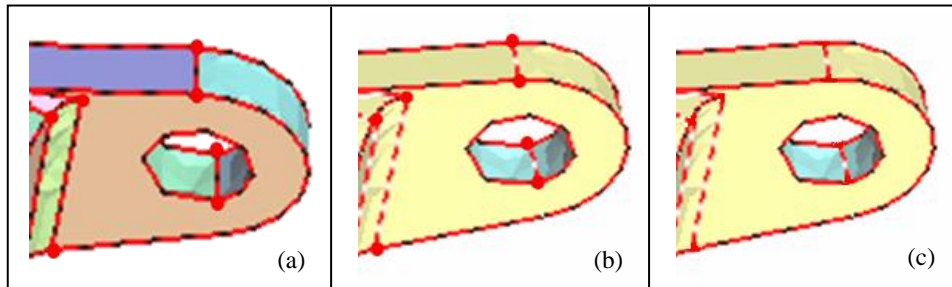


**Figure 3.11:** Example of HLT-Vertex isolated in a partition. Its invalidation does not imply further modifications of the boundary decomposition.



**Figure 3.12:** Example of invalidation of a HLT-Vertex shared by two valid polyedges. (a) Initial configuration, where the HLT-Vertex is valid; (b) Invalidation of the HLT-Vertex associated with the merging of its adjacent polyedges.

Obviously, the boundary decomposition may evolve (by using the operator described in section 3.2.2). Then, it could happen that a HLT-Vertex that in the initial configuration was the extreme of more than two valid polyedges, becomes adjacent to only two valid polyedges. In this case, we fall in case b and the polyedges can be invalidated. Figure 3.13 shows an example. At first, the HLT-vertices of the shape model particular showed in Figure 3.13(a) are linked to three valid polyedges. Then, the boundary decomposition evolves (see Figure 3.13(b)) and the HLT-Vertices become adjacent only to two valid polyedges, thus falling in case b. Only at this point the HLT-Vertices can be invalidated (see Figure 3.13(c)).



**Figure 3.13:** Example of HLT-Vertices linked to more than two polyedges. (a) The HLT-Vertices invalidation is not possible; (b) The boundary decomposition has evolved and some HLT-Vertices are adjacent to only two valid polyedges. In this case, their invalidation becomes possible; (c) The HLT-Vertices have been invalidated.

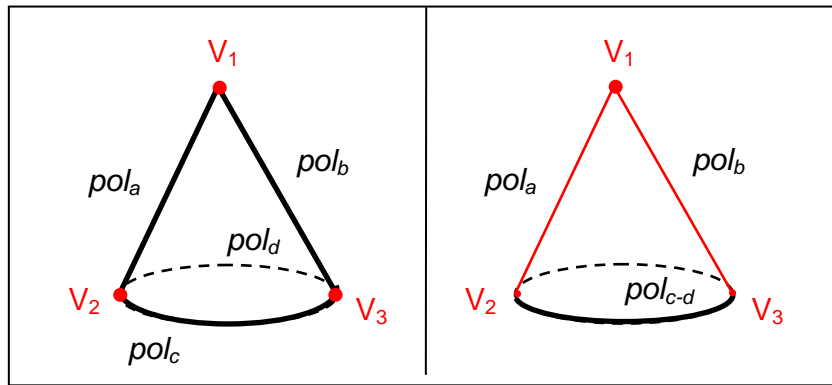
### 3.2.2 Polyedge invalidation operator

The polyedge invalidation can be performed either alone or together with the invalidation of its two extreme HLT-Vertices, depending on the algorithm driving the polyedge invalidation (algorithms will be detailed later on in this section) and on the occurring configuration. Actually, it could also happen that, together with the polyedges invalidation, only one extreme HLT-Vertex is allowed to be invalidated. An example is showed in Figure 3.14, which is related to a cone, where the

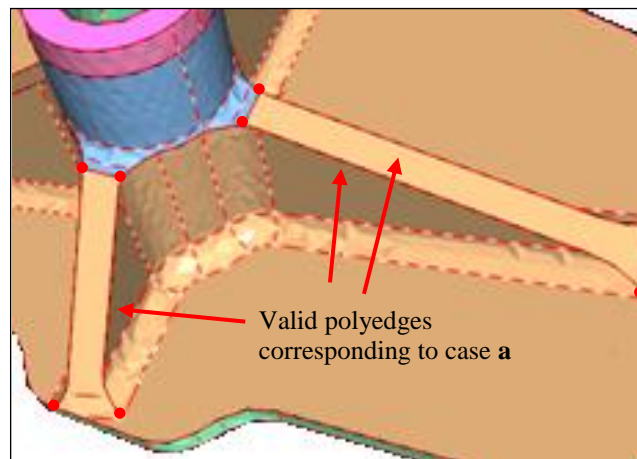
polyedges  $pol_a$  and  $pol_b$  are invalidated, but the HLT-Vertex  $V_1$  belonging to both of them is kept valid, since it is meaningful for describing the cone.

Different configurations may occur:

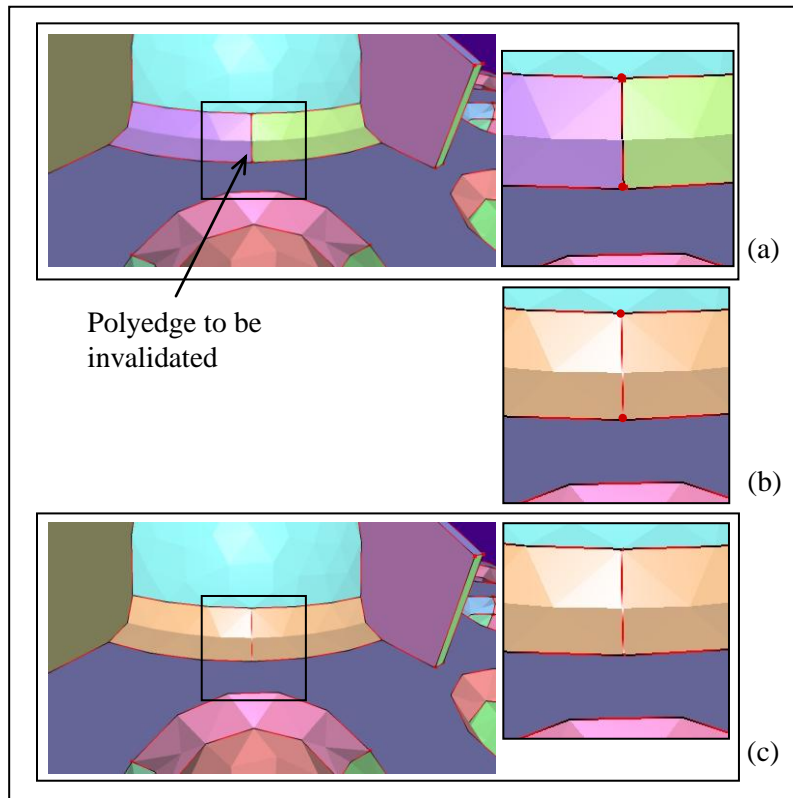
- a. The polyedge is immersed in a partition or it is connected to the partition boundary through only one of its extreme HLT-Vertices. This situation, as case (a) of section 3.2.1, corresponds to an arbitrary non-manifold configuration and usually results from previous modifications of the boundary decomposition. An example is shown in Figure 3.15. If together with the polyedges its extreme HLT-Vertices are invalidated, their invalidation corresponds either to the case (a) of the previous section, when the HLT-Vertex is immersed into the partition, or to the case (b), if it belongs to the partition boundary;



**Figure 3.14:** Example of polyedge invalidation where one of the extreme HLT-Vertices, i.e.  $V_1$ , is not allowed to be invalidated.

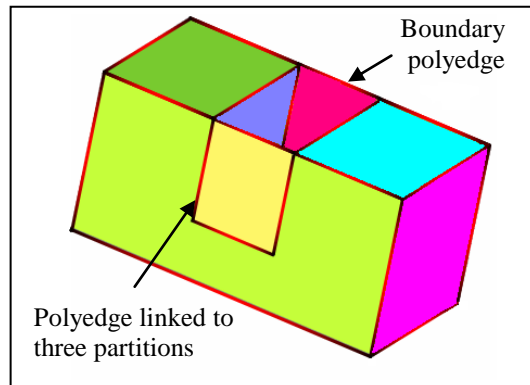


**Figure 3.15:** Example of valid polyedges connected to the partition boundary through only one of their extreme HLT-Vertices (courtesy of EADS CCR).



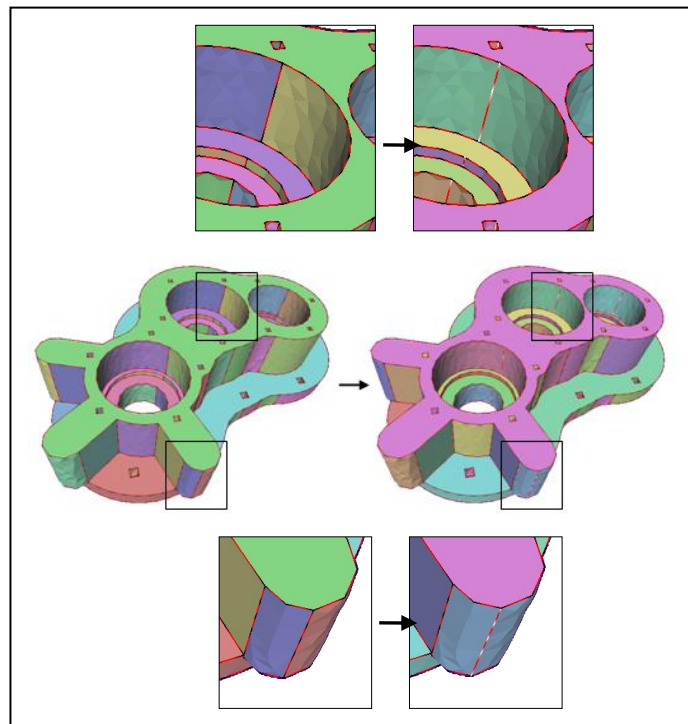
**Figure 3.16:** Example of invalidation of a polyedge shared by two partitions. (a) Initial configuration, where the polyedge is valid; (b) Invalidation of the polyedge and merging of its adjacent partitions; (c) Invalidation of the HLT-Vertices related to the invalidated polyedge, which falls into case (b) of the section 3.2.1.

- b. The polyedge is shared by exactly two partitions (see Figure 3.16). In this case, besides the invalidation of the polyedge, a merging of the two partitions is required. The resulting partition is formed by the two initial HLT-Faces. If the HLT-Faces have common geometric parameters, this information is propagated onto the new partition. If the two HLT-Vertices at the extreme of the invalidated polyedge are allowed to be invalidated, the HLT-Vertices invalidation falls into case (b) of the previous section. Therefore, a merging of polyedges connected to them is also required;
- c. The polyedge is linked to only one partition, i.e. it is a boundary edge, or to more than two partitions (see the example of Figure 3.17). In this case, it is not allowed to be invalidated, unless a degenerated configuration occurs, where the polyedge represents more boundaries of a partition.



**Figure 3.17:** Example of polyedges that are not allowed to be invalidated.

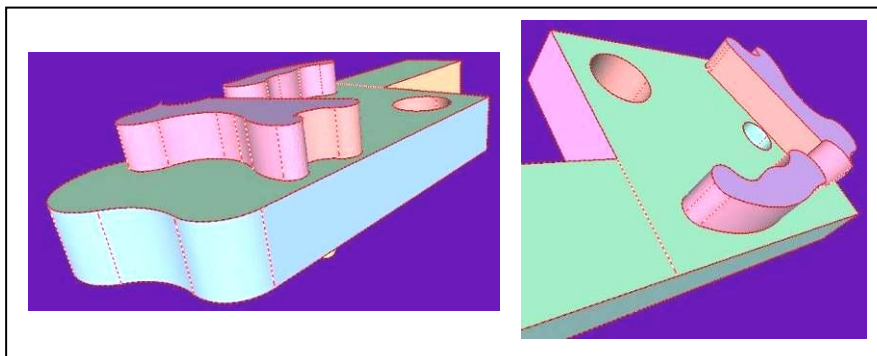
Based on the above considerations, in manifold configurations HLT-Vertex invalidation occurs together with polyedge merging and polyedge invalidation occurs together with partition merging.



**Figure 3.18:** Example of automated boundary decomposition modification obtained by merging partitions that have the same geometrical type and surface parameters.

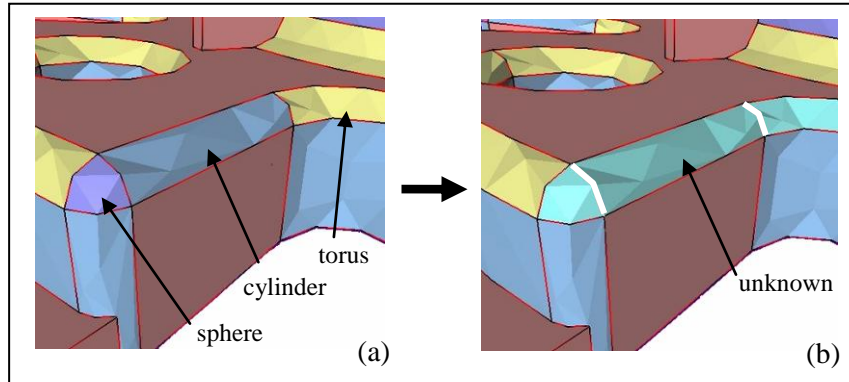
The operators described above can be applied either in an interactive way or through an automated process. Below, examples are given of the algorithms developed to automatically modify the boundary decomposition:

- Merging of partitions characterised by identical shape types and same geometrical parameters. This operation is useful when the considered partitions have the same underlying geometric surface. This algorithm allows the user to obtain a boundary decomposition that fits with the maximal surface decomposition criterion. The geometric information can be propagated onto the new boundary decomposition (see section 2.3.3), which therefore will embed an information intrinsic to the shape object and will be closer to the user's perception. Figure 3.18 shows an example of modifications of boundary decomposition obtained by applying this principle;
- Merging of partitions of the same geometrical type, sharing a smooth surface connection. This operation enables one to identify partitions possible location of particular functional links with other components, which could slip by using the previous algorithm. Obviously, some geometric parameters related to the initial partitions can no longer be associated to the resulting partition, e.g. axis and radius of cylindrical partitions;
- Merging of partitions sharing smooth surface connections, whatever their type (see an example in Figure 3.19). This operation allows one to highlight the sharp edges of the component. During the present merging operation, not only geometrical parameters related to the initial partitions, but also information about geometrical type of surfaces can be lost on the new created partition. When a polyedge separating two partitions of different geometrical types is invalidated, the geometrical type of the new partition is set to unknown. See an additional example in Figure 3.20.



**Figure 3.19:** Example of automated modification of the boundary decomposition obtained by merging partitions with smooth surface connection.

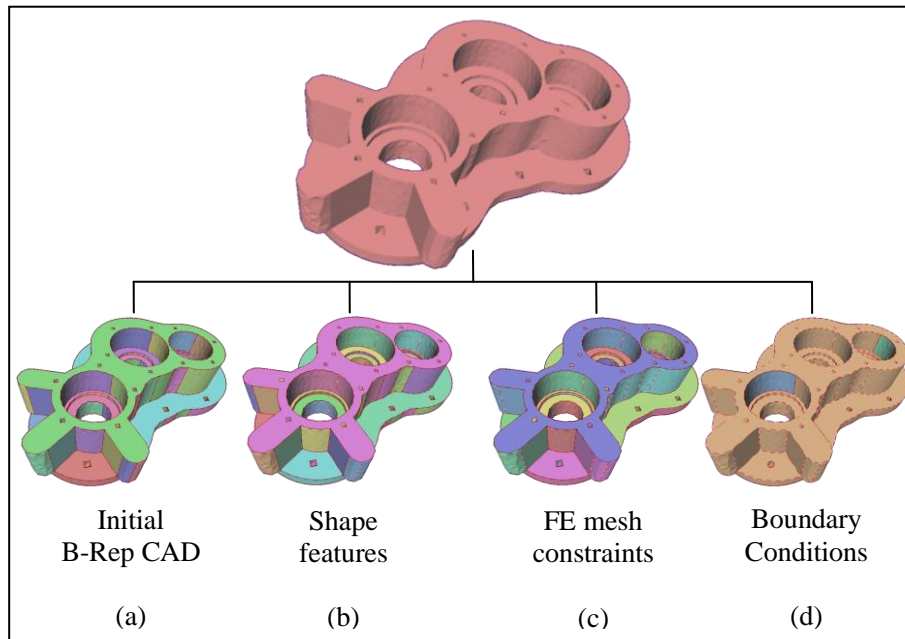




**Figure 3.20:** Example of merging of partitions where the information related to the surface geometric type is transformed.

Operations aiming at modifying the boundary decomposition can be included in algorithms devoted to specific purposes, e.g. for the characterization of shape form features that will be described in section 3.4.

The concept of multiple topological layers, introduced in section 2.3.3, allows to define several topological layers, each one corresponding to a specific boundary domain decomposition corresponding to specific user's needs (see as example Figure 3.21). In the following of this chapter, it will be described how proper boundary decompositions can be associated to shape simplification operators acting on the polyhedral shape representation, in order to achieve a more efficient process of shape simplification. Indeed, suited boundary decompositions can be obtained to specify the location of shape features over the object (see Figure 3.21(b)) or BCs related to the considered mechanical problem (see Figure 3.21(c)). Moreover, in the next chapter, we will see how the specification of BCs over the domain boundary can be also used to associate this information to a FE mesh; and how a proper boundary decomposition can support the definition of FE meshing constraints (see Figure 3.21(d)).



**Figure 3.21:** Different topological layers of a shape model that have associated specific boundary domain decompositions: (a) boundary decomposition coinciding with the one of the input B-Rep NURBS model; (b) boundary decomposition highlighting shape features; (c) boundary decompositions related to BCs of the mechanical problem of interest; (d) boundary decomposition expressing some FE mesh constraints.

### 3.3 Exploiting the two shape representations

The efficiency of the shape transformation process has been improved taking advantage of the mixed shape representation. Indeed, the polyhedral simplification operators described in section 3.1, when combined with the operators described in section 3.2, have allowed us to set up new methods for shape transformations.

#### 3.3.1 Simplification process constrained by boundary representations

The simplification process making use of the skin operator, which acts on the polyhedral model and has been described in section 3.1, can be constrained to respect the topological layers related to the boundary representations obtained by using operators introduced at section 3.2. Valid polyedges and HLT-Vertices are considered as constraints of the shape transformation process. The two rules to be respected during the application of a skin operator are the following ones:

- a. A vertex of the polyhedron that is associated to a valid HLT-Vertex cannot be removed;
- b. The polyedges must be consistently rebuilt, i.e. their polyhedral description must be based on a subset of the polyhedral vertices initially defining the polyedge and respect the initial connectivity scheme. This allows that

adjacency relations between partitions are preserved during the simplification process.

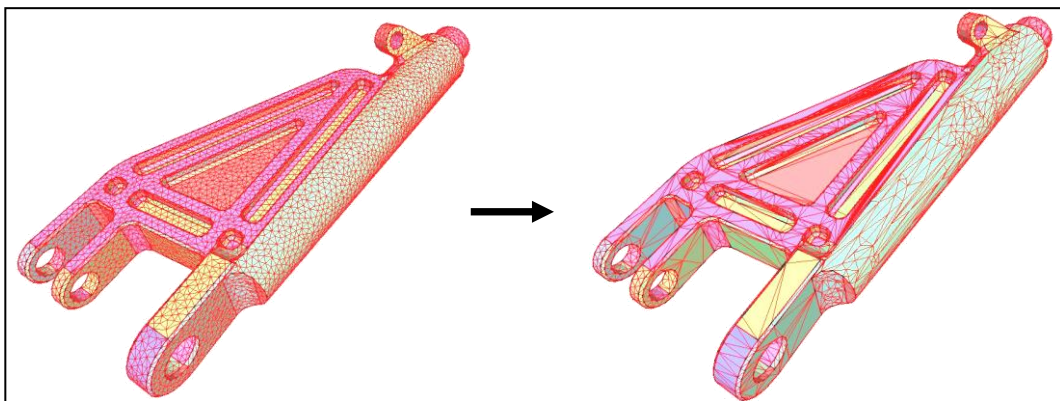
The topology of the domain boundary, defined by the user and reflecting his/her objectives, is preserved during the shape simplification process. This allows the information that the user deems as essential for downstream applications to be propagated on the adapted model. In contrast, when some polyedges become invalid following a transformation of the domain boundary decomposition, the vertex removal process applied to the polyhedron model is not constrained by the content of the HLT data structure. An example of a shape simplification process constrained by the boundary representation is given in Figure 3.22.

However, the possibility still exists of performing a “free remeshing”, i.e. simplifying the polyhedron without constraining it to respect the boundary representation expressed by the HLT entities. In this case, a vertex of the polyhedron associated to a HLT-Vertex could be removed or a polyedge could not be fully rebuilt. In this latter case:

- The corresponding polyedges are invalidated;
- Partitions that are adjacent to invalidated polyedges are merged.

Even in the case where a free remeshing is performed, the topological information contained in HLT data structure, together with the geometric information associated to it, is preserved if possible and transferred on the simplified model.

In the current implementation the explicit representation of the topology associated to the multiple topological layers is not available yet. At present, the information corresponding to the various topological layers is attached to a single layer by means of attributes. Therefore, the shape transformation process is constrained only by one boundary representation, which embeds all the data and therefore conveys the information associated to different concepts, e.g. location of shape features and FE meshing constraints.



**Figure 3.22:** Results of a shape simplification process constrained by the boundary representation (courtesy of EADS CCR).

### **3.3.2 Localised simplification process**

A further way of combining the polyhedron decimation with the decomposition of the boundary of the object domain is to focus the simplification process on some specific areas of the model, e.g. partitions, identified through automatic or semi-automatic processes.

The information contained in the HLT representation allows high-level reasoning on the model to automatically detect particular areas, e.g. set of partitions constituting shape form features of the object. This topic will be discussed in details in the next section.

In addition, the user can get an object boundary decomposition that allows areas with a specific meaning in relation with the hypotheses of the considered mechanical problem, such as location of BCs, to be highlighted.

In this way it is possible to limit the simplification process either to concentrate on specific partitions or to forbid the simplification on the selected areas and elements.

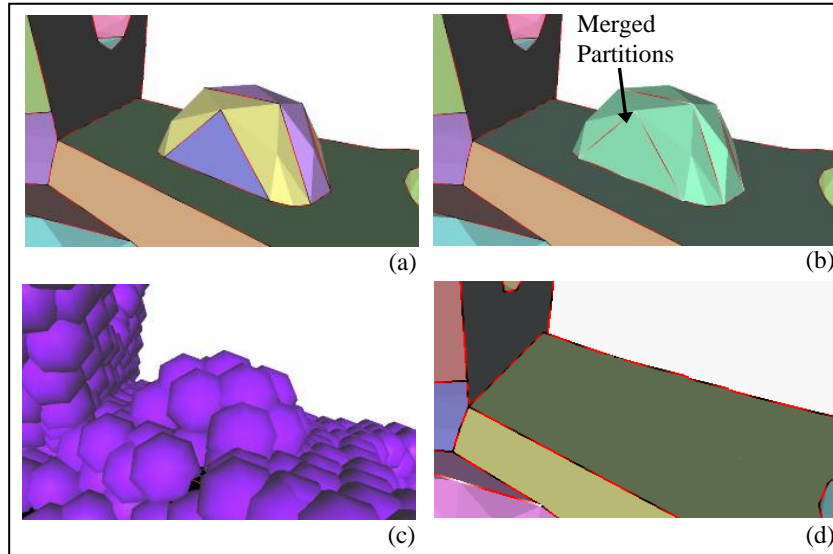
## **3.4 Identification and removal of simplification details**

As illustrated in the previous sections, the mixed shape representation has allowed us to develop operators supporting the identification and selection of shape sub-domains on the model.

These shape sub-domains can represent the geometric location of a priori simplification details (see section 2.5). They are represented on the shape model based on the mixed shape representation by means of a suited boundary decomposition, which can be obtained either interactively by the user or with automatic identification algorithms. Information about the various types of the identified shape sub-domains can be attached to this boundary decomposition.

The boundary decomposition expressing the location of shape sub-domains over the boundary of the object domain will act as a constraint during the shape simplification process, as introduced in section 3.3.1.

However, mechanical criteria have to be considered as additional constraint during the shape simplification process, e.g. the geometric envelope binding shape transformations or the maximum variation of volume during the shape simplification process. Only the satisfaction of these constraints allows one to consider the shape sub-domains identified as a priori simplification details and therefore to validate their removal.



**Figure 3.23:** Suppression of a shape sub-domain constrained by the boundary decomposition and by the geometric envelope acting as a mechanical criterion.

Figure 3.23 shows an example, where the validation of a shape sub-domain as a priori simplification detail and then its removal is validated by an a priori mechanical constraint specified through the definition of a geometric envelope. Figure 3.23(a) shows the initial configuration, where the boundary decomposition corresponds to that one of the input B-Rep NURBS model coming from a CAD system. The user interactively modifies the boundary decomposition by means of operators described in section 3.2. Thus, the shape sub-domain shown in Figure 3.23(b) will correspond to a single partition. The removal of all the polyhedron vertices belonging to the shape sub-domain is validated, since the geometric envelope, acting as a constraint during the shape simplification process (see Figure 3.23(c)), is large enough and the boundary decomposition obtained does not constraint the polyhedron entities belonging to the shape sub-domain. Figure 3.23(d) shows the result of the simplification process, during which the protrusion has been completely removed and



### 3.4.1 Interactive identification of simplification details

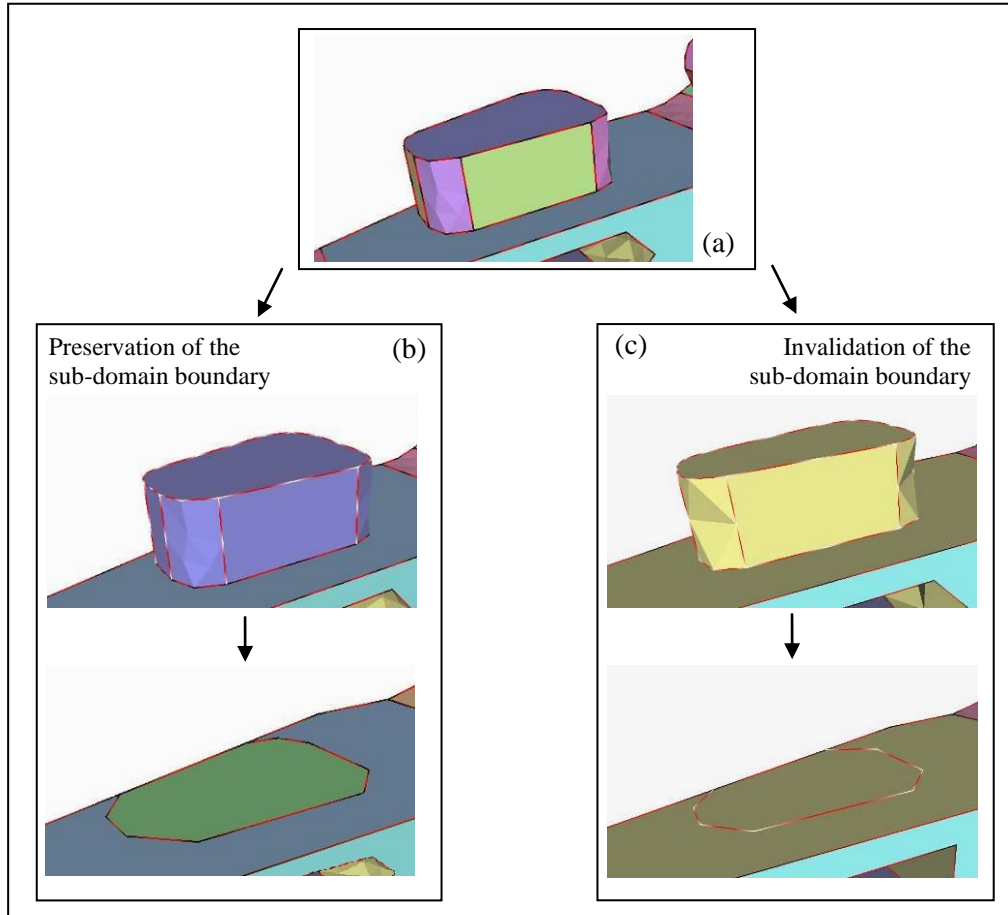
As introduced above, the identification of the shape sub-domains possible location of a priori simplification detail can be performed directly by the user. Using this approach, the expert interactively selects and merges the partitions of the boundary domain that are part of the shape sub-domain he/she has in mind. The new partition corresponding to the shape sub-domain can be labelled as *general shape sub-domain* and, if the a priori criterion used allows its suppression, it can be removed from the shape model. The opening created by the sub-domain removal in the model needs to be filled, i.e. new polyhedral entities have to be added according to the curvature of the surrounding area. Anyway, if we use the operators described in section 3.1.1, the polyhedral faces belonging to the shape sub-domain being removed are not suppressed in one step. Indeed, when a skin operator is applied, the shape simplification consists in the iterative application of the vertex removal

operator, which goes with a local remeshing of the polyhedron star around the removed vertex. The need to fill the gap in the polyhedron occurs only if the topological operator, described in section 3.1.2, is applied.

When removing a shape sub-domain, the user has two choices, which are consistent with his/her simulation objectives:

- a. Invalidating the polyedge at the boundary of the shape sub-domain. In this case, the polyhedral faces of the remeshed area will belong to the adjacent partition and no explicit indication of the location of the simplified shape sub-domain will be kept on the boundary decomposition of the simplified shape;
- b. Leaving valid the polyedge at the boundary of the removed sub-domain. In this case, the remeshed polyhedral faces will belong to a distinct partition. Therefore, the boundary decomposition of the simplified model will keep as a trace of the suppressed shape sub-domain.

An example is shown in Figure 3.24. The aim here is to remove the shape protrusion of Figure 3.24(a). In the case where the created set of faces defines an area with some specific meaning, e.g. the locus of boundary conditions, this area must be considered as a specific partition and the polyedges at the boundary must not to be invalidated (Figure 3.24(b)). In contrast, if no additional constraint exists, the polyedges at the boundary of the sub-domain are invalidated, therefore. the polyhedral faces belonging to the remeshed area will be part of the surrounding partition (Figure 3.24(c)).



**Figure 3.24:** (a) The shape sub-domain forming a protrusion has to be removed; (b) Simplification of the shape sub-domain which preserves the corresponding boundary on the simplified model; (c) Simplification of the shape sub-domain with invalidation of the corresponding boundary.

### 3.4.2 Automatic identification of shape features as simplification details

In product design, some specific recurrent shape elements exist that are functionally meaningful but whose removal may have low impact with respect to the product mechanical behaviour. This applies for instance to specific shape features such as holes and fillets, that can be easily detected by exploiting the capabilities of the mixed shape representation.

#### 3.4.2.1 Hole form features

In the following, we describe the algorithms used to identify and simplify partitions corresponding to hole form features.

## Hole definition

Holes are very common features in the real-world mechanical parts. According to the classification introduced in [SM95], holes are subtractive volume features, which originate either two distinct openings (*Passage*), or one opening in the part volume (*Depression*). We can use this classification to state a first important distinction between through holes, i.e. passages, and blind holes, i.e. depressions. Then, for each one of these categories, the characterization of a hole can be done according to different respects.

An algorithm for the identification of through holes on polyhedral models has been introduced in section 3.1.2. Although it is able to count the number of through holes in a shape model, it could fail in identifying the exact location of through holes, and in addition, it is not sufficient for an unambiguous determination of hole boundaries. Moreover, it provides no method for the identification of blind holes. However, the algorithm detailed in section 3.1.2 stays useful when we deal with object models whose B-Rep information is not available, e.g. scanned data or already existing meshes.

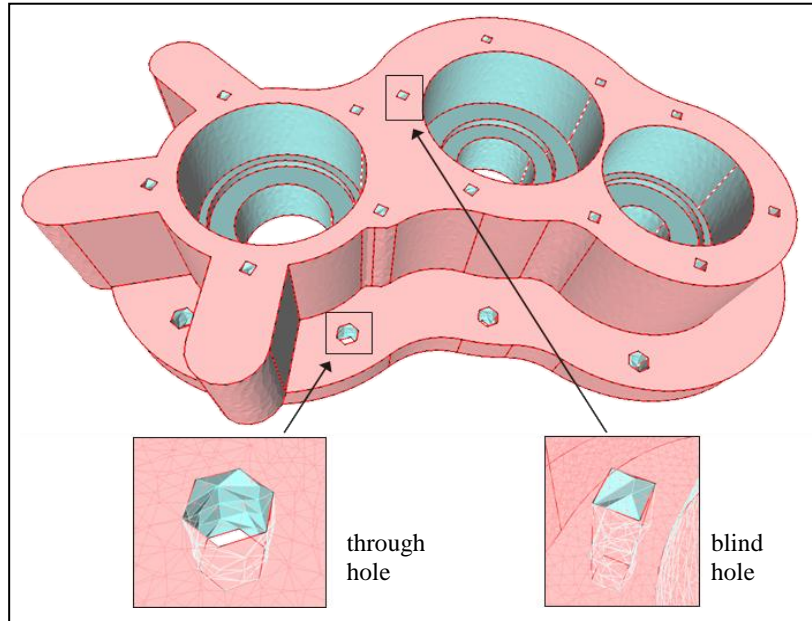
Based on the above considerations, we conclude that, in the case where a B-Rep model is available, alternative methods should be considered exploiting the geometric and topologic information it contains. The topology of the B-Rep representation is helpful for the characterization of holes. Boundary of holes could correspond to internal loops of B-Rep faces, if the faces forming the candidate holes have concurrent normals. Then, all elements belonging to the hole could be identified with a recursive algorithm [DFG98]. If the hole does not end up with a bottom, it will be through.

However, the topology of the B-Rep representation is still not sufficient to get a robust characterization of holes. As we will detail below in this section, sometimes the boundary of the hole does not correspond to an internal loop. In these cases, additional considerations, still based on the topology of the model could be made. Anyway, a robust identification of holes is possible only when a complementary geometric aspect is taken into account.

The discussion about the boundary delimitating a hole is still open (see Figure 3.9). Additional criteria can be taken into account, driven for example by technological [CFL01] or manufacturing considerations (see for example [CTB\*98]), or also by mechanical ones, e.g. a threshold value of the allowed volume variation in the case where the hole is suppressed, or also by a combination of both of them.

Technological and manufacturing considerations could also support the characterization of other types of holes that can be found on the model, e.g. pockets into the model corresponding to milling operations.





**Figure 3.25:** Examples of through and blind holes identified on a shape model.

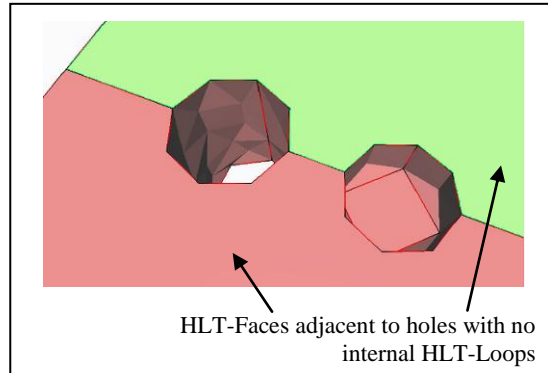
In our software environment based on the mixed shape representation, notwithstanding the algorithm used for hole identification, once holes have been identified, their partitions are labelled as *hole form features*. In addition, if the lateral side of a hole is formed by more than one partition, these partitions are merged together to produce an intrinsic representation of the hole. An example of hole form features identified in a shape model is shown in Figure 3.25, where both through and blind holes have been identified.

### Hole identification

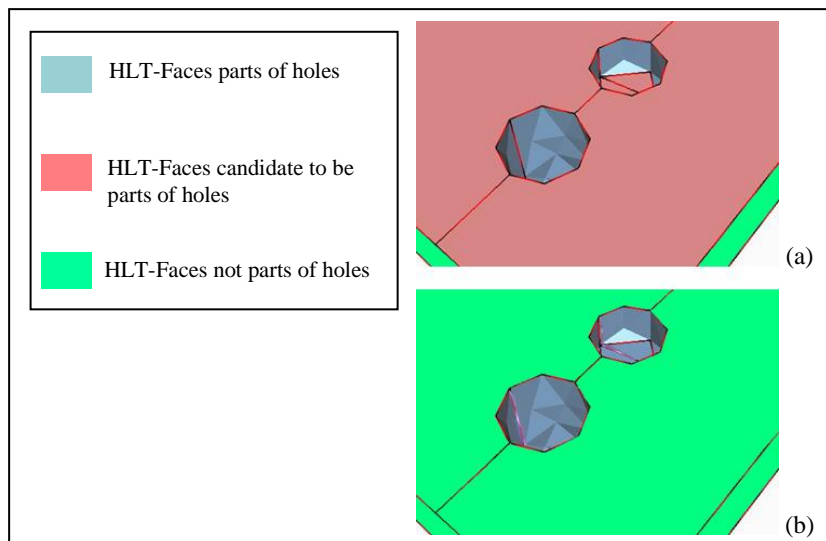
The process of hole identification begins by looking for candidate HLT-Faces adjacent to HLT-Faces that define the walls of holes. This is achieved by selecting all the HLT-Faces that have a number of internal HLT-Loops strictly greater than zero, i.e. HLT-Faces containing at least one interior loop. Next, for each internal loop of the recovered HLT-Faces, the algorithm looks at all the HLT-Faces adjacent through an external edge. Among all the HLT-Faces retrieved, only those actually part of depressions are kept. Next, the algorithm looks recursively at all the HLT-Faces adjacent to the retrieved ones through external loops.

However, it can also happen that a hole boundary is formed by two HLT-Edges and therefore has two adjacent HLT-Faces. In this case, the algorithm would fail since these faces do not have internal HLT-Loops (see Figure 3.26). Moreover, often the hole sides are formed by more HLT-Faces sharing smooth connections, and therefore we can no longer rely on the topologic algorithm above described to retrieve the whole hole. To overcome this weakness, an alternative algorithm can be used in the case where information about geometrical types of HLT-Faces is available. This algorithm identifies holes having cylindrical, conical, toroidal or spherical sides, which include all the typical configurations of holes in mechanical components. The algorithm selects, among all the HLT-Faces having these

geometric types, those having the normal directed towards the surface's axis. Then, for each set of HLT-Faces sharing smooth connections, the number of boundaries is estimated. The set of HLT-Faces corresponds to a hole side only if the number of boundaries is larger than two. Indeed, single holes have exactly two boundaries, while holes having more than two boundaries are part of crossing holes.



**Figure 3.26:** Example of failure of the algorithm looking for internal HLT-Loop when identifying holes.



**Figure 3.27:** Process of identification of the elements of the bottom of holes when the hole side shares one boundary with more than one HLT-Faces: (a) detection of faces candidate to be parts of a hole; (b) Identification of HLT-Faces which are actually parts of holes.

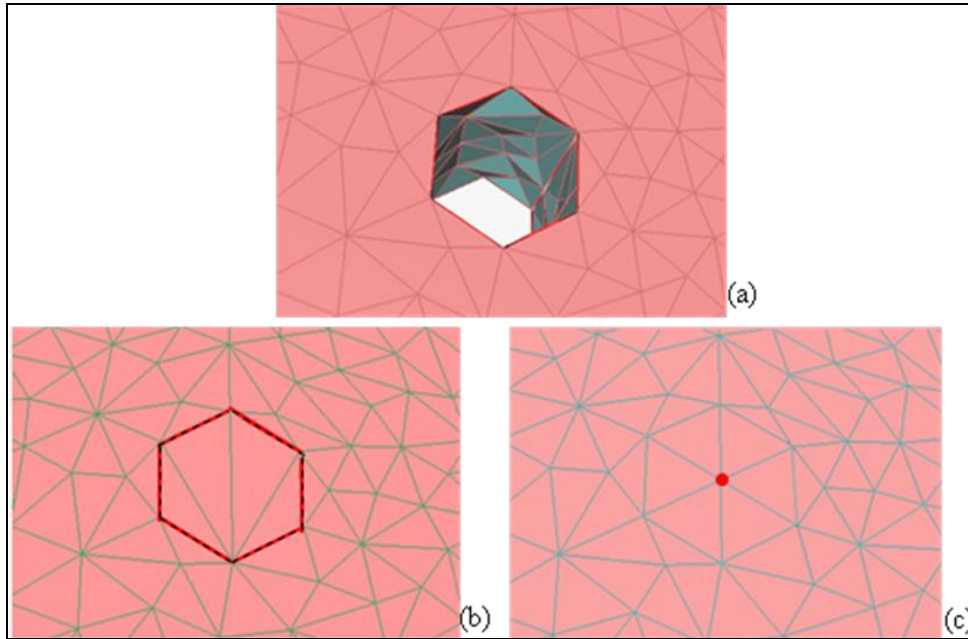
The bottom of blind holes is generally retrieved during the process stage above described, except when the topological analysis is not sufficient. In this case, a complementary algorithm is needed. The search for the bottom is performed in two steps. First, all HLT-Faces adjacent to the hole sides not through internal loops are classified as possible parts of hole bottoms and included in a list (see Figure 3.27(a)). Then, among all the faces recovered in the list, we will consider as part of a hole only HLT-Faces that are adjacent solely to hole sides and to other HLT-Faces of the list (see Figure 3.27(b)).

From the analysis of the above detailed algorithm, it turns out that sometimes the topology provided by the HLT data structure is not enough to detect all holes in the model, and complementary information is needed. In particular, we used geometric information in complement to the topological one. Then, from the combination of the topology and the geometry available in the mixed shape representation, we can assert to be able to characterize all the holes in the shape model having cylindrical, conic, toroidal and spherical sides, and all their combinations leading to compound holes. Complementary information would be useful to unambiguously characterize other possible configurations. For example, holes can be regarded also considering a manufacturing point of view, and a distinction could be done between drilled holes and milled holes. Anyway, at present, the subject lies outside the scopes of this work.

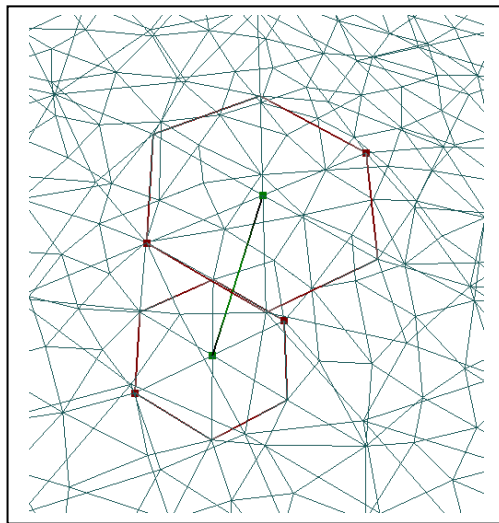
### **Hole simplification**

If the a priori criterion used validates the simplification of the hole, a geometrical operator replaces the set of polyhedron faces belonging to the side of the hole with new polyhedron faces. In this way, we close openings in the polyhedral model (one if the hole is blind, two in the case of through holes), caused by the side removal and corresponding to the hole boundaries. As already explained in section 3.4.1, the user has the choice whether invalidating or not the polyedges corresponding to the boundaries between the removed hole and its adjacent partitions, according to his/her simulation objectives. Figure 3.28 shows an example. The through hole depicted in Figure 3.28(a) is suppressed, and the two openings in the polyhedron caused by its removal are closed by defining suited remeshing schemes. Here, the user has made the choice of keeping active the polyedges corresponding to the removed hole boundary (see Figure 3.28(b)).

In the case where the user prefers to preserve information about the removed hole to take into account simulation requirements, he/she can benefit from a further option offered by the hole simplification process. If the hole has a cylindrical side whose axis is known, or if the hole contour is arbitrary and its barycentre lies inside the contour, a new HLT-Vertex and a corresponding polyhedron vertex can be inserted in the remeshing scheme of the gaps created in the polyhedron by the hole removal (see Figure 3.28(c)). Moreover, when a through hole is suppressed, it is possible to insert a polyedge that joins the two HLT-Vertices inserted in the remeshed model, therefore providing a linear idealization of the simplified hole (see Figure 3.29).



**Figure 3.28:** Hole simplification: (a) A through hole; (b) Remeshing of the gap caused by a hole removal; in this case the polyedge corresponding to the hole boundary is kept active; (c) Process of idealisation, where the partition corresponding to the remeshed area has been substituted with a new vertex at its centre.



**Figure 3.29:** Creation of polyedges that idealize the suppressed through holes.

### 3.4.2.2 Fillet form features

#### Fillet definition

Blends are commonly used in mechanical part design and provide a smooth transition between different surfaces of a solid. They improve the safety for handling a component, enhance its strength by reducing stress concentrations and

provide a better aesthetic appearance. From the machining point of view, blends can be either generated due to the tip radius of the cutting tool or machined on purpose with a suited filleting cutting tool [HPR00]. From the design point of view, their design usually follows that of primary features such as slots, holes and pockets, and is obtained by smoothing their sharp corners and edges [SM95]. In this way, continuous smooth blends are created.

Blending may either remove or add material on a component, depending on the convexity of the model local to the blend. A blend on a convex entity removes material from the model to round off the entity, whereas a blend on a concave edge adds material to the model.

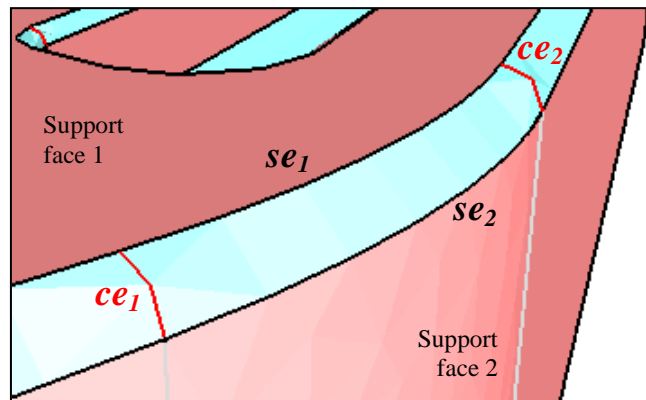
In the case of vertices, we refer to the blend as a *corner*. A *corner* replaces a vertex with a smooth face tangent to the faces incident to the vertex.

The edge blends replace edges by a face tangent to the two faces adjoining the edge. Edge blends replacing convex edges are called *rounds*, whereas those replacing concave edges are referred as a *fillets* [NBB07]. In the rest of the manuscript, we will refer to both kinds of edges blends as *fillets*.

When the edge is replaced by a face non tangent to the incident faces is called *chamfer*.

### Fillet identification

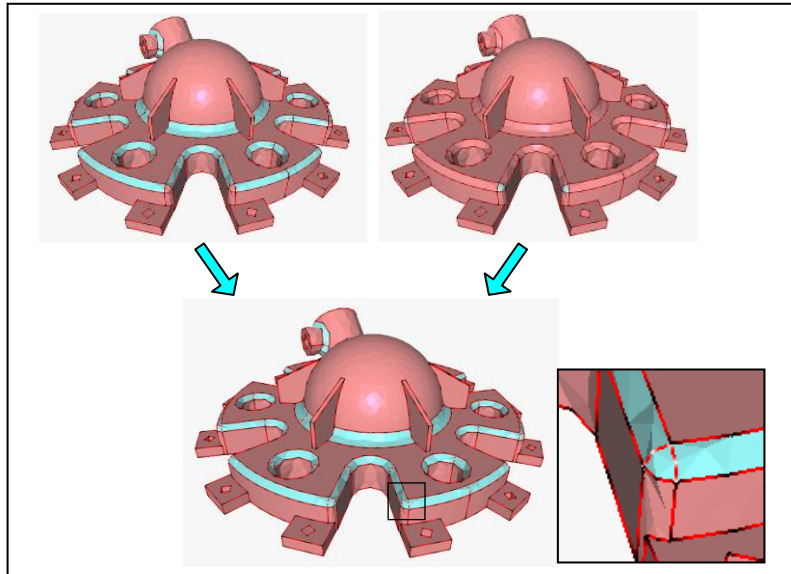
In our work we will focus on fillets whose radius is constant, which represent the most common configuration in real-world mechanical parts. We will adopt the notation of [ZM02], who named **support faces** the faces adjacent to a fillet, i.e. those that the fillet blends. The edges of contact between the fillet faces and the support faces are called **spring edges** ( $se$ ). Often, a series of adjacent edges are smoothed in a single filleting operation forming a series of fillet faces, i.e. a blend chain. The edges that connect adjacent faces in a blend chain are called **cross edges** ( $ce$ ). Figure 3.30 illustrates the concepts explained above.



**Figure 3.30:** Terminology used for the fillet form features.

Here, we provide an algorithm identifying blending chains in a shape model based on the mixed shape representation. Blending chains are identified by using information recovered from the B-Rep NURBS description. The blend identification technique used is similar to the one indicated by [JD03]. Fillet faces are detected by collecting information about the curvature variation across each B-Rep NURBS face, which in the mixed shape representation is a HLT-Face. Potential spring and cross HLT-edges are detected looking at curvatures according to directions along and perpendicular to each HLT-edge, computed on its two adjacent HLT-Faces. If the curvature along the edge is equal for both faces, the edge is a cross edge. If the curvature of the face candidate as fillet, in the direction normal to the edge, is greater than the curvature of the adjacent face, the edge is classified as a spring edge. A HLT-Face is a fillet if it contains two cross HLT-edges and two spring HLT-edges. Additional checks are mandatory, e.g. faces adjacent to the candidate fillet along the spring edges are not allowed to define an angle equal to either  $0^\circ$  or  $180^\circ$ . The algorithm described is broad in identifying fillets. Therefore, some additional geometric criteria could be used to correctly select the candidate HLT-Faces. For example, it would be possible to set a minimum threshold value of the ratio  $\frac{se}{ce}$  (the two  $se$  have the same order of magnitude since we consider only constant radius fillets), in order to retain as fillets only the partitions which have one dominant dimension.

In addition to edge fillets, the algorithm is able to identify corners. If both concave and convex edges converge to the vertex, the corner has a toroidal shape (we note it as *t\_corner*). Therefore, a *t\_corner* is a toroidal HLT-Face having three HLT-Edges classified as *ce* and one HLT-Edge classified as *se*. At the contrary, if the edges incident to the vertex are all of the same type (either convex or concave) either the corner has a spherical shape and is bounded only by three HLT-Edges or it corresponds to a revolution surface and it is bounded by four HLT-Edges (we note it as *c\_corner*). Therefore, spherical *c\_corners* are HLT-Faces whose three HLT-Edges are classified as *ce*, while *c\_corners* formed by revolution surfaces have three HLT-Edges classified as *ce* and one HLT-Edge classified as *se*.



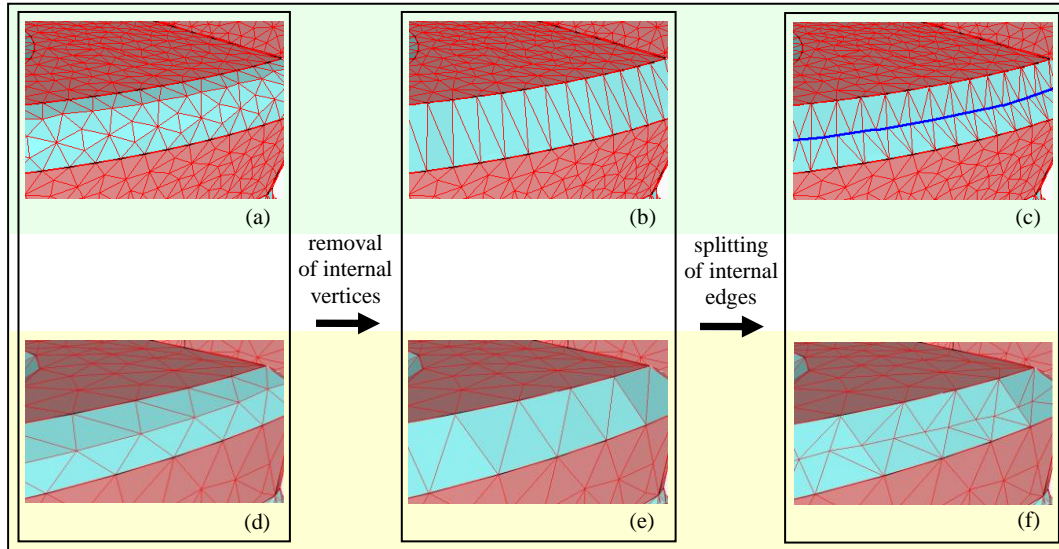
**Figure 3.31:** Fillets and corners merged to create blend chains. In the mixed shape representation, they have labelled as *fillet form features*.

Once the fillet and corner identification has been performed, the corresponding partitions can be merged along cross edges in order to obtain chained blends. This is the typical example of automatic merging of partitions, introduced at section 3.2.2. Thanks to the merging operation, each single blend chain will correspond to a single partition of the model. All partitions of a blend chain are labelled as *fillet form feature*. An example of blending area identification is showed in Figure 3.31, where corners are also included into the blend chains. Once blending areas have been identified, simplification operations can be performed on them in order to modify locally their shape.

### **Fillet simplification**

The operator for fillet simplification benefits from the possibility of concentrating the simplification process on specific partitions, i.e. those labelled as *fillet form feature*, thus taking into account the semantics added to the model during the fillet identification stage.

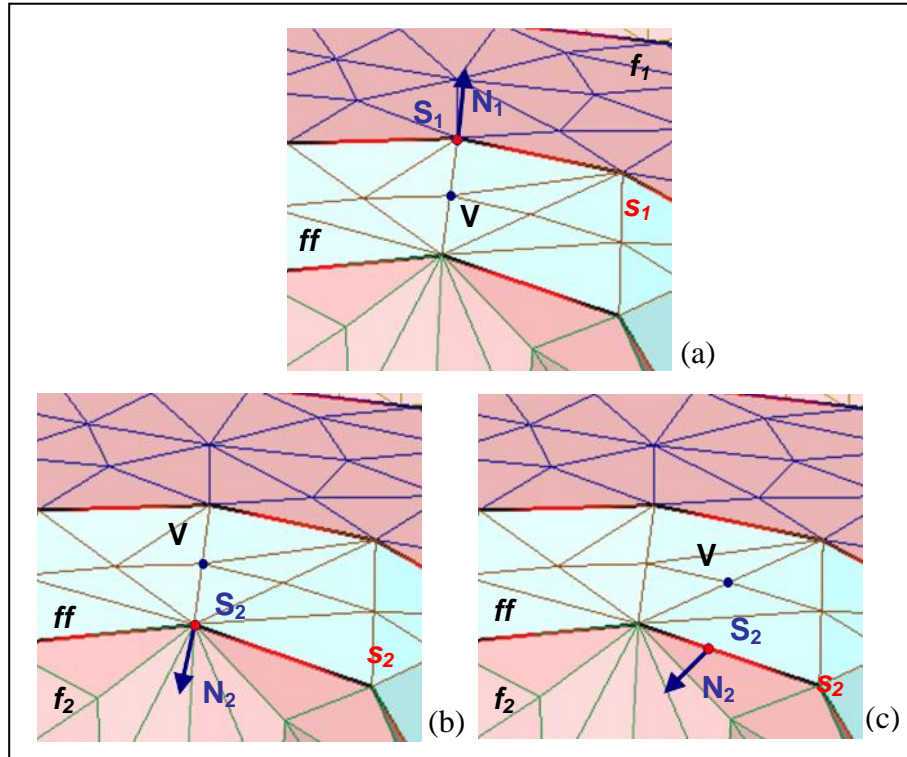
The removal of a fillet corresponds either to the removal or to addition of a volume sub-domain, depending on whether the fillet is concave or convex. However, only the use of a complementary mechanical criterion can decide whether the fillet can be simplified or not. When the mechanical criterion used is an a priori map of FE sizes, which acts as a geometric envelope bounding the simplifications, the fillet can be removed only if it is completely immersed in the envelope. However, the user could also arbitrarily decide to suppress the fillet, relying exclusively on his/her expertise.



**Figure 3.32:** Standardisation operations applied to two fillet form features corresponding to the same boundary decomposition but tessellated with different parameters: (a)(d) The model is tessellated with different parameters; (b)(e) Result of the removal of the vertices internal to the fillets; (c)(f) Result after splitting the internal edge of the fillet.

The tessellation of a fillet form feature depends on the initial domain discretization. Therefore, before simplifying the fillet, a standardisation operation is performed on the polyhedral model, which leads all fillets to have a single layer of internal nodes. To reach this configuration, a simplification operation is performed first, removing all vertices inside the partition that defines the fillet. Removing all the internal vertices of a fillet makes plane the partition labelled as *fillet form feature*. Therefore, the fillet becomes equivalent to a chamfer, and the simplification algorithm, which we are going to detail could be as well used for simplifying chamfer form features. After removing vertices internal to the partition defining the fillet, an enrichment operator is applied. All edges that belong to the fillet partition and that are not part of its boundary polyedges are split. Whenever an edge is split, new polyhedron entities are created. A new vertex is placed in the middle of the edge split; simultaneously, each of the existing triangles is split into two new triangles (more details about the enrichment operator will be found in chapter 4). The enrichment process is applied only once such that, when it ends, the fillet partition will have exactly one layer of internal nodes. In Figure 3.32, the standardisation operations are applied to two fillet form features (see Figure 3.32(a) and Figure 3.32(d)). The fillets belong to two shape models that have the same boundary decomposition corresponding to the one of the initial B-Rep NURBS model, but have been tessellated with different tessellation parameters. We can notice that, at the end of the standardisation process, the fillet will contain only one layer of internal vertices (see Figure 3.32(c) and Figure 3.32(f)), no matter the starting tessellation. However, the standardisation process is applied only to vertices internal to the feature. Therefore, the number of vertices belonging to the fillet spring edges will depend on the initial tessellation parameters also in the standardised configuration.





**Figure 3.33:** Different configurations occurring when estimating normals for determining the position of a fillet point. (a)  $\vec{N}_1$  is the normal to the planar face  $f_1$ ; (b)  $\vec{N}_2$  is the weighted sum of the normals of the triangles belonging to  $f_2$  and incident to  $S_2$ ; (c)  $\vec{N}_2$  is the normal of the triangle of  $f_2$  adjacent to the polyhedron edge which  $S_2$  belongs to.

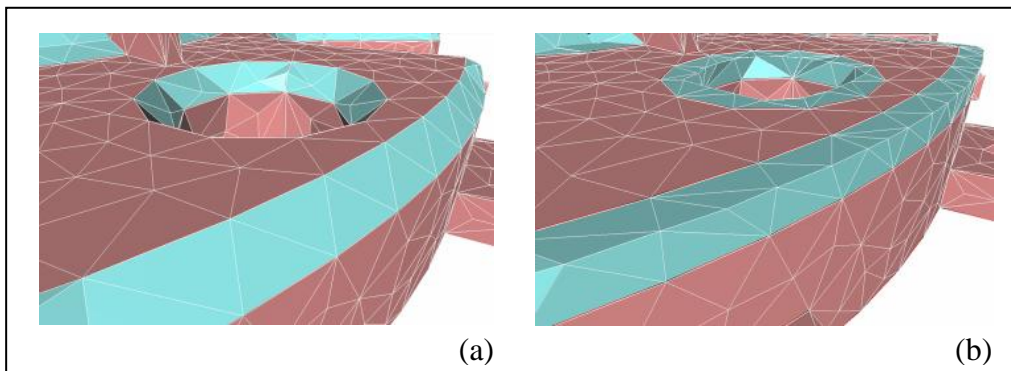
The fillet simplification consists in moving the new layer of polyhedron vertices, created with the enrichment operations, onto the ideal sharp edge that the fillet under consideration blends, similarly to the approach proposed by [ABR\*03]. It means that a new position needs to be defined for each new internal vertex. To this end, for each vertex  $V$  internal to a fillet, its closest points  $S_1$  and  $S_2$  on the two associated spring HLT-edges  $s_1$  and  $s_2$  are recovered. Then, for each  $S_i$ , with  $i \in \{1, 2\}$ , the normal  $N_i$  to the support HLT-face  $f_i$ , adjacent to the fillet HLT-face  $ff$  along the edge  $s_i$ , has to be estimated. If the face  $f_i$  is planar, the normal is invariant for each point on the plane, and its coordinates are easily recovered thanks to the information contained into the mixed shape representation (see Figure 3.33(a)). In a general case, two situations can occur:

- if the point  $S_i$  coincides with a polyhedron vertex (see Figure 3.33(b)), the normal is estimated as a weighted sum of the normals of the triangles of  $f_i$  incident to this vertex;
- if the point  $S_i$  does not lie on the extremity of a polyhedron edge but is internal to it (see Figure 3.33(c)), its normal is considered as equal to the normal of the polyhedron face adjacent to the polyhedron edge where it lies, since it belongs to the support face  $f_i$ .

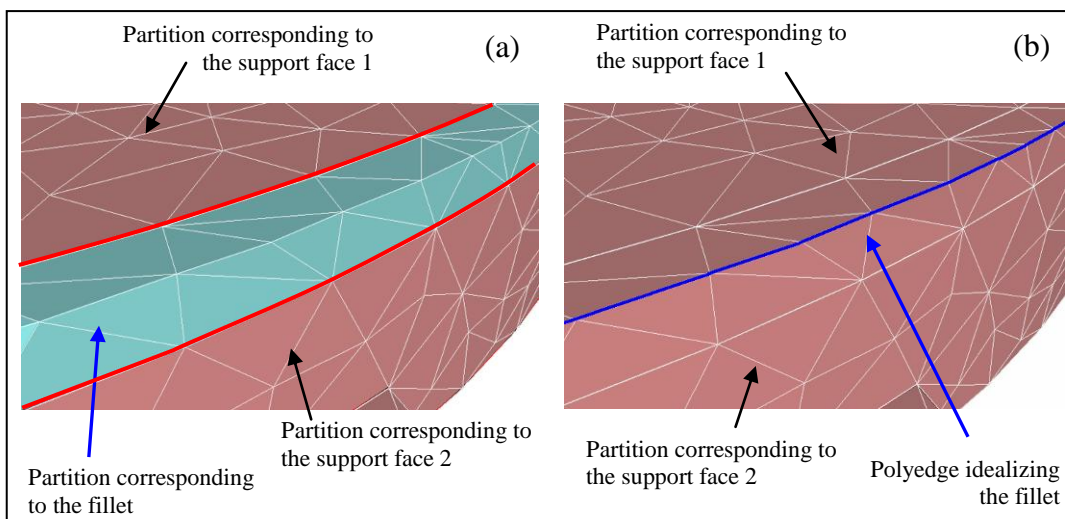
The plane having  $\vec{N}_1$  as normal and passing through  $S_1$  and the plane having  $\vec{N}_2$  as normal and passing through  $S_2$  will be estimated. Then, the vertex  $V$  will be repositioned on the line resulting from the intersection of these two planes, which approximates the sharp edge associated to the two blended surfaces  $f_1$  and  $f_2$ .

Figure 3.34 shows the design particular of a shape model where a fillet simplification is performed. Partitions corresponding to fillets are the blue ones.

The fillet simplification could be considered as an idealization operation, which idealizes the partition corresponding to the fillet form feature in the polyedge corresponding to the sharp edge created when simplifying the fillet. In this case, the polyhedral faces belonging to the fillet partition should be moved to the partitions corresponding to the fillet's support faces. Figure 3.35 shows an example of fillet idealization.



**Figure 3.34:** Example of a model before (a) and after (b) the fillet simplification.



**Figure 3.35:** Two different possibilities for polyhedron partitioning after fillet removal: (a) The partition corresponding to the fillet remains valid (b) The partition corresponding to the fillet is idealized into a polyedge corresponding to the created sharp edge.

### 3.5 Storing data for later reuse

In section 2.1, we highlighted the importance and the utility of an efficient propagation of information and of a clear track of the performed shape modifications during the process of mechanical model generation. Keeping trace of the shape simplifications could be useful for different applications, such as a posteriori mechanical analyses or reinsertion of meaningful shape details.

The operations that enable to store the information about the shape modifications performed can be categorised in two classes:

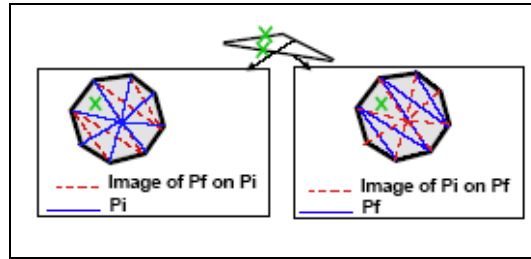
- storage based on the chronology of modifications, which will be described in sections 3.5.1 and 3.5.2,
- storage of the geometrical information linked to the removed shape sub-domains, which will be discussed at section 3.5.3.

Actually, a clear distinction between the two classes is not possible, and often both kinds of actions are performed in order to obtain an efficient storage of the information about shape simplifications applied.

#### 3.5.1 Storing operations based on the polyhedral representation

The first simple operation that can be performed to keep track of shape modifications is to maintain the history of the shape evolutions due to the application of simplification operators. In particular, when a skin operator is applied, new remeshing schemes are provided around each removed vertex. Therefore, at each stage of the skin simplification operator, new polyhedral faces can be created and existing ones can be suppressed. We associate to a face of the polyhedral model, additional information on the stage of the process corresponding to its creation, if it was not part of the initial polyhedral model, and to its suppression, if it is not part of the final simplified model.

Another interesting possibility we offer during the application of the skin operator is the use of a one-to-one mapping function, providing for any point  $N$  of the initial polyhedron  $P_i$  its position on the final one  $P_f$ . This mapping function is based on *reciprocal images* introduced by [FML04]. At each step of a vertex removal process, during the application of the skin operator, the image of a polyhedron  $P_i$  on the target polyhedron  $P_f$  is built by “mapping” the position of initial star polyhedron on its corresponding remeshed area. In the same way, the image of  $P_f$  on  $P_i$  can be generated. This is the reason why the term *reciprocal images* is used (see Figure 3.36). Reciprocal images provide an appropriate and robust mapping of shape evolutions, which allows one to evaluate criteria on local shape modifications.



**Figure 3.36:** Reciprocal images, mapping the initial polyhedron  $P_i$  on the final one  $P_f$ , and vice versa.

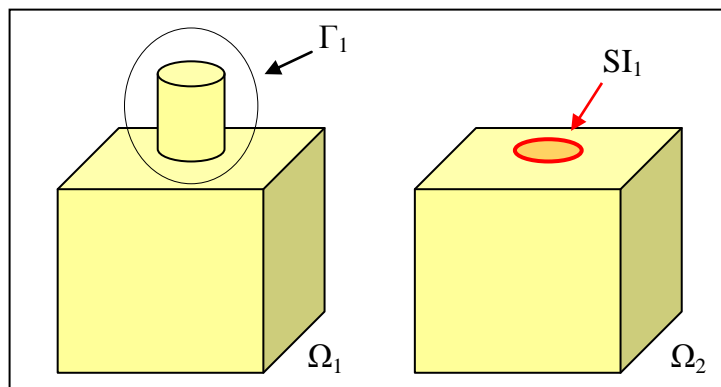
### 3.5.2 Storing operations exploiting the mixed shape representation

Efficient methods for propagating and storing the information on shape modifications can be devised by using the mixed shape representation.

The mixed shape representation allows one to characterise the transition between two different states of the shape model, by specifying the *shape interface* between them. At present, the concept of shape interface has been defined with regard to two shape versions  $\Omega_1$  and  $\Omega_2$  corresponding to the shape before and after the removal of some disconnected shape sub-domains  $\Gamma_i$ . A shape interface  $SI_i$  is the surface characterizing the separation between two different volume shape domains.

Let us consider the removal of a volume sub-domain  $\Gamma_1$  shown in Figure 3.37. Then, the Shape Interface  $SI_1$  is the surface resulting from the intersection between the simplified domain  $\Omega_2$  and the removed sub-domain  $\Gamma_1$ :

$$SI_1 = \Omega_2 \cap \Gamma_1. \quad (3.10)$$



**Figure 3.37:** Explicative example of the concept of shape interface.

Properties of shape interfaces can be easily deduced. If we call  $\Gamma_{1\text{-open}}$  the surface domain corresponding to the boundary of the volume sub-domain  $\Gamma_1$ , which was actually part of the shape domain  $\Omega_1$ , we have:

$$\partial\Gamma_1 = \Gamma_{1\text{-open}} \cup SI_I, \quad (3.11)$$

$$\partial\Omega_2 = (\partial\Omega_1 - \Gamma_{1\text{-open}}) \cup SI_I. \quad (3.12)$$

It should be noticed that, in the case of topological detail removal (see section 3.1.2), the shape interface will be formed by two disconnected surface domains, i.e.  $SI_i = SI_i^A \cup SI_i^B$ .

Thanks to the mixed shape representation, shape interfaces due to shape variations can be characterized on the shape model. A typical example is that provided in Figure 3.24. When suppressing the shape sub-domain related to the protrusion in the figure, the polyedge forming its boundary could be kept valid. In this case, it will surround a new created partition. This partition is the  $SI$  of the suppressed shape sub-domain on the simplified shape model.

However, if the user wants to retain the possibility of performing undo operations or readapting the model after a posteriori analyses, the information about *shape interfaces* needs to be completed with the storage the removed shape sub-domains data, as detailed in the next section.

### 3.5.3 Storing data on removed sub-domains

The user has the possibility to store and therefore to recover the shape information about removed sub-domains. In our context, it is useful when an a posteriori mechanical analysis is performed and therefore the removal of a priori simplification details has to be validated. Moreover, in the case where the a posteriori analysis shows that the removal of a shape sub-domain affects the accuracy of the FE analysis results, the availability of the related shape data allows its reinsertion into the simplified model.

Two kinds of recovery of information related to the a priori simplification details exist:

- a. Recovering of shape sub-domains identified as a priori simplification details during the shape simplification process.

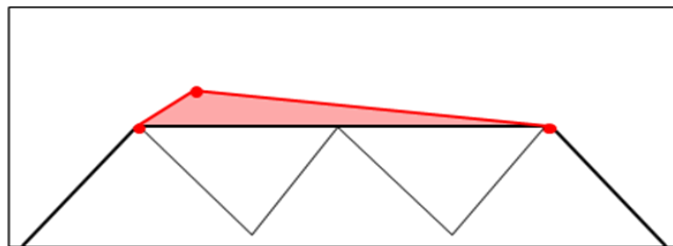
In this case, each shape sub-domain has a precise label. For example, the automatic algorithms detecting holes and fillets form features, which has been described in section 3.4.2, identify them with specific labels, respectively *hole form feature* and *fillet form feature*. Therefore, when removing the shape sub-domains, we can store them as independent shape sub-domains while keeping their semantic meaning, and recover them when necessary;

- b. Recovering of shape sub-domains after the simplification process.

This kind of recovering could be performed based on reasoning on the polyhedral entities. This is particularly useful in the case where no mixed shape representation is available and the simplifications are performed only by using polyhedral operators. In section 3.5.1, we have detailed that it is possible to associate to each polyhedral face some information about the stage of its creation and its removal. This means that we are able to store the information about faces of the initial polyhedral model no longer existing in the simplified one. These data could be retrieved in an a posteriori stage and could be exploited to compute the single sub-domains that have been removed.

When shape simplifications are performed by using the skin operator described in section 3.1.1, it can happen that the modifications correspond mainly to shape adjustments rather than to actual shape detail removals (Figure 3.38). If only slight shape refinements occurred, corresponding shape modifications are so small that their recovery is not useful for the application of the a posteriori mechanical criterion that we use in our approach (see section 2.4.3 and, for a complete description, chapter 5). Indeed, if we generated a FE mesh on the volume sub-domain corresponding to these shape modifications, we would obtain FE sizes too small or FE shapes too flattened. Therefore, we can ascribe the error on the FE solution due to these shape modifications as a discretization error.

To detect which are the areas corresponding to actual shape modifications, after recovering all the faces belonging to the initial polyhedron but no longer to the simplified polyhedral model, we apply some criteria measuring distances and angles between the recovered faces and the simplified model. The user can set threshold values in order to select only polyhedral faces belonging to areas where large shape changes have occurred. Once recovered the targeted faces, their adjacencies are analyzed in order to generate independent a priori simplification details.



**Figure 3.38:** Configuration where the shape modification (corresponding to the coloured area) is considered as a shape adjustment and thus no need for recovering the corresponding volume sub-domain exists.







### **Toward a FE mesh generator: Approach and operators**

*Once obtained the simplified shape domain related to a mechanical model, we need to generate the simulation model where performing the behaviour analysis, i.e. the model containing the FE mesh. The process of FE mesh generation is part of the overall process of reference model generation occurring at the interface between the design and the behaviour simulation PVs. Current industrial practice often relies on different software environments for CAD and FE mesh generation. Then, we have made some efforts aiming at the integration of a surface FE mesh generator in the software environment based on the mixed shape representation. To this end, we have developed some operators that produce a FE surface mesh starting from the polyhedral model resulting from the shape simplification process described in the previous chapter.*

---

#### **4.1 Integration of a surface FE mesh generator in the software environment based on the mixed shape representation**

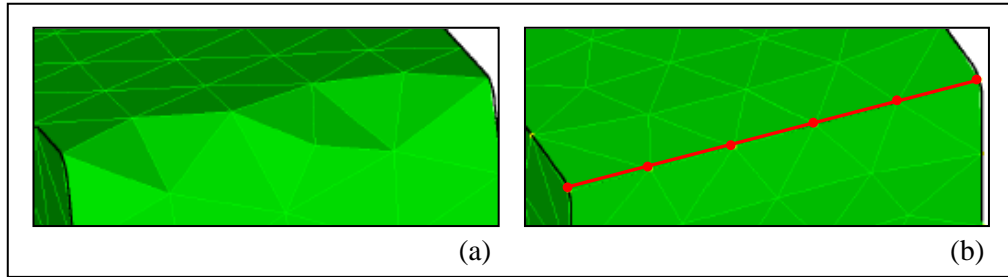
As described in chapter 2, in our approach we have chosen to explicitly define a mechanical model before generating the actual reference model for the behaviour simulation PV, which is based on a FE mesh. However, as explained in section 2.6, in the current industrial practice a shift of software environment occurs just after the generation of the mechanical model (see Figure 2.17), and the simulation model is generated into the same software environment where the behaviour simulation PV task is performed, i.e. the FEA solver. No standard file format is existing that allows FE meshing constraints to be exchanged, and therefore the change of software environment leads to losses of the information embedded into a mechanical model that can be useful to define constraints monitoring the generation of the FE mesh. In an ideal situation, the change of software environment should correspond to the transfer from the PV Interface In to the Current PV kernel (see Figure 2.16), so that the whole process devoted to the generation of the simulation model occurs in the PV Interface In, and the input of the Current PV kernel contains the FE mesh related to the mechanical problem under evaluation.

In this work, some efforts have been made to move towards the situation above described, considered as more efficient. The software environment based on the

mixed shape representation uses polyhedral models, and more precisely triangulations, which are shape models very close to surface FE meshes. Therefore, we developed some operators, which will be detailed in this chapter, able to transform a generic triangulation into an actual FE surface mesh. Indeed, as introduced in section 1.4.3, the triangular elements forming a FE surface mesh must have a shape as much regular as possible. Moreover in our approach, we explicitly define FE meshing constraints on the polyhedral model, in order to obtain a surface FE mesh that meets the hypotheses related to the studied mechanical problem. If shape information contained in the initial design model is available, e.g. geometric type of B-Rep surfaces and their adjacency relationships, the use of the mixed shape representation and the possibility of setting up multiple topological layers improve the efficiency of the meshing constraints definition. For example, it is possible to decompose the shape domain boundary so that the sharp edges of the shape model or the location of BCs are clearly highlighted. However, some efforts have been also made in order to define meshing constraints as attributes even if only the polyhedral representation is available.

In the case of surface shape domains, our developed operators would allow the user to obtain the actual FE mesh where performing the behaviour analysis. In the proposed approach, we always consider volume domains and we do not take into account idealizations of the model in lower dimensional sub-domains. Since we always deal with volume models, once meshed the boundary of the considered shape model, the generation of the desired tetrahedral FE mesh needs to be performed. At present, the current software prototype does not integrate a volume mesh generator. Therefore, once obtained the surface FE mesh, we still need to transfer it into another software environment that will generate the volume mesh (see Figure 2.18).

However, a proper mesh of the model boundary is a good starting point for the generation of the volume FE mesh. Indeed, most of the tetrahedral meshing techniques mesh first the boundary of the shape domain and, only later on, propagate the mesh toward the interior of the volume domain (see section 1.5.5). Moreover, all the hypotheses related to the mechanical problem to be solved have been translated into constraints expressed over the boundary of the domain, and have been already take into account when adapting the shape domain and generating the surface FE mesh. Therefore, no loss of information about the mechanical problem to solve occurs during the transfer process between the two different software environments.



**Figure 4.1:** FE mesh generation on a volume model: (a) the FE mesh does not respect the sharp edges of the shape model; (b) a FE mesh has been generated that respect the sharp edges of the shape model.

In this chapter, we will detail the developed shape operators that contribute to the generation of a surface FE mesh. They act through two different classes of constraints at the same time:

- a. Constraints defining the desired size and shape of the triangular faces of the FE mesh. These constraints are easily deduced by knowing the coordinates of mesh vertices, and therefore can be set directly on the polyhedron model, without the support of additional topological layers. For example, we can set either the maximum value of edge size allowed over the FE mesh or the minimum value of the angle that the mesh edges are allowed to form;
- b. Constraints that make use of the mixed shape representation to express the hypotheses related to the mechanical problem to be solved. We can further distinguish two kinds of constraints belonging to this class:
  - Discretization constraints allowing the mesh generator to place mesh nodes in the curvilinear space defining the boundary of the mechanical model. If a mixed shape representation is available, we explicit these constraints by defining proper decompositions of the model boundary, which allow feature edges to be highlighted over the shape model. In addition, it is possible to exploit CAD information about the geometry of B-Rep surfaces when placing mesh nodes. On the contrary, if only the polyhedral representation is available, we can consider the polyhedral model resulting from the simplification process as a shape reference for the generation of the surface FE mesh, as we will describe in section 4.2.

By setting these constraints, we avoid large discretization errors, because we are able to respect sharp edges of the model (see Figure 4.1) and constraint mesh nodes to be placed in the curvilinear space defining the boundary of the domain model;

- Constraints that allow algorithms to transfer the information about BCs from the mechanical model to the FE surface mesh. We are able to specify areas that correspond to locations of BCs by means of suited boundary decompositions. The meshing algorithms are monitored so that surface mesh elements do not cross the boundary of BCs locations.

Then, we are able to specify which are the surface mesh elements belonging to these areas, and hence we can attach to them the information about BCs.

## 4.2 Generation of FE meshing constraints based on the shape domain of the mechanical model

The shape domain of the mechanical model will be act as topological and geometric reference during the generation of the surface FE mesh. Two different cases can be taken into account:

- a. The mixed shape representation is available.

In this case, proper boundary decompositions can be defined thanks to the concept of multiple topological layers. We will constraint the algorithms generating the FE meshes to respect the boundary decompositions that have been defined in the various topological layers of the mixed shape representation. As introduced in chapter 3, at present, we have the topological representation of only one boundary decomposition, and therefore the topological constraints we set are simple. Involving many topological layers is part of perspectives.

From a geometric point of view, the CAD information about the geometry of B-Rep surfaces will be useful to place nodes on the surface FE mesh. It can be related either to the geometric parameters of analytical surfaces, such as planes, cylinders, cones, or to the parametric parameters  $N_u$ ,  $N_v$  (see Eq. (1.1)) of the NURBS surfaces.

- b. No mixed shape representation is available.

In this case, as introduced in section 4.1, the only reference for the successive stages of mesh generation is the shape of the polyhedron resulting from the previous shape simplification process, which we name **base polyhedron**. Indeed, we assume that the simplifications performed on the initial shape model have produced a valid shape where generating the surface FE mesh. Therefore, when generating the surface FE mesh, we use the polyhedron model as topological and geometric reference to define FE meshing constraints.

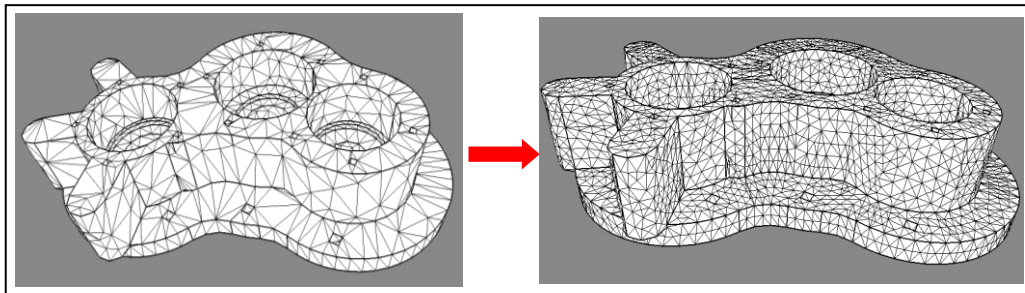
To this end, we have implemented a data structure that allows the operators to maintain the correspondence between nodes of the surface FE mesh and faces of the base polyhedron. We attach to each node of the surface FE mesh the information about the corresponding face of the base polyhedron by means of the node's barycentric coordinates on that face. In the case where the node lies on an edge of the base polyhedron, we will attach to it the information related to both faces of the base polyhedron adjacent to this edge.

### 4.3 Mesh enrichment operator

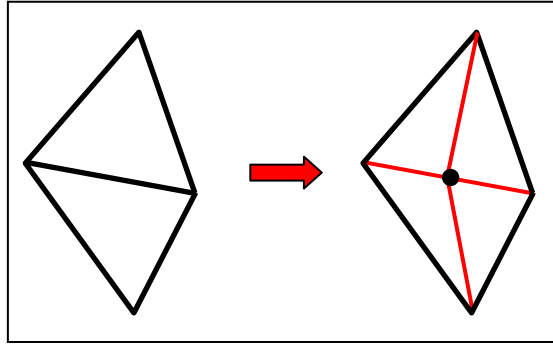
The decimation operators described in chapter 3 allows one to simplify the shape domain associated to an initial design model, while taking into account simultaneously the mechanical hypotheses related to the problem of interest. However, the map of FE sizes that we use as a priori mechanical criterion defines a geometric envelope inside which the polyhedron has to stay, and no care about the shape quality of triangles is taken when decimating the polyhedron. Therefore, the polyhedron resulting from a decimation process cannot be considered as a surface FE mesh and some additional treatments are necessary to get a proper FE mesh.

The first operator to be applied aims at increasing the number of vertices of the polyhedral model to meet the FE discretization requirements. Indeed, the polyhedron resulting from a shape simplification process contains triangles whose dimensions, in planar or nearly planar areas, are larger than those required by the map of FE sizes (see Figure 4.2). Only once enriched the polyhedron, actual mesh optimization operators (described in section 4.3) can be applied.

The enrichment operator allows the user to target the desired size of the surface mesh elements by specifying a threshold value of the edge length. At the moment, the algorithm has been only implemented for generating a surface FE mesh whose elements have a constant size, so a single value  $l_e$  is provided for the desired edge length. When applying the operator, all the polyhedron edges longer than  $l_e$  are subdivided. Whenever an edge is cut, new polyhedron entities are created. A new vertex is placed in the middle of the cut edge; while each one of the existing triangles is split into two new triangles (see Figure 4.3). The enrichment process is re-iterated until all edges get shorter than  $l_e$ . In the future, the FE map of sizes should be the effective criterion to monitor arbitrary mesh generation.



**Figure 4.2:** Example of mesh enrichment operator.



**Figure 4.3:** Enrichment operator cutting an edge of the polyhedron.

The information associated to the initial shape model can be propagated on the enriched one. The propagation operator distinguishes the two situations detailed in section 4.2.

If the mixed shape representation is available, the high-level information embedded into the mechanical model is preserved during the enrichment process. Algorithms constrain the enriched polyhedron to respect the boundary decompositions that have been defined in the topological layers of the mixed shape representation. Indeed, whenever an edge is cut, for each topological layer we aim at preserving the related partition, which is updated by adding the new polyhedron entities created and removing the old ones. If the edge to be cut belongs to a polyedge, the enrichment process updates the related polyedge and the partitions on both sides of the cut edge.

In the case where no mixed shape representation is available, we use the base polyhedron as topological reference during the generation of the surface FE mesh. As introduced in section 4.2, we compute, for each node added onto the mesh, its barycentric coordinates on the face(s) of the base polyhedron on which the node lies. In this way, a correspondence is achieved between nodes of the FE meshes and the base polyhedron.

#### 4.4 Operators for FE mesh optimization

Operators aiming at optimizing the surface FE mesh can be subdivided in two main categories:

- **Smoothing operators**, which reposition nodes in order to improve the mesh quality;
- **Swapping operators**, which change the connections among nodes without modifying their position.

For each one of the two categories, some algorithms have been developed.

#### 4.4.1 Smoothing operator

A smoothing operator balances the position of polyhedron vertices to obtain a more regular surface mesh. We developed a smoothing algorithm based on the force density method [SCH74]. It is based on the method proposed in [Noe94] for the optimization of surface FE meshes forming an open domain.

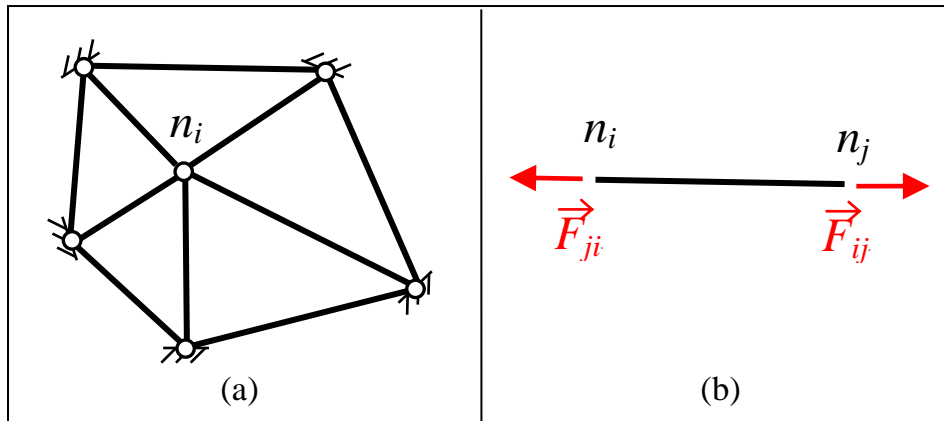
For each node  $n_i$  of the surface mesh, the edges attached to it are assimilated to a bar network, whose bars are under traction, such that:

- The mesh nodes coincide with the nodes of the network;
- The mesh edges coincide with the bars of the network.

A bar  $B$  creates a mechanical interaction between its extreme nodes  $n_i$  and  $n_j$ , whose coordinates are  $\overrightarrow{X}_i$  and  $\overrightarrow{X}_j$ , which can be characterized by:

$$\overrightarrow{F}_{ji} = q_j(\overrightarrow{X}_j - \overrightarrow{X}_i), \quad (4.1)$$

where  $q_j$  is called force density of the bar  $B$ ,  $q_j \geq 0$ . Therefore,  $\overrightarrow{F}_{ji}$  represents the interaction between  $n_i$  and  $n_j$ . It is proportional to the bar length and defines the traction force in this bar. Figure 4.4 shows the bar network associated to a mesh node  $n_i$ , and the mechanical interaction occurring between  $n_i$  and  $n_j$ .



**Figure 4.4:** (a) Bar network which the polyhedron star surrounding a node  $n_i$  is assimilated to; (b) Mechanical behaviour of each bar belonging to the network.

A new position  $\overrightarrow{X}_i'$  is searched for the node  $n_i$  under examination, while the locations of the nodes  $\overrightarrow{X}_j$  neighbouring the considering one are considered as fixed (see Figure 4.4(a)). The new location of the node will minimize the external force  $\overrightarrow{R}_i$  required to get its static equilibrium. The static equilibrium of a node is guaranteed if:

$$\sum_{j=1}^m \overrightarrow{F}_{ji} + \overrightarrow{R}_i = \vec{0}, \quad (4.2)$$

where  $j \in \{1, \dots, m\}$  spans all the nodes adjacent to  $n_i$ , and  $\overrightarrow{F}_{ji}$  is given by Eq. (4.1). Then, we can convert Eq. (4.2) in:

$$\overrightarrow{X}_i = \frac{\sum_{j=1}^m q_j \overrightarrow{X}_j + \overrightarrow{R}_i}{\sum_{j=1}^m q_j}. \quad (4.3)$$

Therefore, the position of static equilibrium for each node  $n_i$  depends on:

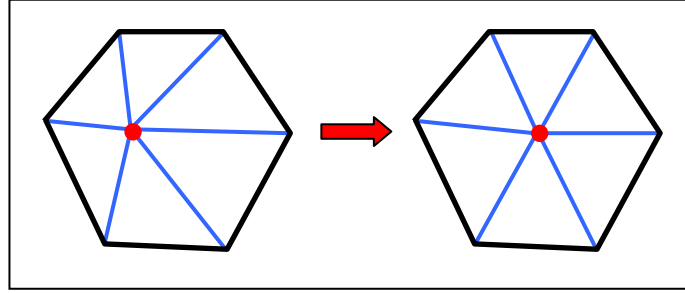
- the force densities  $q_j$  of the bars linked to the node  $n_i$ ;
- the locations of the nodes  $n_j$  adjacent to  $n_i$ ;
- the external force  $\overrightarrow{R}_i$  applied to the bar network.

To provide an isotropic mesh behaviour, a constant unitary density for all the nodes of the polyhedron, i.e. for each node  $n_j$ ,  $q_j = 1$ . This is consistent if the objective is to obtain an equilateral mesh everywhere over the boundary domain. Therefore, Eq. (4.3) can be simplified to produce:

$$\overrightarrow{X}_i = \frac{\sum_{j=1}^m \overrightarrow{X}_j + \overrightarrow{R}_i}{m}. \quad (4.4)$$

As mentioned before, when studying the static equilibrium of the bar network associated to a node  $n_i$ , we assume the locations of its adjacent nodes  $n_j$  to be fixed. Therefore, the only parameter on which we can act to determine a new position  $\overrightarrow{X}_i'$  for the node  $n_i$  is the external force  $\overrightarrow{R}_i$  that is applied to the bar network. The more regular (i.e. equilateral) the triangles around  $n_i$ , the lower the force  $\overrightarrow{R}_i$ . When the bar network reaches some symmetric configurations,  $\overrightarrow{R}_i$  may vanish.





**Figure 4.5:** Mesh smoothing operator balancing the position of mesh nodes.

Due to the above consideration, the new location  $\overline{X}'_i$  of the node  $n_i$  should minimize the external force  $\overline{R}_i$ . However, we cannot just set it to  $0$ , since we have to take into account the surface  $S$  from which the FE mesh is generated, which is related to the shape of the mechanical model. This shape acts as a constraint when moving mesh nodes and the equivalent mechanical problem becomes equivalent to slide the node  $n_i$  without friction on this reference surface. This implies that the node  $n_i$  is forced to move in the parametric space of the domain boundary. Only in the case where the polyhedral star lies on a planar domain, the resulting  $\overline{R}'_i$  will be null and the new location  $\overline{X}'_i$  of the node  $n_i$  will coincide with the barycentre of the planar set of triangles attached to node  $n_i$  (see Figure 4.5).

If a mixed shape representation is available, the definition of FE meshing constraints profits by the high-level information embedded into the shape of the mechanical model. Three main configurations may occur:

- a. The node  $n_i$  lies inside a partition, i.e. neither it is part of a polyedge nor it coincides with a HLT-Vertex. Since the geometric information related to the partition is known, when moving  $n_i$  in the parametric space of the domain boundary  $S$ , we can reason into the parametric space  $\Omega(u_i, v_i)$  of the surface  $S$  underlying it. Therefore, the position of  $n_i$ , can be expressed as a function of the parametric space  $\Omega(u_i, v_i)$  coordinates, i.e.  $\overline{X}_i = \overline{P}(u_i, v_i)$ . We define an orthogonal reference frame in the plane tangent to  $S$  at the point  $\overline{P}(u_i, v_i)$ , having  $\overline{T}_i^1$  and  $\overline{T}_i^2$  as basis vectors and  $\overline{N}_i$  as normal.  $\overline{T}_i^1$  and  $\overline{T}_i^2$  are the partial derivatives of  $\overline{P}(u_i, v_i)$  with respect to  $u$  and  $v$  respectively, and  $\overline{N}_i$  is the normal vector at that point. Then, when estimating the position  $\overline{X}'_i = \overline{P}'(u_i, v_i)$ , the constraints prescribed in order that  $n_i$  moves along  $S$  are:

$$\overline{R}_i \cdot \overline{T}_i^j = 0, \quad (4.5)$$

where  $j \in \{1, 2\}$ . This means that the external force  $\overline{R}'_i$  in the new position  $\overline{X}'_i$  will be directed along the normal  $\overline{N}_i$ .

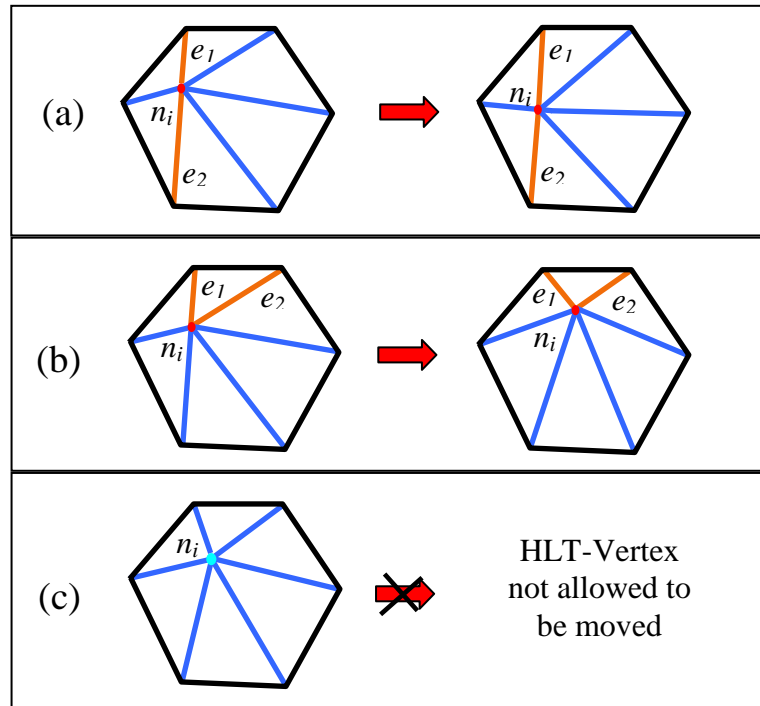
- b. The node  $n_i$  is part of a valid polyedge in the topological layer devoted to the definition of FE meshing constraints, but it does not coincide with a HLT-Vertex. In this configuration, we will find that two edges  $e_1$  and  $e_2$ , adjacent to  $n_i$  are parts of the polyedge. Two situations may occur:

- $e_1$  and  $e_2$  are collinear. Then,  $n_i$  will move along their direction  $\vec{T}_e$  (see Figure 4.6(a)), i.e.:

$$\vec{R}_i \cdot \vec{T}_e = 0 . \quad (4.6)$$

- $e_1$  and  $e_2$  are not collinear. Then,  $n_i$  will be forced to move according to one of their directions,  $\vec{T}_{e_1}$  and  $\vec{T}_{e_2}$  (see Figure 4.6(b)). Actually, we apply Eq. (4.6) for both directions and we choose moving along that providing the smallest value for  $\vec{R}_i'$  ;

- c. The node  $n_i$  is associated to a valid HLT-Vertex. In this case, it is not allowed to move (see Figure 4.6(c)).



**Figure 4.6:** Smoothing in presence of constraints prescribed by some topological layers of the mixed shape representation: (a) Two edges  $e_1$  and  $e_2$  concurrent at the node  $n_i$  to be moved are part of a valid polyedge and are collinear, while  $n_i$  is not associated to any valid HLT-Vertex; (b) Two edges  $e_1$  and  $e_2$  concurrent into the node  $n_i$  to be moved are part of a valid polyedge and are not collinear, while  $n_i$  is not associated to any HLT-Vertex; (c) The node  $n_i$  is associated to a valid HLT-Vertex.

If only the polyhedral representation is available, the meshing constraints acting during the smoothing algorithm are of the same kind of those described above, in the case where the mixed shape representation was available, except that the shape reference for retrieving meshing constraints is the base polyhedron. Therefore, when replacing nodes, we move in the piecewise parametric space defined by the triangular faces of the base polyhedron, and have to take into account its edges. Here, we assume that a mesh enrichment operator has been previously applied, and therefore we know which face of the base polyhedron each node  $n_i$  belongs to. Three main configurations can occur:

- a. The node  $n_i$  lies inside a face  $f$  of the base polyhedron. In this case, we constrain it to move onto the planar surface of  $f$ . The position  $\overline{X}_i'$  will be estimated so that  $\overline{R}_i'$  vanishes, i.e.  $n_i$  will move toward the barycentre of the bar network related to its attached triangles;
- b. The node  $n_i$  belongs to an edge  $e$  of the base polyhedron, shared by two faces  $f_1$  and  $f_2$ . In this case, an additional constraint is considered. The node  $n_i$  is free to move over the parametric space defined by the boundary of the base polyhedron (see Eq. (4.6)) only in the case where:

$$H_{1,2} \leq H_{\max} , \quad (4.7)$$

where  $H_{1,2}$  is the mean curvature at the edge  $e_b$ , computed with Eq. (3.7), and  $H_{\max}$  is a threshold value of the mean curvature set by the user. In contrast, if  $H_{1,2}$  turn out to be too large,  $n_i$  will be constrained to move along the direction defined by  $e$ .

- c. The node  $n_i$  coincides with a vertex  $v$  of the base polyhedron. Then, the mean curvature  $H_{k,k+1}$  at the edges adjacent to  $v$  is estimated, still using Eq. (3.7), and compared with the user-defined threshold value  $H_{\max}$ . The node  $n_i$  will be allowed to move only along directions of edges whose  $H_{k,k+1}$  is smaller than  $H_{\max}$ .

An iteration of the mesh smoothing algorithm corresponds to solve the mechanical local problem for all the mesh nodes. At present, the effective convergence of the nodes in a final position is not checked. However, we have noticed that, when several iterations are performed, the nodes significantly move only during the first iterations. Therefore, a number maximum of iterations is set by the user.

#### 4.4.2 Swapping operator

The swapping algorithm that we use has been developed based on the one presented in [LLV05]. It swaps polyhedron edges  $e$ , shared by two faces  $f_1$  and  $f_2$  of the mesh, while trying to minimize a cost function. By default, the minimization of our cost function  $F_{eq}$  will aim at providing a mesh whose triangles are as equilateral as possible (see Figure 4.7). To this end, we define  $F_{eq}$  as the value of the largest angle

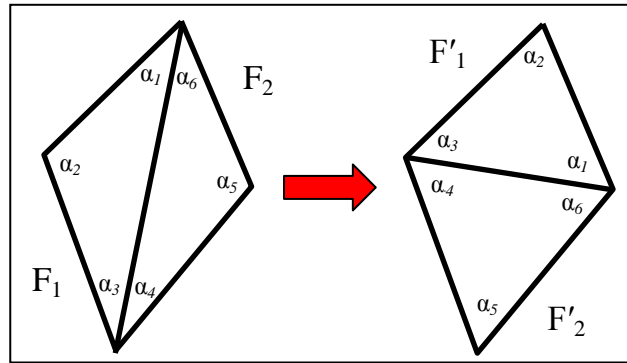
$\alpha_i$  at the vertices of the faces  $f_1$  and  $f_2$ . The function  $F_{eq}$  is computed before and after the swapping of an edge  $e$ , and the configuration is chosen that minimizes it:

$$\min F_{eq} = \min (\max \alpha_i), \quad (4.8)$$

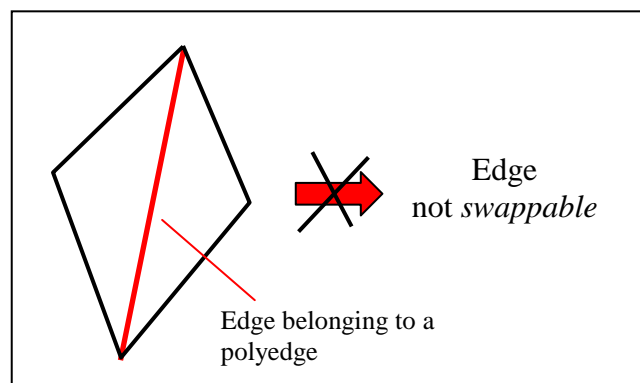
with  $i \in \{1, \dots, 6\}$ . Therefore, we swap an edge  $e$  only if the swapping operation leads to diminish  $F_{eq}$ . The swapping operator is iteratively applied to all edges of the mesh until no more edge can be swapped.

Also for this operator, additional constraints are considered in order to take into account the shape of the mechanical model and the corresponding discretization constraints.

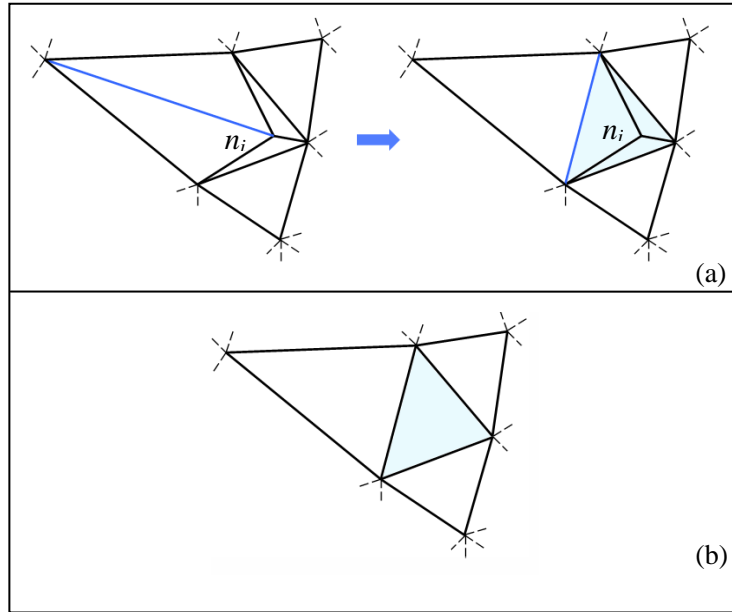
If the mixed shape representation is available, we will be constrained by the boundary decomposition defined in topological layers devoted to the definition of FE meshing constraints. This means that mesh edges that are part of polyedges will not be swapped. In this way, we can preserve feature edges (see Figure 4.8).



**Figure 4.7:** Example of edge swapping and parameters used by the function  $F_e$ .



**Figure 4.8:** The swapping operator does not swap edges belonging to polyedges.



**Figure 4.9:** (a) The edge swapping operator has led to a configuration where a node  $n_i$  has only three faces adjacent; (b) The node  $n_i$  is removed and the gap produced in the mesh is remeshed with a triangular face.

If no mixed shape representation is provided, the edge swapping operator will be constrained by the base polyhedron so that the shape of the base polyhedron is preserved. An edge  $e$  lying on an edge  $e_b$  of the base polyhedron will be swapped only if Eq. (3.7) is satisfied, where  $H_{1,2}$  is still the mean curvature at the edge  $e_b$ , and  $H_{max}$  is the user-desired threshold.

Edge swapping could lead to mesh configurations where a node has only three edges adjacent (see Fig. 4.9(a)). This configuration is not desirable in a surface FE mesh since it does not contribute to the regularity of the mesh, which instead should be formed by triangles as equilateral as possible. To overcome this problem, an algorithm has been developed, which acts after that an iteration of the swapping operator has been performed (each swapping operation corresponds to apply once the swapping operator to all mesh edges). The developed algorithm allows one to suppress all the mesh nodes  $n_i$  that have only three edges adjacent and remesh the gap in the mesh produced by the node removal with a triangular face (see Fig. 4.9(b)).

## 4.5 Combination of operators for preparing a surface FE mesh

In our approach, the final surface FE mesh will be obtained through a combination of the operators described up to here:

- a. The first step consists in applying a combination of the enrichment and swapping operators, described in sections 4.3 and 4.4.2 respectively. Iterations of mesh enrichment and swapping alternate. They will be constrained by parameters that the user gives as input: the value  $l_e$  of the

desired edge size will constrain the enrichment operator, and the maximum angle  $\alpha_{max}$  at edges will allow to compute the threshold value of the mean curvature  $H_{k,k+1}$ ;

When applying the swapping operator, constraints detailed at section 4.4.2 are considered, which allow the shape of the model to be preserved. In addition, at this stage, in the case where no mixed shape representation is available, and therefore swapping constraints are defined on the base polyhedron, the swapping of edges  $e$  that lie on edges  $e_b$  belonging to the base polyhedron is not allowed at all.

After an iteration of the swapping algorithm, nodes  $n_i$  having only three edges adjacent are suppressed, as described in section 4.4.2.

- b. Actually, the enrichment operator splits edges until some edges exist, whose size is longer than  $l_e$ . Anyway, several iterations of the mesh enrichment operator could produce a surface FE mesh having some edges shorter than the prescribed size  $l_e$ . Therefore, after the enrichment and swapping processes described at step (a), a skin transformation operator (see section 3.1.1) is applied to the polyhedral model. It removes polyhedral vertices that are adjacent to edges shorter than  $l_e$  and remeshes the polyhedron, possibly creating edges having better suited sizes. The candidate vertices to the removal are sorted according to an increasing size of their adjacent edges, i.e. vertices that are the extremities of the smallest edges will be removed first. The remeshing criterion employed is the equilaterality one, since the objective is to obtain a surface FE mesh as regular as possible. The edge size is constrained during the mesh simplification, so that we are not allowed to create remeshing configurations having edges longer than  $l_e$ . To improve the mesh regularity, we can also constraint the minimum angle that edges have to form when remeshing.
- c. Once obtained a mesh whose edges respect the desired size  $l_e$ , the shape of mesh triangular elements can be optimized with alternate iterations of smoothing and swapping algorithms, described in section 4.4. This combined process allows one to restore a “local” Delaunay property on the surface mesh [ILT\*01]. In the case where no mixed shape representation is available, the swapping is still constrained by the base polyhedron but, differently from what has been described at step (a), edges  $e$  lying on edges  $e_b$  of the base polyhedron are allowed to be swapped, according to the method detailed in section 4.4.2.

Once obtained the surface FE mesh, if we have performed the meshing operators constrained by specific topological layers defining BCs, we will have a finite number of FE triangles for each partition of the considered layer. Then, we can attach to them the information about BCs and transfer the obtained model to the software environment where the volume FE mesh will be generated through a file format suited to exchange data about meshes and BCs. At present, the automatic association of BCs to the elements of the surface mesh has not been performed yet, but it is part of current perspectives.







# A posteriori criterion characterizing the influence of shape variations on FE results

*A posteriori criteria support the choice of valid simplifications performed on a shape model when moving from design to behaviour simulation PV. To this end, an a posteriori mechanical criterion is here introduced, which evaluates the impact of a priori shape simplifications, performed over the shape of a design model, on the FE analysis results related to the corresponding simulation model. The developed criterion, which can be applied to problems of FE static analysis of linear elastic structures or FE thermal problems for stationary linear conduction, can be integrated into an adaptive modelling process. Therefore, a priori choices performed during the preparation of a simulation model can be validated or, at the contrary, the shape of the simulation model needs to be adapted in order to provide more accurate simulation results.*

---

### 5.1 Advantages of using an a posteriori criterion

As highlighted in section 2.4.3, a priori criteria are not guaranteeing the choice of the correct simplifications to perform on a shape model, and the use of an a posteriori criterion can help in overcoming such limitations. It can be applied once some FE results are available and therefore can be based on parameters that are objective, since they are directly related to the mechanical behaviour of the analysis model.

Consequently, in our approach we have chosen to tune the a priori mechanical hypotheses related to a component shape by applying an a posteriori mechanical criterion. Such a criterion allows the evaluation of the influence of shape simplifications on FE results, at a global scale [FML\*06b]. It can be applied to the shape sub-domains removed during a shape simplification process with the support of some a priori mechanical criteria, and whose shape models have been stored.

The a posteriori mechanical criterion gives an indication of the mechanical influence of each a priori simplification detail. In this way, choices made during the a priori stage of model preparation are evaluated. They can be either validated or rejected, thus requiring a modification of the shape of the behaviour simulation model to provide more accurate simulation results.

The use of the developed a posteriori mechanical criterion guarantees to get accurate information about the mechanical behaviour of the model while using a shape support simple enough if compared to the shape of the initial design model. Indeed, building the FE mesh on a simplified shape support provides remarkable time and complexity reduction. At the same time, the application of the a posteriori mechanical criterion is not expensive both from time and computation points of view, especially if it is incorporated into an automated process.

## **5.2 Principles of the a posteriori mechanical criterion**

Currently, the developed a posteriori criterion can be applied to problems of FE static computation of linear elastic structures or FE thermal problems for stationary linear conduction.

The criterion has two main advantages:

- It can be applied to any type of simplification, i.e. it does not matter whether the analysis shape sub-domain is derived from to a continuous shape variation or not;
- It can be applied to sub-domains having arbitrary size and shape, i.e. it works with large shape sub-domains too.

These characteristics allow the use of the a posteriori criterion to be generalized to all situations where:

- Some FE results coming from a linear static analysis are available, which are related to a simplified version of the design model;
- The original shape of the design model has been partially modified, and the modifications consist in the addition or subtraction of shape sub-domains having arbitrary shape and size.

The generality of these hypotheses and the broad application field of the a posteriori criterion make it a useful tool in various scenarios and phases of the product development process, as discussed in details in the next chapter.

### 5.2.1 Error on the FE solution due to sub-domain removals

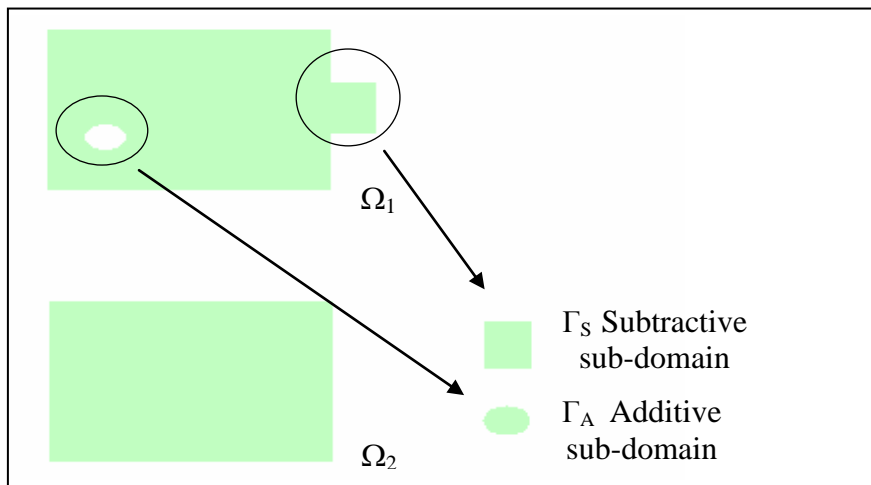
We briefly explain how the a posteriori criterion is computed by referring to the example shown in Figure 5.1.  $\Omega_1$  is the initial shape domain, i.e. the one before the sub-domain removal, and  $\Omega_2$  indicates the simplified one. Shape changes, corresponding to the differences between  $\Omega_1$  and  $\Omega_2$ , can be classified according to the variation of the geometric domain:

- Shape sub-domains  $\Gamma_A$  of additive type;
- Shape sub-domains  $\Gamma_S$  of subtractive type.

Let us assume that the solution of the initial FE problem over the domain  $\Omega_1$  returns the displacement field  $\vec{U}_1$ , the stress field  $\bar{\bar{\sigma}}_1$  and the strain field  $\bar{\bar{\varepsilon}}_1$ . We call  $\partial\Omega_1$  the boundary of  $\Omega_1$ . Accordingly,  $\vec{U}_2$ ,  $\bar{\bar{\sigma}}_2$  and  $\bar{\bar{\varepsilon}}_2$  are the solution fields of the simplified FE problem on the domain  $\Omega_2$ , having  $\partial\Omega_2$  as boundary.

We can assume that the simplified FE problem matches exactly the first one, i.e. the error  $e$  between the FE solutions related to  $\Omega_1$  and to  $\Omega_2$  vanishes, if:

- On the intersection of the two domains, i.e.  $\partial(\Omega_1 \cap \Omega_2)$ , the initial and simplified problem solutions are equal;
- Over the sub-domain  $\Gamma_A$ , the stress and strain fields,  $\bar{\bar{\sigma}}_2$  and  $\bar{\bar{\varepsilon}}_2$  respectively, are equal to zero;
- Over the sub-domain  $\Gamma_S$ , the stress and strain fields,  $\bar{\bar{\sigma}}_1$  and  $\bar{\bar{\varepsilon}}_1$  respectively, are equal to zero.



**Figure 5.1:** Simplification example: an initial domain  $\Omega_1$  and the corresponding simplified domain  $\Omega_2$ . To produce  $\Omega_2$ , the sub-domain  $\Gamma_A$  is added and the sub-domain  $\Gamma_S$  is removed from  $\Omega_1$ .

To estimate the influence of the above shape modifications, we need to assess:

- The difference  $(\vec{U}_1 - \vec{U}_2)$  over the common sub-domain  $(\Omega_1 \cap \Omega_2)$ ;
- The stress field  $\bar{\bar{\sigma}}_2$  over  $\Gamma_A$ ;
- The stress field  $\bar{\bar{\sigma}}_1$  over  $\Gamma_S$ .

The above quantities are measured by using an energy norm (see section 1.8.1). The corresponding error,  $e$ , is given by:

$$2e^2 = \int_{\Omega_1 \cap \Omega_2} (\bar{\bar{\sigma}}_1 - \bar{\bar{\sigma}}_2) : (\bar{\bar{\varepsilon}}_1 - \bar{\bar{\varepsilon}}_2) d\Omega + \int_{\Gamma_A} \bar{\bar{\sigma}}_2 : \bar{\bar{\varepsilon}}_2 d\Omega + \int_{\Gamma_S} \bar{\bar{\sigma}}_1 : \bar{\bar{\varepsilon}}_1 d\Omega. \quad (5.1)$$

We can simplify Eq. (5.1), assuming that the boundaries of the simplified sub-domains  $\Gamma_A$  and  $\Gamma_S$  are free, i.e. no BCs are applied. Here,  $\vec{n}_\Omega$  is the normal vector pointing outward  $\Omega$ , and  $\vec{f}_d$  is the volumetric fields of forces applied to  $\Omega$ . Therefore:

$$2e^2 = \int_{\Gamma_A} \vec{f}_d \cdot \vec{U}_2 d\Omega + \int_{\Gamma_S} \vec{f}_d \cdot \vec{U}_1 d\Omega + \int_{\partial\Gamma_A} [\bar{\bar{\sigma}}_2 \cdot \vec{n}_{\Gamma_A}] \cdot \vec{U}_1 d\partial\Omega + \int_{\partial\Gamma_S \cap \partial\Omega} [\bar{\bar{\sigma}}_1 \cdot \vec{n}_{\Gamma_S}] \cdot \vec{U}_2 d\partial\Omega. \quad (5.2)$$

**Proof:**

Here, we provide a proof of the passage from Eq. (5.1) to Eq. (5.2), related to an additive sub-domain  $\Gamma_A$ . Eq. (5.1) is reduced to:

$$2e^2 = \int_{\Omega_1} (\bar{\bar{\sigma}}_1 - \bar{\bar{\sigma}}_2) : (\bar{\bar{\varepsilon}}_1 - \bar{\bar{\varepsilon}}_2) d\Omega + \int_{\Gamma_A} \bar{\bar{\sigma}}_2 : \bar{\bar{\varepsilon}}_2 d\Omega. \quad (5.3)$$

Considering Green's theorem and integration by parts, we obtain Eq. (5.4):

$$2e^2 = \int_{\Omega_1} \text{div}[\bar{\bar{\sigma}}_1 - \bar{\bar{\sigma}}_2] \cdot (\vec{U}_2 - \vec{U}_1) d\Omega - \int_{\Gamma_A} \text{div}[\bar{\bar{\sigma}}_2] \cdot \vec{U}_2 d\Omega + \int_{\partial\Omega_1} [(\bar{\bar{\sigma}}_1 - \bar{\bar{\sigma}}_2) \cdot \vec{n}_{\Omega_1}] \cdot (\vec{U}_1 - \vec{U}_2) d\partial\Omega + \int_{\partial\Gamma_A} [\bar{\bar{\sigma}}_2 \cdot \vec{n}_{\Gamma_A}] \cdot \vec{U}_2 d\partial\Omega. \quad (5.4)$$

We can subdivide the boundary of the domain into two parts  $\partial\Omega^*$  or  $\partial\Omega^{**}$  where, respectively, the surface traction  $\vec{F}_d$  and the displacement  $\vec{U}_d$  are given. On each domain, local equations connect the stresses, the volumetric field of forces  $\vec{f}_d$  and the boundary load  $\vec{F}_d$ :

$$\operatorname{div}[\bar{\bar{\sigma}}] = \vec{f}_d \quad \text{on } \Omega; \quad (5.5)$$

$$\bar{\bar{\sigma}} \cdot \vec{n}_\Omega = \vec{F}_d \quad \text{on } \partial\Omega^*; \quad (5.6)$$

$$\vec{U} = \vec{U}_d \quad \partial\Omega^{**}. \quad (5.7)$$

Eq. (5.5) gives  $\operatorname{div}[\bar{\bar{\sigma}}_1] = \operatorname{div}[\bar{\bar{\sigma}}_2] = -\vec{f}_d$  on  $\Omega_1$ , and therefore the first integral of Eq. (5.4) vanishes. The boundary  $\partial\Omega_1$  can be subdivided in  $\partial\Omega_1 \cap \partial\Omega_2$  and  $\partial\Omega_1 \cap \partial\Gamma_A$ . On  $\partial\Omega_1 \cap \partial\Gamma_A$ ,  $\vec{n}_{\Omega_1}$  is directed oppositely to  $\vec{n}_{\Gamma_A}$ . We can transform (5.4) in:

$$\begin{aligned} 2e^2 = & \int_{\Gamma_A} \vec{f}_d \cdot \vec{U}_2 d\Omega + \int_{\partial\Omega_1 \cap \partial\Omega_2} [(\bar{\bar{\sigma}}_1 - \bar{\bar{\sigma}}_2) \cdot \vec{n}_{\Omega_1}] \cdot (\vec{U}_1 - \vec{U}_2) d\partial\Omega + \\ & - \int_{\partial\Omega_1 \cap \partial\Gamma_A} [(\bar{\bar{\sigma}}_1 - \bar{\bar{\sigma}}_2) \cdot \vec{n}_{\Gamma_A}] \cdot (\vec{U}_1 - \vec{U}_2) d\partial\Omega + \int_{\partial\Gamma_A} [\bar{\bar{\sigma}}_2 \cdot \vec{n}_{\Gamma_A}] \cdot \vec{U}_2 d\partial\Omega. \end{aligned} \quad (5.8)$$

This expression can be simplified since:

- On  $\partial\Omega_1 \cap \partial\Omega_2$ , the BCs involve  $\vec{U}_1 = \vec{U}_2$  or  $\bar{\bar{\sigma}}_1 \cdot \vec{n}_{\Omega_1} = \bar{\bar{\sigma}}_2 \cdot \vec{n}_{\Omega_2}$  and the second integral of (5.8) vanishes;
- We consider as free the boundary of the removed sub-domain  $\Gamma_A$ , i.e.  $\bar{\bar{\sigma}}_1 \cdot \vec{n}_{\Gamma_A} = \vec{0}$ ;
- For additive sub-domains  $\Gamma_A$ , we have  $\partial\Omega_1 \cap \partial\Gamma_A \equiv \partial\Gamma_A$ .

Therefore, we will obtain:

$$2e^2 = \int_{\Gamma_A} \vec{f}_d \cdot \vec{U}_2 d\Omega + \int_{\partial\Gamma_A} \bar{\bar{\sigma}}_2 \cdot \vec{n}_{\Gamma_A} \cdot \vec{U}_1 d\partial\Omega. \quad (5.9)$$

When both additive and subtractive sub-domains are considered, a similar demonstration will provide the more general Eq. (5.2).

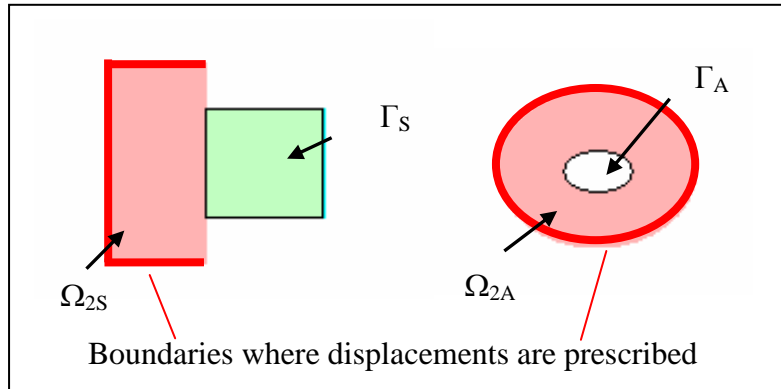
The error  $e$  is an absolute error. A more meaningful relative error,  $\eta$ , can be expressed in terms of the strain energy of the problem, as:

$$\eta^2 = \frac{e^2}{\frac{1}{2} \int_{\Omega_1} Tr[\bar{\bar{\sigma}}_1 : \bar{\bar{\varepsilon}}_1] d\Omega} \cong \frac{e^2}{\frac{1}{2} \int_{\Omega_2} Tr[\bar{\bar{\sigma}}_2 : \bar{\bar{\varepsilon}}_2] d\Omega} \quad (5.10)$$

### 5.2.2 Error approximation by means of a FE local problem

The equations introduced in the above section express the error generated by shape sub-domain removals on the FE results over the domain  $\Omega_2$ . It is clear that the computation of the error  $e$  based on Eq. (5.2) implies the FE solution over the initial domain  $\Omega_1$ . Now, since our goal is to avoid solving the initial FE problem, this initial solution is unknown. Therefore, we estimate it by using a local FE computation. To this end, sub-domains  $\Omega_{2S}$  or  $\Omega_{2A}$ , surrounding a subtractive or an additive sub-domain respectively, are retrieved. Figure 5.2 shows an example, with reference to the sub-domains removed in Figure 5.1.

From a geometric point of view, the FE mesh associated to the local FE problem is formed by the Boolean union of the FE meshes related to the removed sub-domain and its neighbourhood. According to the removed sub-domain's type we have  $\Delta_S = \Gamma_S \cup \Omega_{2S}$  or  $\Delta_A = \Gamma_A \cup \Omega_{2A}$ .



**Figure 5.2:** Neighbouring sub-domains,  $\Omega_{2S}$  and  $\Omega_{2A}$ , for the local FE computations around  $\Gamma_S$  and  $\Gamma_A$  respectively, related to the example of Figure 5.1.

From a mechanical point of view, we calculate the stiffness of the FE mesh associated to the local FE problem with different methods, depending on whether the removed sub-domain is subtractive or additive. The case of shape simplifications of subtractive type implies a reduction of the initial model  $\Omega_1$  following the subtraction of a sub-domain  $\Gamma_S$ , where  $\Gamma_S$  and the neighbouring sub-domain  $\Omega_{2S}$  are adjacent. The stiffness of  $\Delta_S$  is computed as the sum of the stiffnesses of the FE meshes of  $\Gamma_S$  and  $\Omega_{2S}$ . In contrast, the case of shape simplifications of additive type, implies an increase of the initial model  $\Omega_1$  due to the addition of a sub-domain  $\Gamma_A$ .  $\Gamma_A$  is completely immersed in its neighbouring sub-domain  $\Omega_{2A}$  and the stiffness of  $\Delta_A$  is computed as the difference between the stiffnesses of the FE meshes of  $\Omega_{2A}$  and  $\Gamma_A$ .

The BCs of the FE local problems over  $\Delta_S$  and  $\Delta_A$  are given by the displacement field  $\vec{U}_2$ , which results from the FE computation over the simplified problem  $\Omega_2$ . Bold lines in Figure 5.2 correspond to the boundaries where displacements from the field  $\vec{U}_2$  are applied.

FE local computations allow us to give an estimation of the relative error,  $e_{est}$ , as:

$$2e_{est}^2 = \int_{\Gamma_A} \vec{f}_d \cdot \vec{U}_2 d\Omega + \int_{\Gamma_S} \vec{f}_d \cdot \vec{U}_{1est} d\Omega + \int_{\partial\Gamma_A} [\bar{\bar{\sigma}}_2 \cdot \vec{n}_{\Gamma_A}] \cdot \vec{U}_{1est} d\partial\Omega + \int_{\partial\Gamma_S \cap \partial\Omega} [\bar{\bar{\sigma}}_{1est} \cdot \vec{n}_{\Gamma_S}] \cdot \vec{U}_2 d\partial\Omega. \quad (5.11)$$

The corresponding relative value  $\eta_{est}$  is the proposed a posteriori mechanical criterion, which is evaluated over the domain  $\Omega_2$  as,

$$\eta_{est}^2 \cong \frac{e_{est}^2}{\frac{1}{2} \int_{\Omega_2} Tr[\bar{\bar{\sigma}}_2 : \bar{\bar{\varepsilon}}_2] d\Omega}. \quad (5.12)$$

We note that the aim of the a posteriori criterion is to provide a global indication, i.e. to estimate the influence of the removal of a shape sub-domain on the distribution of strain energy over the entire model. Therefore, it mainly addresses the compliance of the structure. No precise information is intended to be provided about the influence of the sub-domain removal on the stresses or strains in its neighbourhood. However, the results of the FE local computation in the neighbourhood of each removed sub-domain, i.e. on  $\Delta_S$  or  $\Delta_A$ , could be exploited to evaluate the local distribution of mechanical fields in the area where the shape sub-domain has been suppressed.

### 5.3 Generation of the FE local problems

The developed a posteriori criterion has been integrated into an automatic adaptive modelling process whose aim is to provide the Reference Model for the Simulation PV. The complete process proposed in our approach is summarised in Figure 5.3, where we refer to mechanical models and simulation models as  $\Omega_2$ , although  $\Omega_2$  represents only the shape support of the models, which actually contains additional data, e.g. BCs.

In what follows, we will indicate a generic sub-domain as  $\Gamma_{Xi}$ , with  $i \in \{1, \dots, n\}$  and  $X \in \{A, S\}$ , where  $n$  is the number of shape sub-domains removed from the initial model  $\Omega_1$  and  $X$  expresses whether the sub-domain is additive or subtractive. Shape sub-domains  $\Gamma_{Xi}$  are supposed to be disconnected, i.e. for  $i \neq j$   $\Gamma_{Xi} \cap \Gamma_{Xj} = \emptyset$ .

The influence on the FE results of the removal of each sub-domain  $\Gamma_{Xi}$  is:

$$\eta_{est-i}^2 = \frac{e_{est-i}^2}{\frac{1}{2} \int_{\Omega_2} Tr[\bar{\bar{\sigma}}_2 : \bar{\bar{\varepsilon}}_2] d\Omega}, \quad (5.13)$$

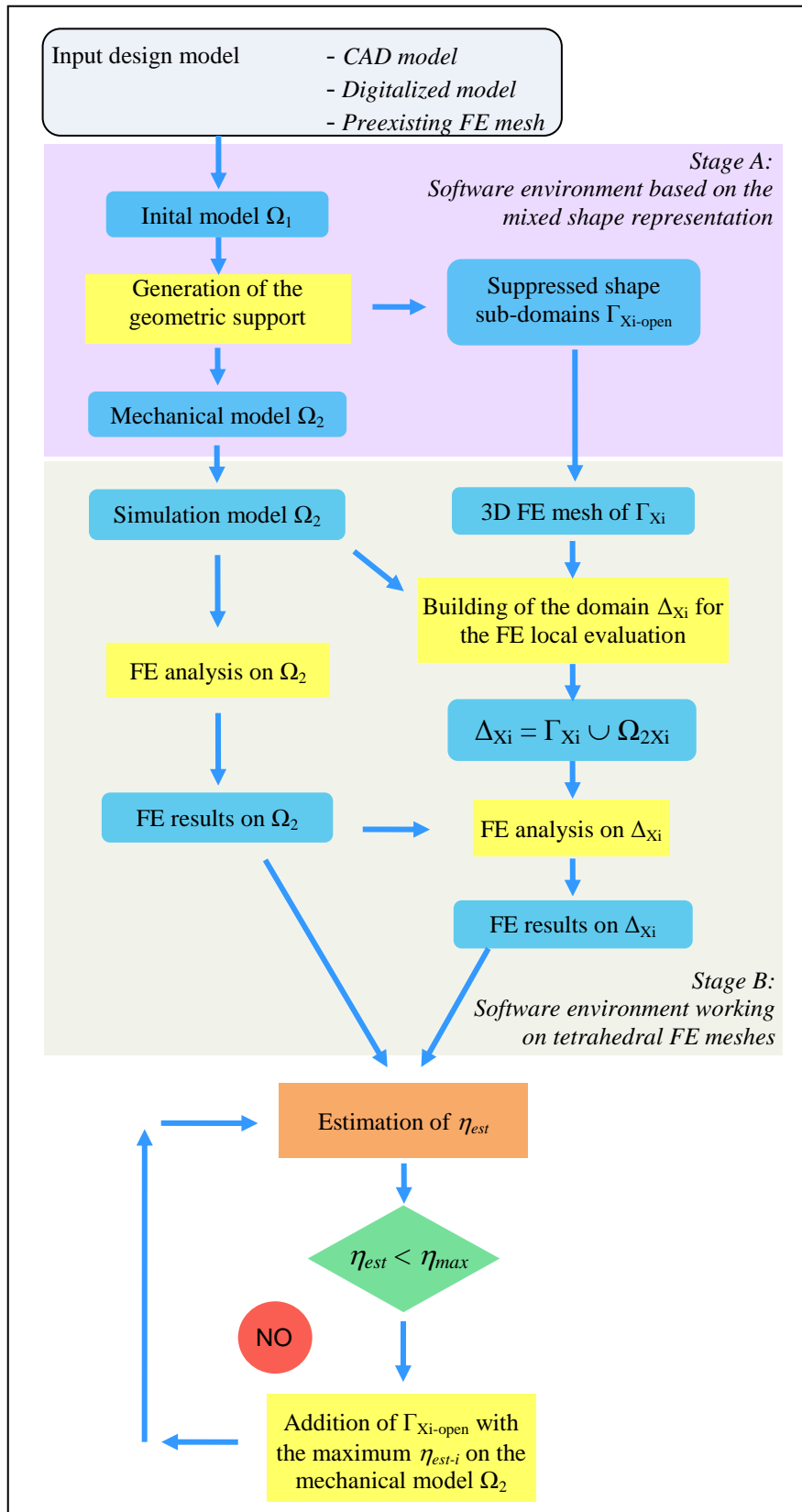
while the value of the a posteriori criterion  $\eta_{est}$ , which is the global influence given by the removal of the  $n$  sub-domains, can be estimated using the generalization of Eq. (5.12),

$$\eta_{est} = \sqrt{\sum_{i=1}^n \eta_{est-i}^2}. \quad (5.14)$$

In the context of an automatic adaptive modelling process, the a posteriori mechanical criterion acts as a constraint (see section 2.4.1). Indeed, it returns a value of  $\eta_{est}$  that, if compared with a threshold value  $\eta_{max}$  set by the user, allows one to establish if some  $\Gamma_{Xi}$  needs to be reintegrated into the shape of the simulation model (see the example provided in section 0).

To simplify the notations, we will use the same symbols both in the software environment based on the mixed representation (stage A of Figure 5.3) and in the software environment based on FE tetrahedral meshes (stage B of Figure 5.3). For example, the symbol  $\Omega_2$  can be used for referring both to the simplified model, i.e. the shape domain of the mechanical model resulting from the process of generation of the geometric support at the stage A, and to the corresponding FE mesh, i.e. the shape domain of the simulation model where the FE analysis is performed at the stage B.





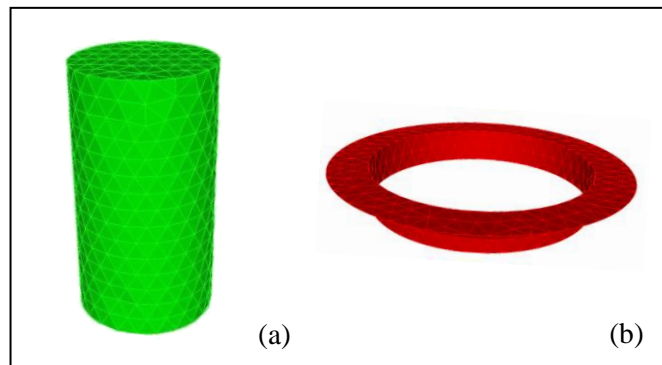
**Figure 5.3:** Use of the proposed a posteriori criterion in the scope of an adaptive modelling process.

### 5.3.1 Obtaining closed sub-domains for the application of the a posteriori criterion

In section 2.5, we highlighted the different geometric meanings that are associated to a simplification detail, depending on the considered phase of the process. Shape sub-domains corresponding to a priori simplification details are open sub-domains formed by sets of polyhedral faces. In contrast, in order to generate a FE mesh during the a posteriori process, volumetric sub-domains are needed, which correspond to the local volumetric difference between the shape domains  $\Omega_1$  and  $\Omega_2$ . Therefore, additional operations are necessary to close the open sub-domains removed and stored during the a priori shape simplification process. In the following,  $\Gamma_{Xi}$  will represent a volumetric sub-domain targeted by the a posteriori analysis, and  $\Gamma_{Xi-open}$  will be its corresponding open sub-domain.

If the removal of the sub-domain  $\Gamma_{Xi-open}$  has not changed the genus of the model, the open set of faces that needs to be closed has only one boundary, while in the case of single through holes, i.e. holes increasing by one the genus of the volume model, two boundaries needs to be closed to obtain a volume.

If the boundary of the recovered sub-domain  $\Gamma_{Xi-open}$  lies on a plane (it is typically the case of sub-domains corresponding to removed circular holes), the remeshing is easy and no need exists for looking at the simplified model to close the open sub-domain. In a general case, an a priori simplification detail corresponds to the difference between the initial model and the simplified one. Therefore, the shape of the simplified model  $\Omega_2$  needs to be kept into account for getting a faithful reconstruction of the volume corresponding to the removed shape sub-domain  $\Gamma_{Xi-open}$ . For example, to obtain closed shape sub-domains  $\Gamma_{Xi}$  corresponding to fillet form features, the faces belonging to the partition where the fillet lied before its suppression can be stored together with the set of polyhedral faces  $\Gamma_{Xi-open}$ . The two sets of faces share boundary polyedges that have the same discretization. Therefore, a closed polyhedron can be obtained by merging double edges belonging to the two boundaries. Figure 5.4 shows two examples of closed shape sub-domains resulting from a hole removal (Figure 5.4(a)) and a fillet removal (Figure 5.4(b)).



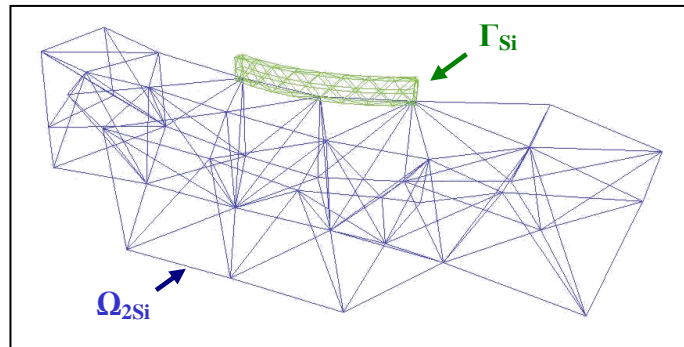
**Figure 5.4:** Closed shape sub-domains corresponding to a hole (a) and a fillet (b) removal.

### 5.3.2 Generation of FE local problems

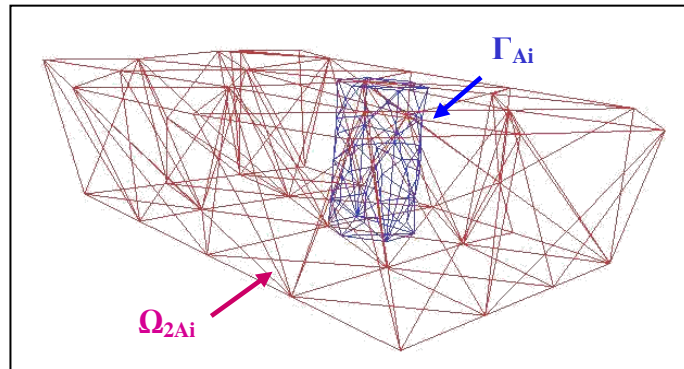
For each removed sub-domain  $\Gamma_{Xi}$ , the a posteriori mechanical criterion is computed on the basis of:

- The FE results obtained when solving the FE problem of analysis on a simplified domain  $\Omega_2$ ;
- The FE results of a local FE problem built around the sub-domain  $\Gamma_{Xi}$ .

In order to build the local FE problem, a subset of the FE mesh describing  $\Omega_2$ , noted as  $\Omega_{2Xi}$ , is needed. It is formed by the subset of FE elements of  $\Omega_2$  closest to  $\Gamma_{Xi}$ , and can be defined using a finite number of layers of FE elements in the neighbourhood of  $\Gamma_{Xi}$ . Geometric criteria could be set to determine the size of the sub-domain  $\Omega_{2Xi}$ . They could be defined proportionally to the size of  $\Gamma_{Xi}$ . However, since in the neighbourhood of a removed sub-domain  $\Gamma_{Xi}$  the FE sizes of  $\Omega_2$  are generally larger than those of  $\Gamma_{Xi}$ , a small number of FE layers should always provide a  $\Omega_{2Xi}$  whose boundary is enough distant from  $\Gamma_{Xi}$  to achieve a correct transfer of the mechanical fields. Tests performed until now show that the use of 2 or 3 layers of FE provides an acceptable solution to generate  $\Omega_{2Xi}$ .



**Figure 5.5:** Example of a FE mesh  $\Omega_{2Si}$  built around a simplified sub-domain  $\Gamma_{Si}$  for the FE local computation over the resulting sub-domain  $\Delta_{Si}$ .



**Figure 5.6:** Example of a FE mesh  $\Omega_{2Ai}$  built around a simplified sub-domain  $\Gamma_{Ai}$  for the FE local computation over the resulting sub-domain  $\Delta_{Ai}$ .

$\Delta_{X_i} = \Gamma_{X_i} \cup \Omega_{2X_i}$  is the sub-domain where the FE local analysis is performed, whose stiffness is computed as described in section 5.2.2. Since  $\Gamma_{X_i}$  and its neighbouring sub-domain  $\Omega_{2i}$  are two domains geometrically independent, we face a situation where vertices of FE elements of  $\Gamma_{X_i}$  are not coinciding with vertices of FE elements of  $\Omega_{2X_i}$ , and FE elements corresponding to  $\Gamma_{X_i}$  and  $\Omega_{2X_i}$  may be disconnected or interpenetrating. As a result, the FE mesh of  $\Delta_{X_i}$  is non-conform (see Definition 1.2). Figure 5.5 and Figure 5.6 show two examples of domains  $\Delta_{X_i}$ , where the shape sub-domains removed  $\Gamma_{X_i}$  and their neighbouring sub-domains  $\Omega_{2X_i}$  are highlighted. It appears clearly that the resulting mesh  $\Delta_{X_i}$  is non-conform. To enforce the continuity of  $\Delta_{X_i}$  from a mechanical point of view, the FE meshes of  $\Gamma_{X_i}$  and  $\Omega_{2X_i}$  are linked through a linear kinematic relation.

The BCs of the local FE problem related to  $\Delta_{X_i}$  are obtained when solving the FE problem on  $\Omega_2$ . Indeed, they are the displacements of the nodes at the boundary of the domain  $\Omega_{2X_i}$ , which are internal to the domain  $\Omega_2$ , i.e. are not part of the  $\Omega_2$  boundary. Once set the simulation model related to  $\Delta_{X_i}$ , the FE local problem is solved, and, for each shape sub-domain  $\Gamma_{X_i}$ , the a posteriori mechanical criterion  $\eta_{est-i}$  is evaluated.

Therefore, following our approach, if  $n$  is the number of the shape sub-domains  $\Gamma_{X_i}$  corresponding to a priori simplification details, we have to solve  $(n + 1)$  FE problems. However, each removed sub-domain  $\Gamma_{X_i}$  has a simple shape that is easy to mesh, and moreover its surrounding sub-domain  $\Omega_{2X_i}$  is formed by few FE of  $\Omega_2$ , which are easily retrievable. Therefore, solving the  $n$  problems related to the all the removed  $\Gamma_{X_i}$  is not computationally expensive. In addition, the integration of the a posteriori mechanical criterion in the whole model preparation process reduces the time required to set up the FE local problems, since the shape models  $\Gamma_{X_i-open}$  related to the removed sub-domains are automatically provided during the a priori shape simplification process, as it has been described in the previous chapter, and can be easily closed, as described in section 5.3.1.

## 5.4 Validation of the a posteriori mechanical criterion

In what follows, some examples are provided which allow us to validate the use of the a posteriori mechanical criterion above introduced.

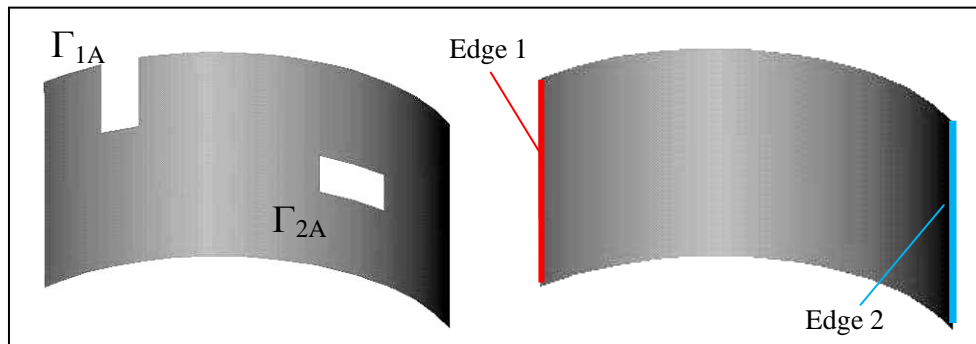
### 5.4.1 Effectiveness of the a posteriori mechanical criterion

The first kind of tests we performed aimed at evaluating how much the estimation of the FE solution over the initial domain  $\Omega_1$ , provided by solving the local FE problems, influences the accuracy of the error computation. To this end, for each shape sub-domain  $\Gamma_{X_i}$  a priori removed, we computed an effectiveness index  $\gamma_i = \eta_{est-i} / \eta_i$  (see Eq. (1.6)) providing a value of the reliability of the error estimation. We always got values of the effectiveness index  $\gamma_i$  close to 1, which corroborates the reliability of the proposed a posteriori criterion. Moreover, our main interest is

the criterion ability in evaluating the mutual influence of sub-domain removals on FE results. The above requirements are fulfilled in all the tests we performed. In what follows, we describe two simple examples.

### a. Shell problem

Figure 5.7 depicts a linear FE thermal analysis with stationary conduction applied to a shell domain. We imposed a constant temperature along Edge 1 and a conduction flux along Edge 2. For each a priori simplification detail corresponding to the shape sub-domain  $\Gamma_{Xi}$ , with  $i \in \{1, 2\}$ , a FE analysis is performed on the object domain  $\Omega_2$  resulting from its removal, and the influence  $\eta_{est-i}$  of the sub-domain suppression is estimated with Eq. (5.12). Since the proposed model has a simple geometry, the FE solution on the initial domain  $\Omega_1$  can be easily computed and, consequently, for each removed shape detail  $\Gamma_{Xi}$ , it is possible to compute the relative error  $\eta_i$  given by Eq. (5.3) and compare it with its estimation  $\eta_{est-i}$ , by using an effectiveness index  $\gamma_i$  given by Eq. (1.6). Results are shown in Table 5.1.



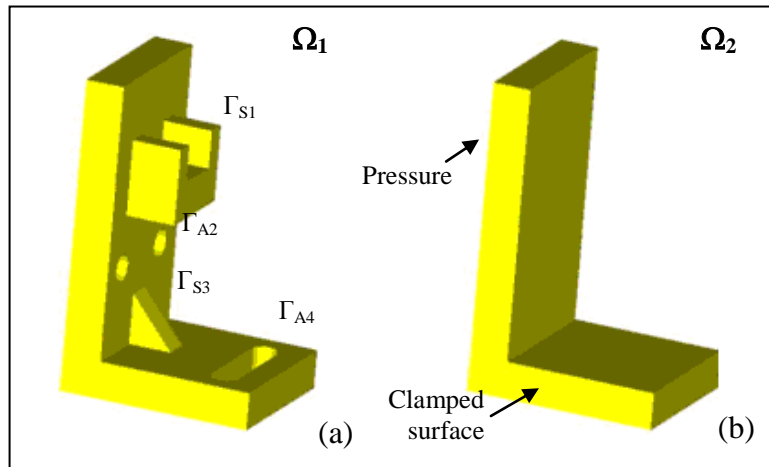
**Figure 5.7:** (a) Initial domain  $\Omega_1$ ; (b) Simplified domain  $\Omega_2$  and related BCs.

	$\eta_i$	$\eta_{est-i}$	$\gamma_i$
$\Gamma_{A1}$	8.00 %	8.31 %	1.04
$\Gamma_{A2}$	14.5 %	14.8 %	1.02

**Table 5.1:** Comparison between the effective influence ( $\eta_i$ ) and its estimation ( $\eta_{est-i}$ ) for each removed shape sub-domain  $\Gamma_{Xi}$  related to the object in Figure 5.7.

### b. 3D problem of FE linear static analysis

The example here illustrated is related to a 3D problem of FE linear static analysis. Figure 5.8 depicts the initial domain  $\Omega_1$  and the simplified one  $\Omega_2$ , where a priori simplification details  $\Gamma_{X_i}$ ,  $i \in \{1, \dots, 4\}$ , have been removed. The bottom surface of the object is clamped and a uniform pressure is applied over the left hand side surface. As in the previous example, for each a priori simplification detail corresponding to the shape sub-domain  $\Gamma_{X_i}$ ,  $\eta_{est-i}$  provides an estimation of the influence of the sub-domain removal that can be compared with the effective influence  $\eta_i$ . Results are shown in Table 5.2.



**Figure 5.8:** (a) Initial domain  $\Omega_1$ ; (b) Simplified domain  $\Omega_2$  and related BCs.

	$\eta_i$	$\eta_{est-i}$	$\gamma_i$
$\Gamma_{S1}$	16.7 %	19.2 %	1.15
$\Gamma_{A2}$	1.10 %	1.14 %	1.04
$\Gamma_{S3}$	38.7 %	43.6 %	1.13
$\Gamma_{A4}$	27.7 %	27.6 %	0.996

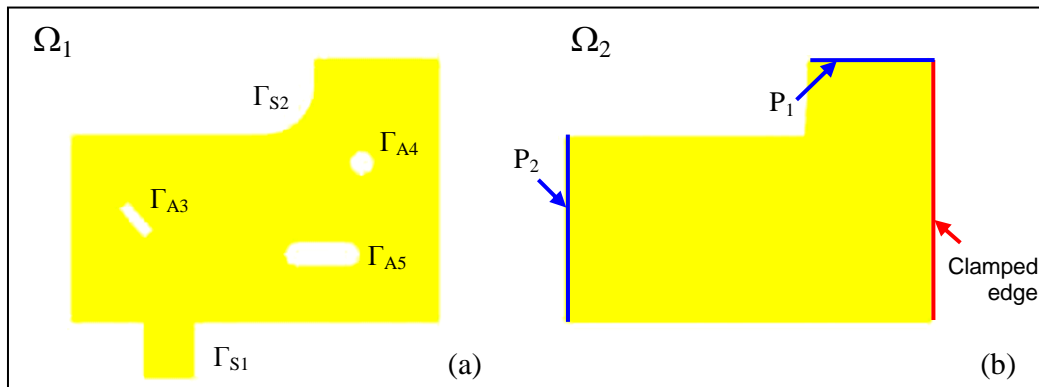
**Table 5.2:** Comparison between the effective influence ( $\eta_i$ ) and its estimation ( $\eta_{est-i}$ ) for each shape sub-domain  $\Gamma_{X_i}$  related to the object in Figure 5.8.

### 5.4.2 Influence of the discretization error on the criterion accuracy

All the equations provided in section 5.2 refer to the exact solutions of the mechanical problem; whereas the values of the mechanical fields we provide for their solution are obtained through a FE analysis. Therefore, when applying the a posteriori mechanical criterion, actually we neglect the discretization errors due to the use of a FE solving method. Anyway, tests performed by using different discretizations for the same domains have revealed that discretization errors weakly influence the accuracy of the proposed a posteriori mechanical criterion, and do not alter the information about the mutual influence of sub-domain removals. Hereafter, an example of linear FE thermal analysis with stationary conduction applied to a 2D model is illustrated, where the a posteriori criterion is applied to two different discretizations of the shape.

Figure 5.9 depicts the 2D FE problem and the prescribed BCs. Blue edges indicate pressure areas and the red one is clamped. The initial domain  $\Omega_1$  is modified by removing some a priori simplification details corresponding to the sub-domains  $\Gamma_{X_i}$ , with  $i \in \{1, \dots, 5\}$ . Table 5.3 provides the values  $\eta_{est-i}$  related to each removed sub-domain  $\Gamma_{X_i}$  and compares them with their corresponding  $\eta_i$ .

In order to test the impact of the discretization error, we reapplied the a posteriori criterion over FE meshes with halved FE sizes. Results are shown in Table 5.4. We can notice that the order of magnitude of the values returned by  $\eta_{est-i}$  and their mutual influences stay the same. This corroborates the fact that the user is able to evaluate the influence of each sub-domain removal  $\Gamma_{X_i}$  by not minding the discretization errors.



**Figure 5.9:** (a) Initial domain  $\Omega_1$ ; (b) Simplified domain  $\Omega_2$  and related BCs.

	$\eta_i$	$\eta_{est-i}$	$\gamma_i$
$\Gamma_{S1}$	5.29 %	4.70%	0.888
$\Gamma_{S2}$	7.81 %	7.75 %	0.992
$\Gamma_{A3}$	11.8 %	12.6 %	1.07
$\Gamma_{A4}$	8.06 %	8.83 %	1.10
$\Gamma_{A5}$	21.4 %	20.0 %	0.935

**Table 5.3:** Estimation  $\eta_{est-i}$  and effective influence of removal  $\eta_i$  for each shape sub-domain  $\Gamma_{X_i}$  suppressed in the problem of Figure 5.9.

	$\eta_i$	$\eta_{est-i}$	$\gamma_i$
$\Gamma_{S1}$	3.61 %	4.00%	1.11
$\Gamma_{S2}$	8.19 %	8.95 %	1.09
$\Gamma_{A3}$	11.8 %	12.8 %	1.08
$\Gamma_{A4}$	9.75%	8.83 %	0.906
$\Gamma_{A5}$	22.8 %	18.2%	0.798

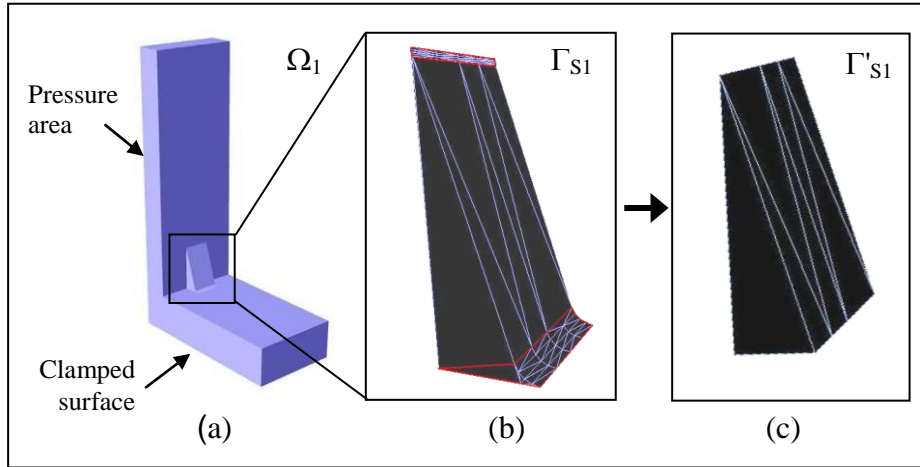
**Table 5.4:** Estimation  $\eta_{est-i}$  and effective influence of removal  $\eta_i$  for each shape sub-domain  $\Gamma_{X_i}$  suppressed in the problem of Figure 5.9. The element sizes of FE meshes have been halved in comparison with those used when obtaining values of Table 5.3.

### 5.4.3 Influence of the mesh non-conformity

As highlighted at section 0, the FE mesh of  $\Delta_{X_i}$  is not conform. Anyway, the tests illustrated above proved the effectiveness of the a posteriori criterion.

To further prove that the non-conformity of the domain  $\Delta_{X_i}$  does not significantly influence the accuracy of the a posteriori criterion, we performed even tests where the shape detail  $\Gamma_{X_i}$  and its surrounding sub-domain  $\Omega_{2X_i}$  not only generated a non-conform mesh, but were separated from each other. Still in these cases, the information returned by the mechanical criterion was accurate. This situation may occur when the operator building the domain  $\Delta_{X_i}$  suppresses automatically some faces lying between the sub-domain  $\Gamma_{X_i}$  and the simplified domain  $\Omega_2$ , since they could be harmful for the construction of the FE mesh related to  $\Gamma_{X_i}$ .





**Figure 5.10:** Example of modification of a domain  $\Gamma_{Xi}$  to simplify the generation of its FE mesh: (a) Initial model  $\Omega_1$ ; (b) Sub-domain  $\Gamma_{S1}$  initially recovered. Two fillet areas are present; (c) Sub-domain  $\Gamma_{S1}'$  actually used to build the local FE problem, where the fillet areas have been removed.

Figure 5.10 shows an example. Figure 5.10(a) depicts the initial model  $\Omega_1$  with the BCs related to the mechanical problem to be solved. The a priori simplification detail initially recovered corresponds to the sub-domain  $\Gamma_{S1}$  of Figure 5.10(b). Some polyhedral faces belonging to the sub-domain  $\Gamma_{S1}$ , which correspond to the two fillet areas delimited in Figure 5.10(b) by red edges, would complicate the generation of the FE mesh  $\Gamma_{S1}$ , since they would require very small FE elements. Therefore, they can be suppressed by the sub-domain  $\Gamma_{S1}$  generating a modified sub-domain  $\Gamma_{S1}'$  (see Figure 5.10(c)).  $\Gamma_{S1}'$  will be used for building a FE local problem and the resulting domain  $\Delta'_{S1}$  will be formed by two non-adjacent sub-domains.

$\eta$	$\eta_{est}$ (related to $\Gamma_{S1}$ )	$\eta'_{est}$ (related to $\Gamma_{S1}'$ )
48.99 %	49.90 %	52.44 %
	$\gamma_I$	$\gamma_I'$
	1.02	1.07

**Table 5.5:** Results of the mechanical a posteriori criterion  $\eta_{est}$  related to the a priori simplification detail ( $\Gamma_{S1}$  with and  $\Gamma_{S1}'$  without fillet areas) removed when solving the mechanical problem of Figure 5.10.

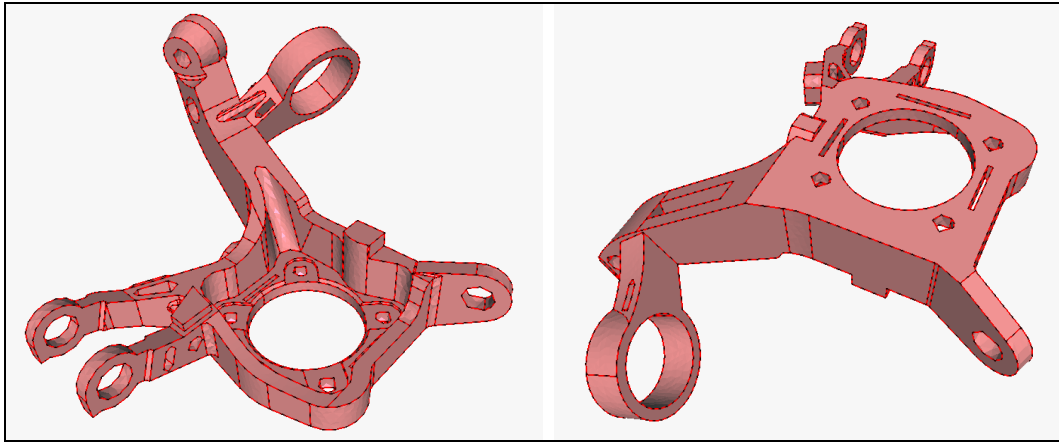
Table 5.5 shows the results obtained during the a posteriori analysis.  $\eta'_{est}$  is the value of the a posteriori criterion related to the shape sub-domain  $\Gamma_{S1}$  actually retrieved, while  $\eta_{est}$  is the one that would have been obtained if the fillet areas were not automatically suppressed. Both values are compared with the effective relative error  $\eta$  caused by the shape simplification performed. The related effectiveness indexes  $\gamma_l$  and  $\gamma_l'$  have similar orders of magnitude, and therefore we can conclude that the mesh non-conformity does not influence the criterion's effectiveness.

## 5.5 Adaptive modelling process

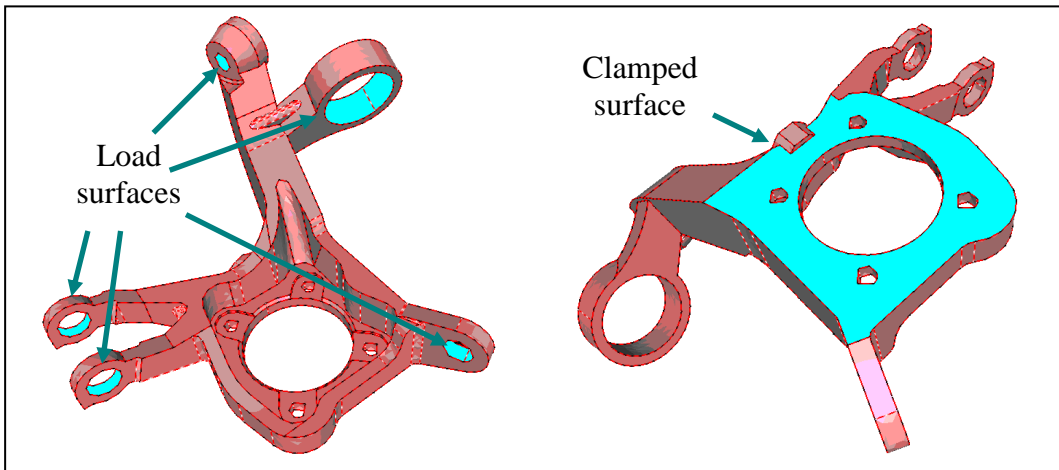
Thanks to the FE computation over  $\Delta_{Xi}$ , the a posteriori mechanical criterion  $\eta_{est}$  estimates the global influence of the removal of sub-domains  $\Gamma_{Xi}$  on the FE solution related to  $\Omega_2$ . A low value of  $\eta_{est-i}$  does not contribute to the global  $\eta_{est}$  and therefore confirms that  $\Gamma_{Xi}$  corresponds to a simplification detail. On the contrary, if a shape sub-domain  $\Gamma_{Xi}$  proves to have an important influence  $\eta_{est-i}$  over the mechanical behaviour of the component, it cannot be considered as an effective simplification detail and the simulation model must be modified to obtain a suitable one that allows a well-tuned FE simulation. As introduced at section 1.8.2, the adaptation of the model can be performed either at the level of the simulation model, i.e. directly on the FE mesh or on the shape domain related to the mechanical model, in the case where it is explicitly defined (see Figure 1.28). In our approach, the shape domain of the mechanical model is explicitly defined in the software environment based on the mixed representation, where shape simplifications occur during the a priori process. Therefore, the adaptations of the model due to the a posteriori analysis are performed at the level of the mechanical model (see Figure 5.3).

### 5.5.1 Redefinition of the geometric domain of the mechanical model

Hereafter, an example is presented where the shape of the simulation model (and hence of the mechanical model) needs to be redefined according to results provided by the a posteriori mechanical criterion  $\eta_{est}$ . Figure 5.11 illustrates the shape of the initial design model  $\Omega_1$ , and Figure 5.12 shows the shape domain  $\Omega_2$  of the mechanical model obtained during the a priori stage of shape simplification together with the related BCs. The bottom surface is clamped and a non-uniform pressure is applied on some partitions. Blue partitions correspond to the location of BCs.

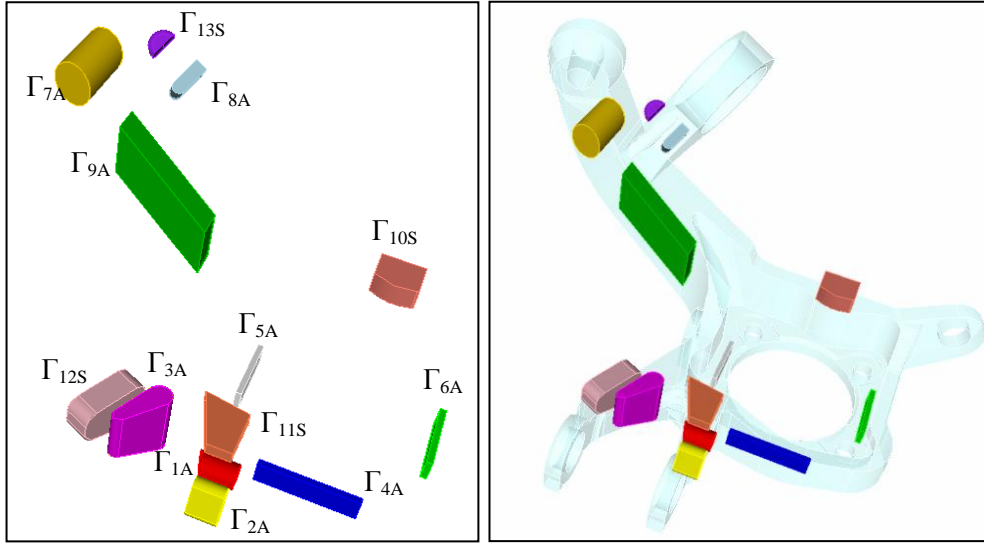


**Figure 5.11:** Initial model  $\Omega_1$ .



**Figure 5.12:** Mechanical model formed by the shape domain  $\Omega_2$  and BCs of the mechanical problem related to the considered component.

During the first stage of shape simplification, the shape sub-domains  $\Gamma_{X_i}$ ,  $i \in \{1, \dots, 13\}$ , which are both of subtractive and additive type, are considered a priori simplification details and therefore suppressed (see Figure 5.13). After performing a FE analysis on the simplified domain  $\Omega_2$ , the influence  $\eta_{est-i}$  caused by the removal of each  $\Gamma_{X_i}$  is computed, and the a posteriori criterion  $\eta_{est}$  is evaluated. Table 5.5 reports the values  $\eta_{est-i}$ .

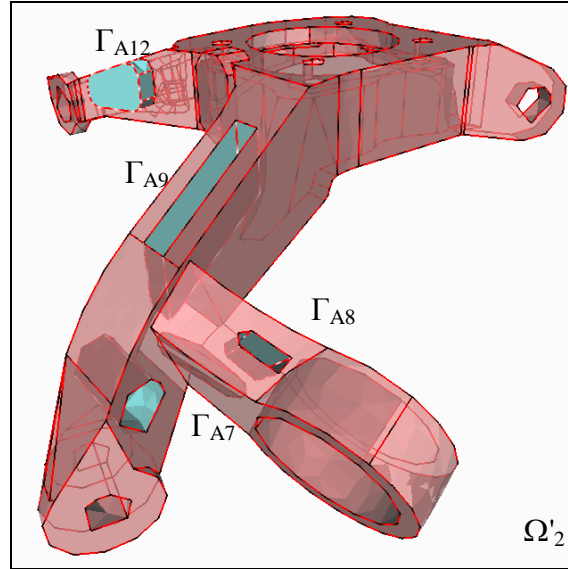


**Figure 5.13:** Shape sub-domains  $\Gamma_{Xi}$  removed from the initial design model  $\Omega_1$  of Figure 5.11.

	$\Gamma_{A1}$	$\Gamma_{A2}$	$\Gamma_{A3}$	$\Gamma_{A4}$	$\Gamma_{A5}$	$\Gamma_{A6}$	$\Gamma_{A7}$
$\eta_{est-i}$	4.472 %	3.406 %	11.32 %	1.414 %	2.236 %	1.304 %	10.28 %
	$\Gamma_{A8}$	$\Gamma_{A9}$	$\Gamma_{S10}$	$\Gamma_{S11}$	$\Gamma_{S12}$	$\Gamma_{S13}$	
$\eta_{est-i}$	11.37 %	11.92 %	1.612 %	8.124 %	14.40 %	3.521 %	

**Table 5.5:** Values of  $\eta_{est-i}$  for each sub-domain  $\Gamma_{Xi}$  of Figure 5.13.

The corresponding value provided by the a posteriori mechanical criterion is  $\eta_{est} = 28.88\%$ . Then, as introduced in section 0, the user can set a threshold value of accuracy  $\eta_{max}$  that is consistent with his/her objectives. Hence, depending on the threshold value  $\eta_{max}$  specified, it is decided whether some shape sub-domain  $\Gamma_{Xi}$  have to be reinserted into the model or not, i.e. whether if there are some sub-domains  $\Gamma_{Xi}$  that do not represent actual simplification details. With regards to the considered example, in the case where the user requires an accuracy characterized by  $\eta_{max} = 30\%$ , the shape domain  $\Omega_2$  does not need to be adapted, thus confirming that each  $\Gamma_{Xi}$  is an actual simplification detail. Otherwise, if an accuracy  $\eta_{max} = 15\%$  is required, the sub-domains  $\Gamma_{A3}$ ,  $\Gamma_{A7}$ ,  $\Gamma_{A8}$ ,  $\Gamma_{A9}$  and  $\Gamma_{S12}$  must be reinserted into  $\Omega_2$ , in order to obtain a mechanical model with a suitable shape domain  $\Omega'_2$  (see Figure 5.14), which provides a  $\eta_{est} = 11.01\%$ .



**Figure 5.14:** Shape domain  $\Omega'_2$  corresponding to the adapted simulation model, where a threshold value  $\eta_{max} = 15\%$  has been set.

### 5.5.2 Shape operators for the simulation model re-adaptation

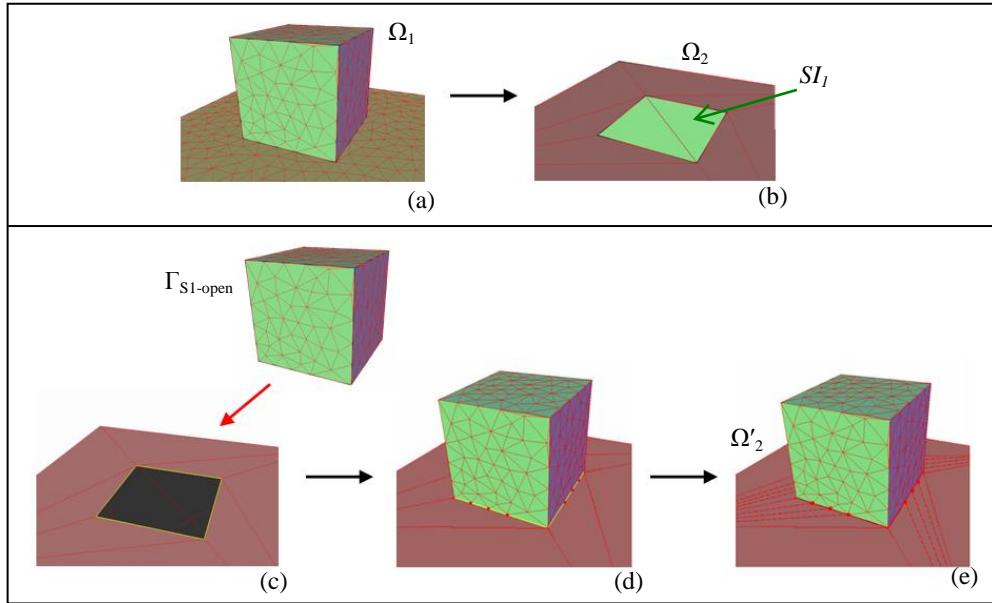
As mentioned above, the adaptation of the domain  $\Omega_2$  according to the results provided by the a posteriori mechanical criterion is performed at the level of the mechanical model, i.e. in the software environment based on the mixed representation. Polyhedral faces added to the removed sub-domains  $\Gamma_{Xi-open}$  to obtain volumetric sub-domains are not useful for reinsertion purposes. However, only open sets of polyhedral faces removed, i.e.  $\Gamma_{Xi-open}$ , which are faces belonging to the initial model  $\Omega_1$ , will be useful to readapt the shape domain of the mechanical model  $\Omega_2$ .

At the moment, an adaptation of the domain  $\Omega_2$  is possible if geometric information about shape interfaces (see section 3.5.2) is available.

As a hypothesis, we consider only shape sub-domains  $\Gamma_{Xi-open}$  that have been removed in one step, i.e. the shape interface resulting from the sub-domain removal is simply the sub-domain boundary (see Fig. 3.24(b)). This is consistent with the hypothesis of disconnection of shape sub-domains  $\Gamma_{Xi-open}$  that has been mentioned at section 0. Each sub-domain  $\Gamma_{Xi-open}$  is associated either to one (if the sub-domain removal does not change the object's genus) or two shape interfaces (in the case where the sub-domain removal changes the object's genus, i.e. the sub-domain is a simple through hole). Each shape interface  $SI_i$  consists in a partition  $P_i$  created when suppressing the sub-domain  $\Gamma_{Xi-open}$ , which is formed by some polyhedral faces. To reinsert the shape sub-domain  $\Gamma_{Xi-open}$  in the shape model  $\Omega_2$ , the polyhedral faces forming  $SI_i$  are removed and replaced by the polyhedral faces belonging to  $\Gamma_{Xi-open}$ . Obviously, as regards through holes, we have two shape interfaces  $SI_i^A$  and  $SI_i^B$ , and therefore two distinct partitions  $P_i^A$  and  $P_i^B$ , and, in the case where of reinsertion of  $\Gamma_{Xi-open}$ , both partitions have to be removed.

Actually, the tessellation of the sub-domain  $\Gamma_{Xi-open}$  could not conform to that one of the shape domain  $\Omega_2$  to be adapted. Indeed, during the a priori shape simplification process we are under the hypothesis of one-step sub-domains removal. After suppressing the sub-domains  $\Gamma_{Xi-open}$  further modifications of the domain  $\Omega_2$  could be occurred, which did not consist in shape modifications but modified only the tessellation of the model. Indeed, if the skin detail operator is not subjected to further constraints, such as user-prescribed edge sizes, it will be allowed to remove polyhedral vertices as long as the simplified polyhedron stays within the geometric envelope. This means that for example, in planar areas, configuration of minimal tessellation will be obtained, i.e. the tessellation of the simplified model will have less polyhedral vertices than the initial one. Therefore, if this case occurs, the polyhedral edges at the boundary of the shape interface  $SI_i$  do not coincide with those belonging to the boundary of the sub-domain  $\Gamma_{Xi-open}$ . To overcome this problem, an easy solution is to preserve the tessellation of  $\Gamma_{Xi-open}$ , while subdividing its faces adjacent to  $\Omega_2$  in order to create new edges that enable the conformity of the polyhedron.

Figure 5.15 illustrates all the process above described. In Figure 5.15(a), a particular configuration of a shape model in the software environment based on the mixed shape representation is showed. During the shape simplification process, a sub-domain  $\Gamma_{S1-open}$  is removed, which produces a shape interface  $SI_I$  and implies the creation of the partition  $P_I$  on the simplified model (see Figure 5.15(b)). If the user aims at reinserting  $\Gamma_{S1-open}$  in order to obtain an adapted shape model  $\Omega'_2$ , the polyhedral faces belonging to  $P_I$  are removed (see Figure 5.15(c)) and replaced by those forming the sub-domain  $\Gamma_{S1-open}$ . Looking at Figure 5.15(d), we can notice that the tessellation of  $\Gamma_{S1-open}$  and that one of the shape domain  $\Omega_2$  do not conform. Therefore, the edges at the boundary of  $SI_I$  are replaced with the edges at the boundary of  $\Gamma_{S1-open}$ , and new edges are added to the polyhedral model in order to make conform of the polyhedral representation of  $\Omega'_2$  (see Figure 5.15(e)).



**Figure 5.15:** Illustration of the re-adaptation process of the shape domain  $\Omega_2$  related to the mechanical model: (a)(b) A priori shape simplification process, where the initial shape domain  $\Omega_1$  is transformed in  $\Omega_2$  by removing the shape sub-domain  $\Gamma_{S1-open}$ ; (c) Need for readapting the shape domain  $\Omega_2$  by reinserting the shape sub-domain  $\Gamma_{S1-open}$ ; (d) Non-conformity of the tessellation corresponding to the readapted shape domain  $\Omega'_2$ ; (e) Setting-up the conformity of the polyhedral representation of  $\Omega'_2$ , where some faces of the polyhedral model are split by adding new polyhedral edges.

## 5.6 Conclusions

In the present chapter, an a posteriori mechanical criterion has been introduced, which can be applied to problems of FE static analysis of linear elastic structures or FE thermal problems for stationary linear conduction. We have given evidence of the effectiveness of this criterion in evaluating the impact of a priori shape simplifications, performed over the shape of a design model, on the FE analysis results related to the corresponding simulation model.

At present, the proposed a posteriori criterion is applicable only in the case where the simplifications performed consist in the addition of the removal of disconnected sub-domains. However, it would be interesting to test its validity in more general situations, where the removed sub-domains are adjacent. Some clues about this subject will be given in the next chapter, where the use of the a posteriori criterion will be envisaged also in other scenarios occurring during a PDP, where the above situation may likely occur.

Moreover, we aim at further automating the re-adaptation process of the simulation model based on the indications provided by the a posteriori mechanical criterion, since the examples of re-adaptation set up are still linked too much restrictive hypotheses.





# Application of the a posteriori mechanical criterion to different scenarios of PDP

*The a posteriori mechanical criterion introduced in the previous chapter could be usefully exploited in other scenarios occurring during a PDP. Indeed, the criterion provides information about the mechanical behaviour of a component without performing a new FE analysis at each shape modification. Therefore, it allows one to save the time usually dedicated to the preparation of the geometric support of a mechanical model, and does not require a strong involvement of the stakeholders participating to the behaviour simulation PV. Here, several scenarios are identified, and the potentials and requirements of the a posteriori mechanical criterion in each scenario are analyzed.*

---

## 6.1 Towards a stronger integration of the engineering analysis in a PDP

As highlighted in section 1.4.1, behaviour analysis can provide useful indications at different stages of a PDP and helps the stakeholders of the design PV to meet the desired product specifications. However, the stakeholders involved in design and behaviour analysis PVs have different skills and are used to operate with different software environments. Each time that a behaviour analysis has to be performed, several communication processes between the PVs must take place, in terms both of communication between PVs' stakeholders and model exchange between their different software environments. In addition, when moving from the design to the behaviour simulation PV, the shape domain of the design model needs to be simplified and prepared. The related process of simulation model generation can be time consuming, even if it is related to a simple component or to modifications of an existing product.

Despite the efforts to set up concurrent engineering approaches (see section 1.1), the use of a FE analysis is often limited to the classical PDP scenario, and therefore is performed at a product development stage where product's specifications have been already stated. In this case, a FE analysis is only able to validate product's performances and does not bring rapid and meaningful design improvements. In contrast, a "continuous" use of behaviour analysis could provide information about the mechanical behaviour during the PDP and lead to better design choices and a more efficient PDP.

An efficient tool allowing the designer to perform the different kinds of analyses described in section 1.4.1 should fulfil the following requirements:

- a. Reduce communication between design and behaviour simulation PVs classically required for a FE analysis;
- b. Be of simple use for the stakeholders of design PV, who usually do not have a technical background suited for performing engineering analyses;
- c. Shorten the time scheduled for obtaining FE analysis results, thanks to a reduction of the time required by the process of simulation model preparation;
- d. Be reliable, i.e. be able to go beyond a qualitative information providing data about mechanical fields;
- e. Exploit when possible information about the mechanical behaviour returned by a FE analysis previously performed.

## 6.2 Use of the a posteriori criterion during PDP

On the basis of the arguments discussed in the previous section, we propose here to use the a posteriori mechanical criterion described in chapter 5 as a rapid and effective tool providing useful information about the mechanical behaviour of an object to be designed [FLM\*06]. In particular, the a posteriori criterion satisfies the requirements *a-e* listed in section 6.1.

As detailed in section 5.2, the a posteriori mechanical criterion can be used in all PDP scenarios where FE results are available. These FE results have been computed on shape versions obtained by adding or subtracting shape sub-domains with arbitrary size and shape from an initial shape version. Then, several PDP scenarios have been identified, where the a posteriori mechanical criterion may be usefully applied.

In the classical PDP scenario, for which the a posteriori mechanical criterion has been conceived and that has been described in chapter 5, the two shape versions  $\Omega_1$  and  $\Omega_2$  are related to the design model and to the mechanical model respectively. When using the a posteriori criterion, we are able to evaluate whether the FE results related to the mechanical model describe with sufficient accuracy the mechanical behaviour of the initial design model, i.e. if shape modifications performed during the mechanical model preparation process affect or not the accuracy of FE analysis results.

The scenarios analyzed in sections 6.3 and 6.4 are based on the idea of exploiting the FE results related to a previous shape version of a design model to describe the mechanical behaviour of a shape version at a different design stage. This allows us to decide whether the mechanical behaviour of two shape domains can be considered as equivalent or not. A new FE analysis is then required only in

the case where a shape sub-domain  $\Gamma_{X_i}$ , which characterizes a shape difference between two shape versions, shows a significant influence over the FE results accuracy. The shape domain  $\Omega_2$  to consider for the FE analysis will be adapted by including the influent shape sub-domain  $\Gamma_{X_i}$ . In these scenarios, the a posteriori mechanical criterion acts as an influence indicator (see section 2.4.1), and can be considered as a tool integrated into the design PV and supporting the design decision making process. It provides a useful indication about the mechanical behaviour of different shape versions of a component with a limited number of FE analyses that are always performed on shape models that are much easier to mesh and solve.

In section 6.5, we investigate the possibility of using the a posteriori mechanical criterion in a scenario where no shape changes have occurred. A typical example is related to the situation where an existing component is used in a new product. In this case, it is subject to new BCs, and a new FE analysis must be performed to estimate its mechanical behaviour. By using the a posteriori mechanical criterion, the user of the behaviour simulation PV may evaluate if it is possible to exploit the shape domain related to the mechanical model with old BCs when generating the mechanical model with the new ones. Also in this scenario, time compression is provided. Indeed, the exploitation of an existing shape domain for generating a mechanical model with new mechanical hypotheses, i.e. new BCs, allows the user to avoid the preparation process of a new shape domain, which would be time expensive.

### **6.3 The behavioural modeller paradigm**

In the early stages of the PDP, the final product shape is unknown and design choices may significantly influence several aspects, such as costs, performances, reliability, security, environmental impact. During the design process of a component performed in a CAD system, an engineering analysis can help in evaluating the performances of different design variants and consequently support the design decision making process (see section 1.4.1). However, analyses performed at this stage of the PDP often do not make use of FE techniques, but rather analytical methods (see section 1.4.2) that are no longer adequate when the complexity of the design model increases, and return only a qualitative response. In addition, even if a FE analysis is performed, rarely consistency is met between the design models, whose shapes continuously evolve, and the shape models used in the behaviour simulation PV. Indeed, generating a new simulation model at each shape modification of the design model would be too time consuming and require an important involvement of the behaviour simulation PV's stakeholders.

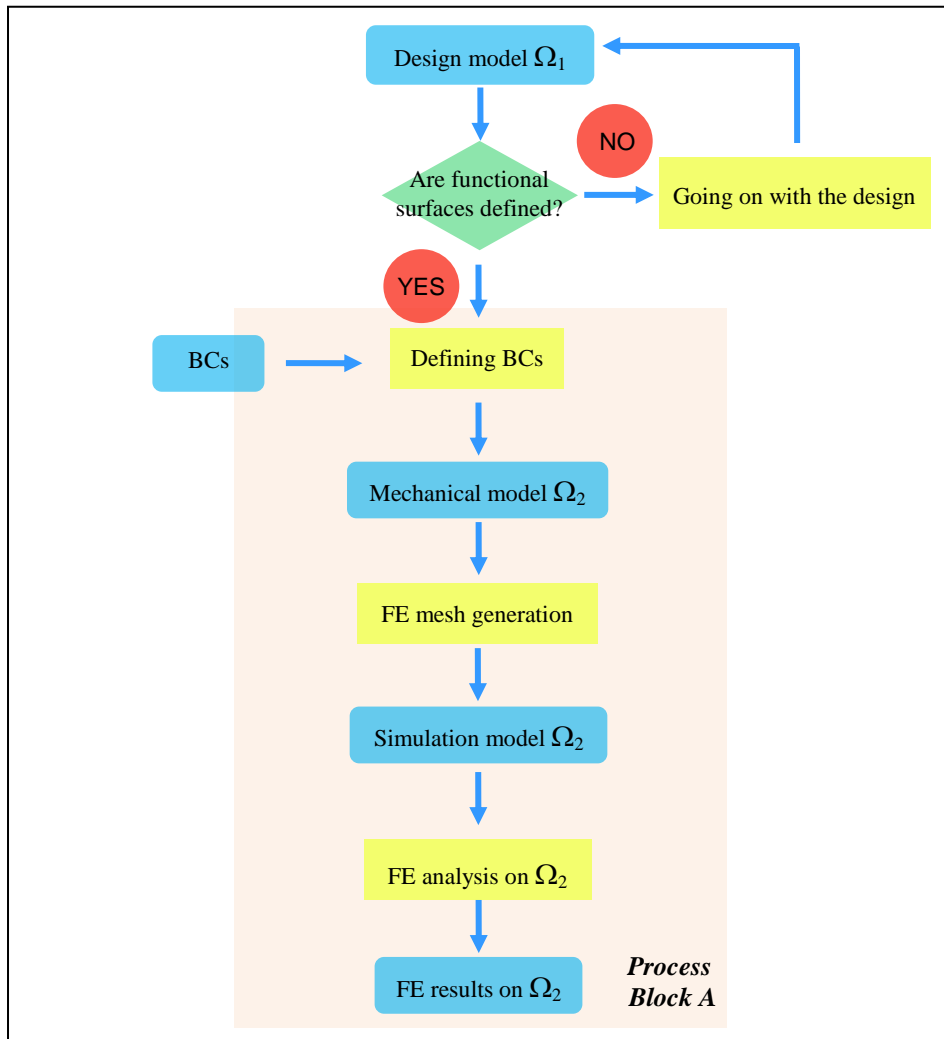
In this context, the a posteriori mechanical criterion could be a useful tool. It could allow the user of the design PV to take into account the mechanical behaviour of the product during the design task and orient design choices, while the shape of the object is generated through incremental shape changes based on CAD modelling primitives. For this reason, we call this scenario the “*behavioural modeller paradigm*”.

The *behavioural modeller paradigm* is based on the two following key hypotheses:

1. Once the rough shape of a component has been obtained, the designer modifies it through volumetric changes based on design primitives. The shape generation process is of constructive type, i.e. the rough shape is iteratively refined by means of addition or subtraction of shape sub-domains. This is a reasonable hypothesis, since the use of a constructive shape generation process is a common approach when moving from conceptual to detailed design. Indeed, it reflects the decisions taken during the product definition, where more precise product data, which take into account several kinds of requirements, e.g. functional and manufacturing, are incorporated into the component shape as soon as they are available. In contrast, the proposed paradigm is not applicable to free-form surface design, where the generation of a volume object is based on the incremental generation and assembly of elements of its surface boundary;
2. Key areas of the component shape correspond to functional surfaces, and are assumed to be defined during the early stages of the modelling process. This hypothesis is also justified, since functional surfaces play a crucial role in defining the objectives of a component and strongly influence the overall shape. In addition, these surfaces usually represent the interface of a component with the other assembly components, and therefore they are likely the locations of BCs.

Based on the above hypotheses, we can conclude that, in early design stages, it is possible to access simultaneously simple shapes and, since the functional surfaces have been already defined, most part of the location of BCs.

We assume that, at the beginning of the modelling process, no precise information about functional surfaces and consequently about the location of BCs is known. Anyhow, as introduced above, functional surfaces are the zones whose shape has to be determined first. Once their location is defined, it is possible to define component's BCs and therefore perform a first FE analysis (see Figure 6.1).



**Figure 6.1:** Definition of a first FE problem on the component being designed. The process related to the process block A can be regarded as integrated in a Design PV.

At this stage of the design process, the shape  $\Omega_1$  of the component is rather simple, and does not need to be adapted when generating the simulation model. Therefore in this situation, the shape domains  $\Omega_1$  and  $\Omega_2$ , related to the design model and the mechanical model respectively, coincide. The generation of the FE mesh on a simple shape support could be successfully performed even by a stakeholder of the design PV, who typically has only basic skills about FE meshing and analysis processes. In this context, the behaviour analysis may become a quite transparent process, which could be regarded as integrated into the design PV.

Although at this stage of the design process many details of the component shape are still unknown, differences with the final version are generally not substantially important, and the FE analysis results give valid indications about the mechanical behaviour of the component. The solution of the FE problem over the shape domain  $\Omega_2$  can be considered as a reference solution for the subsequent shape evolutions occurring during the modelling process. FE results provide an insight on some mechanical parameters associated to the model, e.g. strain energy or displacements, otherwise hard to estimate, and therefore the designer gains

information about the areas characterized by large values of such mechanical parameters. In the case where he/she can trust the available FE results, he/she can profit by this information considering it as an additional criterion driving all subsequent design choices.

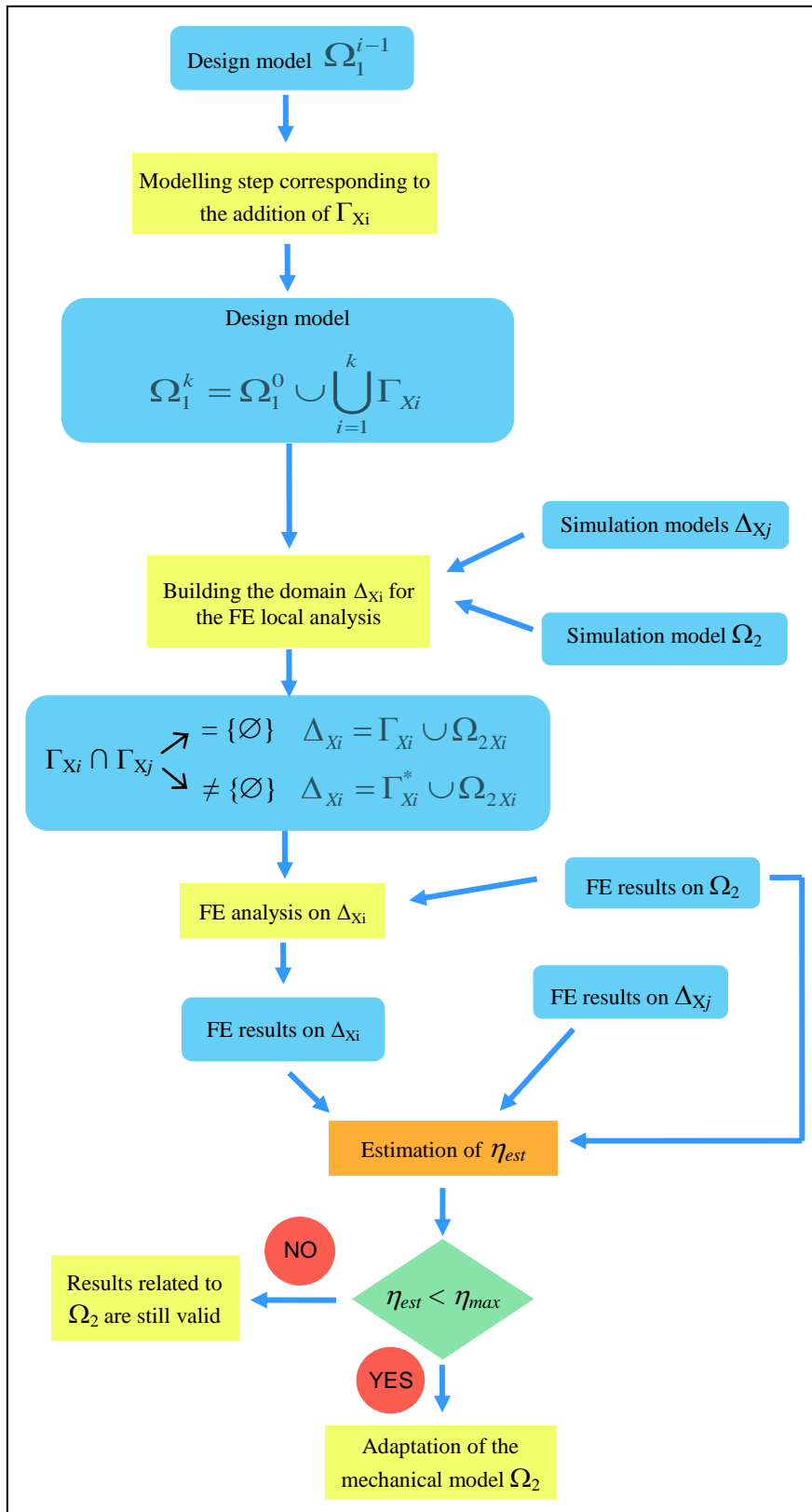
Then, the need emerges for verifying the consistency between the FE results related to the shape domain  $\Omega_2$  and the successive evolutions of the component shape during the design process. The a posteriori mechanical criterion turns out to be useful for this purpose. Starting from the shape domain  $\Omega_1$  which the FE results are related to, the a posteriori mechanical criterion can be applied for each subsequent modification of the design component shape in a CAD environment (see Figure 6.2). The a posteriori criterion will evaluate the influence of each shape sub-domain  $\Gamma_{X_i}$ <sup>1</sup> added to  $\Omega_1$  on the FE results related to  $\Omega_2$ . For each new design primitive occurring in the CAD modeller, we can define the corresponding volume sub-domain,  $\Gamma_{X_i}$ . Hence, the component shape, after the modelling step  $k$ , is

$$\Omega_1^k = \Omega_1^0 \cup \left[ \bigcup_{i=1}^k \Gamma_{X_i} \right], \text{ where } i = \{1, \dots, k\}. \text{ The shape domain } \Omega_1^0 \text{ coincides}$$

with the shape domain of the mechanical model, i.e.  $\Omega_1^0 = \Omega_2$ , as long as the a posteriori criterion ensures that FE results related to  $\Omega_2$  give reliable information about the component's mechanical behaviour. The sub-domain  $\Gamma_{X_i}$  has a simple shape, since it is related to a single CAD modelling operation, and therefore the related FE mesh can be generated in an easy and transparent way. Indeed, if the shape of  $\Gamma_{X_i}$  is simple, its FE mesh will not contain deformed FEs. However, in the case where this occurs, it would be possible to build a polyhedral model of the  $\Gamma_{X_k}$ , which could be easily simplified in order to eliminate areas harmful for the FE mesh generation, as occurring in the example detailed in section 5.4.3. Following the procedure described in section 5.3.2, we must identify  $\Omega_{2X_i}$ , which is a subset of the FE mesh describing  $\Omega_2$  and is formed by the subset of FE elements of  $\Omega_2$  closest to  $\Gamma_{X_i}$ . Then, the sub-domain  $\Delta_{X_i} = \Gamma_{X_i} \cup \Omega_{2X_i}$  is obtained and the a posteriori mechanical criterion can be automatically evaluated without any additional burden for the user, since  $\Delta_{X_i}$  is a small-scale FE model that can be solved quickly to keep up with an interactive environment.

---

<sup>1</sup> We keep the notation used in chapter 5, where the depressions are called additive sub-domains and the protrusions subtractive sub-domains, even if in this scenario the opposite notation would be more meaningful.



**Figure 6.2:** Use of existing FE results to describe the mechanical behaviour of a component during its design process. The process scheme following the Method B and related to the modelling stage  $k$  is detailed.

However, this scenario exhibits some differences if compared to the classical one that has been described in chapter 5, because the sub-domains  $\Gamma_{X_i}$  are not always disconnected. Indeed, if a sub-domain  $\Gamma_{X_i}$  is entirely or partially built on the boundary of  $\Gamma_{X_j}$ , we could have  $\Gamma_{X_i} \cap \Gamma_{X_j} \neq \{\emptyset\}$ , where  $j \in \{1, \dots, (i - 1)\}$ . Then, each sub-domain  $\Gamma_{X_i}$  has no longer a neighbourhood defined solely by  $\Omega_{2X_i}$ , but in addition to  $\Omega_{2X_i}$  the neighbouring sub-domain contains a set of sub-domains  $\Sigma_{X_i-j}$ , where:

- For each  $j$  such that  $\Gamma_{X_i} \cap \Gamma_{X_j} \neq \{\emptyset\}$ , the corresponding  $\Sigma_{X_i-j} \neq \{\emptyset\}$  is formed by the sub-part of the sub-domain  $\Gamma_{X_j}$  closer to  $\Gamma_{X_i}$ ;
- For each  $j$  such that  $\Gamma_{X_i} \cap \Gamma_{X_j} = \{\emptyset\}$ , we have  $\Sigma_{X_i-j} = \{\emptyset\}$ .

In the case where we have some  $\Sigma_{X_i-j} \neq \{\emptyset\}$ , the generation of the local FE problem is different in comparison with the typical one described in section 5.3.2. We describe two different methods that can be used for generating  $\Delta_{X_i}$ . However, at present, no further tests have been performed to assure the reliability of the Method A.

#### - Method A

The first option consists in estimating the influence of  $\Gamma_{X_i}$  independently from that of sub-domains  $\Gamma_{X_j}$  adjacent to it. Then, when looking for the 2-3 FE layers closer to the sub-domain  $\Gamma_{X_i}$ , we may retrieve a neighbourhood formed by the sub-domain  $\left[ \bigcup_{j=1}^{i-1} \Sigma_{X_i-j} \right] \cup \Omega_{2X_i}$ . The sub-domain  $\Delta_{X_i}$  where the a posteriori criterion is applied becomes:

$$\Delta_{X_i} = \Gamma_{X_i} \cup \left[ \bigcup_{j=1}^{i-1} \Sigma_{X_i-j} \right] \cup \Omega_{2X_i}. \quad (6.1)$$

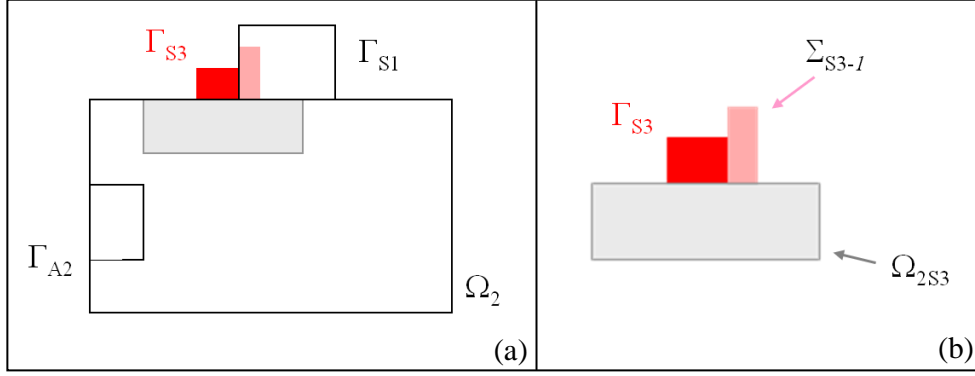
When setting BCs of  $\Delta_{X_i}$ , we have to consider:

- The FE results on  $\Omega_2$  in order to define BCs over  $\Omega_{2X_i}$ ;
- The FE results on  $\Delta_{X_j}$  in order to define BCs over  $\Sigma_{X_i-j}$ , where  $j$  spans all the configurations such that  $\Gamma_{X_i} \cap \Gamma_{X_j} \neq \{\emptyset\}$ .

Although the non-conformity of  $\Delta_{X_i}$  increases if compared to the classical scenario of application of the a posteriori criterion, the use of kinematic BCs to bind all the sub-domains contributing to  $\Delta_{X_i}$  is still applicable.

At worst, in the case where  $\Gamma_{X_i} \cap \Omega_2 = \{\emptyset\}$ , it could also happen that the neighbourhood of  $\Gamma_{X_i}$  contains solely sub-domains  $\Sigma_{X_i-j}$ , while  $\Omega_{2X_i} = \{\emptyset\}$ .





**Figure 6.3:** Simple 2D example of retrieval of a sub-domain  $\Delta_{X_i}$  by using the method described in the Method A: (a) When adding the sub-domain  $\Gamma_{S_3}$  to the shape domain  $\Omega_2$ , we find that  $\Gamma_{X_3} \cap \Gamma_{X_1} \neq \{\emptyset\}$ . Therefore, the neighbourhood of  $\Gamma_{S_3}$  is formed by  $\Sigma_{X_{3-1}} \cup \Omega_{2X_3}$ ; (b) The local FE problem is performed on a sub-domain  $\Delta_{X_3} = \Gamma_{X_3} \cup \Sigma_{X_{3-1}} \cup \Omega_{2X_3}$ .

A simple 2D example of the generation of a sub-domain  $\Delta_{X_i}$  with the Method A is given in Figure 6.3. At the step  $i = 3$  of the incremental shape modelling process, the sub-domain  $\Gamma_{S_3}$  is added. For each  $j \in \{1, 2\}$ , the intersection between  $\Gamma_{S_3}$  and  $\Gamma_{X_j}$  is estimated. Since we have  $\Gamma_{X_3} \cap \Gamma_{X_1} \neq \{\emptyset\}$ , when looking in the neighbourhood of  $\Gamma_{X_3}$ , we find a sub-domain  $\Sigma_{X_{3-1}} \cup \Omega_{2X_3}$ . Then, the sub-domain where performing the local FE problem is  $\Delta_{X_3} = \Gamma_{X_3} \cup \Sigma_{X_{3-1}} \cup \Omega_{2X_3}$ . The BCs related to  $\Omega_{2X_3}$  are retrieved by looking at FE results over  $\Omega_2$ , while those related to  $\Sigma_{X_{3-1}}$  by looking at FE results over  $\Delta_{X_1}$ .

#### – Method B

At present, no tests have been in fact performed regarding situations where  $\Gamma_{X_i} \cap \Gamma_{X_j} \neq \{\emptyset\}$  for some  $j \in \{1, \dots, (i - 1)\}$ . Therefore, in the case where it occurs, we evaluate the influence  $\eta_{est-i}$  based on the sub-domain formed by

$$\Gamma_{X_i}^* = \Gamma_{X_i} \cup \left[ \bigcup_j \Gamma_{X_j} \right], \text{ where } j \text{ spans all the configurations where } \Gamma_{X_i} \cap \Gamma_{X_j} \neq$$

$\{\emptyset\}$ . The resulting sub-domain  $\Delta_{X_i}$  is:

$$\Delta_{X_i} = \Gamma_{X_i}^* \cup \Omega_{2X_i}, \quad (6.2)$$

where the sub-domain  $\Omega_{2X_i}$  is built in the neighbourhood of  $\Gamma_{X_i}^*$ , and BCs are set considering only the FE results related to  $\Omega_2$ . The process scheme of Figure 6.2 refers to this configuration.

By means of the proposed behaviour modeller paradigm, the designer is able to determine, during the component modelling process, the impact of shape changes from a mechanical behaviour point of view. Therefore, it is possible to have available the information about the mechanical behaviour while the component

shape is evolving, without performing a FE analysis on the whole component for each shape modification, and hence without effective need for moving to a behaviour simulation PV.

The solution of the local FE problem related to the sub-domain  $\Delta_{X_i}$  provides the user with two kinds of criteria:

1. The global influence  $\eta_{est}$  (see Eq. 5.14) of all his/her  $n$  modelling steps on the FE analysis results over the shape domain  $\Omega_2$ . The FE analysis performed at the early stage of design on the shape model  $\Omega_2$  is used as long  $\eta_{est} < \eta_{max}$ , where  $\eta_{max}$  is the accuracy threshold value set by the user. In the case where, when adding a shape sub-domain  $\Gamma_{X_i}$ , the threshold value  $\eta_{max}$  is exceeded, the designer can decide whether redefining the shape domain  $\Omega_2$  of the mechanical model.

It should be noticed that, in the case where we have some  $j \in \{1, \dots, (i-1)\}$  such that  $\Gamma_{X_i} \cap \Gamma_{X_j} \neq \{\emptyset\}$  and we use the Method *B* to generate the sub-domain  $\Delta_{X_i}$ , when solving the local FE problem related to  $\Delta_{X_i}$  we actually assess the influence due to the union  $\Gamma_{X_i}^*$  of the sub-domains interconnected. Therefore, when estimating  $\eta_{est}$ , we do not have to consider no longer the  $\eta_{est-j}$  related to the sub-domains  $\Gamma_{X_j}$  that have been incorporated into  $\Gamma_{X_i}^*$ .

The adapted sub-domain  $\Omega'_2$  will include the  $\Gamma_{X_z}$  providing the maximum value of  $\eta_{est-z}$ , i.e.  $\Omega'_2 = \Omega_2 \cup \Gamma_{X_z}$ . Then, a new FE analysis related to  $\Omega'_2$  has to be performed, and the successive steps of shape modelling will refer to the new FE analysis results.

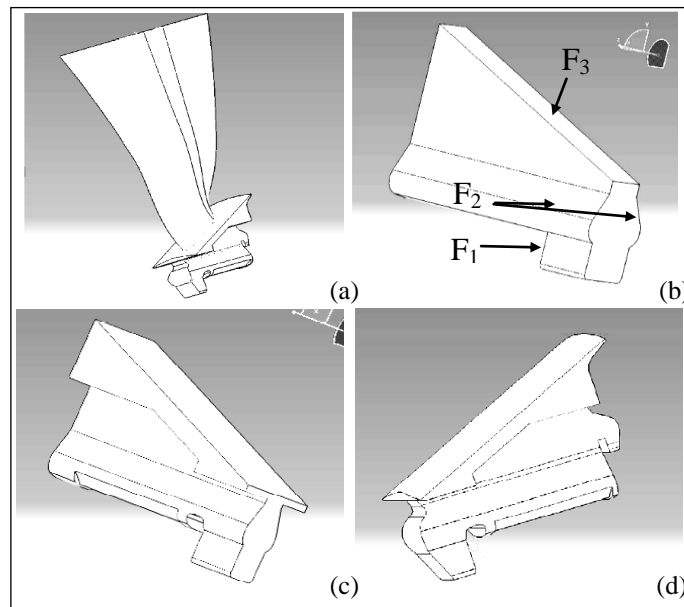
In particular, if  $\Gamma_{X_z}$  is interconnected with other shape sub-domains, the shape sub-domain  $\Omega_2$  will be adapted as it follows:

- In the case where the Method *B* is chosen, a  $\Gamma_{X_z}^*$  has been retrieved. Then,  $\Omega'_2 = \Omega_2 \cup \Gamma_{X_z}^*$ ;
- In the case where the Method *A* is chosen, the influence of  $\eta_{est-k}$  has been estimated independently from that of the other sub-domains, and therefore we can adapt the shape of the mechanical model as  $\Omega'_2 = \Omega_2 \cup \Gamma_{X_z}$ . The influence  $\eta_{est-z}$  of the sub-domains  $\Gamma_{X_z}$  adjacent to  $\Gamma_{X_i}$  need to be reassessed based on the FE results related to the adapted shape domain  $\Omega'_2$ , since its contribution to  $\eta_{est}$  could have changed. However, it could be happen that  $\Gamma_{X_z} \cap \Omega_2 \neq \{\emptyset\}$ . In such particular case, we are obliged to reinsert  $\Gamma_{X_z}$  together with its adjacent sub-domains, thus falling in the case where  $\Omega'_2 = \Omega_2 \cup \Gamma_{X_z}^*$ .

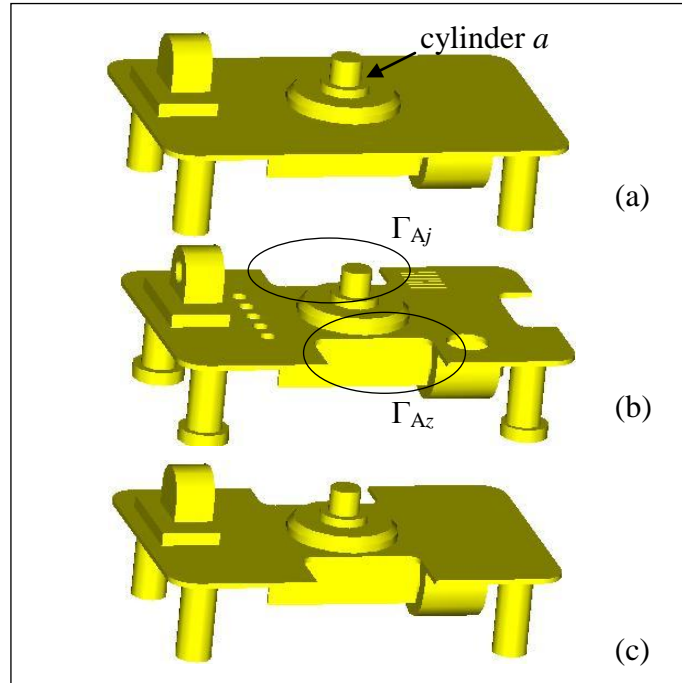
2. The influence  $\eta_{est-k}$  of his/her last  $k$  modelling step on the FE analysis results over the shape domain  $\Omega_2$  (or  $\Omega'_2$  if it has been previously adapted). This

value can be considered as an influence indicator assessing the influence of the shape sub-domain on FE results. The user could even decide not to insert the shape sub-domain  $\Gamma_{X_k}$  into the design model  $\Omega_1$ . In this case, the shape domain  $\Omega_2$  of the mechanical model stays valid in describing the mechanical behaviour of the component  $\Omega_1$ . This is the typical case of behaviour analysis supporting decision making process, where FE results can influence the product modelling and validate or not design alternatives.

Figure 6.4 shows a practical example of design process that could be monitored by the a posteriori criterion, which is related to a fan blade. The fan blade to be modelled is depicted in Figure 6.4(a). Figure 6.4 (b) shows the design model at the stage where we are able to define functional surfaces  $F_1$ ,  $F_2$  and  $F_3$ . Based on some idealizations about the BCs linked to the aerodynamic part attached to  $F_3$ , a first FE analysis can be performed by using this shape domain. BCs idealization is justified because the a posteriori criterion provides a global clue rather than quantifying the local distribution of mechanical fields. Figure 6.4(d) shows the shape model at the end of the design stage, which has been enriched with shape details. The a posteriori mechanical criterion, applied each time that a shape sub-domain  $\Gamma_{X_i}$  is added, could estimate the global shape sub-domain influence over the FE results available, therefore allowing the designer to interactively monitor the mechanical behaviour of the component along its shaping process.



**Figure 6.4:** Example of component design where the a posteriori criterion could be usefully employed: (a) fan blade to be designed; (b) component shape after the first three modelling steps, where the functional surfaces have appeared; (c) intermediate design stage; (d) final shape at the end of the modelling process.



**Figure 6.5:** Example of adaptation of the shape domain  $\Omega_2$  of a mechanical model during the component design: (a) Shape domain  $\Omega_2$ , where the initial FE analysis is performed; (b) Shape domain  $\Omega_1^n$  related to the final design model, which has been obtained through  $n$  CAD modelling step; (c) Shape domain  $\Omega'_2$ , where the use of the mechanical criterion with a threshold value  $\eta_{max} = 10\%$  has highlighted the need for redefining the simulation model.

Figure 6.5 shows an example of the application of the a posteriori criterion in this scenario. Figure 6.5(a) illustrates the domain  $\Omega_2$  where a first FE analysis is performed. The table legs are clamped and a pressure is applied on the cylindrical surface  $a$ . Figure 6.5(b) depicts the final design of the component. If we set a threshold value  $\eta_{max} = 30\%$ , when applying the a posteriori mechanical criterion  $\eta_{est}$ , it is possible to verify that the addition of any sub-domain  $\Gamma_{Xi}$  guarantees the desired accuracy. If we set a threshold value  $\eta_{max} = 10\%$ , the two sub-domains  $\Gamma_{Aj}$  and  $\Gamma_{Az}$  indicated in Figure 6.5(b) become prominent and decisions about their design need to be taken. If the user considers essential the insertion of the corresponding shape features in the design model, a new model  $\Omega'_2$  must be investigated, as support of the behaviour simulation model,  $\Omega'_2 = \Omega_2 \cup \Gamma_{Xj} \cup \Gamma_{Xz}$  (see Figure 6.5(c)).

#### 6.4 Consistency of mechanical models when designing component variants

The developed a posteriori mechanical criterion could be usefully employed also when different shape versions of an existing component need to be designed. The criterion is applicable if data related to the PDP of a first designed version of the component are available, including shape models and FE analysis results.

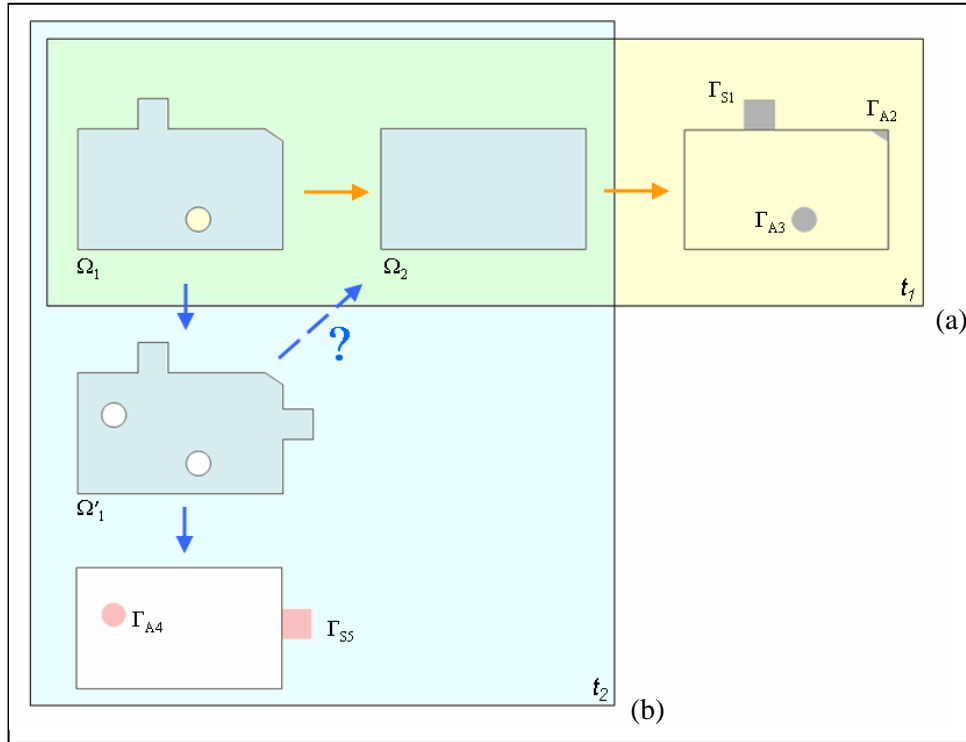
Compared to the two scenarios analysed in section 6.3 and in chapter 5, the chronological distance between the first FE analysis, performed at a time  $t_1$ , and the assessment of the mechanical influence of shape modifications, performed at a time  $t_2$ , is much larger. Anyway, this has no influence on the effectiveness of the a posteriori mechanical criterion.

$\Omega_1$  is the shape domain characterizing the initial shape of the design model and  $\Omega'_1$  is the one related to its new design version. Here, the aim is to state whether the FE analysis results, which describe the mechanical behaviour of a design model having the shape domain  $\Omega_1$  and are related to the shape domain  $\Omega_2$  of the corresponding mechanical model, are still accurate or not, despite the shape modifications occurred on the component when moving from  $\Omega_1$  to  $\Omega'_1$ . The shape details removed during the transition from  $\Omega_1$  to  $\Omega_2$  are the sub-domains  $\Gamma_{X_j}$ , with  $j \in \{1, \dots, n\}$ . We suppose that at the time  $t_1$ , the shape models of the sub-domains  $\Gamma_{X_j}$  have been retrieved and stored, and their influences  $\eta_{est-i}$  on FE results related to  $\Omega_2$  have been estimated.

First, it is necessary to characterize the shape differences between the two different shape versions of the component. These differences are related to the shape domains of the design models, i.e.  $\Omega_1$  and  $\Omega'_1$ . Here, we assume that the shape modifications performed over  $\Omega_1$  to obtain the new shape version  $\Omega'_1$  are rather local, so that it is possible to characterize them through disconnected sub-domains  $\Gamma_{X_i}$ , with  $i \in \{(n+1), \dots, p\}$ .

Figure 6.6 illustrates a simple 2D example of retrieval of sub-domains  $\Gamma_{X_j}$  and  $\Gamma_{X_i}$ , while the whole scenario will be illustrated in Figure 6.7. In Figure 6.6(a), the retrieval of shape sub-domains  $\Gamma_{X_j}$  performed at the time  $t_1$ , with  $j \in \{1, 2, 3\}$ , is highlighted. The sub-domains  $\Gamma_{X_j}$  correspond to shape modifications performed when moving from  $\Omega_1$  to  $\Omega_2$ . Then, at the time  $t_2$  (see Figure 6.6(b)), a new shape version  $\Omega'_1$  of the component is designed. The user wonders whether the FE results available, computed on the shape domain  $\Omega_2$ , are still able to describe with enough accuracy the mechanical behaviour of the new shape version  $\Omega'_1$  of the design model. Then, he/she characterizes the shape differences between  $\Omega_1$  and  $\Omega'_1$  by retrieving the sub-domains  $\Gamma_{X_i}$ , with  $i \in \{4, 5\}$ .

The characterization of shape differences between shape models is a complex scientific topic, which can be addressed by using different methods.



**Figure 6.6:** Simple 2D example of retrieval of sub-domains  $\Gamma_{X_j}$  and  $\Gamma_{X_i}$ : (a) When moving from  $\Omega_1$  to  $\Omega_2$  at the time  $t_1$ , the sub-domains  $\Gamma_{X_j}$ , with  $j \in \{1, 2, 3\}$ , are suppressed; (b) A new shape version  $\Omega'_1$  of the component is designed. Sub-domains  $\Gamma_{X_i}$ , with  $i \in \{4, 5\}$ , characterizing the shape differences between  $\Omega_1$  and  $\Omega'_1$ , are retrieved.

Most of the methods existing in the literature are mainly devoted to shape similarity assessments. Surveys about the existing techniques can be found in [CGK03], and, more recently, in [HLK06]. However, they are not completely suited to our specific needs. Indeed, we need to know local exact shape differences between two design models that have been designed in a CAD environment. More precisely, two distinct situations may occur in our context:

- The generation of the shape domain  $\Omega_1$  and the shape modification process generating the shape domain  $\Omega'_1$  are performed into the same CAD environment. Then, the shape differences between the two versions of the components can be retrieved by exploiting the information contained into the two history trees. However, this task is not trivial. Even in the case where the history trees are available, the only explicit representation is that related to the final model, and automatic operators should be set up in order to retrieve shape differences. Although some CAD systems, e.g. SolidWorks [Sol] have developed ad hoc tools that are able to perform this task;
- The generation of the shape domain  $\Omega_1$  and the shape modification process generating the shape domain  $\Omega'_1$  are performed into different CAD software environments. Tools cited above are subjected to the tolerances of CAD modellers and could therefore fail. An alternative approach could consist in

retrieving shape differences in the software environment based on the mixed shape representation. Since we have CAD data available, it is possible to obtain HLT representations of  $\Omega_1$  and  $\Omega'_1$  together with their polyhedral ones. In particular, we could use the same tessellation parameters when generating the polyhedral models. Then, shape differences could be retrieved by reasoning on the information provided by both the available representations. In addition, working in the software environment based on the mixed shape representation, we are able to easily store shape models of sub-domains  $\Gamma_{X_i-open}$ , with  $i \in \{(n+1), \dots, p\}$ , that characterize shape differences between models  $\Omega_1$  and  $\Omega'_1$ , and to generate their corresponding volume versions  $\Gamma_{X_i}^2$  to be used during the a posteriori analysis. It should be anyway noticed that, at present, no actual operator has been developed taking into account the above considerations.

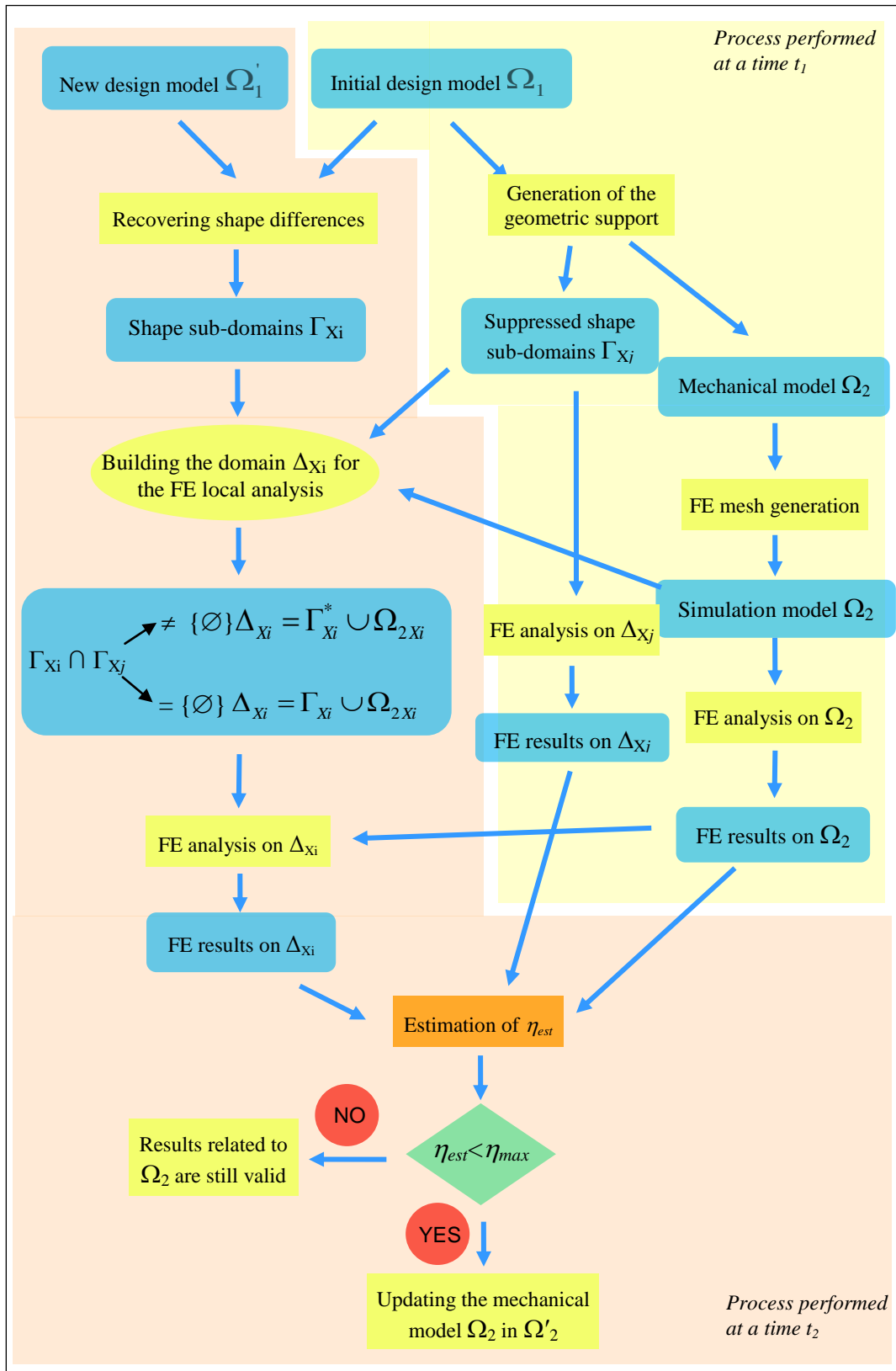
Estimating the influence  $\eta_{est-i}$  on the FE results related to  $\Omega_2$  of the shape sub-domains  $\Gamma_{X_i}$  inserted on  $\Omega_1$ , where  $i \in \{(n+1), \dots, p\}$ , leads to different situations, depending on the geometric interactions between the sub-domains  $\Gamma_{X_i}$ , the shape domain  $\Omega_2$  and the shape sub-domains  $\Gamma_{X_j}$ , with  $j \in \{1, \dots, n\}$ , which were removed when moving from  $\Omega_1$  to  $\Omega_2$ :

- If shape modifications performed on  $\Omega_1$  are such that  $\Gamma_{X_i} \cap \Gamma_{X_j} = \{\emptyset\}$  for each  $j \in \{1, \dots, n\}$ , the domain where estimating the influence  $\eta_{est-i}$  has the classical formulation,  $\Delta_{X_i} = \Gamma_{X_i} \cup \Omega_{2X_i}$ ;
- If shape modifications performed on  $\Omega_1$  are such that  $\Gamma_{X_i} \cap \Gamma_{X_j} \neq \{\emptyset\}$  for some  $j \in \{1, \dots, n\}$ , considerations analogous to those of section 6.3 should be done.

Based on the above analysis, we are able to know how much the available FE results are still accurate despite the shape modifications  $\Gamma_{X_i}$  occurring on the component when moving from  $\Omega_1$  to  $\Omega'_1$ . If  $\eta_{est}$  indicates that some  $\Gamma_{X_i}$  strongly influences the FE results, the shape domain  $\Omega_2$  of the mechanical model needs to be redefined analogously as detailed in section 6.3. The process scheme of this process showed in Figure 6.7 follows Method *B*.

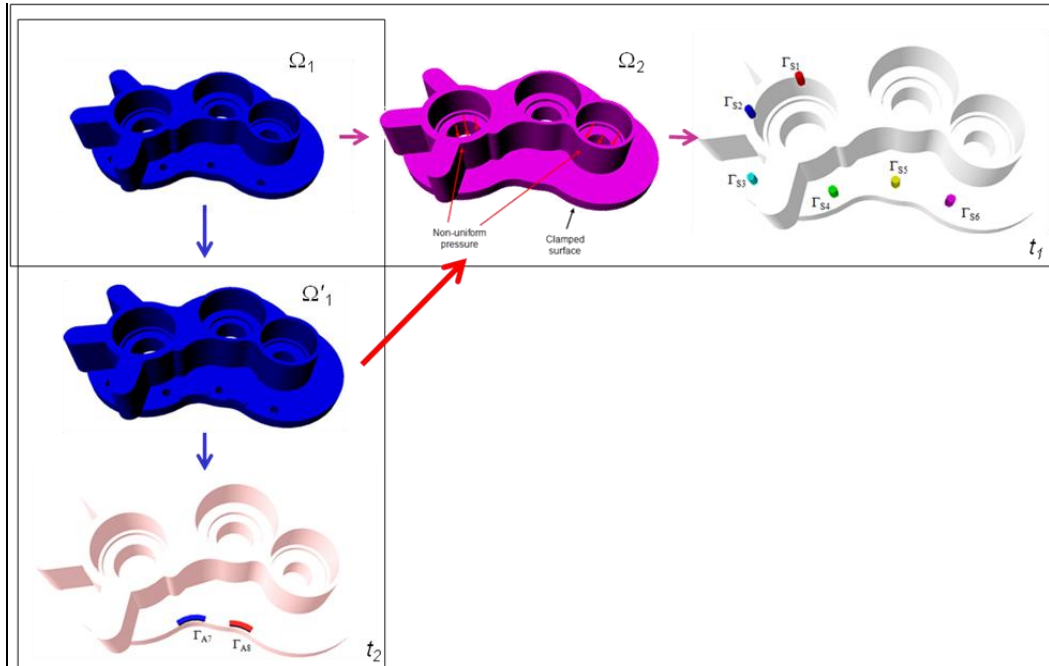
---

<sup>2</sup> For sake of simplicity, in the description of the processes presented in Figure 6.7, and also later on in Figure 6.9, we imply the distinction between a shape sub-domain  $\Gamma_{X_j-open}$  and its corresponding closed version  $\Gamma_{X_j}$ , and we make always reference to  $\Gamma_{X_j}$ .



**Figure 6.7:** Process evaluating whether the mechanical model having the shape domain  $\Omega_2$  is able to reliably describe the mechanical behaviour of the two different variants  $\Omega_1$  and  $\Omega'_1$  of a design model. The process scheme is related to Method B.





**Figure 6.8:** Example of shape differences recovery between different design versions  $\Omega_1$  and  $\Omega'_1$  of the same component. The a posteriori criterion confirms that the mechanical model  $\Omega_2$  is valid to describe the mechanical behaviour of  $\Omega'_1$ .

Figure 6.8 shows an example of this scenario. Here, the mechanical model  $\Omega_2$  generated at a time  $t_1$  has provided high-accurate FE results, i.e.  $\eta_{est} = 0.05\%$ . At a time  $t_1$ , a different version of the component is designed,  $\Omega'_1$ , where some shape modifications have occurred. Shape differences between the two models are recovered, which are characterized by the shape sub-domain  $\Gamma_{A7}$  and  $\Gamma_{A8}$ . The a posteriori criterion confirms that the global influence of the shape variations occurred is negligible on the FE results related to  $\Omega_2$ , and therefore we can still use this mechanical model to describe the mechanical behaviour of  $\Omega'_1$ .

## 6.5 Impact of BCs modification on the shape of a mechanical model

This scenario is related to a situation differing from those ones considered in this chapter, where the objective was to state if the mechanical behaviour of two different design models could be described by the same mechanical model. Here, no modifications of the design model shape have occurred, but a new behaviour analysis needs to be performed with new BCs associated. Therefore, we need to redefine the mechanical model by considering the new mechanical hypotheses. This often happens in an industrial context, when an initial component having a shape  $\Omega_1$  is candidate for being reused in a new product. In this case, generating a mechanical model that exploits the shape domain  $\Omega_2$  related to a mechanical model previously defined could provide remarkable time saving. Anyway, it could happen that the initial simulation model related to the shape domain  $\Omega_2$  does not include some

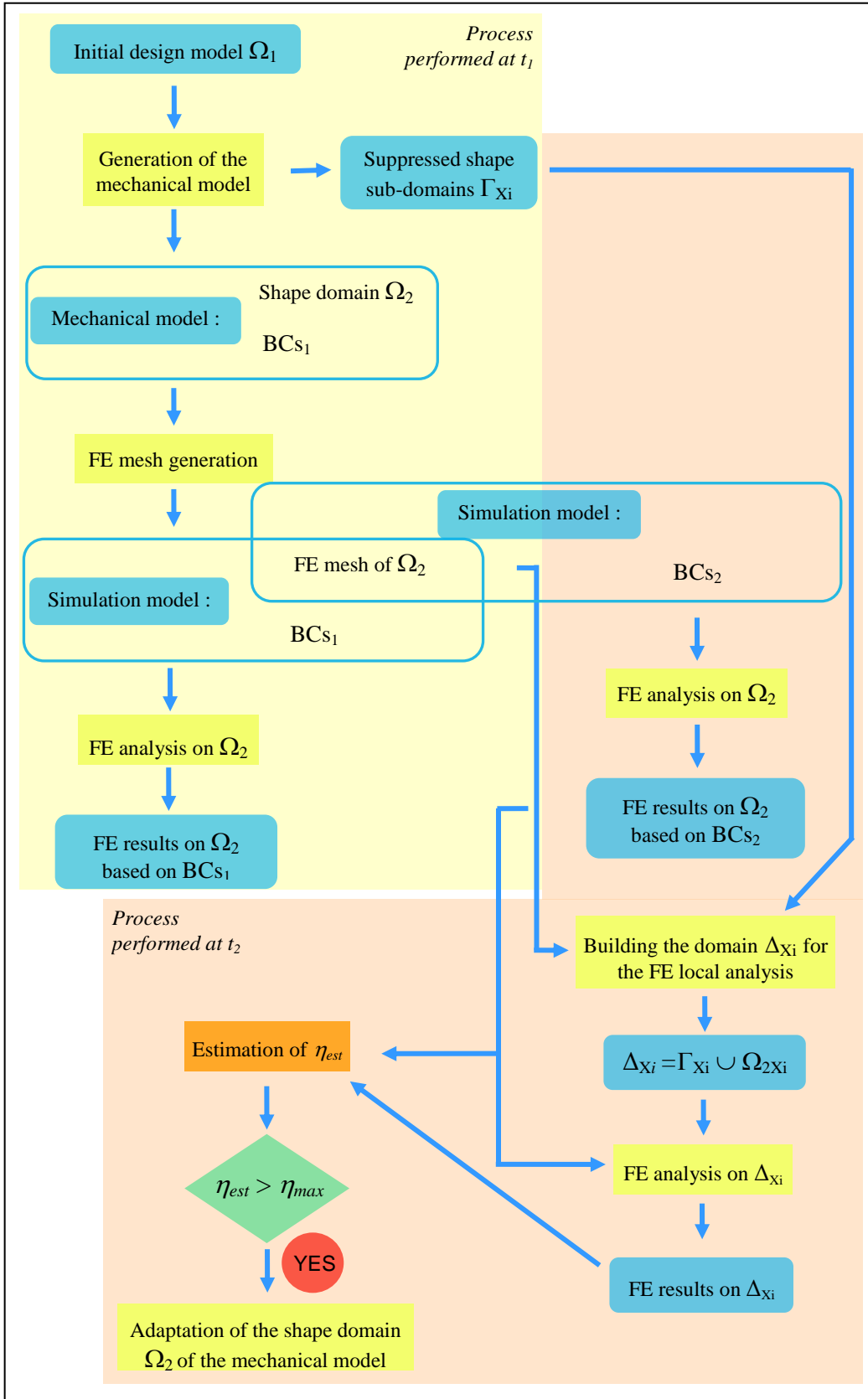
shape sub-domains  $\Gamma_{X_i}$  that are essential in order to obtain accurate FE results. Therefore, the use of the a posteriori mechanical criterion can be useful here for evaluating whether using the shape domain  $\Omega_2$  for the generation of the new mechanical model still returns reliable FE analysis results. Obviously, the a posteriori analysis can be performed only if data related to the shape domain  $\Omega_2$  related to the old mechanical model are available.

We assume that the modifications of the BCs do not imply the creation of new shape sub-domains. We perform a new FE analysis on the shape domain  $\Omega_2$  by considering the new BCs, and we evaluate the influence of each shape sub-domain  $\Gamma_{X_i}$  on the new FE analysis results by applying the a posteriori mechanical criterion to the sub-domain  $\Delta_{X_i}$ . If some shape sub-domains  $\Gamma_{X_i}$  exhibit a significant influence on the new FE results, we need to prepare a new mechanical model, whose shape domain  $\Omega_2$  keeps into account all the sub-domains  $\Gamma_{X_i}$  that are needed.

The adapted shape domain of the mechanical model will be  $\Omega'_2 = \Omega_2 \cup \left[ \bigcup_j \Gamma_{X_j} \right]$ ,

where  $j$  spans the  $\Gamma_{X_j}$  that make  $\eta_{est} > \eta_{max}$ .

Figure 6.9 shows the process flow summarizing the analysis of the scenario where the BCs part of the mechanical and the simulation models are changing. BCs<sub>1</sub> are those related to the first FE analysis, while BCs<sub>2</sub> are the new ones.



**Figure 6.9:** Use of the same shape domain  $\Omega_2$  for performing different FE analyses.

## 6.6 Conclusions

The use of the a posteriori criterion introduced in chapter 5 has been investigated in different situations that may occur in a PDP. The objective is to show that the a posteriori criterion could provide a useful mechanical criterion driving the designer choices without performing a complete FE analysis at each shape modification. Due to its user-friendliness, in some cases the use of the a posteriori criterion could be considered as transparent for the user and integrated into a design PV, where the stakeholders have only basic competences about FE meshing and computations. This would make it as an actual CAE tool providing substantial support to the stakeholder of the Design PV.

However, at present, the study of the scenarios detailed in this chapter has been limited to an analysis stage. Requirements of the a posteriori criterion and differences among the different scenarios have been detailed. An actual implementation and effective tests are part of perspectives.





## Conclusions and Perspectives

The present work has been motivated by the need of processing product shapes for the communication between the various activities of a Product Development Process. This is particular important when moving between different PDP activities which make use of different shape descriptions and representations, are associated to application-specific information, and are performed by different stakeholders. In this context, the concepts of Product View and of Product View Reference Model naturally emerged.

In particular, we have investigated the transfer of information from the design to the behavior simulation PVs, and the needs in terms of shape modeling capabilities and formalization of the information involved in defining a mechanical problem. In the proposed approach, an intermediate model, i.e. the mechanical model, is generated at the interface between the considered PVs. The definition of the mechanical model consists in characterizing an appropriate shape domain, based on the hypotheses and objectives related to the specific mechanical problem. Therefore, the knowledge related to the mechanical analysis to be performed is a primary factor in the generation process of the mechanical model.

In this work, we propose a general framework for translating some of the problem hypotheses and objectives in terms of shape constraints driving the generation of the mechanical model or, later in the process, of the simulation model. Moreover, in some cases, we show how to set a correspondence between model shape modifications and mechanical hypotheses associated to the problem.

The key elements of the proposed methodology are:

- The use of the **mixed shape representation**. When CAD data are available, it combines the polyhedral representation with the B-REP NURBS data. The shape transformation operators act on the polyhedral model, the master model, in order to be more generic and robust. In addition, when the input model comes from a CAD system, we represent the B-Rep NURBS and the polyhedral models simultaneously. This allows us to exploit geometric and topological information of higher level than those contained in the polyhedral model. The efficiency of the shape processing operators is therefore improved, and the complexity of the detail identification tasks needed during the preparation of simulation models is reduced;
- The setting of the concept of **multiple topological layers**. This allows the association of additional data to the shape models in the software environment based on the mixed shape representation. By using the HLT data structure introduced in [HAM06], it is possible to obtain any decomposition of the object boundary, including arbitrary non-manifold ones. Therefore, by means of convenient boundary decompositions, each

one associated to a dedicated topological layer, it is possible to define the geometric and topological support for attaching additional information to the model. These boundary decompositions can act as constraints when adapting the shape domain of the mechanical model;

- Integration of a **surface FE mesh generator** in the software environment based on the mixed shape representation. In this way, an explicit formulation of the constraints related to a mechanical problem in terms of shape could be useful not only in adapting the shape domain of the mechanical model, but also for generating the actual simulation model, i.e. the FE mesh;
- The characterization of **simplification details** as shape sub-domains that can be suppressed without influencing the mechanical behaviour of the associated shape model. A mechanical criterion, either a priori or a posteriori, is applied in relation to a volume shape variation, characterized in terms of shape sub-domain. Thus, the removal of a simplification detail is consistent with the hypotheses related to the particular mechanical problem, and can be considered as an additional way of characterizing mechanical hypotheses over the shape domain of the mechanical model;
- The use of an **a posteriori mechanical criterion** that characterizes the influence of shape variations on FE results. In the proposed approach, the removal of shape sub-domains during the a priori shape adaptation process is validated in an a posteriori stage, by using a specific a posteriori mechanical criterion. Actual simplification details are only those whose impact over the FE results, computed by means of such criterion, turns out to be negligible. In contrast, if the a posteriori criterion does not validate the removal of some shape sub-domains, the shape domain of the mechanical (simulation) model must be adapted in order to provide more accurate simulation results. The a posteriori mechanical criterion can be useful not only during the adaptation of the shape domain where performing the behaviour analysis, but also in additional scenarios of a PDP. Indeed, it can be applied whenever some FE results are available, computed on shape versions obtained by adding or subtracting shape sub-domains of arbitrary size and shape from an initial shape version of the design model.

The results obtained in this thesis work demonstrate the feasibility and the potentialities of our approach in improving the integration of the design and the behaviour simulation PVs. To make the devised framework fully operative in real engineering environments several issues must be addressed:

- Complete implementation of the concept of multiple topological layers. At present, the explicit representation of the multiple topological layers is still not implemented, and their related information is attached to a single boundary representation by means of attributes. Therefore, the shape transformations are constrained only by one boundary representation, which conveys the information associated to different concepts;



- Further investigation of the concept of shape interface, which at present has been considered only in the case of simple configurations. Transfer of shape interfaces during the shape adaptation should be studied. A meaningful definition of shape interfaces has to be devised also in the case where differences between shape models consist in several connected volume sub-domains;
- Exploitation of the basic operators modifying the boundary decomposition. This is needed in order to define new operators for the automatic identification, removal and storage of shape sub-domains. Indeed, at present, automatic operators have been set up only in the case of holes and fillet form features, while the identification of other shape sub-domains, e.g. small protrusions, is performed interactively by the user;
- Integration in our software environment of a volume FE mesh generator. At present, only a surface FE mesh generator has been integrated. The execution of the entire process at the interface between the Design and Behaviour Simulation PVs in the same software environment would significantly improve the integration between these PVs;
- Further automation of the re-adaptation process of the simulation model, based on the indications provided by the a posteriori mechanical criterion. At present, only simple examples of re-adaptation have been considered, which are still associated to restrictive hypotheses limiting the reachable configurations. This would allow us to consider the a posteriori criterion as an effective constraint, which aims at providing the shape of the mechanical model without excessive burden for the user;
- Testing and automating the use of the a posteriori criterion in all the analysed PDP scenarios. The availability of an actual CAE tool during the design process would provide substantial support to the designer. Indeed, it would allow him/her to explore more design alternatives, thus contributing to more effective design decisions;

Finally, from a more general point of view, a promising perspective is the extension of the concepts introduced in the present framework, in view of the integration of any PVs occurring during a PDP.



## Bibliography

- [Aba] Abaqus, from Dassault Systemes, <http://www.simulia.com/>.
- [ADS06] Aifaoui N., Deneux D., Soenen R., *Feature-based interoperability between design and analysis processes*, Journal of Intelligent Manufacturing, Vol. 17, pp. 13–27, 2006.
- [ABD\*98] Armstrong C.G., Bridgett S.J., Donaghy R.J., McCune R.W., McKeag R.M., Robinson D.J., *Techniques for Interactive and Automatic Idealisation of CAD Models*, Proceedings of the 6<sup>th</sup> International Conference on Numerical Grid Generation in Computational Field Simulations, University of Greenwich, , pp. 643–662, 1998.
- [ABR\*03] Attene M., Falcidieno B., Rossignac J., Spagnuolo M., *Edge-sharpener: recovering sharp features in triangulations of non-adaptively re-meshed surfaces*, Proc. of the Eurographics ACM SIGGRAPH Symposium on Geometry Processing, ACM Press, pp. 62-69, 2003.
- [ABS91] Arabshahi S., Barton D.C., Shaw N.K., *Towards Integrated Design and Analysis*, Finite Elements in Analysis and Design, Vol. 9(4), pp. 271-293, 1991.
- [Aia98] AIAA (American Institute of Aeronautics & Astronautics), *Guide for the Verification and Validation of Computational Fluid Dynamics Simulations*, AIAA G-077-1998, 1998.
- [Alt93] Altिंग L., *Life-cycle design of products: A new Opportunity for Manufacturing Enterprises*, *Concurrent Engineering: Automation, Tools and Techniques*, Edited by Andrew Kusiak, John Wiley & Sons, 1993.
- [AO93] Ainsworth M. and Oden J.T., *A unified approach to a posteriori estimation using element residual methods*, Numer. Math., Vol. 65, pp. 23-50, 1993.
- [AO97] Ainsworth M. and Oden J.T., *A posteriori error estimation in finite element analysis*, Computer Methods in Applied Mechanics and Engineering, Vol. 142(1-2), pp. 1-88, 1997.
- [Arm94] Armstrong C.G., *Modelling Requirements for Finite-Element Analysis*, Computer-Aided Design, Vol. 26(7), pp. 573-578, 1994.
- [ARM\*95] Armstrong C.G., Robinson D.J., McKeag R.M., Li T.S., Bridgett S.J., Donaghy R.J., McGleenan C.A., *Medials for Meshing and More*, Proceedings of 4<sup>th</sup> International Meshing Roundtable, Sandia National Laboratories, pp. 277–288, 1995.
- [BBB00] Belaziz M., Bouras A., Brun J.M., *Morphological analysis for product design*, Computer-Aided Design, Vol. 21(5-6), pp. 377–88, 2000.
- [BB94] Blacker T. and Belychko T., *Superconvergent patch recovery with equilibrium and conjoint interpolant enhancements*, Int. J. Numer. Methods Engrg., Vol. 37, pp. 517-536, 1994.

- [BDK98] Barequet G., Duncan C.A., Kumar S., *RSVP: A geometric toolkit for controlled repair of solid models*, IEEE Trans. on Visualization and Computer Graphics, Vol. 4(2), pp. 162-177, 1998.
- [Boi95] Boix E., approximation linéaire des surfaces de R3 et applications, PhD thesis, Ecole Polytechnique, Paris, 1995.
- [Bou97] Bousquet J., Détection et élimination d'irrégularités sur les surfaces manipulées en CAO, PhD thesis, Université de Nantes, Nantes, 1997.
- [BN03] Broonsvoort W.F., Noort A., Multiple-view feature modelling for integral product development, Computer-Aided Design, Vol. 36, pp. 929-946, 2003.
- [BK94] Beitz W., Kuttner K.H., *Handbook of Mechanical Engineering*, Springer Verlag, 1994.
- [BR78] Babuska I. and Rheinboldt W.C., *A posteriori error estimates for the finite element method*, Int. J. Numer. Methods Engrg, Vol. 12, pp. 1597-1615, 1978.
- [BR82] Babuska I. and Rheinboldt W.C., Computational error estimates and adaptive processes for some non linear structural problems, Comp. Math. Appl. Mech. Engng., Vol. 34, pp. 895-937, 1982.
- [BSU\*94] Babuska I., Strouboulis T., Upadhyay C.S., Gangaraj SK., Copps K., Validation of a posteriori error estimators by numerical approach, Int. J. Numer. Methods Engrg., Vol. 37, pp. 1073-1123, 1994.
- [BS96] Butlin G. and Stops C., *CAD data repair*, Proceedings 5<sup>th</sup> International Meshing Roundtable, Sandia National Lab., pp. 7-12, 1996.
- [BWS03] Beall M.W., Walsh J., Shephard M.S., *Accessing CAD geometry for mesh generation*, Proceedings of 12<sup>th</sup> International Meshing Roundtable, Sandia National Laboratories, 2003.
- [BZ97] Boroomand B. and Zienkiewicz O.C., *Recovery by Equilibrium in Patches (REP)*, International Journal For Numerical Methods In Engineering, Vol. 40, pp. 137-164, 1997.
- [Cad] Cadfix, <http://www.cadfix.com>
- [Cat] CATIA V5, Dassault Systemes, <http://www.3ds.com/products-solutions/brands/CATIA>.
- [CFL01] Chen K.Z., Feng X.A. and Lu Q.S., *Intelligent dimensioning for mechanical parts based on feature extraction*, Computer Aided Design, Vol. 33(13), pp. 913-1022, 2001.
- [CGK03] Cardone A., Gupta S.K., and Karnik M., *A survey of shape similarity assessment algorithms for product design and manufacturing applications*, ASME Journal of Computation and Information Science in Engineering, Vol. 3, pp. 109-118, 2003.
- [Che06] Cheutet V., *Towards semantic modelling of free-form mechanical products*, PhD Thesis, INPG, Grenoble, 2006.
- [Cim] Cimne, <http://gid.cimne.upc.es/index.html>

- [CLP\*92] Coorevits P., Ladevèze P., Pelle J.P., Rougeot P., Some new applications of a method for the control and optimization of finite element computations, *New Advances in Computational Structural Mechanics*, Elsevier, pp.205-217, 1992.
- [CLP95] Coorevits P., Ladevèze P., Pelle J.P., An automatic procedure with a control of accuracy for finite element analysis in 2D elasticity, *Computer Methods in Applied Mechanics and Engineering*, Vol. 121, pp.91-120, 1995.
- [CL01] Corney J., Lim T., *3D modelling with ACIS*, Saxe-Coburg Publications, ISN 1-874672-14-8, 2001.
- [Cui98] Cuillière J., Automatic triangulation of 3d parametric surfaces, *CAD*, vol. 30, pp. 139-149, 1998.
- [CM99] Cuilliere J. et Maranzana R., *Automatic and a priori refinement of three-dimensional meshes based on feature recognition techniques*, *Advances in Engineering Software*, vol. 30(8), pp. 563-573, 1999.
- [CS04] Choi S.W., Seidel H-P., *Linear one-sided stability of MAT for weakly injective 3D domain*, *Computer-Aided Design*, Vol. 36(2), pp. 95–109, 2004.
- [CTB\*98] Chep A., Tricarico L., Bourdet P., Galantucci L., Design of object-oriented database for the definition of machining operation sequences of 3D workpieces, *Computers Ind. Engng.* Vol. 34(2), pp. 257-279, 1998.
- [Den98] Deneux D., *Introduction to assembly features: an illustrated synthesis methodology*, *Journal of Intelligent Manufacturing*, A. Kusiak (ed.), Vol. 10(1), 1998.
- [DFG98] De Martino T., Falcidieno B., Giannini F., *Integrated feature-based modelling in concurrent engineering*, *Proceedings of TMCE 98 : 2nd International Symposium on Tools and Methods for Concurrent Engineering*, Manchester, UK, I. Horvath, A. Taleb-Bendiab (eds.), pp. 381 - 392, 1998.
- [DKK\*05] Date H., Kanai S., Kishinami T., Nishigaki I., Dohi T., *High-Quality and Property Controlled Finite Element Mesh Generation From Triangular Meshes using the Multiresolution Technique*, Vol. 5(4), pp. 266-276, 2005.
- [DMB\*96] Donaghy R.J., McCune W., Bridgett S.J., Robinson D.J., McKeag R.M., *Dimensional Reduction for Analysis Models*, *Proceedings of 5<sup>th</sup> International Meshing Roundtable*, Sandia National Laboratories, pp. 307–320, 1996.
- [Dri06] Drieux G., *Procédés d'élaboration de maquettes numériques en aval du processus de conception: préparation de maquettes aval pour le développement de produits*, PhD thesis, INPG, Grenoble, 2006.
- [DSG97] Dey S., Shephard M.S., Georges M.K., *Elimination of the adverse effects of small model features by the local modification of automatically generated meshes*, *Engineering with Computers*, pp. 134-152, 1997
- [DT84] Dhatt G. and Touzot G., *The Finite Element Method Displayed*, Wiley, New York, 1984.
- [Eds95] EDS, Parasolid Business Unit, *Overview of Parasolid*, 1995.

- [EWW96] Eshete T., Werner F., Wthrich C.A., *Checking Boundary Non-Manifoldness of Solid Objects for Steel Construction*, 4th International Conference in Central Europe on Computer Graphics and Visualization (WSCG'96), Plzen, Czech, pp. 388-398, 1996.
- [FCF\*08] Foucault G., Cuillière J-C., François V., Léon J-C., Maranzana R., *Adaptation of CAD model topology for finite element analysis*, Computer Aided Design, Vol. 40(2), pp. 176-196, 2008.
- [FG82] Firby P.A. and Gardiner C.F., *Surface topology*, ed. Ellis Horwood limited, 1982.
- [Fin01] Fine L., *Processus et méthodes d'adaptation et d'idéalisation de modèles dédiés à l'analyse de structures mécaniques*, PhD thesis, INPG, Grenoble, 2001.
- [FL05] Fine L., Léon J-C., *A new approach to the preparation of models for F.E. analyses*, Int. Journal of Computer and Applications, Vol. 23(2/3/4), pp. 166-184, 2005.
- [FML04] Foucault G., Marin P., Léon J-C., *Mechanical criteria for the preparation of finite element model*, Proceeding of 13<sup>th</sup> International Meshing Roundtable, Williamsburg, USA, 2004.
- [FML\*06a] Ferrandes R., Marin P.M., Léon J-C., Giannini F., *Preparation of Finite Element Models: The Use of an a Posteriori Mechanical Criterion*, Proc. of Eurographics Italian Chapter, Catania, 2005.
- [FML\*06b] Ferrandes R., Marin P.M., Léon J-C., Giannini F., *Evaluation of simplification details for an adaptive shape modelling of components*, Proc. of 5th Int. Conf. on Engineering Computational Technology, N°2 ISBN 1-905088-11-6, Las Palmas de Gran Canaria, Spain, 2006.
- [FLM\*06] Ferrandes R., Léon J-C., Marin P.M., Giannini F., *An analysis of product development process configurations where an a posteriori FE criterion improves simulation models consistency*, Proc. of ASME CIE Int. Conf., Philadelphia, USA, 2006.
- [FLM\*07] Ferrandes, R., Léon, J-C., Marin, P.M., Giannini, F., Falcidieno, B., *Semantic operators for Handling shape sub-domains in FE model preparation*, Proc. ASME CIE Int. Conf., Las Vegas, USA, 2007.
- [LFG08] Léon, J-C., Ferrandes R., Giannini, F., *Shape processing and reasoning for multiple product views: key issues and contributions to a general framework*, Proc. Proc. of the 9th Biennial ASME ESDA Conference , Haifa, Israel, 2008.
- [Fou07] Foucault G., *Adaptation de modèles CAO paramétrés en vue d'une analyse de comportement mécanique par éléments finis*, PhD thesis, Université du Québec, Montreal, 2007
- [Fra65] Fraeijs de Veubeke B., *Displacement and equilibrium models in the finite element method, Stress Analysis*, Ed. O. C. Zienkiewicz and G. S. Holister, Publ. John Wiley & Sons, Chapter 9, pp. 145-197, 1965.
- [Fra98] François V., *Méthode de maillage et de remaillage automatique appliquée à la modification de modèle dans le contexte de l'ingénierie simultanée*. PhD thesis, Université Henri Poincaré, Nancy, 1998.

- [FRL00] Fine L., Remondini L., Léon J-C., *Automated Generation of FEA models through idealization operators*, International Journal for Numerical Methods in Engineering, Vol. 49(1-2), pp. 83-108, 2000.
- [GKZ\*83] Gago J.P., Kelly D.W., Zienkiewicz O.C., Babuska I., *A posteriori error analysis and adaptive processes in the finite element method: part II – Adaptive mesh refinement*, Int. Journ. Num. Meth. Engng., Vol.19, pp.1621-1656, 1983.
- [Gam] Gambit, from Fluent Inc., <http://www.fluent.com/software/gambit/index.htm>
- [GB04] Grätsch T. and Bathe K-J., *A posteriori error estimation techniques in practical finite element analysis*, Computers & Structures, vol. 83(4-5), pp. 235-265, 2004.
- [GB97] George P.L., Borouchaki H., *Triangulation de Delaunay et maillage*, ed. Hermes, 1997.
- [Geo91] George P.L., *Génération automatique de maillage: applications aux méthodes d'éléments finis*, Collection Recherches en Mathématiques Appliquées, Masson, 1991.
- [GLM\*92] Gastine J.L., Ladevèze P., Marin P., Pelle J.P., *Accuracy and optimal meshes in finite element computation for nearly incompressible materials*, Computer Methods in Applied Mechanics and Engineering, Vol. 94(3), pp. 303-315, 1992.
- [GS07] Gopalakrishnan S.H. and Suresh K., *A formal theory for estimating defeaturing-induced engineering analysis errors*, Computer-Aided Design, Vol. 39(1), pp. 60-68, 2007.
- [HAC03] Hornus S., Angelidis A., Cani M.P., *Implicit modelling using subdivision-curves*, The Visual Computer, Vol. 19(2/3), pp. 99-104, 2003.
- [Ham06] Hamri O., *Method, models and tools for Finite Element model preparation integrated into a product development process*, PhD thesis, INPG, 2006.
- [HC03] Haimes R. and Crawford C., *Unified geometry access for analysis and design*, Proceedings of 12th International Meshing Roundtable, Sandia National Laboratories, pp. 21-31, 2003.
- [HD07] Hui A., De Floriani L., *A two-level topological decomposition for non-manifold simplicial shapes*, Proceedings of the 2007 ACM symposium on Solid and physical, Beijing, China, pp. 355-360, 2007.
- [HLG\*04] Hamri O., Léon J-C., Giannini F., Falcidieno B., *From CAD models to F.E. simulations through a feature-based approach*, Proceedings of ASME DETC Int. Conf., Salt Lake City, Utah, 2004.
- [HLG\*06] Hamri O., Léon J-C., Giannini F., Falcidieno B., Poulat A., Fine L., *Interfacing product views through a mixed shape representation. Part I: Data structures and operators*, Proceedings of Virtual Concept 2006, Playa Del Carmen, Mexico, 2006.
- [HLK06] Hong T., Lee K., Kim S., *Similarity comparison of mechanical parts to reuse existing designs*, Computer-Aided Design, Vol. 38(9), pp. 973-984, 2006.

- [HPR00] Han J.H., Pratt M., Regli W.C., *Manufacturing Feature Recognition from Solid Models: A Status Report*, IEEE Transactions on Robotics and Automation, Vol. 16(6), pp. 782-796, 2000.
- [Ide] SDRC IDEAS Master Series, Structural Dynamics Research Corporation (now UGS), <http://www.ugs.com>.
- [IIS\*01] Inoue K., Itoh T., Shimada A., Furuhata T., Shimada K., *Face Clustering of Large-scale CAD Model for Surface Mesh Generation*, Computer-Aided Design, Vol. 33(3), pp. 251–261, 2001.
- [ILT\*01] Indermitte C., Liebling Th.M., Troyanov M., Cl  men  on H., *Voronoi diagrams on piecewise flat surfaces and an application to biological growth*, Theoretical Computer Science Vol. 263, pp. 263–274, 2001.
- [Imb79] Imbert J., *Analyse des structures par   l  ments finis*, Editions Cepadues, 1979.
- [JC88] Joshi S., Chang T.C., *Graph-based heuristic for recognition of machined features from a 3D solid model*, Computer-Aided Design, Vol. 20(2); pp. 58-66, 1988.
- [JD03] Joshi N. and Dutta D., *Feature simplification techniques for freeform surface models*, Journal of Computing and Information Science in Engineering, vol. 3, 2003.
- [JH02] Jiao X. and Heath M.T., *Feature Detection for Surface Meshes*, Numerical Grid Generation in Computational Field Simulations, The International Society of Grid Generation, pp. 705–714, 2002.
- [JPB95] Jones M.R., Price M.A., Butlin G., *Geometry management Support for Auto-Meshing*, Proceeding of the 4<sup>th</sup> Sandia International Meshing Roundtable Conference, pp.153-164, 1995.
- [JPS93] Jo H.H., Parsei H.R., Sullivan W.G., *Principles of concurrent engineering*, Concurrent engineering, contemporary issues and modern design tools, edited by H.R. Parsei and W.G. Sullivan, Chapman & Hall, 1993.
- [Kel84] Kelly D.W., *The self-equilibration of residuals and complementary a posteriori error estimates in the finite element method*, Int. J. Numer. Methods in Engrg, Vol. 20, pp. 1491-1506, 1984.
- [Kim92] Kim Y.S., *Recognition of form features using convex decomposition*, Computer-Aided Design, Vol. 24(9), pp. 461-76, 1992.
- [KS97] Kurowski P.M., Szabo B.A., *How to find errors in finite-element models*, Machine Design, Edited by Paul Dvorak, 1997.
- [Kus93] Kusiak A., *Concurrent engineering: automation, tools and techniques*, Wiley Inter Science, 1993.
- [Lad75] Ladev  ze P., *Comparaison de mod  les de milieu continu*, PhD Thesis, Universit   P. et M. Curie, Paris, 1975.
- [LAP\*05] Lee K.Y., Armstrong C.G., Price M.A., Lamont J.H., *A small feature suppression/unsuppression system for preparing B-Rep models for analysis*,



- Proceedings of the ACM symposium on solid and physical modelling, Cambridge (Massachusetts), 2005.
- [Lee99] Lee K., *Principles of CAD/CAM/CAE systems*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 1999.
- [Leo91] Léon. J-C., *Modélisation et construction de surfaces pour la CFAO*, Hermes Sciences Publications, 1991.
- [LGT01] Lua Y., Gadha R., Tautges, T.J., *Feature based hex meshing methodology: feature recognition and volume decomposition*, Computer-Aided Design Vol. 33, pp. 3-24, 2001.
- [LLV05] Lesage D., Léon J-C., Véron P., *Discrete curvature approximations and segmentation of polyhedral surfaces*, International Journal of Shape Modelling, Vol. 11(2), pp. 217-252, 2005.
- [LL83] Ladevèze P. and Leguillon D., *Error estimate procedure in the finite element method and applications*, SIAM J. Numer. Anal., Vol. 20, pp. 485-509, 1983.
- [LL02] Li B. and Liu J., *Detail feature recognition and decomposition in solid model*, Computer-Aided Design, Vol. 34, pp. 405-414, 2002.
- [LP84] Ladevèze P., Pelle J.P., Error estimation methods for vibrations of elastic structures, Proc. of Int. Conf. On Accuracy and Adaptive Refinements in Finite Element Computation (ARFEC), Lisbon, pp.217-226, 1984.
- [LR97] Ladevèze P., Rougeot P., *New advances on a posteriori error on constitutive relation in finite element analysis*, Comput. Methods Appl. Mech. Eng., Vol. 150, pp. 239–249, 1997.
- [Man88] Mantyla M., *An introduction to solid modelling*, Computer Science Press, 1988.
- [MCC98] Mobley A.V., Carroll M.P., Canann S.A., *An object oriented approach to geometry defeaturing for Finite Element Meshing*, Proceeding of the 7th International Meshing Roundtable, Dearborn, USA, 1998.
- [Mer98] Mer S., *Les mondes et les outils de la conception. Pour une approche sociotechnique de la conception du produit*, PhD thesis, INPG, Grenoble, 1998.
- [MF05] Marin P.M., Ferrandes R., *Adaptive modelling of CAD model for linear static computation*, Proceedings of ADMOS Conference, Barcelona, 2005.
- [Mor85] Mortenson M.E., *Geometric Modelling*, John Wiley, 1985.
- [MV97] Morris A.J., Vignjevic R., *Consistent finite element structural analysis and error control*, Computer Methods in applied mechanics and engineering, Elsevier Science, vol. 140, pp.87-108, 1997.
- [NBB07] Nyirenda P.J, Bidarra R., Bronsvoort W.F., *A Semantic Blend Feature Definition*, Computer-Aided Design & Applications, Vol. 4(6), pp. 795-806, 2007.

- [NFT\*03] Novotny A.A., Feijóo R.A., Taroco E., Padra C., *Topological-Shape Sensitivity Analysis*, Computer Methods in Applied Mechanics and Engineering, Vol. 192(7), pp. 803-829, 2003.
- [Noe94] Noël F., *Mailleur auto-adaptatif pour des surfaces gauches en vue de la conception intégrée*, PhD thesis, INPG, Grenoble, 1994.
- [ODR\*89] Oden J.T., Demkowicz L., Rachowicz W., Westermann T.A., *Toward a universal h-p adaptive finite element strategy. Part 2: A posteriori error estimation*, Computer Methods in Applied Mechanical Engineering, Vol. 77, pp.113-180, 1989.
- [Pro] ProEngineer, Parametric Technology Corporation, <http://www.ptc.com>.
- [PSB95] Price M., Stops C., Butlin G., *A Medial Object Toolkit for Meshing and Other Applications*, Proceedings of 4<sup>th</sup> International Meshing Roundtable, Sandia National Laboratories, pp. 219–229, 1995.
- [RB03] Razadan A. and Bae M., *A hybrid approach to feature segmentation of triangle meshes*, Computer Aided Design, Vol. 23, pp. 783–789, 2003.
- [RBO02] Ribó R., Bugeda G., Onate E., Some algorithms to correct a geometry in order to create a finite element mesh, Computer-Aided Design, vol. 80, pp. 1399-1408, 2002.
- [Req80] Requicha A.A.G., *Representations of solid objects – theory, methods, and systems*, ACM Computing Surveys, Vol. 12(4), pp. 437-464, 1980.
- [Req99] Requicha, A.G, *Geometric Modelling, A first course*, 1999.
- [Rez96] Rezayat M., *Midsurface abstraction from 3D solid models: general theory and applications*, Computer-Aided Design, Vol. 28(11), pp. 905–15, 1996.
- [RG98] Rosenman M.A. and Gero J.S., *CAD modelling in multidisciplinary design domains*, Lecture Notes in Computer Science, Vol. 1454, pp. 335-347, 1998.
- [RHG01] Ribelles J., Heckbert P.S., Garland M., Stahovich T., Srivastava V., *Finding and removing features from polyhedra*, Proceedings of ASME design engineering technical conference, Pittsburgh (PA, USA), 2001.
- [Rod94] Rodriguez R., *Some remarks on Zienkiewicz-Zhu estimator*, Int. J. Numer. Methods in PDE, Vol. 10, pp. 625-635, 1994.
- [RT95] Reddy J.M., Turkiyyah G.M., *Computation of 3D skeletons using a generalized Delaunay triangulation technique*, Computeir-Aided Design, Vol. 27(9), pp. 677–94, 1995.
- [Rup95] Ruppert J., *A Delaunay Refinement Algorithm for quality 2Dimensional Mesh Generation*, Journal of Algorithms, pp.548-585, 1995.
- [SAR96] Sheehy D.J., Armstrong C.G., Robinson D.J., *Shape description by medial surface reconstruction*, IEEE Trans. Visualization and Computer Graphics, Vol. 2(1), pp. 62–72, 1996.
- [SBC\*00] Sheffer A., Blacker T., Clements J., Bercovier M., *Virtual topology operators for meshing*, Int. J. of Computational Geometry and Applications, Vol. 10(3), pp. 309-331, 2000.

- [SBO98] Shephard M.S., Beall M.W., O'Bara R.M., *Revisiting the elimination of the adverse effects of small features in automatically generated meshes*, Proceedings of 7<sup>th</sup> International Meshing Roundtable, Sandia National Laboratories, SAND98-2250 pp.347-364, 1998.
- [SCH74] Schek H. J., *The force density method for form finding and computation of general networks*, Computers Methods in Applied Mechanics and Engineering, Vol. 3, pp. 115-134, 1974.
- [SG90] Sakurai H., Gossard D.C., *Recognizing shape features in solid models*, IEEE Computer Graphics Applications, Vol. 10(5), pp. 22-32, 1990.
- [Sha88] Shah, J.J., *Feature transformations between application-specific feature spaces*, Computer-Aided Engineering Journal, Vol. 5(6), pp. 247-255, 1988.
- [She96] Shewchuk J.R., *Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator*, First Workshop on Applied Computational Geometry, Philadelphia, Pennsylvania, ACM, pp. 124-133, 1996.
- [She01] Sheffer A., *Model Simplification for Meshing Using Face Clustering*, Computer-Aided Design, Vol. 33, pp. 925-934, 2001.
- [SM95] Shah J.J., Mantyla M., *Parametric and Feature Based CAD/CAM: Concepts, Techniques and Applications*, John Wiley & Sons, Inc., 1995.
- [Sol] SolidWorks, SolidWorks Corporation, <http://www.solidworks.com>.
- [Spa93] Spatial Technology Inc., *ACIS Reference Manual*, Boulder, CO.
- [SSP00] Sakkalis T., Shen G., ätrikalakis N.M., *Representational validity of boundary representation models*, Computer-Aided Design, Vol. 32, pp. 719-726, 2000.
- [Suh90] Suh N.P., *The principles of design*, Oxford University Press, 1990.
- [SY84] Shephard M. and Yerry M., *Automatic mesh generation for three dimensional solids*, International Journal of Numerical Methods in Engineering, vol. 20, pp. 1965-1990, 1984.
- [Sza96] Szabo B.A., *The problem of model selection in numerical simulation*, Advances in Computational Methods for Simulation, B.H.V. Topping, Civil-Comp Press, pp. 9-16, 1996.
- [STEP] *STEP - Standard for the Exchange of Product model data*, International Standard ISO/DIS 10303-214, application protocols, Part 203, 214, integrated resources, Part 41, 1995.
- [SZ99] Sokolowski J., Zochowski A., *On Topological Derivative in Shape Optimization*, SIAM journal on control and optimization, Vol. 37(4), pp. 1251-1272, 1999.
- [Tau01] Tautges T., *Automatic Detail Reduction for Mesh Generation Applications*, Proceedings of 10<sup>th</sup> International Meshing Roundtable, Sandia National Laboratories, pp. 407-418, 2001.

- [TGS07] Turevsky I., Gopalakrishnan S.H., Suresh K., *Generalization of topological sensitivity and its application to defeaturing*, Proc. of ASME CIE Int. Conf., Las Vegas, USA, 2007.
- [Tim30] Timoshenko S., *Strength of Materials, Part I-II*, D. Van Nostrand Company, 1<sup>st</sup> Edition, 1930.
- [TJ03] Tate S.J., Jared G.E.M., *Recognising symmetry in solid models*, Computer-Aided Design, Vol. 35(7), pp.673-92, 2003.
- [Tro99] Troussier N., *Contribution à l'intégration du calcul mécanique dans la conception de produits techniques: proposition méthodologique pour l'utilisation et la réutilisation*, PhD thesis, Université Joseph Fourier, Grenoble, 1999
- [Ver99] Verfurth R., *A review of a posteriori error estimation techniques for elasticity problems*, Computer Methods in Applied Mechanical Engineering, Vol. 176, pp. 419-440, 1999.
- [VL98] Véron P., Léon J-C., *Shape Preserving Polyhedral Simplification with Bounded Error*, Computers & Graphics, Vol 22(5), pp. 565-585, 1998.
- [VMB98] Vignjevic R., Morris A.J., Belagundu A.D., *Towards high fidelity finite analysis*, Advances in Engineering Software, Elsevier Science, Civil-Comp Press, Edinburgh, vol. 9(79), pp. 655-665, 1998.
- [VSK01] Venkataraman S., Sohoni M., Kulkarni V., *A graph-based framework for feature recognition*, Proceedings of 6<sup>th</sup> ACM symposium on solid modelling and applications, ANN Arbor, MI, pp. 194-205, 2001.
- [VS02] Venkataraman S. and Sohoni M., *Reconstruction of feature volumes and feature suppression*, Proceedings of the 7<sup>th</sup> ACM symposium on Solid modeling and applications, ACM Press, pp. 60-71, 2002.
- [WA93] Wiberg N.E. and Abdulwahab F., *Patch recovery based on superconvergent derivatives and equilibrium*, Int. J. Numer. Methods Engrg., Vol. 36, pp. 2703-2724, 1993.
- [Wei88] Weiler K.J., *The radial-edge structure: A topological representation for non-manifold geometric boundary representations*, Geometric modelling for CAD applications, North Holland, pp. 3-36, 1988.
- [Zie77] Zienkiewicz O.C., *The Finite Element Method*, Mac Graw Hill, 1977.
- [ZM02] Zhu H. and Menq C.H., *B-REP model simplification by automatic fillet/round suppressing for efficient automatic feature recognition*, Computer-Aided Design, Vol. 34(2), pp. 109-123, 2002.
- [ZT00] Zienkiewicz O.C. and Taylor R.L., *Errors, recovery processes and error estimates*, The finite element method, Vol. 1(14), Butterworth-Heinemann, 2000.
- [ZZ87] Zienkiewicz O.C. and Zhu J.Z., *A simple error estimator and adaptive procedure for practical engineering analysis*, Int. J. Numer. Methods in Engrg, Vol. 24, pp. 337-357, 1987.

- [ZZ92] Zienkiewicz O.C. and Zhu J.Z., *Superconvergent patch recovery (SPR) and adaptive finite element refinement*, Computer Methods in Applied Mechanical Engineering, Vol. 101, pp. 207-224, 1992.
- [ZZ95] Zhang Z. and Zhu J.Z., *Analysis of the superconvergent patch recovery technique and a posteriori error estimator in the finite element method (I)*, Computer Methods in Applied Mechanical Engineering, Vol. 123, pp. 173-187, 1995.
- [ZZ98] Zhang Z. and Zhu J.Z., *Analysis of the superconvergent patch recovery technique and a posteriori error estimator in the finite element method (II)*, Computer Methods in Applied Mechanical Engineering, vol. 163, pp. 159-170, 1998.