Integration of CAD, FEA and Topology Optimization through a Unified Topological Model

Jean-Christophe Cuillière[1] and Vincent Francois[2]

[1]Université du Québec à Trois-Rivières, Canada, cuillier@uqtr.ca
[2]Université du Québec à Trois-Rivières, Canada, francois@uqtr.ca

ABSTRACT

We have been involved in research work in the field of finite element analysis (FEA) integration with computer aided design (CAD) for several years and have developed several concepts and tools that have aroused interest and shown efficiency. In the meantime, both the evolution of our research developments (on topics like geometry comparison, geometry reconstruction and simplification, mixed-dimensional analysis and topology optimization) and the evolution of CAD systems and CAD kernels made us reconsider our database organization. This led to the design of an original development environment and database organization referred to as the Unified Topological Model (UTM). The main interests of this new CAD/FEA database organization is its ability to tackle multi-platform CAD/FEA integration (handling geometries coming from different CAD kernels), mixed-dimensional modeling and analysis (3D solid geometry mixed and integrated with surface geometry and curvilinear geometry) and topology optimization (TO) procedures. The paper presents the structure of this new research development environment and the original concepts underlying it. The UTM environment is strongly designed around object-oriented computer programming concepts and it is focused towards generality, modularity and ability to evolve. The paper also briefly presents some of the most important features and algorithms that have been integrated, at this point, into the UTM environment.

Keywords: CAD, FEA, integration, topology optimization, mixed-dimensional.

1. INTRODUCTION

Despite the recent appearance of new analysis methods such as isogeometric analysis [17] (IGA), finite element analysis (FEA) is still intensively and increasingly used in the product development process. One of the reasons that explain this success is the fact that FEA is always easier to use and by the way, much more productive. This evolution can be attributed to a huge research effort that led to many advances at different levels of its implementation and to the ongoing enhancement of computer systems capabilities. Among these advances, the integration of FEA with computer aided design (CAD) systems represents a cornerstone. It has indeed made possible the achievement of major improvements in productivity by significantly contributing to the reduction of time required to build a FEA model from a detailed product definition CAD model.

In fact, although FEA data input may seem relatively simple with the use of integrated systems, it still remains tedious to perform for complex models and, in many cases, CAD model preparation for FEA still represents the most significant effort in time along the whole FEA process. Indeed, managing complex models usually requires using systems that are only dedicated to FEA, which means not integrated with CAD, because they allow a total control when building the FEA model. However, even if these platforms are very flexible, as they are not integrated with CAD, its use is tedious and sometimes very complex. Consequently, finding the best of both worlds consists of introducing flexibility inside integrated systems. A significant effort has been put towards this direction for the last ten years, with the objective of improving the functionality of CAD-FEA integrated systems in easily creating versatile and complex models but this effort remains an ongoing progress and many further advances are still to come. Moreover, the recent development of IGA based approaches [1,17], which basically consists of using the same shape functions for geometry and analysis, even if very promising due to the fact that analysis is performed on the "exact geometry", still faces the issue of isogeometric model generation for 3D domains with arbitrary geometry and topology.

The research of our team falls within this general context. We have developed several tools that are integrated with CAD with the objective of deriving FEA models from complex CAD models as easily, efficiently and rapidly as possible [8–11,13]. This integration is performed by using specific concepts and data structures, referred to as the Unified Topological Model (UTM) and the objective of this paper is presenting the concepts and data structures that underlie the UTM along with its main features. This paper is organized as follows. In Sec. 2 we introduce CAD/FEA integration topics underlying this work. After a presentation of possible CAD/FEA integration approaches, Sec. 3 outlines the purpose of developing our Unified Topological Model (UTM). Basic concepts upon which our UTM is based are then introduced in Sec. 4 and its classes and methods are detailed in Sec 5. The input/output format associated with the UTM is briefly presented in Sec. 6 and finally, several illustrative results obtained with the UTM are presented in Sec. 7. The paper ends with a short conclusion about potential enhancements that can be foreseen for the UTM.

2. RESEARCH CONTEXT

2.1. CAD/FEA Integration

Our research work is intended towards the development of new concepts and methods that are focused on improving CAD/FEA integration. The objective is increasing in how fast a relevant and accurate FEA model can be obtained from a CAD detailed product model and how rapidly design changes made in CAD models can be derived to FEA models involved, and this at any stage of the product development process with CAD. More generally, our work is focused on minimizing time and effort implied in product development and product adaptation with CAD and FEA. This integration between CAD and FEA technologies is closely and mainly related to CAD-FEA data representation, integration and management within the same environment. CAD-FEA data integration is not a new concept in itself and many authors have proposed interesting ideas and structures towards this objective. The main idea is gathering and integrating within a unique data structure all the information that is necessary for describing a part as a solid model and all the information that is necessary for performing efficient and accurate FEA on it. Then the idea is setting up tools that automatically maintain the consistency of this integrated data structure throughout subsequent modifications of the design. Even if a lot of achievements have been made for the last 15 years, there is still space for improvement and the rapid evolution of CAD-FEA and topology optimization (TO) technologies makes that new requirements and new constraints arise.

For various reasons, the first constraint in the design of an integrated CAD-FEA database is its necessary ability to support multi CAD platforms. In fact, commercial CAD systems evolve so quickly that, in the long run, it is absolutely necessary for the kernel of our database to be independent from these commercial CAD systems. We will explain in what way the organization of classes and methods in our research development platform prevents our work from being affected by a change in the CAD system we use and how it avoids rewriting a great part of the code. One of the first developments that led us towards developing a new CAD-FEA data structure is automatic mesh pre-optimization (pre stands for before any FEA) [11]. This concept consists of preparing a mesh for FEA from a solid CAD model by automatically including analysis constraints. Overall, these analysis constraints are likely to be derived from the geometric model shapes and features, loads and boundary conditions (BCs) applied, material distribution and analysis objectives. Basically, automatic mesh pre-optimization aims at providing the FEA process, before any FEA, with a mesh that is intended to obtain more accurate results at a lower analysis cost. The practical implementation of the pre-optimization concept required major improvements in automatic mesh generation processes that we have achieved by improving advancing front mesh generation techniques [7,13,14].

Automatic remeshing has also been one of our major developments. It consists of automatically updating an existing mesh when a change is made in the CAD model [10]. In fact, many iterations are typically performed throughout the design process of any product and consequently, CAD models vary continuously. For complex parts, automatic remeshing is extremely powerful (especially when mesh element sizes have been optimized and when geometric topology remains the same). Automatic remeshing requires an integrated CAD/FEA database which gathers all the information necessary to FE analysis and which maintains its integrity throughout the whole design process. It also requires identifying similarities and differences between CAD models in order to be able to retrieve mesh elements [10].

Automatic CAD reconstruction is another development for which CAD/FEA database organization has to be reconsidered [18]. This concept consists of automatically deriving a CAD solid model from FEA results (namely a deformed mesh). This process is particularly useful in the case of elastoplastic FEA such as in the case of the simulation of forming processes. It is also useful when trying to identify interferences in an assembly that are due to the deformation of some parts. Thus, the design of an efficient and integrated CAD/FEA database requires the ability to introduce several deformed configurations for a given part. As mentioned above, it also requires that this introduction is made with respect to the whole model's integrity throughout the design process.

Mixed-dimensional analysis also implies that CAD/FEA database organization should be deeply revisited [8]. Mixed-dimensional analysis is a widely used concept in FEA, which consists in mixing different types of elements (typically solid elements mixed with shell and beam elements) in order to reduce significantly the degrees of freedom for a given problem. This approach is useful and often unavoidable in the analysis of complex systems but, when using commercial FE packages, the lack of automation in the process makes it quite tedious. Consequently, we are presently involved in a research project aiming at the complete automation of the process. The process starts from a CAD model mixing 3D curves, shells and solid volumes on which boundary conditions are directly applied. From this mixed CAD model, a mixed mesh is automatically generated (using specific connection patterns for the transfer of bending and torsion moments). As well as for previous developments mentioned above, the introduction of mixed-dimensional analysis in the design process requires major enhancements in an integrated CAD/FEA database, especially for the support of mixed-dimensional CAD entities.

One of the major problems inherent to the preparation of FEA models from CAD models is the fact that product definition CAD models feature many shape details that are irrelevant for FEA and that in fact contribute to over constrain mesh generation. Moreover, the structure of their Boundary Representation (BREP) may contain tiny edges and small or narrow faces that will also contribute to over constrain mesh generation. Such configurations are both likely to be at the origin of either poorly-shaped elements and/or over-densified meshes, not only increasing the analysis time, but also eventually producing poor simulation results. Using geometry simplification methods and virtual topology concepts [12,20,23] has proven to be very promising but it also has shown that it requires significant improvements and enhancements in CAD/FEA integrated data structures.

For the last ten years, a growing interest and research effort has been put on *topology optimization (TO) methods* [3,6,25]. These methods might be among the most promising tools for the future of product development and optimization with CAD. These methods could even contribute to establish a new paradigm in the way we see the design activity and in the way we are likely to build and use CAD systems in the future. These methods are based on applying iterative finite element analyses to automate the creation and optimization of parts, assemblies and structures. One of the key aspects in bringing TO methods to maturity is its integration with CAD. Towards this objective, we have introduced the concept of *design and non-design geometry* into the UTM, which is one of its last and most promising enhancements.

Consequently, these constraints led us to the design of a new model for CAD/FEA integration. The design of this Unified Topological Model has mainly been achieved by extending Boundary Representation (BREP) concepts on the one hand, and by applying object oriented (OO) principles (particularly polymorphism and encapsulation). The following section briefly introduces BREP principles in order set up the context towards the UTM.

2.2. Boundary Representation (BREP)

Our research work is mainly based on using BREP (Boundary REPresentation) models [19] and most commercial CAD systems are also based on this type of solid modeling structures. One of the key aspects in integrating FEA with CAD consists of linking mesh entities to geometric entities and this is practically achieved by linking these mesh entities (typically nodes, finite elements and subsets of finite elements) with features in the BREP's topological structure (typically vertices, edges, and faces). BREP modeling consists of describing the boundaries of a part while easily differentiating the inside from the outside of the model. However, this model is quite different from a simple surface model. In fact, in addition to describing the entire boundary geometrically, it also holds a topological description of the boundary, which ensures the integrity of the entire data set constituting this model (see Fig. 1). The BREP model describes the boundaries as a juxtaposition of several oriented faces. Each body is composed of one or more faces. Each face is composed of an underlying surface (an entity which describes the geometry of the face) and is bounded by at least one closed

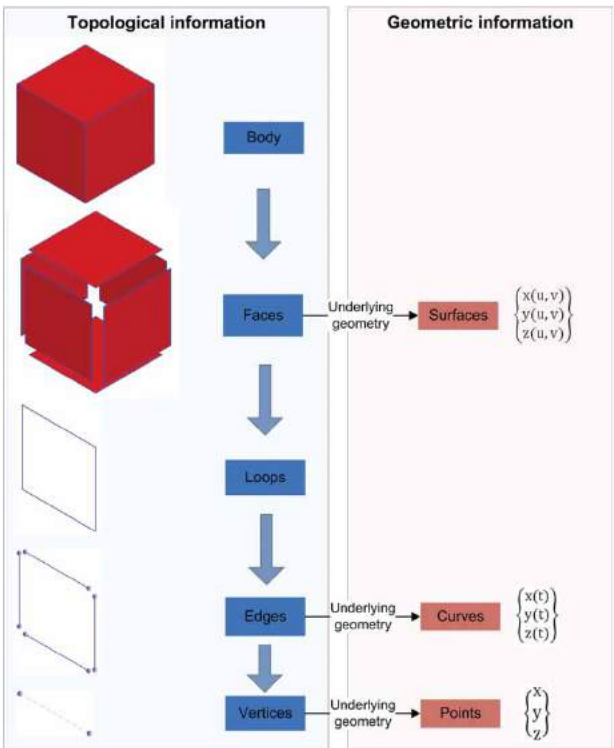

Fig. 1: Topological and geometric entities in a BREP.

and oriented loop (with the exception of specific cases of the sphere and torus). Each contour is formed by a set of edges. Each edge is formed by an underlying curve (entity which describes the geometry of the edge) and is bounded by two vertices and sometimes only one in specific cases. Each vertex is associated with a point that describes its geometry. The integrity of this description is ensured by links between all topological entities and there is no redundancy in the data structure. For example, two neighboring faces are connected by the same edge. A link in the structure must allow one to know that this common edge belongs to the definition of both of these two faces. Most commercial CAD applications make use of this data structure in order to describe solid parts. Several standards have been used over the years in order save the BREP data structure to a file and by the way to allow solid geometry transfer between different CAD systems. We faced various problems when developing mesh generation concepts and algorithms for 3D parts due to the weakness of our BREP based data structures. Our efforts towards overcoming these problems have led to the development of a Unified Topology Model (UTM).

3. LINK WITH PREVIOUS WORK AND ENHANCEMENTS

As for other work in the field of CAD-FEA integration [2,4,16,21,22,24], the first idea on which the UTM is based, is to group under the same data structure, both CAD and FEA entities. As mentioned above, this data structure is built through the extension of BREP concepts. The extension of BREP concepts to the needs of our research as mentioned in section 2 refers to BREP enhancements that are necessary for the support of mixed-dimensional CAD entities (for mixed-dimensional analysis as described above). In fact, the originality of the work presented here mostly follows from this extension and related enhancements of previous work on the subject. This extension could have been tackled using the three following approaches:

- Creating our own extended BREP data structure that is as close as possible to existing standards (IGES, STEP).
- Extending an existing BREP data structure.
- Considering a mixed solution by using the advantages of each of the two previous solutions.

These three solutions have already been investigated by previous research work on the subject [2,4,16,21,22,24]. When trying to integrate CAD and FEA, the first idea that comes to mind is using a structure based on a standard format. Theoretically, both IGES and STEP standards could have been used as they support solid geometry and finite element entities. Practically speaking, STEP is much more promising because it covers a broad range of application fields and life-cycle phases.

Unfortunately, there is a huge difference between the STEP standard itself and the way it is implemented in commercial CAD/FEA systems, if it is. Consequently, data transfer between various commercial CAD/FEA systems through STEP in a unique and universal format is not practically efficient, even if it can be used quite successfully for specific needs [5]. Beall [2] presents the three solutions mentioned above and illustrates them with specific examples. He concludes that the mixed solution is the most efficient, especially for flexibility needs. In [2], the mixed solution is illustrated using the Common Geometry Module library developed by Tautges [22]. This library, aimed at improving flexibility and at being able to adapt to various existing commercial CAD kernels, has indeed been designed starting from the same thoughts than our UTM. It also consists in the definition of a generic BREP model, accessing geometric data in order to implement several algorithms handling non-manifold models as well as virtual topology, parallel computing and numerical analysis. Our experience in the development of the UTM led us to the same conclusions with regard to the choice of the mixed solution: creating our own extended BREP data structure could be an excellent and efficient solution but there are already a large number of BREP implementations on the market and the mathematical background necessary to the model calculations is important and already quite optimal in existing BREP implementations. Engaging in this path seems like a total loss. Extending an existing BREP data structure seems the instinctive solution but working with several different models requires data translation. With regard to geometric data, modeling complex free form surfaces, for example, is specific to each system and translation often results in loss of information. Consequently, the design of the UTM lies between these two approaches, which has shown to be an excellent compromise.

We have already mentioned that the UTM is based on similar principles than those put forward by Tautges [22]. However, the UTM has to be considered as an enhancement of previous efforts in this direction since its structure is genuinely aimed at supporting many different views of a given geometry in the context of various engineering applications along the product design process with CAD. The key features of the UTM, if compared with similar work on the subject are:

- The UTM handles a generic BREP structure but also allows access to other features of the source CAD kernel such as the CAD feature tree information. This is mandatory for research developments taking into account the design intent inherent to a CAD model in a specific application field (for instance automatic mesh density pre-optimization).

- In the UTM, BREP basic principles have been extended in order to be able to introduce beams and shells. Consequently, this allows modeling structures that mix solid, shell and beam geometry through a single data structure. This type of model can then be used for mixed-dimensional FEA [8].
- In the UTM, geometric processing can always be based on exact (genuine) geometry instead of approximated or tessellated geometry (a triangulation for instance) like in [16].
- Moreover, the UTM also allows processing that is based on tessellated geometry because its structure both features an exact representation of geometry and a tessellated representation of geometry. The fact that these two representations are closely integrated makes that at any time, both representations can be used towards the most accurate result at the lowest computational cost.
- Introducing in the UTM a link with a new commercial CAD system is extremely simple and straightforward and by the way the necessary programming effort implied is minimized.
- The UTM features automatic vector based representation capabilities [10]. This type of 3D representations aims at representing a 3D geometric model using sets of vectors that can be used for various geometry processing purposes such as geometry comparisons.
- Virtual topology is implicit in the UTM structure. This means that the impact of virtual topology operators on a model (for FEA geometric simplification needs) only comes in evidence at the meshing step. In fact, virtual topology is practically taken into account by an automatic multi-edges and multi-surfaces advancing front mesh generation system developed by our research team.
- The UTM library aims at being used by various developers (mostly mechanical engineers) after a minimum training effort.

4. BASIC CONCEPTS UNDERLYING THE UTM

The content of an extended BREP data structure can be classified with regard to two types of information: topological information and geometric information. The BREP topological information is related to the definition of data arrangement whereas the BREP geometric information is related to its mathematical definition. Thus, the integration of FEA in the CAD process consists of linking CAD and FEA entities among themselves. To achieve these links, only the BREP topological information has to be considered. Hence, the compromise between the two approaches mentioned above consists of creating a new topology structure that meets the ISO10303 standard in the best possible way (so as to work efficiently on our research) and to create a link toward the geometric information of commercial models. As illustrated in Fig. 2, creating the UTM topology is first achieved by translating an external topology. After this, the link between this external topology and the UTM topology is kept in the UTM data structure. For exact geometry (as opposed to tessellated geometry), the UTM does not explicitly feature any geometric data. Geometric data is obtained in an original and powerful way by encapsulating external geometric functions. Thus, the UTM exact geometric information is derived from the geometric (mathematical) functions of a third party model. For example, solving the reverse problem on a curve or a surface (calculating parameters from Cartesian coordinates of a point lying on a curve or a surface) is made straightforward by encapsulating the appropriate external third party function. Of course, as mentioned below, the drawback of this encapsulation approach, if applied without any adaptation , is a potential performance loss (such as if a large number of separate calls to the external utility for the resolution of the inverse problem have to be made). Nevertheless, if necessary, potential adaptations of the approach can be made, such as previously solving the inverse problem for a discrete set of points and then, solving the inverse problem for any location using an interpolation of these discrete values. This data arrangement concept has been named the Unified Topology Model because it allows for the abstraction of any external topology that supports mixed-dimensional modelling. There are many obvious advantages in the use of these concepts such as the ease conferred in adapting our work on automatic mesh generation to any commercial CAD system on the market. Nevertheless, this generalization has also obviously not been achieved without cost and, as it could be easily predicted, the drawback, when implementing these concepts, is algorithmic performance losses from the perspectives of CPU time and memory requirements. Overall, these CPU and memory performance losses can be seen as anecdotal with regards to the most recent computer capability.

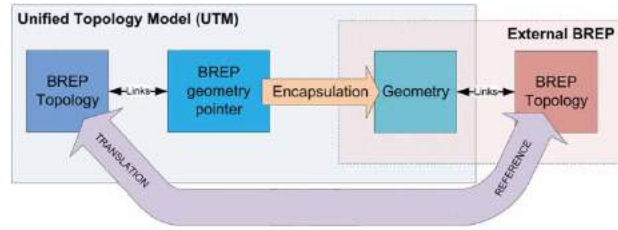

Fig. 2: Basic principles underlying the UTM.

5. PRESENTATION OF THE UTM CLASSES AND METHODS

5.1. General and Common Features of the UTM

By its object-oriented design, the UTM library attempts to extract the maximum functionality from its classes in order to standardize some of the basic classes. This primarily helps to describe a certain number of characteristics, which, unless an exception applies, will be common to all the UTM objects. For example, practically all of the UTM objects bear an identifier (long integer type) that can be used by processes to find a given UTM object or to establish links between UTM objects. Another fundamental characteristic of these objects is that each object knows how to print itself (according to the format considered) and therefore, can be saved on a disc in order to be reused later. This rereading from a file can obviously be done through the same program, or with another program based on the UTM library. The file constitutes a privileged mode of communication and creation of the Unified Topology Model objects. On the other hand, given that the Unified Topology Model is a library of classes to which a programmer may add functionality, the classes are generally designed to be used as base classes in a derivation or encapsulation graph, which is likely to be specific to each application.

5.2. Classification of Entities in the UTM

In order to build our UTM data structure (Fig. 3) we have gathered all data and methods and structured it by considering the following:

- Entities forming the BREP model comprising the following subcategories :
  - Topological entities.
  - Co-topological entities.
  - Geometric entity pointer towards any commercial or standard BREP model.
- FEA entities and physical properties.
- Algorithms.
- An input/output file format and library class viewer.

5.3. Enhancement of the BREP Structure

As mentioned above, the information that is specific to the BREP model is divided into three subcategories. The first subcategory of information found

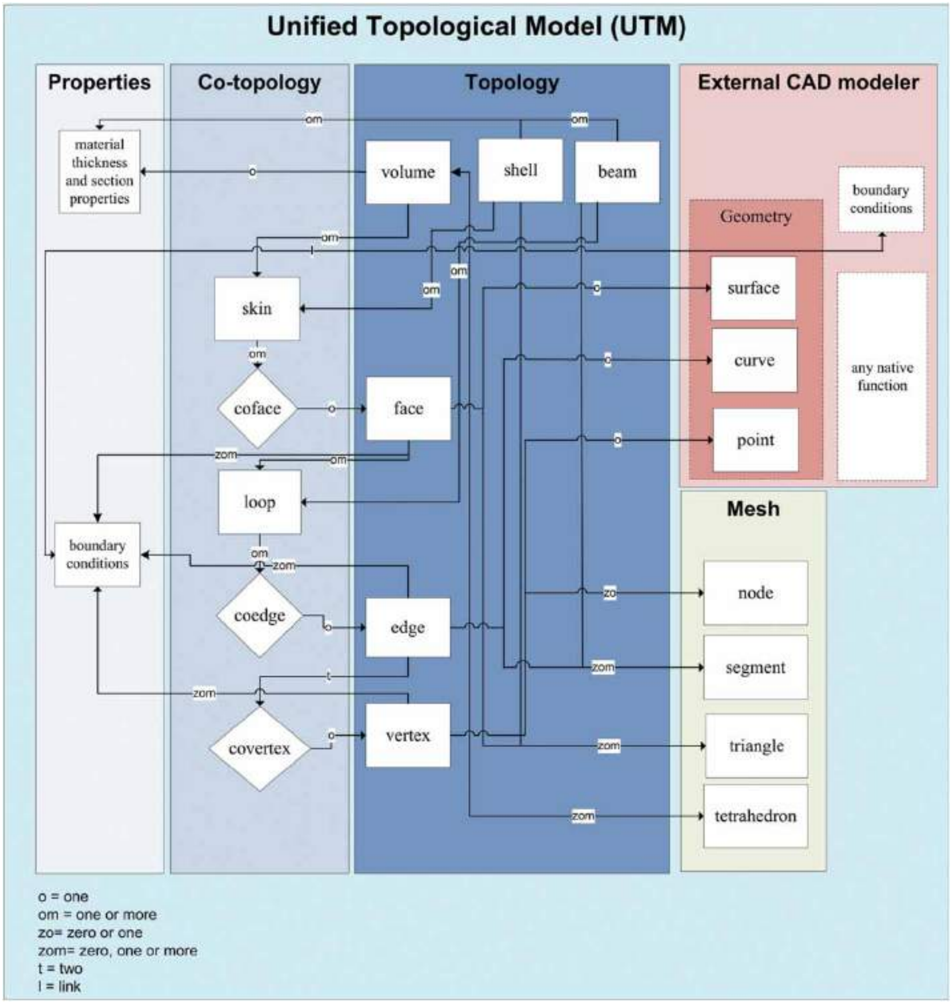

Fig. 3: General architecture of the UTM.

in the model is topology. It is fully consistent with the generic definition of BREP models, as described in section 2.2. This topological information consists of describing an arrangement of the different entities constituting the boundary (or skin) of a 3D part. However, in order to fully describe this arrangement, links must be established between various BREP entities. Also, eliminating redundancy among these entities guarantees the model's integrity.

Also, mixed-dimensional entities have to be introduced in order to complete the design of the extended BREP structure. This is essentially achieved by introducing new types of roots at the head of the classical BREP data structure. Typically, a BREP data structure only features a single type of root which is the body. The extension to mixed-dimensional models consists in adding SHELLS and BEAMS as new types of roots. Then, a SHELL is composed of a BREP SKIN and its underlying geometric and topological entities whereas a BEAM is composed of a BREP LOOP and its underlying geometric and topological entities as illustrated in Fig. 3. Consequently, in opposition to the classical BREP structure of a solid body, this new structure features open SKINS and LOOPS when representing shells and beams.

Another point of interest is that the elimination of redundancy in the BREP's topological definition is mainly achieved through the use co-topology. A co-topological entity (for example a co-edge) is defined as a link with a topological entity (for example an associated edge) along with a Boolean representing its orientation with respect to the associated topological entity (forward or backward with respect to the edge's orientation). By doing this, references can only be made by using co-topological entities and correspondingly, a given topological entity only features once in the BREP structure. This principle also allows us to easily take into account non manifold geometries (we make clear that we refer here to a definition of a manifold part as a part in which each of its edges is always shared by two faces). In the example shown below (see Fig. 4) the part features a straight edge that is shared by four faces, which implies that the object does not conform to the Euler-Poincaré formula. Using co-topological data, manifold and non-manifold parts are treated according to the same method and the non-manifold definition of edges becomes: each edge is shared by an even number of faces through co-edges (co-topological entities associated with an edge). In this case, the BREP edge is shared by 4 BREP faces through 4 BREP co-edges.

In existing commercial BREP models, this co-topology concept is often restricted to the edge's level only. In our work, the co-topology concept has been spread to all levels: vertices, edges and faces. In the case of vertices, there is not much interest, apart from the fact that it standardizes the structure for the three levels of topology (vertices, edges and faces). However, the generalization of BREP co-topology to faces is essential for the study of multi-body parts.

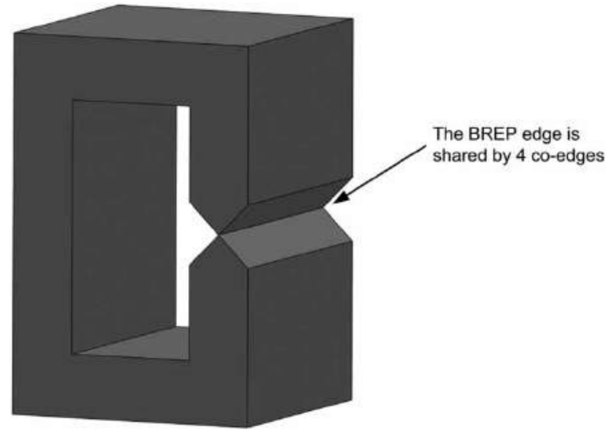

Fig. 4: A non-manifold configuration.

It is notably the case in the modeling of a solid 3D part composed of several different materials. In this case, the material discontinuity is modeled using an interior boundary. The introduction of the BREP co-face (see Fig. 5) allows us to model this discontinuity in a concise and consistent way. In this case, at the interface between two BREP skins, a BREP face is shared by two BREP co-faces. Our UTM structure supports these configurations even if current commercial CAD systems usually don't handle it. As mentioned above, one of the most original features of our UTM structure resides in the way geometry (typically the mathematical representation of curves and surfaces) is handled in the UTM's BREP structure. In fact, there is no geometric information in itself in the UTM structure. The geometry is only represented as references to an external BREP structure (through encapsulation of commercial CAD systems mathematical functions related to the parametric definition of curves and surfaces). The most important benefit of this approach is that it allows taking advantage of the robustness of commercial models in handling complex curves and surfaces. Consequently, there is no need to rewrite any function for managing the curves and surfaces parametric definition. The definition of references and the abstraction of a BREP model in our topology are achieved with the aid of a class (the importation class) and by encapsulating point, curve and surface classes (*Point_ext, Curve_ext* and *Surface_ext* in Fig. 6) of the external BREP. Practically, this can be achieved either by directly accessing each external BREP kernel, which means having a third party partnership agreement with each CAD software company, or by using CAD systems Application Programming Interfaces (API). Our UTM is based on the latter solution (using APIs). This concept is very powerful and it greatly simplifies interfacing the UTM with any new type of external BREP model. Indeed, in our previous developments, interfacing with a new type of BREP model had a major impact on our computer code as it required modifying and/or

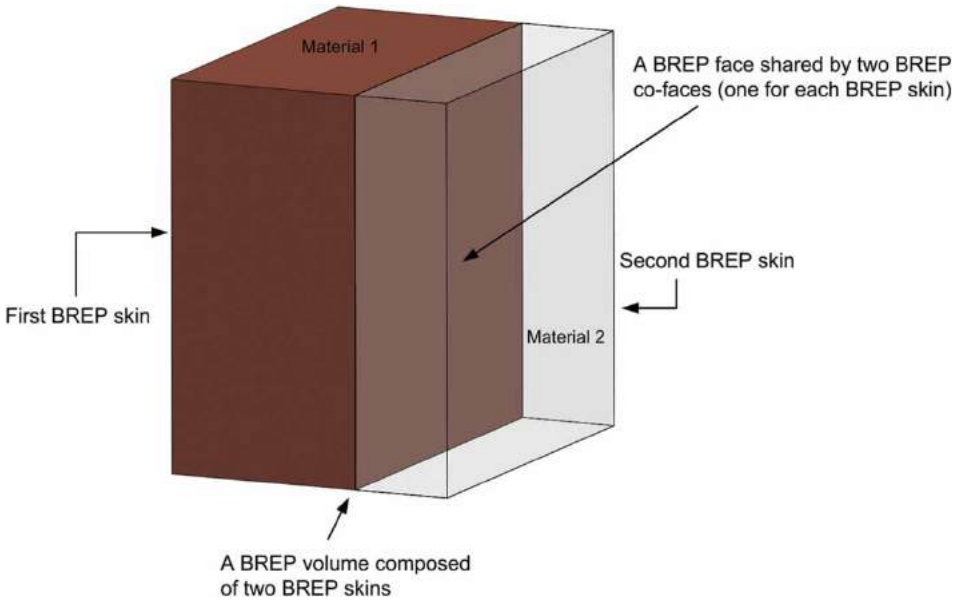

Fig. 5: Definition of a BREP co-face.

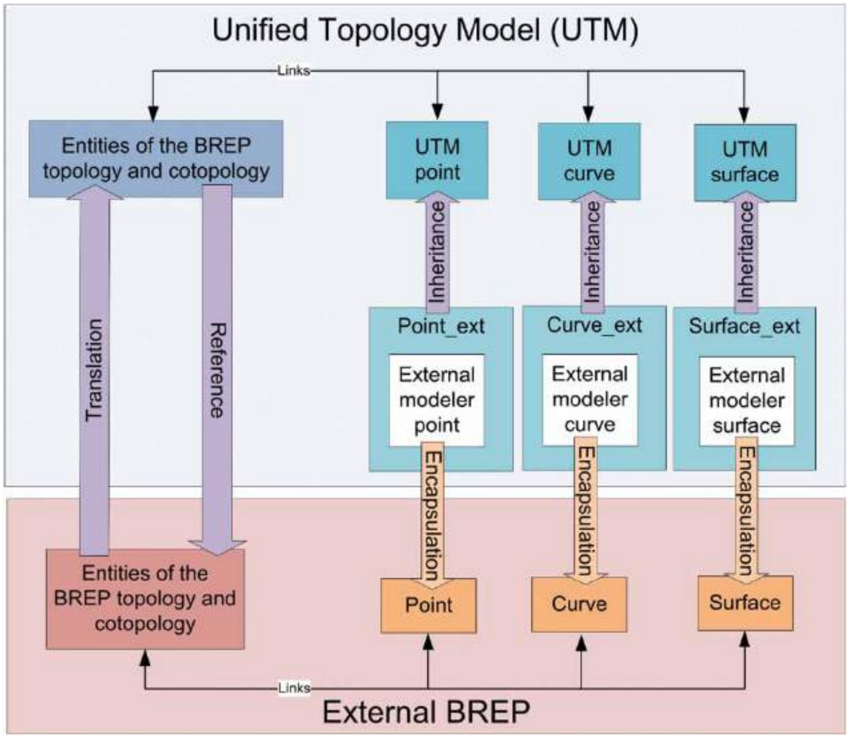

Fig. 6: The external BREP importation class.

rewriting a huge number of lines. Using this encapsulation based approach makes that interfacing the UTM with a new type of external BREP model only requires building four C++ classes: one C++ class to encapsulate the BREP point, one to encapsulate the BREP curve, one to encapsulate the BREP surface and one to translate the original topology into the UTM structure.

5.4. Automatic Mesh Generation

For the last twenty years, an important research effort has been put into the development of efficient, robust and adaptive algorithms for the automatic generation of unstructured grids inside or around complex 3D shapes. On top of various techniques that have been introduced [15], Delaunay based and advancing front based algorithms have led to the design of reliable,

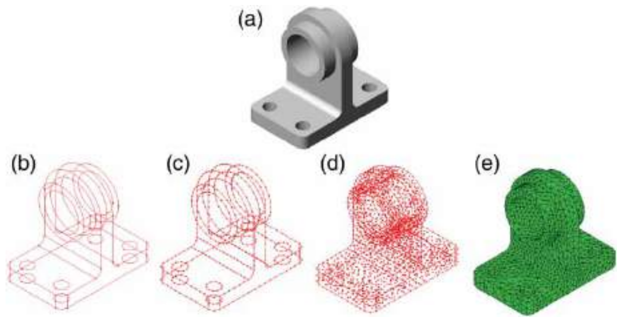

Fig. 7: Hierarchical discretization a) A BREP solid model. b) Nodes on BREP vertices. c) Nodes and segments on BREP edges. d) Triangulation of BREP faces. e) Tetrahedral mesh (with nodes shown only).

adaptive and fully automatic grid generators for triangular and tetrahedral meshes. We have been involved in research work about automatic and adaptive mesh generation for years [7,13,14] which resulted in grid generators introduced in the UTM. Thus, grid generators introduced in the UTM are based on advancing front techniques with the ability to respect various and steep density adaptation constraints. Also, with regard to the UTM, we must underline that the mesh generation process of a 3D solid object, as illustrated in Fig. 7, is performed following steps (hierarchical discretization) that are closely and directly related to the B-Rep data structure (see Fig. 1). Thus, at first, a set of nodes is generated on the B-Rep vertices. Intermediate nodes are then intercalated along the B-Rep edges, eventually with respect to an imposed nodal spacing function (adaptive mesh generation). The next step consists in extracting from the former results and for each of the B-Rep faces, the discretization of its bounding loops. This initiates a triangulation process (based on an adaptive advancing front scheme) which leads, at last, to the object's boundary discretization. The whole process ends up with the generation of tetrahedrons inside of the solid domain through a 3D adaptive advancing front scheme, initiated from the previous triangulation. Consequently, as described in the next paragraph, the integration of finite element entities (nodes, segments, triangles and tetrahedrons) in the UTM data structure is made quite easy and natural. The optimization of the mesh with respect to different quality measures can also be performed at different stages of the whole process (not applied in the mesh shown in Fig. 7). It is also worth noting that, beside general purpose mesh generation tools, many mesh generation and adaptation tools have been developed for very specific needs such as mesh pre-optimization [11], geometry de-featuring and virtual topology [12] and integration of topology optimization methods [9].

5.5. FEA Entities and Physical Properties

The same hierarchical structure is used when considering FEA entities (typically nodes and elements) in our UTM. At present, our UTM considers nodes, linear segments (beam elements), linear triangles (plates, shells) and linear tetrahedrons (volume elements) as FEA entities. However, any other geometric type of FE element (such as quadrangles, hexahedrons and higher order elements) can easily be added if needed. The cornerstone at this stage is the compulsory link between FEA entities and BREP topological entities. It has indeed been proven that fulfilling the actual integration between FEA and CAD information requires a bi-directional link. Therefore, on the one hand, a node is connected to a vertex (or an edge, or a face, or a body), a segment to an edge (or a face, or a body), a triangle to a face (or a body), a tetrahedron to a body. On the other hand, a vertex bears a node, an edge, a list of segments, a face, a list of triangles and a body, a list of tetrahedrons. Additional information that is necessary to FEA (typically material properties and boundary conditions) is also integrated.

5.6. Algorithms

As mentioned above, the UTM design itself is closely object-oriented. Thus, the algorithms are found at the core of all the classes. Algorithms which have been integrated, at this point, into UTM are:

- A priori density map calculations [11].
- Automatic and adaptive mesh generation along curvilinear geometry, over surfaces and inside 3D solid volumes [7,13,14].
- Automatic remeshing in the case of geometric and/or topological modifications in the CAD model [10].
- Automatic CAD reconstruction [18].
- Mixed-dimensional analysis [8].
- Automatic geometry simplification, de-featuring and virtual topology [12,13].
- Automatic comparison between CAD models [10].
- Integration of topology optimization methods [9].

5.7. Input/Output File Format

For practical reasons (typically for CAD-FEA data exchange purposes in the context of our research work), the UTM data structure can be printed in a file using a specific format. This Input/Output file format is closely related to the data structure presented in Fig. 3. It is specific to the UTM and does not aim at introducing a new file format for CAD/FEA data exchange. It includes information about geometry (the extended BREP data structure), FEA (nodes and elements), integration between geometry and FEA, boundary conditions and material properties. As an illustration, the following file content represents the UTM definition (geometry and associated mesh) of a tank (Fig. 8) that features both solid, shell and beam bodies. For practical reasons (size of the printed file in the paper), we only illustrate below information about geometry, boundary conditions and sample elements of the mesh.

```
%1=GEOMETRY(1.000000000000000,SLD,ReservoirMAGiC.Sldprt,mat.dat) ;
%2=BODY(nil,($3),0,0) ;
%3=SKIN($2,($42,$81,$120,$152,$214,$246,$294,$306,$318,$324,$337,$350,$356,$368,$376,$382,
  $390,$400)) ;
%4=SURFACE_SLD(Face90) ;
%5=FACE(Face90,$4,($6),1,0) ;
%6=LOOP($5,($15,$22,$29,$36,$41)) ;
%7=CURVE_SLD(Edge225) ;
%8=POINT_SLD(Vertex145) ;
%9=VERTEX(Vertex145,$8,0) ;
%10=POINT_SLD(Vertex146) ;
%11=VERTEX(Vertex146,$10,0) ;
%12=EDGE(Edge225,$7,$13,$14,1,0) ;
%13=COVERTEX($9,$12,1) ;
.
.
%2396=SHELL(FaceShell108,($2397),0,0) ;
%2397=SKIN($2396,($2408)) ;
%2398=SURFACE_SLD(FaceShell108) ;
%2399=FACE(FaceShell108,$2398,($2400),1,1,((EP,0.01000000000000000))) ;
%2400=LOOP($2399,($2407)) ;
%2401=CURVE_SLD(EdgeShell270) ;
%2402=POINT(-4.000000099522372,4.996003610813204e-16,-1.000000000000000) ;
%2403=VERTEX(nil,$2402,0) ;
%2404=EDGE(EdgeShell270,$2401,$2405,$2406,1,0) ;
%2405=COVERTEX($2403,$2404,1) ;
%2406=COVERTEX($2403,$2404,2) ;
%2407=COEDGE($2404,$2400,1) ;
%2408=COFACE($2399,$2397,1) ;
.
.
%2422=BEAM(EdgeBeam272,($2423),0,0) ;
%2423=LOOP($2422,($2432)) ;
%2424=CURVE_SLD(EdgeBeam272) ;
%2425=POINT_SLD(VertexBeam174) ;
%2426=VERTEX(VertexBeam174,$2425,0) ;
%2427=POINT_SLD(VertexBeam175) ;
%2428=VERTEX(VertexBeam175,$2427,0) ;
%2429=EDGE(EdgeBeam272,$2424,$2430,$2431,1,0) ;
%2430=COVERTEX($2426,$2429,1) ;
.
.
%3189=MESH($1) ;
%3190=NODE($9,-3.400000000000000,-0.9136486133789564,-0.7499999999999999);
.
.
%3472=SEGMENT($12,$3190,$3191) ;
.
.
%4587=TRIANGLE($5,$3192,$3191,$3474) ;
.
.
%15399=TETRA($3170,$3430,$3429,$3471,$10536) ;
END;
```

6. RESULTS OBTAINED WITH THE UTM

6.1. Introduction

As introduced in sections 2.1 and 5.6, different research projects are carried out, based on the

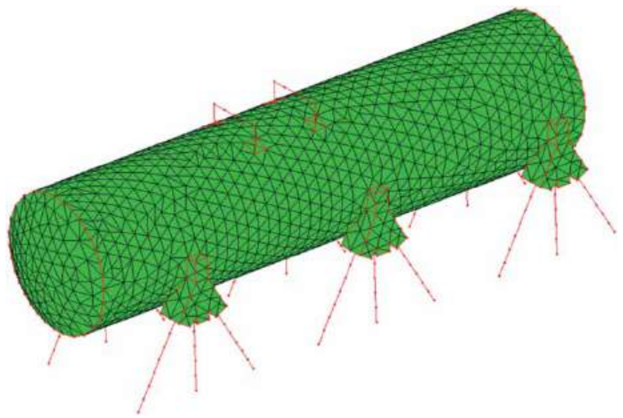

Fig. 8: The UTM mesh of a tank, featuring solid bodies, shells and beams.

UTM structure and underlying concepts. There is a close relationship between these projects and the UTM because most of the UTM features derive from requirements that are inherent to requirements of CAD-FEA integration but also to more specific requirement inherent to these projects. This structure is thus constantly enriched throughout the progress of our research towards integrating CAD-FEA and TO methods. This not only allows foreseeing future developments and enrichments of the UTM but it also provides us with a versatile and powerful development platform for our future research work. Moreover, one of the most interesting features of the UTM is the fact that its algorithms are integrated together instead of being disconnected. This allows studying the interactions between UTM features. For example, this allows studying the influence of pre-optimization on the efficiency of mixed-dimensional models or the influence of varying mesh size distribution on TO results. The following section presents a set of results obtained with some of the UTM features.

6.2. Mesh Pre-optimization

Fig. 9 illustrates automatic mesh density pre-optimization results obtained from the UTM.

This concept consists in, a priori (before any FEA) and automatically refining 3D meshes, based on feature recognition techniques. Form features models are derived into size maps, which represent constraints that must be respected during mesh generation. Form features are either identified from the BREP or from the CAD feature tree information available in the UTM, which makes this feature identification process particularly efficient. The figure illustrates the mesh pre-optimization of a pump housing. For symmetry, only half of the CAD is considered (Fig. 9a). In this case, the size map (in. Fig. 9b and with a cutting plane applied in Fig. 9c) is derived from a surface tolerance criterion. Fig. 9d illustrates the mesh generated along with the projection of the imposed size map on it while Fig. 9e shows the actual size map obtained. The difference (in %) between these two size maps is shown in Fig. 9f (the maximum deviation is 46% on a single node and for most of the mesh the difference is less than 3%).

6.3. Automatic Remeshing

The next example of using the UTM, is shown in Fig. 10 where automatic remeshing is presented on the model of an ejector.

An initial model and a modified model are introduced in Fig. 10a and Fig. 10b where differences between these two models are identified with red circles (a pocket has been added and a hole translated). The initial mesh is shown in Fig. 10c. From this

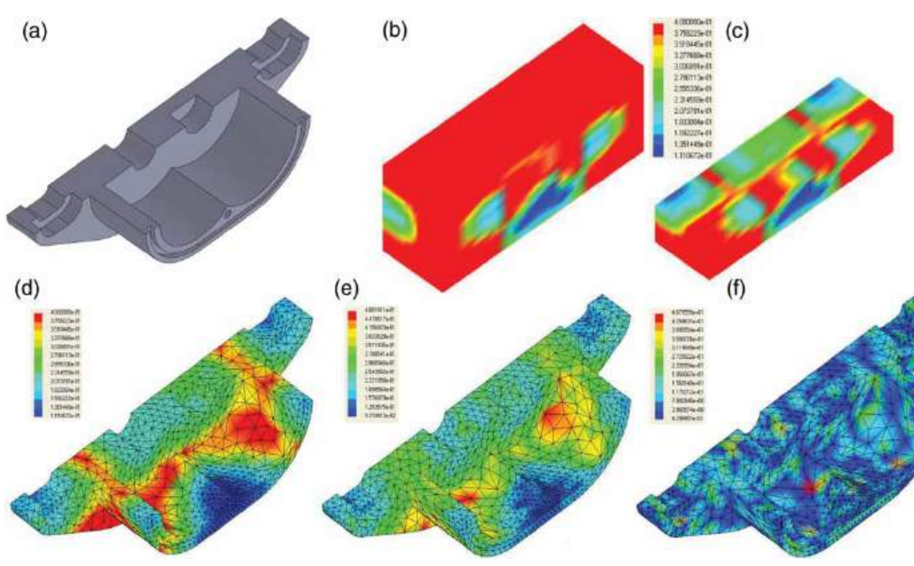

Fig. 9: Mesh pre-optimization with the UTM illustrated on a pump housing (half model).

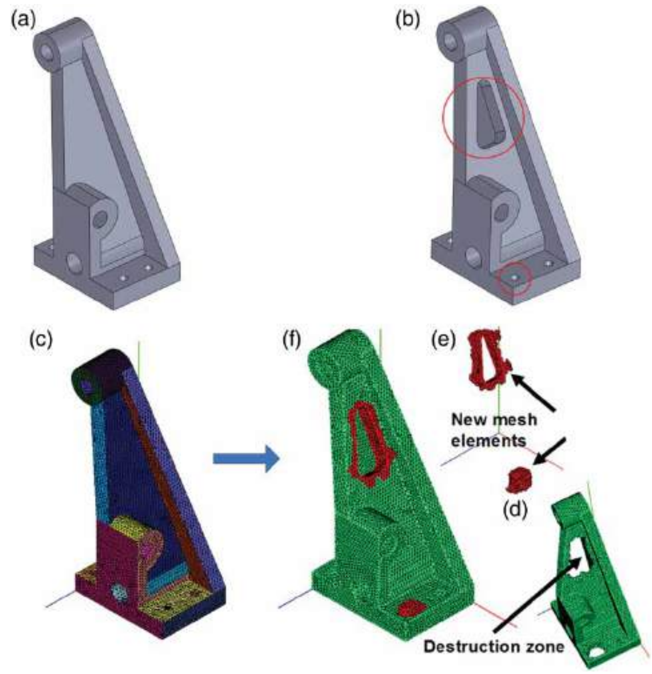

Fig. 10: Automatic remeshing with the UTM.

input, automatic remeshing first consists of automatically identifying differences between the two models and retrieving as many mesh elements as possible from the initial mesh. These elements that have been retrieved from the initial mesh are shown in Fig. 10d. New mesh elements are then created to fit with the new model (illustrated in red in Fig. 10e). These elements are finally joined with elements retrieved from the initial mesh to fulfill the automatic remeshing process (Fig. 10f). The benefit of using such an approach is obvious, especially for initial meshes that have been refined through iterative a posteriori mesh adaptation. We are investigating a very interesting potential extension of automatic remeshing, which is automatic re-analysis. Automatic reanalysis means analyzing the modified model by retrieving as much information as possible from analysis of the initial model. We are presently working on this enhancement.

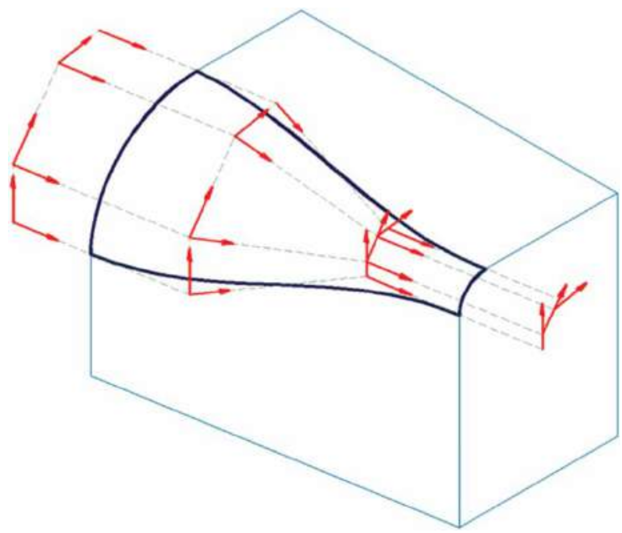

Fig. 11: A vector-based representation of geometry within the UTM.

6.4. Automatic Comparisons between CAD Models

One of the key issues in automatic remeshing is being able to automatically identify and locate the differences between two models. In the UTM, this is performed through the use of a vector-based representation of geometry, which is referred to, in our work, as the *vectorial space* (Fig. 11).

As detailed in [10] it is automatically derived from the BREP and it is used to ensure that the comparison is invariant to affine transformation, which is a very sensitive problem. Fig. 12 presents comparison results obtained on two CAD models of a hook (Fig. 12a). In Fig. 12b and Fig. 12c two views of matching results between the two models are provided and these matching results are identified using color conventions for which two BREP entities (two vertices, two edges or two faces) identified as the same are represented using the same color and BREP entities that are not matching with the other model are black.

6.5. Meshing Constraints Topology and Automatic De-featuring

The capabilities of the UTM with respect to automatic de-featuring and virtual topology (Fig. 13) are also one of its interesting features [12,13].

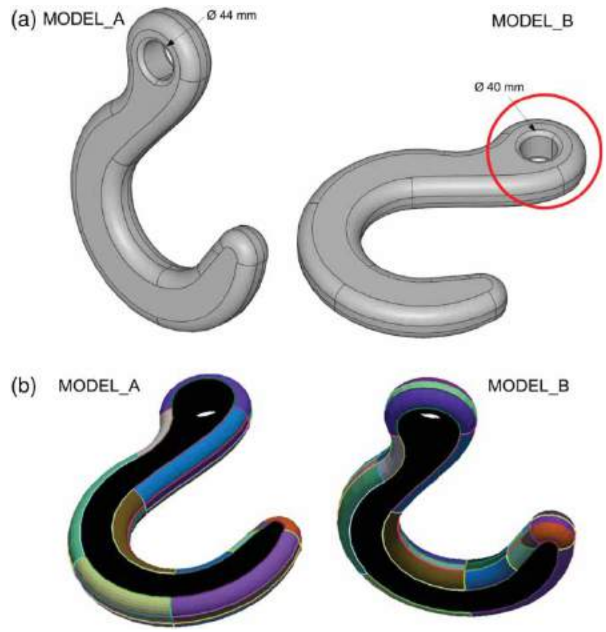

Fig. 12: Comparison between two BREP models with the UTM.

In the UTM, virtual topology is referred to as meshing constraints topology (MCT), which is described with details in [12]. The process consists in identifying BREP shape and topology details that are irrelevant for FEA, building a new BREP geometry and topology (the MCT) and automatically generating a mesh that fits with the MCT. This last procedure is based on specific and automatic mesh generation algorithms over composite edges and composite surfaces, as detailed in [13]. In Fig. 13a, the quarter of a piston is considered and meshed as is, which creates very badly shaped finite elements as illustrated in Fig. 13b. By applying the MCT, a new topology (see Fig. 13c) is automatically derived from the initial topology and mesh generation over composite geometry leads to the final mesh as shown in Fig. 13d. It is easy to understand that this new topology allows avoiding badly shaped elements and, by the way, generating a mesh that is much more suitable for FEA.

6.6. Mixed-dimensional Modeling and Analysis

It has been introduced section 2.1 that the UTM also supports mixed dimensional geometry, which means a mix between curve, surface and volume geometry. Towards this direction, we have seen in section 5.3 that the BREP classical structure has been enriched in the UTM with open SKINs and open LOOPS co-topology features that allow managing mixed-dimensional models like the example shown in Fig. 14 (for beam-volume connexions) in an efficient and consistent way [8]. Automatically meshing this type of mixed-dimensional models also requires solving problems related to mesh continuity and to incompatibility between degrees of freedom for some of the finite elements classically used. These problems are solved in the UTM through the introduction of specific connexion operators as described in [8].

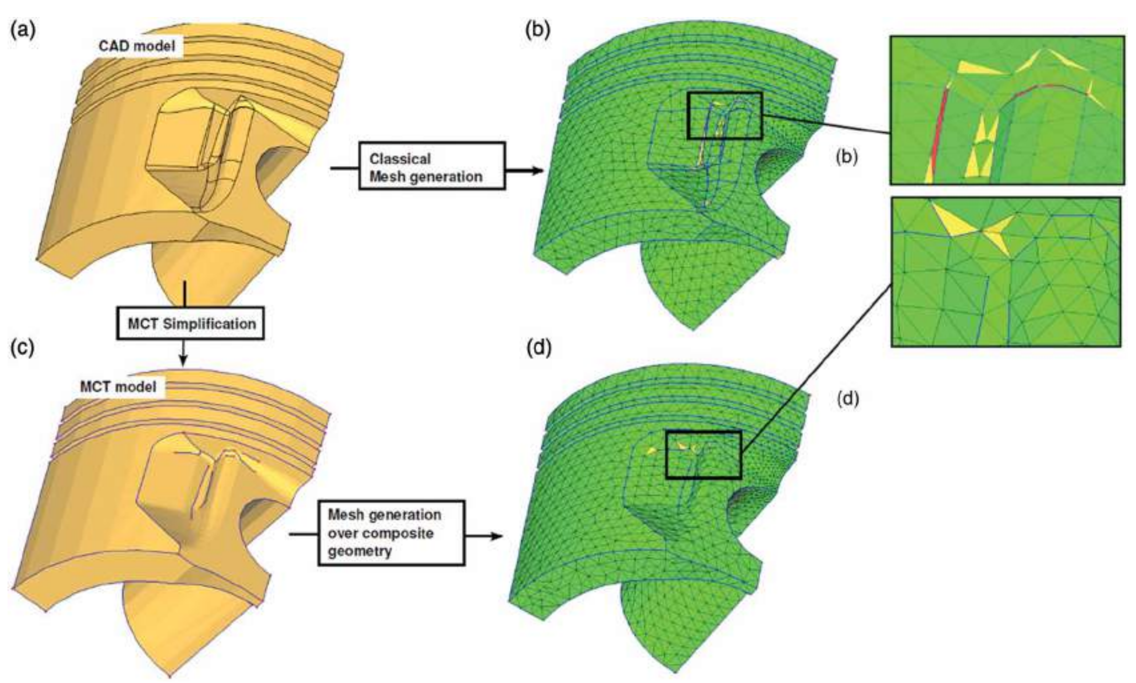

Fig. 13: Meshing constraints topology a) An initial CAD model. b) Mesh inconsistencies generated from the initial model. c) MCT simplified CAD model. d) Solving mesh inconsistencies with MCT simplification and mesh over composite geometry.

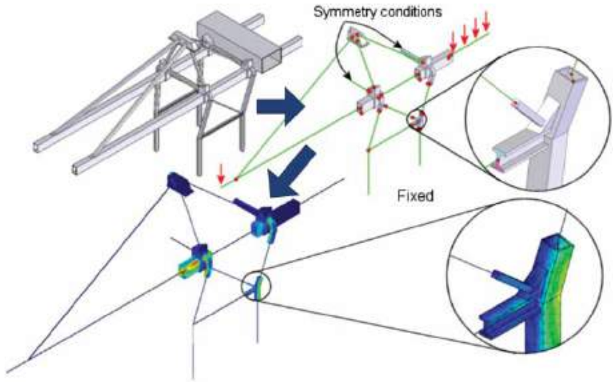

Fig. 14: Mixed-dimensional modeling and analysis.

6.7. Integration of Topology Optimization Methods

The last feature presented in this paper is related to the integration of *topology optimization methods* into the UTM (Fig. 15).

Applying TO methods on a CAD model requires the definition of *design and non-design geometry*. Indeed, when optimizing a component, there is always a subset of its geometry that has to be kept as is. This is mainly due to the fact that most components have relationships with other components and that consequently, at the interface between two components, the material cannot be modified. This material is referred to as *non-design geometry*. A simple illustration of it is the material of a part around mounting holes. Then, for a given component, the material that is not part of the component's non-design material is referred to as the *design material* or, in other words, the material that can be affected by the optimization process. Easily and efficiently defining design and non-design geometry is one of the key issues towards integrating TO with CAD. As illustrated in Fig. 15a and Fig. 15b, in the UTM, design and non-design material is defined using the BREP model of the component

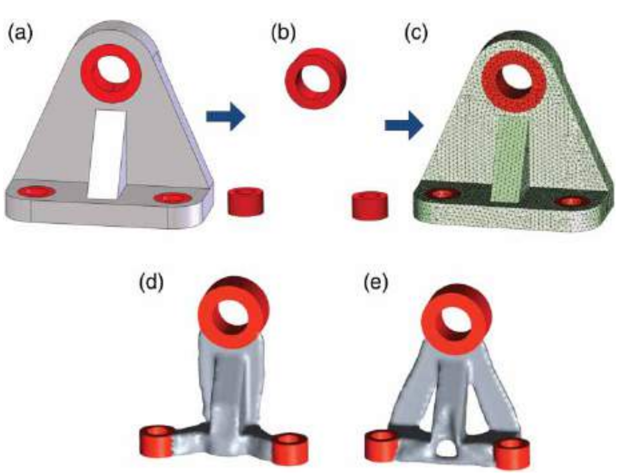

Fig. 15: Integration of topology optimization into the UTM.

to be optimized (Fig. 15a) and a second BREP model associated with non-design material (Fig. 15b). Consequently, the second BREP model (associated with non-design material) is a subset of the first one and it can be derived from it very easily. At this point, it is derived interactively from the first BREP model, which requires user input. However, one of the advantages of defining non-design geometry this way is that it could be derived automatically from the analysis of contact between components in the assembly model. Once defined *design* and *non-design* sub-domains, specific and automatic mesh generation procedures have been introduced in the UTM for the integration of the TO process itself. These procedures ensure that finite elements are tagged as *design* or *non-design* and that the mesh is conformal at the interface between design and non-design sub-domains. Fig. 15c illustrates the mesh that has been generated from data shown in Fig. 15a and Fig. 15b. Then, Fig. 15d and Fig. 15e illustrate two optimization results derived (using two different sets of parameters). For both optimized parts, 80% of design material has been automatically removed through the TO process. In this case, the SIMP method has been used for the TO process itself but many other optimization methods could also have been used successfully.

7. CONCLUSION AND PERSPECTIVES

In this paper, we have introduced the data structure that underlies most of our work. This structure is focused on managing data and tools dedicated to a better integration between CAD, FEA and TO methods. The core of this structure is based on an enrichment of BREP and mesh generation concepts and tools. It allows using models coming from different sources and supports non-manifold geometry and multi-dimensional models. Enhancements of the UTM will come in a natural way through the enhancement of existing tools and through the introduction of new tools along the progress of our work. These enhancements will be facilitated by the fact the design of the UTM is focused towards generality, modularity and ability to evolve. More specifically, introducing automatic reanalysis capabilities in conjunction with remeshing capabilities, as suggested in section 6.2, is one of our most immediate targets. Also, introducing alternative TO methods can be seen as an objective that can be met in quite a short term. Being able to automatically derive functional BREP models from TO results and being able to automatically derive mixed-dimensional models from 3D solid models are also two of our main and most immediate objectives towards contributing to better integrate CAD, FEA and TO methods.

ACKNOWLEDGEMENTS

This study was carried out as part of a project supported by research funding from the Québec Nature and Technology Research Fund and by the Natural Sciences and Engineering Research Council of Canada (NSERC).

REFERENCES

[1] Bazilevs, Y.; Calo, V. M.; Cottrell, J. A.; Evans, J. A.; Hughes, T. J. R.; Lipton, S.; Scott, M. A.; Sederberg, T. W.: Isogeometric analysis using T-splines, Computer Methods in Applied Mechanics and Engineering, (199)5–8, 2010, 229–263. http://dx.doi.org/10.1016/j.cma.2009.02.036

[2] Beall, M. W.; Walsh, J.; Shephard, M. S.: Accessing CAD geometry for mesh generation, presented at the Proceedings of 12th International Meshing Roundtable, Santa Fe, 2003.

[3] Bendsoe, M. P.; Sigmund, O.: Topology optimization - Theory,Methods and Applications, 2nd ed, Springer, Berlin, 2003.

[4] Chand, K. K.; Diachin, L. F.; Li, X.; Ollivier-Gooch, C.; Seol, E. S.; Shephard, M. S.; Tautges, T.; Trease, H.: Toward interoperable mesh, geometry and field components for PDE simulation development, Engineering with Computers, (24)2, 2008, 165–182. http://dx.doi.org/10.1007/s00366-007-0080-z

[5] Charles, S.; Eynard, B.; Bartholomew, P.; Paleczny, C.: Standardization of the finite element analysis data-exchange in aeronautics Concurrent Engineering, Journal of Computing and Information Science in Engineering, (5)1, 2005, 63–66. http://dx.doi.org/10.1115/1.1861474

[6] Chen, J.; Shapiro, V.; Suresh, K.; Tsukanov, I.: Shape optimization with topological changes and parametric control, International Journal for Numerical Methods in Engineering, (71)3, 2007, 313–346. http://dx.doi.org/10.1002/nme.1943

[7] Cuilliere, J.-C.: An adaptive method for the automatic triangulation of 3D parametric surfaces, Computer Aided Design, (30)1998, 139–149.

[8] Cuillière, J.-C.; Bournival, S.; François, V.: A mesh-geometry-based solution to mixed-dimensional coupling, Computer-Aided Design, (42)6, 2010, 509–522. http://dx.doi.org/10.1016/j.cad.2010.01.007

[9] Cuillière, J.-C.; François, V.; Drouet, J.-M.: Automatic mesh generation and transformation for topology optimization methods, Computer-Aided Design, (45)12, 2013, 1489–1506. http://dx.doi.org/10.1016/j.cad.2013.07.004

[10] Cuillière, J.-C.; François, V.; Souaissa, K.; Benamara, A.; BelHadjSalah, H.: Automatic comparison and remeshing applied to CAD model modification, Computer-Aided Design, (43)12, 2011, 1545–1560. http://dx.doi.org/10.1016/j.cad.2011.08.007

[11] Cuilliere, J. C.; Maranzana, R.: Automatic and a priori refinement of three-dimensional meshes based on recognition techniques, Advances in Engineering Software, (30)1999, 139–149.

[12] Foucault, G.; Cuilliere, J.-C.; Francois, V.; Leon, J.-C.; Maranzana, R.: Adaptation of CAD model topology for Finite Element Analysis, Computer Aided Design, (40)2, 2008, 176–196. http://dx.doi.org/10.1016/j.cad.2007.10.009

[13] Foucault, G.; Cuillière, J. C.; François, V.; Léon, J. C.; Maranzana, R.: Generalizing the advancing front method to composite surfaces in the context of meshing constraints topology, CAD Computer Aided Design, (45)11, 2013, 1408–1425. http://dx.doi.org/10.1016/j.cad.2013.05.009

[14] Francois, V.; Cuilliere, J.-C.: An a priori adaptive 3D advancing front mesh generator integrated to solid modeling, Recent Advances in Integrated Design and Manufacturing in Mechanical Engineering, 2003, 337–346.

[15] Frey, P. J.; George, P.-L.: Mesh generation: Application to finite elements, 2008.

[16] Haimes, R.; Crawford, C.: Unified geometry access for analysis and design, presented at the International Meshing Roundtable, 2003.

[17] Hughes, T. J. R.; Cottrell, J. A.; Bazilevs, Y.: Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, Computer Methods in Applied Mechanics and Engineering, (194) 39–41, 2005, 4135–4195. http://dx.doi.org/10.1016/j.cma.2004.10.008

[18] Louhichi, B.; Aifaouli, N.; Hamdi, M.; BenAmara, A.; Francois, V.: An optimization-based computational method for surface fitting to update the geometric information of an existing B-Rep CAD model, International Journal of CAD/CAM, (9)1, 2009.

[19] Mantyla, M.: An introduction to solid modeling, 1988.

[20] Sheffer, A.: Model simplification for meshing using face clustering, Computer Aided Design, (33)2001, 925–934.

[21] Sypkens Smit, M.; Bronsvoort, W. F.: Integration of design and analysis models, Computer-Aided Design and Applications, (6)6, 2009, 795-808. http://dx.doi.org/10.3722/cadaps.2009.795-808

[22] Tautges, T.: CGM: a geometry interface for mesh generation, analysis and other applications, Engineering with Computers, (17)2001, 299–314.

[23] Thakur, A.; Banerjee, A. G.; Gupta, S. K.: A survey of CAD model simplification techniques for physics-based simulation applications, CAD

Computer Aided Design, (41)2, 2009, 65–80. http://dx.doi.org/10.1016/j.cad.2008.11.009

[24] Tierney, C. M.; Nolan, D. C.; Robinson, T. T.; Armstrong, C. G.: Managing Equivalent Representations of Design and Analysis Models, Computer-Aided Design and Applications, (11)2, 2014, 193–205. http://dx.doi.org/10.1080/16864360.2014.846091

[25] Xie, Y. M.; Yang, X. Y.; Liang, Q. Q.; Steven, G. P.; Querin, O. M.: Evolutionary structural optimization, Recent Advances in Optimal Structural Design, 2002, 125.