Amroune, A., Cuillière, J.-C. et François, V. (2022). Automated Lofting-Based Reconstruction of CAD Models from 3D Topology Optimization Results. Computer-Aided Design (Vol. 145, p. 103183).
doi: https://doi.org/10.1016/j.cad.2021.103183.

Automated lofting-based reconstruction of CAD models from 3D topology optimization results

Abdennour Amroune, Jean-Christophe Cuillière, Vincent François

ERICCA, Université du Québec à Trois-Rivières, 3351, boulevard des Forges, Trois-Rivières (Québec) G8Z 4M3, Canada.

Corresponding author: Jean-Christophe Cuillière
jean-christophe.cuilliere@uqtr.ca

Abstract

Topology optimization (TO) has become an integral part of the structural design process in recent years. However, automatically deriving parametrized Computer-Aided Design (CAD) models from TO results still represents a great challenge. In this paper, we present a new fully automatic process aimed at converting 3D TO results that tend towards beam-like structures into solid CAD models. Our reconstruction process starts with curve-skeletonization of the optimized shape. The curve-skeleton obtained is used alongside with a boundary triangulation of the optimized shape to compute closed cross-sections along the skeleton branches and junctions at the intersection between branches. These cross-sections are interpolated with cubic B-spline fitting curves, which are used as a basis for lofting operations to generate CAD surface representations of branches and junctions of the optimized shape. Remaining openings in the optimized shape boundary are closed with filling surfaces. A solid CAD model can be built by sewing together all created surfaces and filling, by the way, the closed boundary that comes out of this process. Finite Element Analysis (FEA) is carried out on both the 3D optimal shape and the CAD solid model derived in order to validate this CAD model. Several case studies are presented to demonstrate effectiveness and usefulness of this new approach.

Keywords

Topology optimization, SIMP, reconstruction, interpretation, CAD, integration, curve-skeleton, beam-like structure.

1. Introduction

With recent advances in computer science and additive manufacturing technologies, structural TO is becoming a very powerful tool to improve and accelerate product design process by automatically generating lightweight, high-performance and innovative structures. Topology optimization (TO) is an autonomous iterative process used to predict the optimal material distribution within a defined design space under specific loads and constraints, in order to

optimally meet a set of performance criteria. TO differs from shape optimization, which only modifies the boundaries of an initial design without changing its topology. Using TO, both shape and topology are modified along the optimization process iterations.

TO has evolved rapidly in the past three decades and has attracted attention of both academia and industry. In fact, it has been successfully applied in a wide range of industries including aerospace, biomedical, automotive etc. The most popular TO methods that have been proposed in the literature can be broadly classified into 4 categories: homogenization methods based on microperforated composite materials [1], material density distribution methods including the well-known SIMP method [2], evolutionary methods such as ESO [3] and BESO [4] methods and boundary variation methods, which involve level set based methods [5] and phase field methods [6]. More details of these approaches can be found in [7]. Among these methods, material density distribution methods, which use elementary density distributions to describe and update the optimal shape along iterations, is the most mature technology due to its computational efficiency and numerical stability. This is the reason why the great majority of TO commercial software are based on implementations of the SIMP method (e.g. Altair OptiStruct, MSC Nastran, TOSCA, SolidWorks etc.).

Regardless of the TO method used, raw TO results cannot be directly used in a product design process. Indeed, TO do not produce CAD models while most engineering applications require CAD models, especially during the various stages of product development and manufacturing. Manually reconstructing CAD models from raw TO results is the common present practice, which is very time-consuming, and which penalizes the design process efficiency. The automatic conversion of TO results into practical CAD models remains a great challenge and represents a broad field of research. A CAD model constructed from TO results must represent an easily editable geometry that faithfully fits the optimal shape as derived from TO. Moreover, the generation of CAD models from TO results should ideally consider manufacturing constraints, so that parts and structures derived from TO can be produced at a reasonable cost. Since manufacturability can be taken into account during the optimization process itself, several "manufacturing oriented" TO methods have been proposed in literature. A very interesting survey and classification of this type of methods can be found in [8]. Achieving this objective allows automatically generating reliable CAD models from TO results and contributes to a better integration of TO along the product design process [9]. The aim of this paper is to present a new and fully automated process for transforming 3D TO raw results that tend towards beam-like structures into 3D CAD solid models. In this work, we use the SIMP method for TO, but our reconstruction process could successfully be adapted for other TO methods.

This paper is organized as follows: section 2 will review and discuss previous work related to the automatic interpretation of TO results. After showing how beam-like TO results can be generated and briefly presenting curve-skeletonization techniques in section 3, a detailed description of the proposed approach is given in section 4. Section 5 presents how finite element

analysis (FEA) is carried out on optimized shapes obtained for validation purposes. Several case studies are presented in section 6 to demonstrate the effectiveness of our method. The last section concludes this paper and introduces perspectives of future work on the method proposed.

2. Review of related work

Different approaches have been proposed in the literature aimed at transforming raw TO results into CAD models. These approaches can be grouped into three main categories according to the strategy used for this transformation. The first category includes methods that are mainly based on lofting CAD operations to generate CAD models from TO results. These methods are thus based on generating and interpolating several cross-sections in 3D. Methods classified in the second category try to derive CAD models by applying Boolean subtractions to an initial volume. These Boolean subtractions try to replicate the voids produced by the TO process via sets of pre-established templates. The third category of methods try to replicate the closed boundary of TO results using sets of free-form surface patches.

Consistently with the first category of methods introduced just above, Marsan and Dutta [10] propose a 3D extension of a 2D method developed by Chirehdast [11]. In their work, cross-sections are first computed as iso-density contours from TO raw results and then interpolated by 3D cubic B-Splines curves. CAD lofting through these cross-section curves allows to generate smoothed CAD models. Based on the same principles, Tang and Chang [12, 13] and Hsu et al. [14, 15] present a process that combines topology and shape optimization in order to generate 3D parametric CAD models with the objective of fulfilling volume fraction and allowable stress objectives. However, these methods require significant user interaction, particularly during the computation of cross-sections. Indeed, location and/or orientation of representative cross-sections is not automatically defined. Moreover, modelling transitions between different branches requires extensive user intervention.

Among methods of the second category, Lin and Chao [16] propose an approach for 2D problems which is later improved by Lin and Chou [17, 18]. This strategy is based on fitting a set of predefined hole templates, with various shapes and sizes, to replicate voids induced by the TO process. Each of these voids is well-positioned and scaled through a shape optimization process that is applied to a hole template. A CAD model of the optimized shape is finally obtained through the Boolean subtraction of these optimized hole templates from the initial design space. An extension of this approach to 3D problems is proposed by Larsen and Jensen [19]. Their process uses the polar map of 2D predefined parametric shapes and selects the best fitted shapes for lofting operation to produce a 3D CAD model for each hole. A CAD model of the optimized shape is finally obtained by applying the Boolean subtraction of all these hole CAD models from the initial design space. One of the major drawbacks of this strategy is that a large set of primitives is needed

to generate accurate CAD models from geometrically complex results. In addition, these methods are semi-automatic and only limited to predefined convex templates. In the same direction, Yi and Kim [20] present a semi-automatic process dedicated to the interpretation of 2D TO results. Using the active contour method and some optimization techniques, their process allows to represent the optimized shape boundary with basic geometric features (lines, circles, arcs and fillets). A parameterized 2D CAD model is derived from these features and used for a subsequent optimization.

The third strategy towards transforming raw TO results into CAD geometry is mainly based on reverse engineering (RE) techniques. These RE techniques allow generating CAD models from point clouds that are typically obtained by 3D scanning. In this context, automatic surface generation tools are used to derive CAD models from TO results. Preprocessing operations are usually needed in this case to reduce the noise level that generally characterizes TO raw results. With respect to this strategy, a three-step reconstruction process has been proposed by Koguchi and Kikuchi [21]. During the first step of this process, a triangulated enclosed iso-density surface model is extracted from TO results by relative density thresholding. In the next step, geometric features such as creases and corners are detected. In the last step of this process, after having converted the triangulated surface mesh extracted in the first step into a quadrangular surface mesh, a CAD model is generated from TO results as biquartic surface splines where features are retained. Similarly, Park et al. [22] present a semi-automatic process that allows extracting free-form B-spline surfaces and characteristic curves from surface mesh models. At first, using the k-means classification method, this process segments the mesh into regions, based on main curvature information. Then, according to several criteria, growth and merging operations are applied to these regions. After straightening its boundaries, each region is converted into a free-form B-spline surface. In the last step, with significant user interaction, characteristic curves (sections and guides) are determined for appropriate regions in order to generate fitted surfaces by lofting. The main drawback of this strategy is that CAD models generated with it are not easily editable and cannot be used as is for further optimization. Moreover, the large number of free-form surfaces generated during reconstruction makes that subsequent FEA is generally complex and costly.

Tubular shapes presented by some TO results have attracted the attention of scientists from the early attempts to automatically interpret TO results [23]. This type of results are typically obtained when low volume fractions are used. Volume fraction refers to the percentage of design material that must be kept at the end of the optimization process, with respect to the initial volume of design material. For such tubular TO result, interpretation usually involves two main phases. At first, a curve-skeleton is extracted from the optimized shape generated by TO. Then this skeleton is converted into a structure, which is made up of bars and/or beams. Interpretation of beam-like TO results is studied as special case in some research work [20, 24], while some other research work is specifically oriented towards dealing with this beam-like TO results [25-28].

Nana et al. [27] propose an automatic process that allows 3D idealized beam structures to be generated from TO tubular results. This process is made up of two modules: a reconstruction module and validation module. In the first module, a skeletonization operation is performed on smoothed optimized shapes in order to extract curve-skeletons. Different branches of these skeletons are then straightened by a normalization operation and the average radius of cross-sections is calculated for each branch. In the second module, a multidimensional FEA model is automatically generated using mini-beams to rigidly connect beam elements to 3D solid elements. Validation of beam structures derived from TO results is performed by comparing FEA results obtained from this multidimensional FEA model with FEA results obtained from the raw optimized shape before smoothing. Although this approach was further improved by Cuillière et al. [25], this process still presents numerous limitations. For example the beam reconstruction process only supports circular sections, which is an important limitation. Additionally, the skeletonization method used in this work [29, 30] generates unsatisfactory results in terms of centeredness for branches and junctions, as well as in preserving topology of the original form.

Kresslein et al. [31] propose an automatic method of recovering cross-sections, as planar point clouds, from beam-like surface mesh models. The starting point of this method is a triangular surface mesh, which can be obtained from a point cloud. A skeletonization process is performed on this triangular mesh in order to extract a well-centered curve-skeleton. Then, after segmentation of this skeleton into unbranched skeletal segments, sets of orthogonal planes are located along each skeletal segment at predefined or adaptive distances. These planes are used to compute intersection points with edges of the surface triangular mesh. These points are filtered and grouped together, resulting in a 2D point cloud for each cross-section. Depending on the application targeted, subsequent post-processing operations can be carried out after obtaining these cross-sections as 2D point clouds. Pattern recognition and digital reconstruction of CAD models are good examples of practical applications of this method.

Despite all methods developed towards automatic reconstruction of CAD models from TO results, research in this area is not mature enough for a full integration of TO in the design process and significant efforts are still needed in this direction. In this paper, we specifically focus on TO results that tend towards beam-like structures and we introduce a new method aimed at automatically deriving CAD models from beam-like TO results. The next section presents how beam-like TO results are generated and how curve-skeletonization is performed on these results, which is a basis for the method presented in this paper.

3. Generation of beam-like TO results and skeletonization

3.1. The SIMP method

As mentioned earlier, in this paper, TO results are obtained with the SIMP (Solid Isotropic Material with Penalization) method. This TO method is based on an iterative process aimed at

determining an optimal distribution of the relative density of a virtual material within a given design space. In the SIMP process itself, the material is considered as virtual since it is affected by this relative density field, which can be seen as virtual distribution of porosity. This iterative process is focused on minimizing the global compliance (which means in other words maximizing stiffness) of a given volume of design material. Each iteration in the process involves a FEA (using Code_Aster™ [32] in this work) and the results obtained from this analysis is used to update the relative density field $\rho(x, y, z)$. Convergence is reached when the relative difference in global compliance between two successive iterations is less than a user defined threshold ($\Delta_{\text{conv}} = 0.5\%$ in this work). The relative element density ρ_e for each finite element e of the mesh used in FEA takes values between 0 and 1, which stand for empty and full material respectively. The elastic modulus of this virtual material (affected by the relative density field) varies continuously and it is expressed using a classical penalization law, which is written as follows for a given finite element :

$$\tilde{E}_e = E \cdot (\rho_e)^p$$

Where $\tilde{E}_e$ is the virtual elastic modulus (affected by the relative density of element e) and E is the actual elastic modulus of the material considered. The penalization coefficient p ($p = 3$ in this work) minimizes the contribution of gray elements (elements with intermediate densities) to the total stiffness, which makes that this penalization steers the optimization solution towards elements that are mostly either solid or void.

Compliance can be defined as a measure of the overall flexibility of a given structure. The SIMP optimization method seeks to find the optimal relative density field $\rho(x, y, z)$ through iterations that minimize this global compliance. The mathematical formulation of this problem is as follows (all quantities affected by the relative density field $\rho(x, y, z)$ are marked with a tilde $\tilde{}$) :

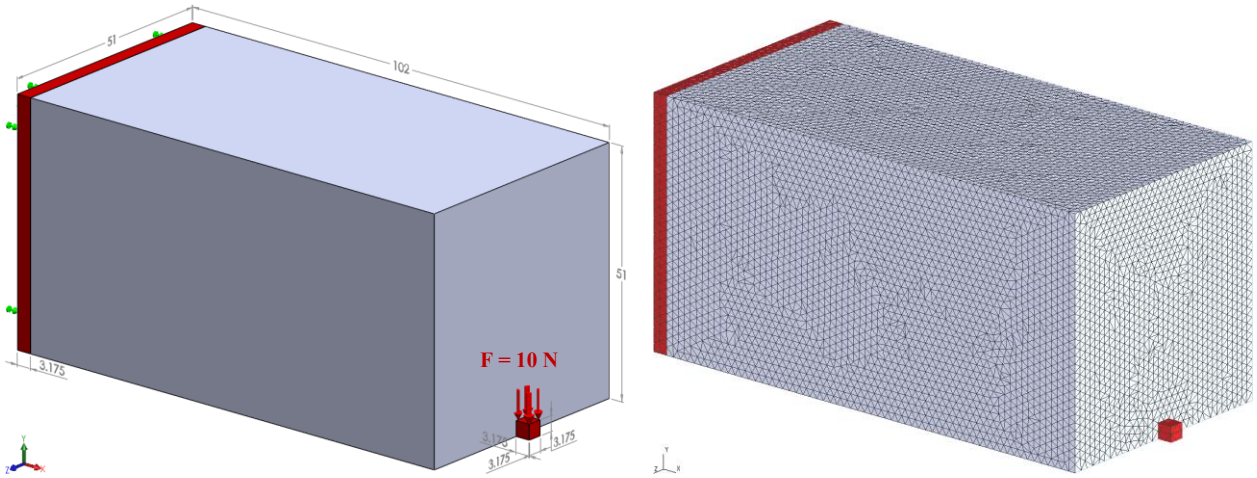
$$\min \tilde{C} = \sum_{e=1}^{N} (\rho_e)^p \cdot \{\tilde{U}\}^t \cdot [K_e] \cdot \{\tilde{U}\}$$

$$\text{with } \begin{cases} [\tilde{K}] \cdot \{\tilde{U}\} = \{F\} \\ \dfrac{\tilde{V}}{V_d} = f \\ 0 \leq \rho \leq 1 \end{cases}$$

Where $\tilde{C}$ is the global compliance for the entire FEA mesh (featuring N elements). $[K_e]$ is the local stiffness matrix associated with element e. $\{\tilde{U}\}$ and $\{F\}$ are respectively the global displacement vector and the global load vector. For each SIMP iteration, equilibrium of the global force-stiffness system and target volume fraction f must be satisfied. f is computed as the ratio between the volume of design material affected by relative density $\rho(x, y, z)$ referred to as $\tilde{V}$, and the volume of the initial design space V_d. Practically, global compliance $\tilde{C}$ is calculated, after each

SIMP iteration, using the total strain energy $\widetilde{W}$ since $\tilde{C} = 2 \cdot \widetilde{W}$. Filtering schemes are applied on the sensitivity of global compliance (referred to as $\partial\tilde{C}/\partial\rho_e$) and/or on the relative density field itself $\rho(x, y, z)$ in order to avoid classical checkerboard effects [33]. The final relative density distribution $\rho(x, y, z)$ is obtained and SIMP iterations stop when the evolution of global compliance reaches convergence. More details about the SIMP method are available on [34]. As mentioned in the introduction, a 3D implementation of the SIMP method is used in this work to perform TO. However, it must be reminded here that many other TO methods could also be used successfully.

The TO process described above is applied on a simple cantilever example. Fig. 1 (a) shows the input model of this cantilever, with an illustration of design material (in gray), non-design material (in red), dimensions (in mm) and boundary conditions (BCs) used in FEA. Non-design material classically refers in TO to material that must be kept unaffected during the optimization process. The actual material elastic modulus is 69 GPa and its Poisson's ratio is 0,33. Fig. 1 (b) shows a mesh of the input model with linear tetrahedrons and based on a uniform element size distribution $d_g = 1.6\,mm$. Tetrahedral meshes used in this work are automatically generated using a specific adaptation (for TO requirements) of the advancing front method [35]. The volume fraction f applied along SIMP iterations for this example is set to 3%. The final distribution of relative density material after SIMP convergence is illustrated in Fig. 1 (c). Once reached this convergence, the rough optimized shape is generated by keeping only mesh elements for which relative density ρ_e is over a threshold, In this case, $\rho_e \geq \rho_{th} = 0.25$ is considered (Fig. 1 (d)). The boundary of this rough optimized shape is extracted as a very noisy triangulation, as shown in Fig. 1 (e). Note that all figures (except Fig. 2) are generated using Gmsh™ [36] and SolidWorks™ [37].



(a) **(b)**

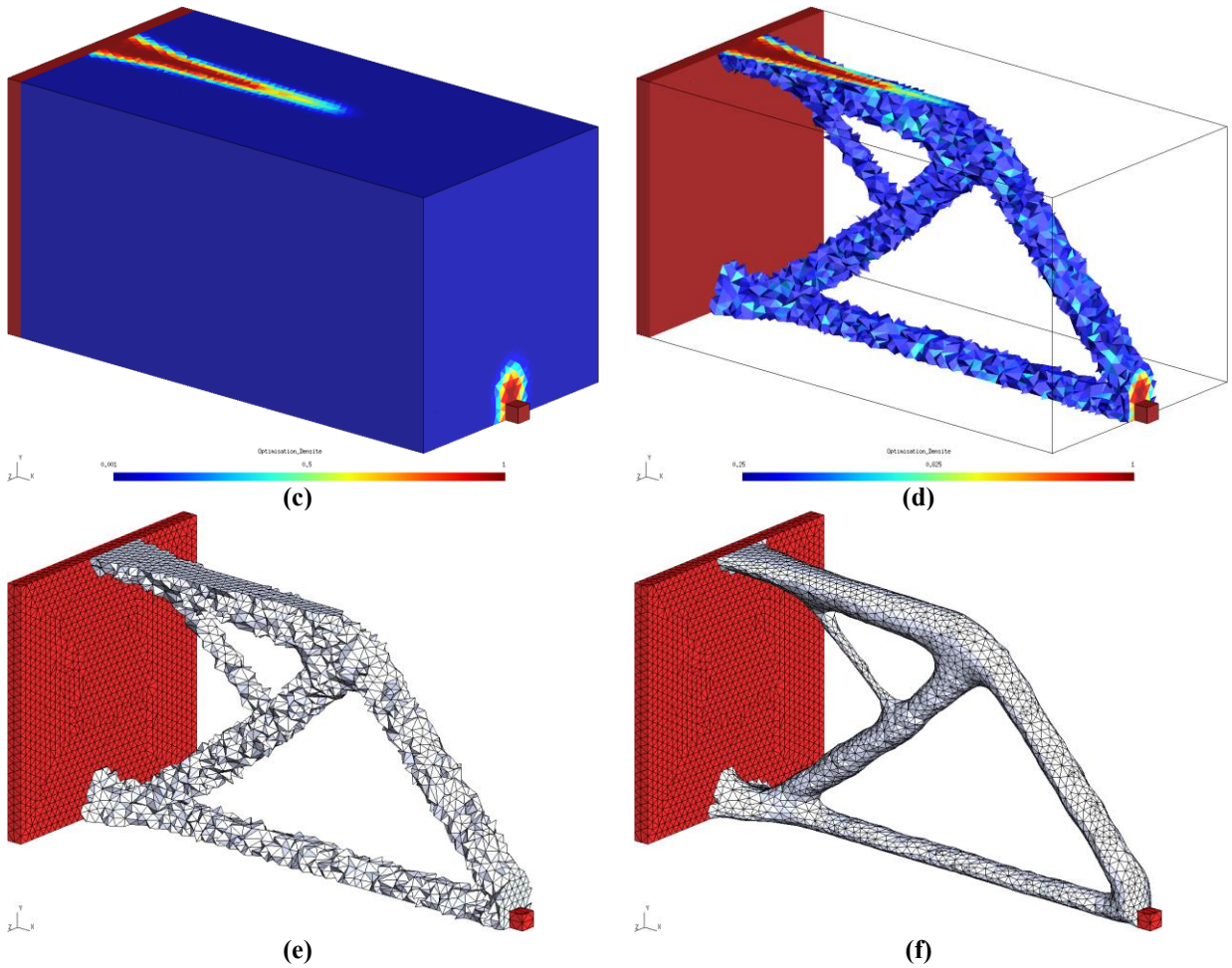


Fig. 1. Cantilever (a) initial problem with design (in gray), non-design (in red), dimensions (in mm) and boundary conditions (b) uniform mesh ($d_g = 1.6\ mm$) (c) relative density distribution after SIMP optimization convergence (d) thresholding result ($\rho_{th} = 0.25$) (e) rough and (f) smoothed boundary triangulation of optimized shape.

3.2. Postprocessing TO results

As illustrated in Fig. 1 (e), the raw boundary triangulation generated from TO results is extremely irregular and needs to be smoothed. In this context, the objective of smoothing should be to produce smooth, untangled and good quality triangulations that closely meet the initial target volume fraction. Many triangulation smoothing techniques available in literature have been explored by our team [38-41] but none of these techniques can be used, as is, to efficiently process the very noisy rough shapes provided by TO. However, combinations of existing triangulation smoothing techniques are likely to produce smooth boundaries while preserving volume and shape. For example, Taubin [40] smoothing can be used to prevent shape shrinkage along with Laplacian-based smoothing [39], which is renowned for its high efficiency for generating very smooth shapes. Applying such a combination of smoothing techniques on the raw TO result shown in Fig. 1 (e) significantly improves the optimized shape, which can be clearly seen in Fig. 1 (f). It is important to note that this smoothing process is only applied to the boundaries of design material

(colored with gray in Fig. 1 (e)). Volume fraction after smoothing is $f_{3D} = 3.2\%$ (compared to the target volume fraction $f = 3\%$). With respect to next steps of the method presented here, this smoothing process also makes curve-skeletonization and cross-sections computations more accurate, which results in a better quality of CAD models generated at the end.

3.3. Curve-skeletonization

The reconstruction process proposed in this paper is mainly based on a skeletonization procedure. In fact, a skeleton preserves the optimized shape topology and enables automation of cross-sections positioning and computation. The fact that we are specifically interested in beam-like TO results suggests an attention to curve-skeletonization methods.

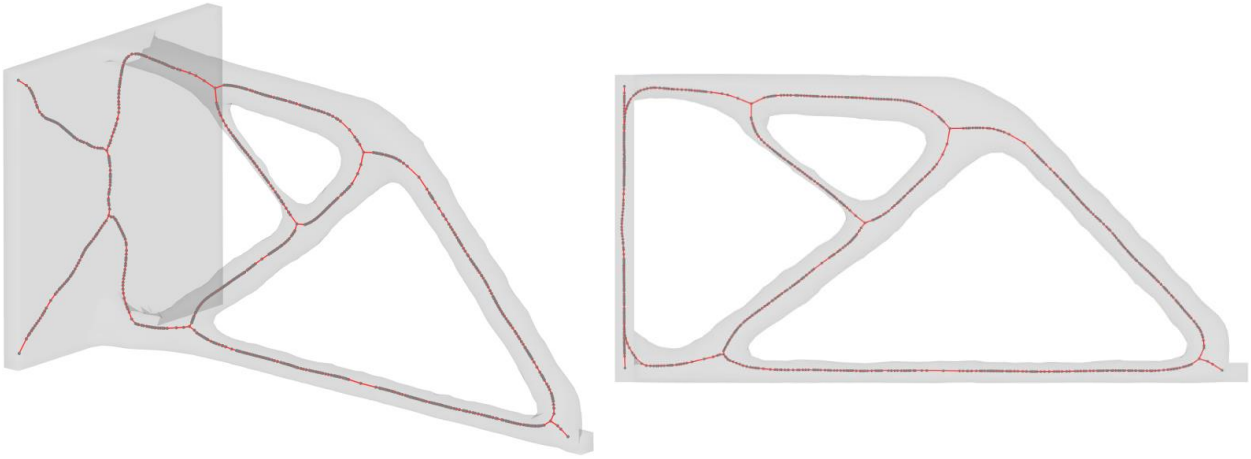
Skeletonization is an automatic process that generates an effective and compact representation of an object while preserving its topologic and geometric information [42]. In other words, the skeleton of a shape is a powerful tool for describing elements of its topology. A skeleton of a 3D shape, also called medial surface or 2D medial axis, is generally composed of surfaces and curves. It can be defined as the centers of largest inscribed balls in the considered 3D shape. A curve-skeleton of a 3D object is a 1D subset of its medial surface. Curve-skeletons are very useful for many applications including computer graphics, medical imaging, object recognition, animation, virtual navigation and more. Depending on the application targeted, different properties of curve-skeletons are considered [43]. In the context of this work, a good curve-skeleton ideally should be homotopic (preserves the original topology), fully connected, insensitive to noise, smooth, included and centered inside the optimal shape. Even though a large variety of skeletonization methods have been proposed in the literature, it is very difficult to fulfill all these requirements in one type of skeletons. The most common curve-skeletonization methods include topological thinning [44, 45], distance field based methods [46], potential field based methods [47, 48], Voronoï diagram based methods [49] and contraction based methods [29, 30, 50]. A very interesting survey on the generation of 3D skeletons can be found in [51].

Among all these methods, contraction-based methods are the most efficient for processing triangulated shapes generated from TO. The principle on which these methods are based consists in iteratively applying Laplacian smoothing on the mesh, which gradually contracts its geometry until a zero-volume skeletal shape is obtained. This contracted mesh is then converted into a curve-skeleton while preserving the original shape topology.

The curve-skeletonization method used in this work is a contraction based method developed by Tagliasacchi et al. [50]. In fact, this method is a new formulation via mean curvature flow (MCF) of the skeletonization method presented in [29]. A given surface is evolving by MCF if each point on that surface moves along its anti-normal with a velocity proportional to its local average curvature. MCF is characterized by its area-minimizing effect, which is used by Tagliasacchi et al. to iteratively contract the input mesh geometry until a full collapsed mesh with a skeletal shape is obtained. Post-processing operations are required to convert this collapsed mesh

to a curve-skeleton. Curve-skeleton generation is controlled by three main parameters $\{w_L, w_H, w_M\}$. Ratio w_L/w_H controls contraction velocity, ratio w_L/w_M controls medial approximation smoothness and w_M controls the intensity of surface attraction to the medial axis. For more details about this curve skeletonization method and its parameters, see reference [50].

This curve-skeletonization method has proven to be very efficient in processing TO results and produce homotopic, smooth and well centered skeletons. Applying this skeletonization method on the smoothed triangulation shown in Fig. 1 (f), allows to automatically generate a curve-skeleton as illustrated in Fig. 2.



(a) **(b)**

Fig. 2. Cantilever curve-skeleton extracted by Tagliasacchi et al. [50] method with $w_L = 1$, $w_H = 0.8$ and $w_M = 1$.

4. Automatic reconstruction of CAD models from beam-like TO results

The full integration of TO into the overall design process is a very ambitious goal given the complexity and scope of this topic. A significant research effort has been devoted to achieving this goal in the last two decades but it still insufficient for such a highly demanding objective. The work presented in this paper mainly aims at this integration by proposing a new fully automatic process for converting 3D TO results that present tubular geometries to parametrized CAD models. An overview of the overall process is given below.

4.1. Overview

Fig. 3 presents a flowchart of the overall reconstruction and adaptation process proposed in this paper. The first steps of this process, from the initial CAD model to the extraction of a curve-skeleton have been discussed in the previous section. In the next step, the extracted curve-skeleton is imported and processed to get it ready for the following step. Once processed this way, the curve-skeleton is gathered with the triangulation of the optimal shape to compute cross-sections

projection points. These cross-sections points are interpolated by cubic B-spline fitting curves, which are then used in lofting operations to generate CAD faces representing branches and CAD faces representing junctions between these branches. A solid model is obtained by assembling all these faces together. FEA results obtained with this solid model are validated with FEA results obtained from the triangulation of the optimal shape. This solid model can be further adapted through iterations on this process to fulfill stress distribution objectives. Indeed, as indicated in Fig. 3, the target volume fraction used in TO can be automatically adapted from FEA results to satisfy stress distribution objectives.

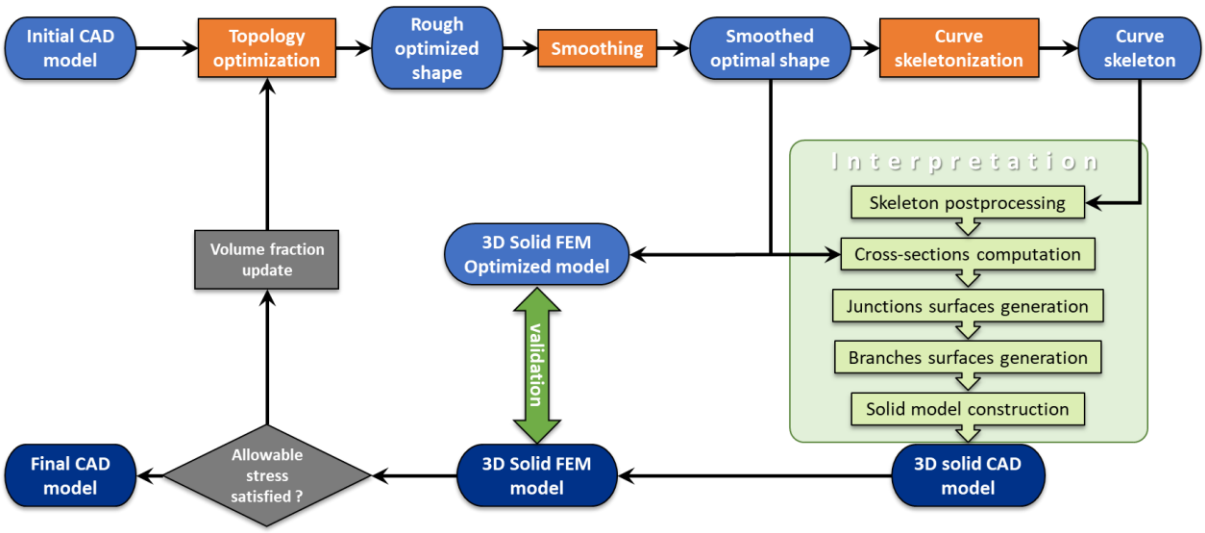


Fig. 3. Flowchart of the overall reconstruction process

The main steps of this reconstruction algorithm will be discussed, with more details, throughout the following sections. The cantilever beam example introduced in section 3 will be used to illustrate the different steps of this approach.

4.2. Processing curve-skeletons

The curve-skeleton presented in section 3.3 (Fig. 2) is generated as sets of ***skeletal nodes*** connected with ***skeletal segments***. Processing operations are then performed on this curve-skeleton as a preparation for the following reconstruction steps. At first, skeletal nodes that are contained in the non-design subdomain are detected and deleted (Fig. 4 (b)). Reconstruction is indeed only required for design material since non-design material remains, by definition, completely unchanged along TO. Consequently, non-design skeletal segments are removed since they have no use in the rest of the process. This leaves the skeleton unconnected with the non-design subdomain. In order to reconnect it, each end node of the curve-skeleton (shown with red points in Fig. 4 (b)) is linked to the nearest design/non-design interface center node with a skeletal segment as illustrated in Fig. 4 (c). The center node of a given design/non-design interface is defined as the centroid of the set of points representing this interface. Finally, all skeleton branches connected to design/non-design interfaces are smoothed using a Laplacian-based smoothing

technique. Fig. 4 (d) illustrates the resulted curve-skeleton after these processing operations (in orange) as well as non-design subdomain (in black) and design/non-design interfaces (in orange).

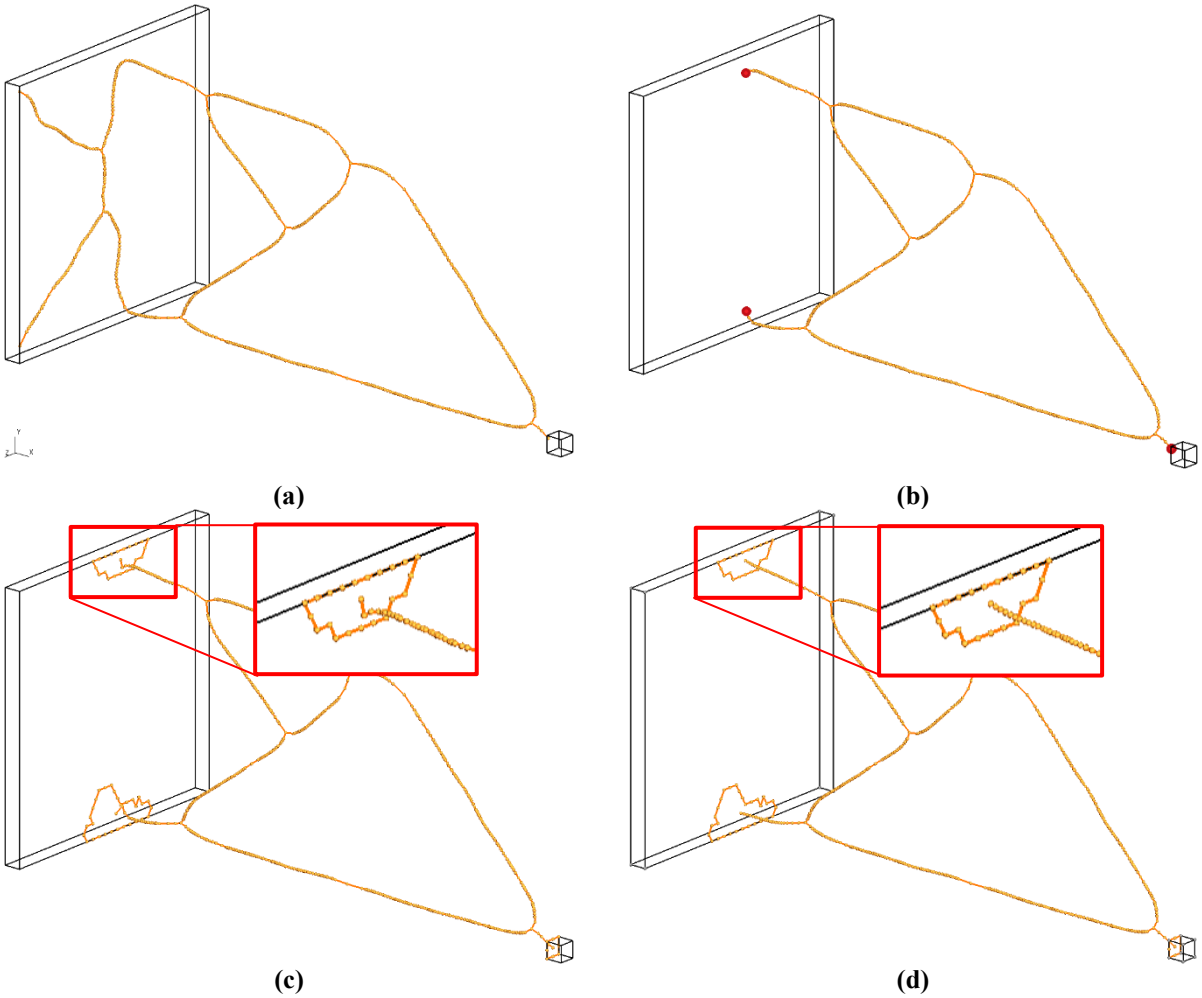


Fig. 4. Skeleton processing steps : (a) importation, (b) deletion of non-design skeletal segments, (c) connecting design skeletal segments with non-design interfaces (d) smoothing connexions with non-design interfaces

4.3. Automatic computation of cross-sections

As mentioned in section 3.3, the curve-skeleton serves as a basis for cross-sections positioning and computation. It is important to note that, at this point in the reconstruction process, each cross-section is defined as a planar point cloud. Each cross-section plane passes through a skeletal node, called the ***positioning node***, and its normal coincides with the ***direction vector*** associated with this node as illustrated in Fig. 5. At each skeletal node (n_i), except for junction nodes, a unit tangent vector ($\vec{d}_i$) is defined. This vector, referred to as ***direction vector***, is parallel to segment $n_{i-1}n_{i+1}$, where n_{i-1} and n_{i+1} respectively are the previous and following nodes along the branch. The direction vector at each connection node with the design/non-design interface Ω (n_1 in Fig. 5) is normal to Ω as illustrated in Fig. 5.

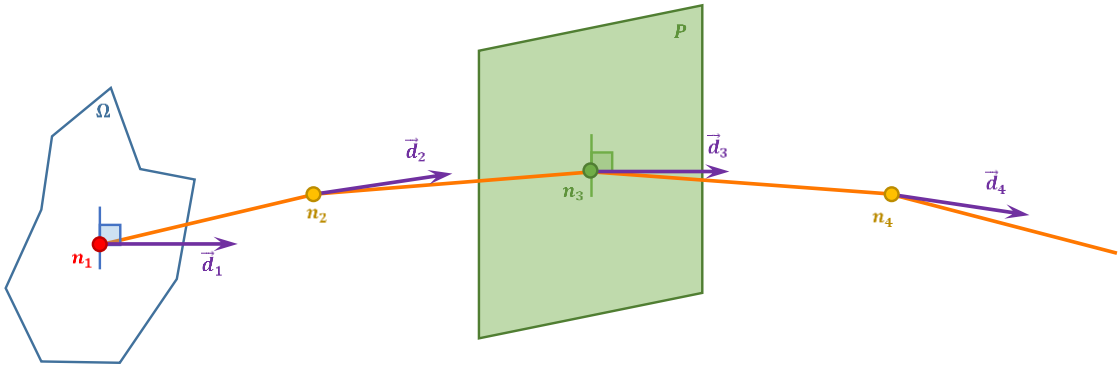
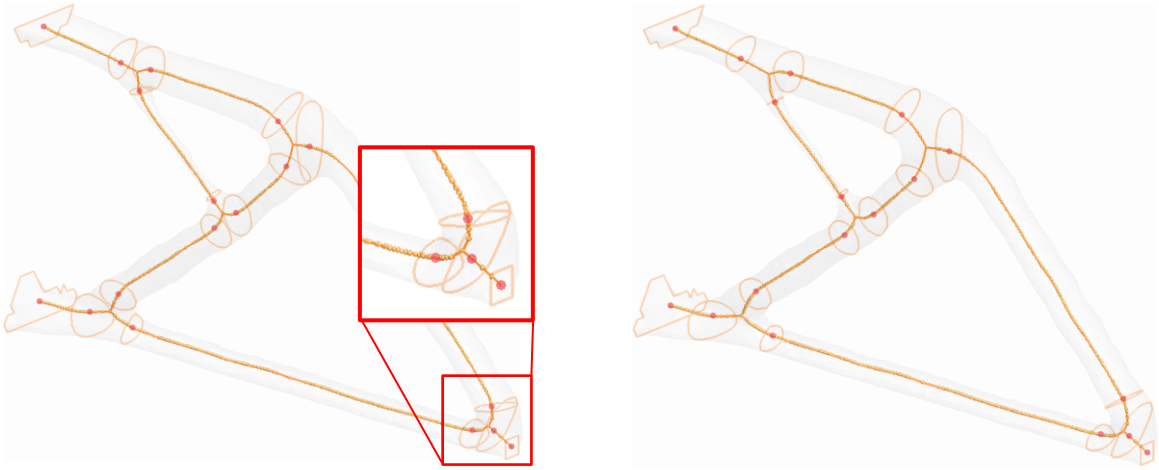


Fig. 5. Direction vectors and cross-section plane

A user-defined distance is used to define ***limit nodes*** at each junction. These nodes allow to segment the optimized shape shown in Fig. 1 (f) into junctions and branches as indicated in Fig. 6 (b). For a given junction, the distance of limit nodes from the junction node is proportional to the average of radii calculated in the middle of all branches attached to this junction. Note that for some values of this distance parameter, limit cross-sections calculated around the junction may intersect (see zoomed junction in Fig. 6 (a)). A good choice of this factor, which is generally between 1.2 and 2.0, is necessary to avoid interference and the problems that may result from it.



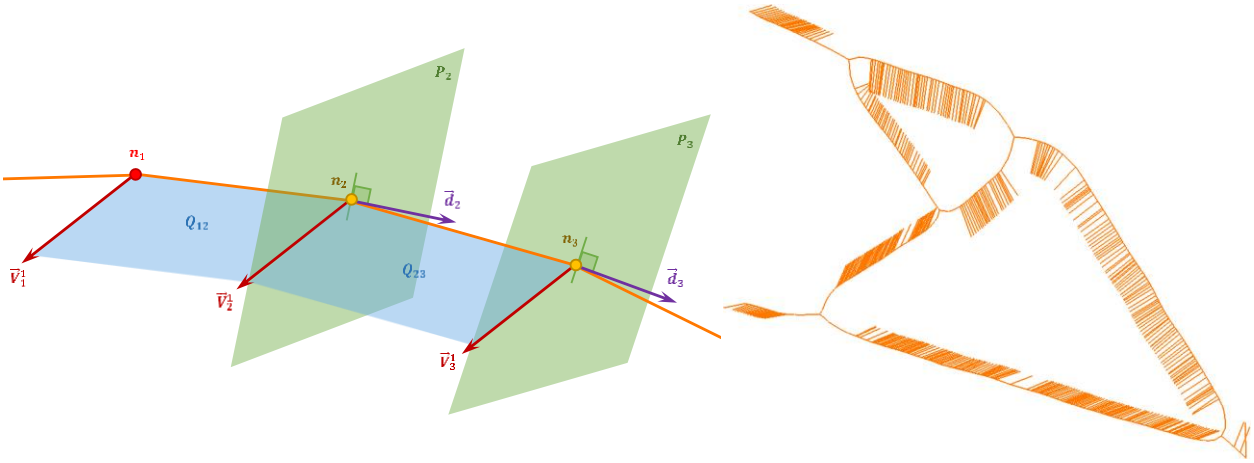
(a) **(b)**

Fig. 6. Different values for distance parameter : (a) 0.9 and (b) 1.3

As mentioned previously, interpolation points associated with a given cross-section are obtained through projecting the positioning node on the smoothed triangulation (see Fig. 8 below) according to a set of projection directions. At a given node, all these projection directions are normal to the direction vector associated with this node. Then, for each cross section, these interpolation points are interpolated by a closed cubic B-spline curve with a G_1 continuity condition for starting and ending points. The starting point (which is also the ending point) of this curve corresponds to the first interpolation point associated with the cross-section. To avoid generating twisted surfaces, starting points of B-spline curves along the same branch must be

properly aligned. This alignment is achieved through an alignment of first projection directions along each branch as shown in Fig. 7 (b).

The calculation of these first projection directions along a given branch, is carried out sequentially and it is based on using intersections between two planes at each step (Fig. 7 (a)). A first projection direction $\vec{V}_1^1$ for a given branch is associated with the first node n_1 along this branch, which is a limit node as introduced earlier (see Fig. 6). This direction is "transferred" to node n_2 as the intersection between plane Q_{12}, formed by $\vec{V}_1^1$ and $\overrightarrow{n_1 n_2}$, and plane P_2, which is perpendicular to direction vector $\vec{d}_2$ at n_2. The resulting first projection direction $\vec{V}_2^1$ at node n_2 is, in turn, "transferred" to node n_3 using the same principle. Repeating this operation for each skeletal node along a branch makes that all first projection directions are nearly aligned and consistent along this branch as shown in Fig. 7 (b).



(a) **(b)**

Fig. 7. Computation of the first projection direction

Thus, the first interpolation point (p_1) in each cross-section sequence (i) is obtained by projecting its positioning node (n_i) on the triangular mesh according to the first direction of projection $\vec{V}_i^1$ defined previously. From this first direction and according to N_p, the number of interpolation points p_j, projection directions $\vec{V}_i^j$ (Fig. 8) are calculated with the following recursive formula:

$$\vec{V}_i^j = \vec{V}_i^{j-1}\cos\theta + \left(\vec{d}_i \times \vec{V}_i^{j-1}\right)\sin\theta \qquad \text{with}: \theta = 2\pi/N_p \quad \text{and} \quad j \in \{2, \ldots, N_p\}$$

For a given cross-section, all interpolation points lie in the same plane, which is normal to direction vector $\vec{d}_i$. In the case shown in Fig. 8, $N_p = 5$, which means that 5 interpolation points are calculated for each cross-section. For complex cross-section shapes, more interpolation points per cross-section can be user-specified.

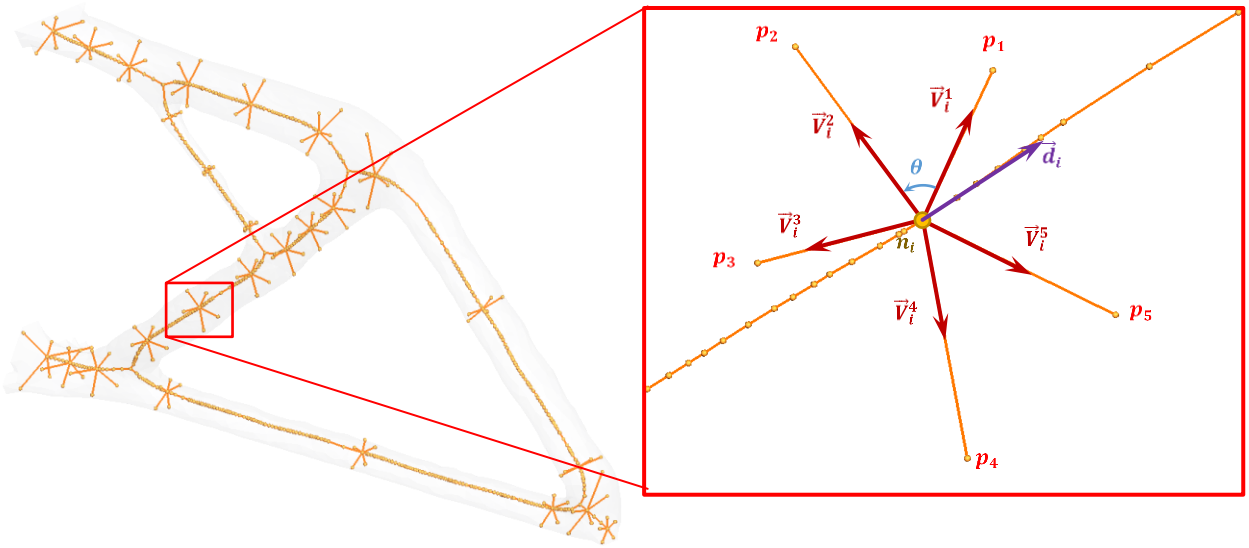
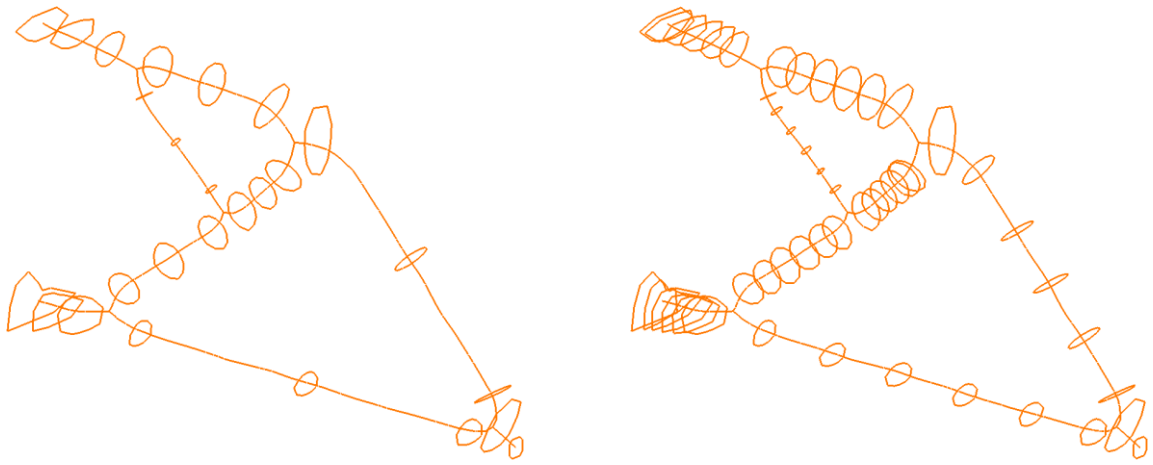


Fig. 8. Computation of cross-sections

Also, depending on user input, a number of equally spaced cross-sections are calculated between branch limits, as sets of interpolation points and curves (Fig. 9). It must be noted that a minimum of two cross-sections per branch are calculated at limit nodes.



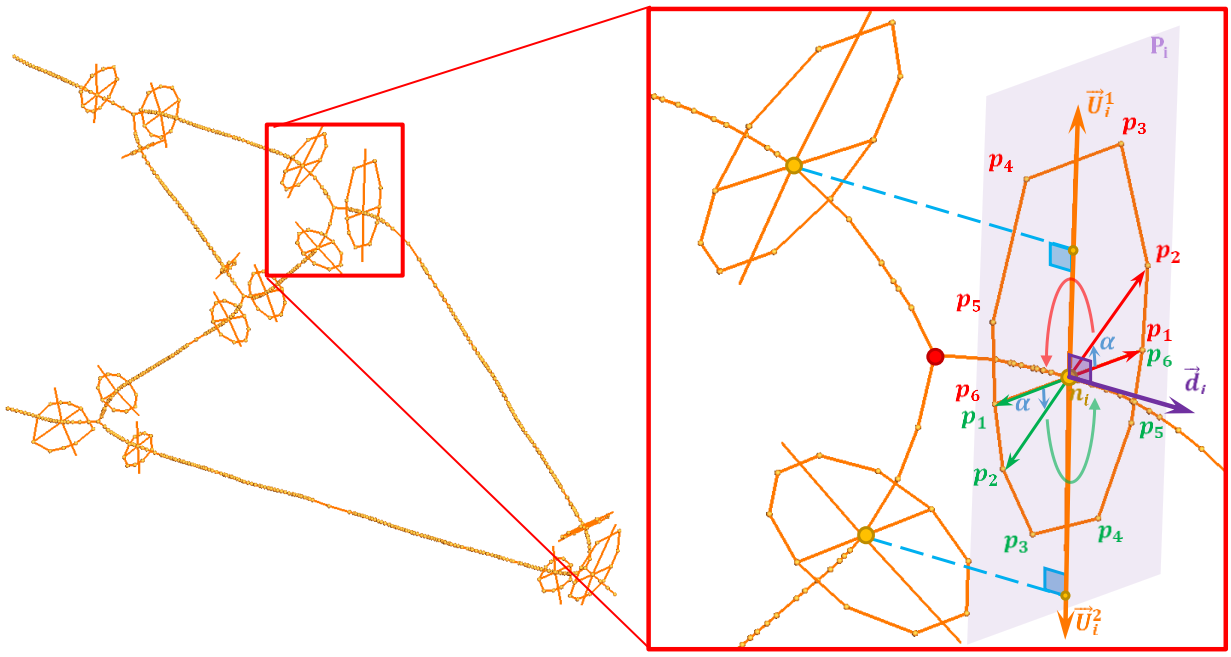
(a) **(b)**

Fig. 9. Computation of (a) 3 and (d) 6 cross-sections per branch

4.4. Automatic generation of junction surfaces

In order to create connecting surfaces between branches, each junction/branch interface is divided into section segments according to the number branches that are attached to the considered junction. This is done by projecting the center node of each adjacent interface of a given junction on associated interface planes (see dotted lines in Fig. 10 (b)), which allows creating vectors (identified as $\vec{U}_i^1$ and $\vec{U}_i^2$ in Fig. 10 (b)) that indicate the location of every segment section. Then each angle between two of these vectors is bisected, which define the projection range of each segment section as illustrated in the zoom of Fig. 10. Interpolation points are computed for each section segment in a similar way as it is done for branch cross-sections. The only difference here

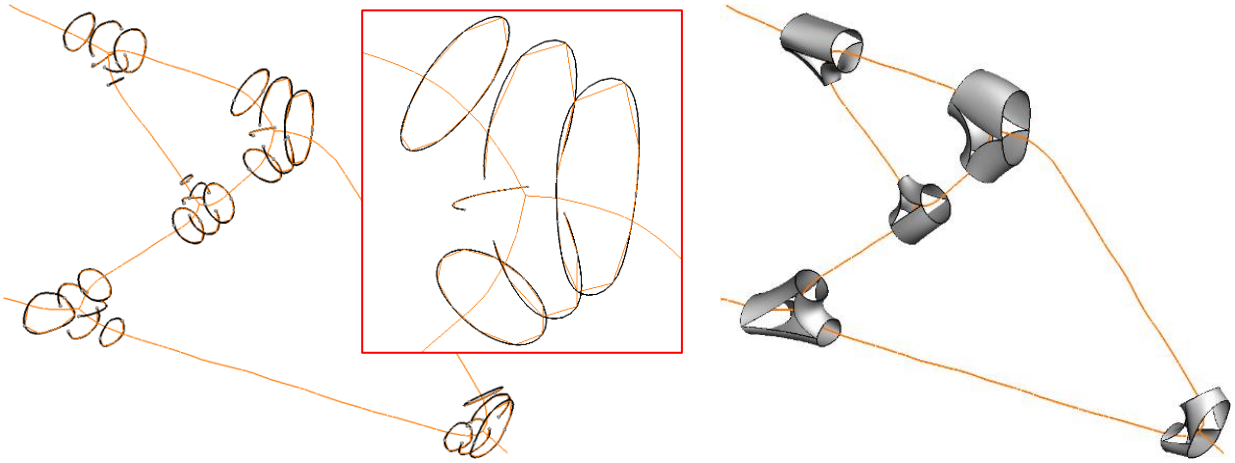
is the range of projections, which depends on the number of section segments in the considered interface. In Fig. 10, every segment section is defined with 6 interpolation points. However, the user can specify any number of interpolation points, since it is greater or equal to 3. Note that all interpolation points of all section segments computed for a given interface are contained in the same plane, which is normal to the direction vector ($\vec{d}_i$ in the example shown in Fig. 10 (b)) of the corresponding limit node (n_i in the example shown in Fig. 10 (b)).



(a) **(b)**

Fig. 10. Calculation of section segments for junction/branch interfaces

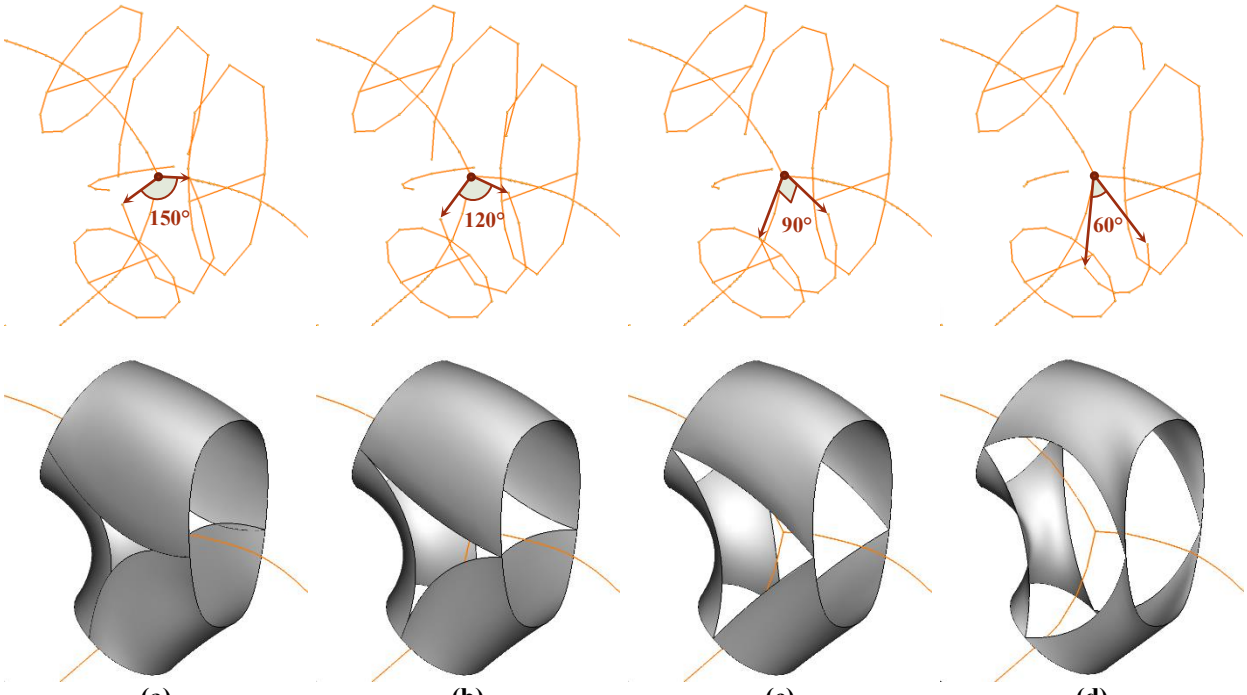
Two section segments are used to connect two branches. Between these two section segments, an intermediate section segment is computed in a similar way, with the same number of interpolation points used in the junction considered. Section segments and intermediate section segments associated with each junction are interpolated by cubic B-spline fitting curves . In this case, the number of control points used is 8 and the number of knots is 12. It should be noted here that a G_1 continuity condition is applied between B-spline curves interpolating section segments of the same junction/branch interface as illustrated in Fig. 11 (a). From these curves, a connecting B-spline surface between each pair of section segments is generated through a lofting operation as shown in Fig. 11 (b). In this case, three lofting operations are performed for each junction.



(a) (b)

Fig. 11. Generation of connecting B-spline surfaces

When generating intermediate section segments, projections of the junction node on the boundary triangulation can be made using various angular ranges as illustrated in Fig. **12**. As it can be clearly seen in Fig. 12, the angular range of projection used for intermediate segment sections computation might lead to overlapping surfaces like in case (a) or leave wide openings in the junction like in case (d), which makes that a good adjustment of this angular projection range may be required.

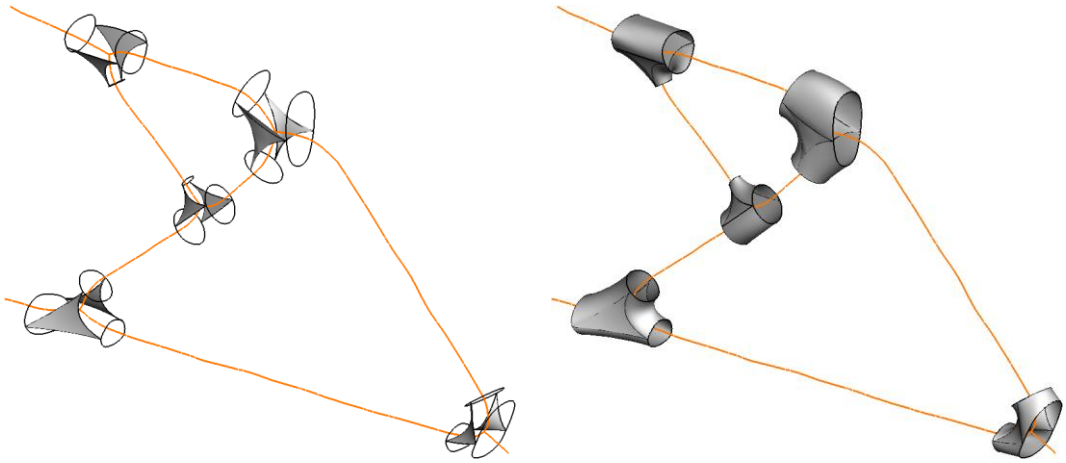


(a) (b) (c) (d)

Fig. 12. Projection range for intermediate section segments : (a) 150°, (b) 120°, (c) 90° and (d) 60°

As seen in Fig. 11 (b), once all connecting surfaces are generated, openings remain between these connecting surfaces. These openings are closed with B-spline filling surfaces as shown in

Fig. 13 (a). Sewing these filling surfaces to connecting surfaces for each junction provides a complete surface representation of each junction as illustrated in Fig. 13 (b).

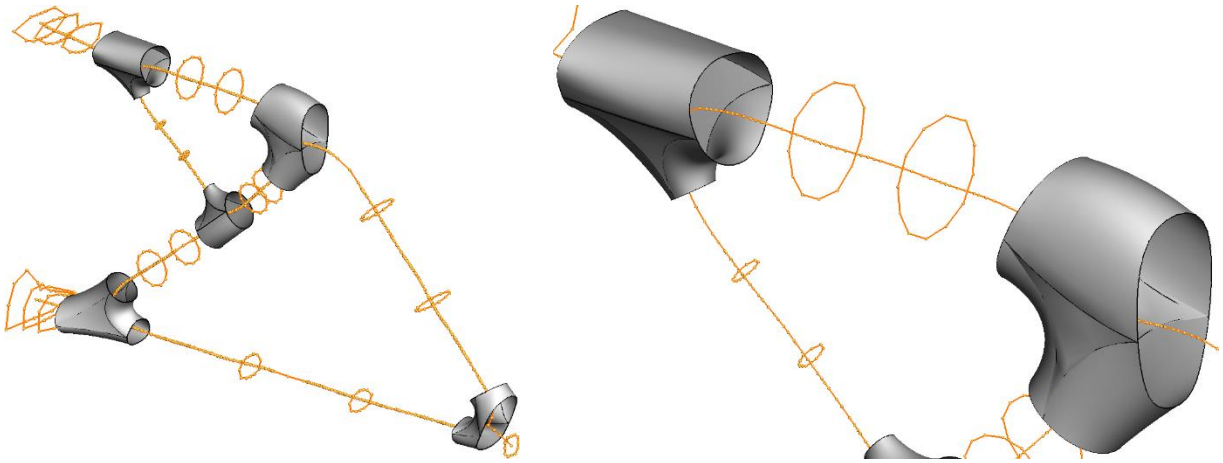


(a) **(b)**

Fig. 13. Junction reconstruction (a) generation of filling surfaces and (b) junctions reconstructed surfaces

4.5. Automatic generation of surfaces for branches

Cross-sections initially calculated as sets of points in section 4.3 are interpolated by closed cubic B-spline curves as shown in Fig. 14 (b). In this case, the number of control points used is 13 and the number of knots is 19. These cross-section B-spline curves along each branch are selected along with corresponding junction interfaces curves to generate, using a loft operation, a bicubic B-spline surface for each branch as illustrated in Fig. 14 (c). It is important to note that control points and knots for both B-spline curves and surfaces are automatically defined. As it can be seen in Fig. 14 (c), the section curves corresponding to design/non-design interfaces are filled with ruled surfaces.



(a)

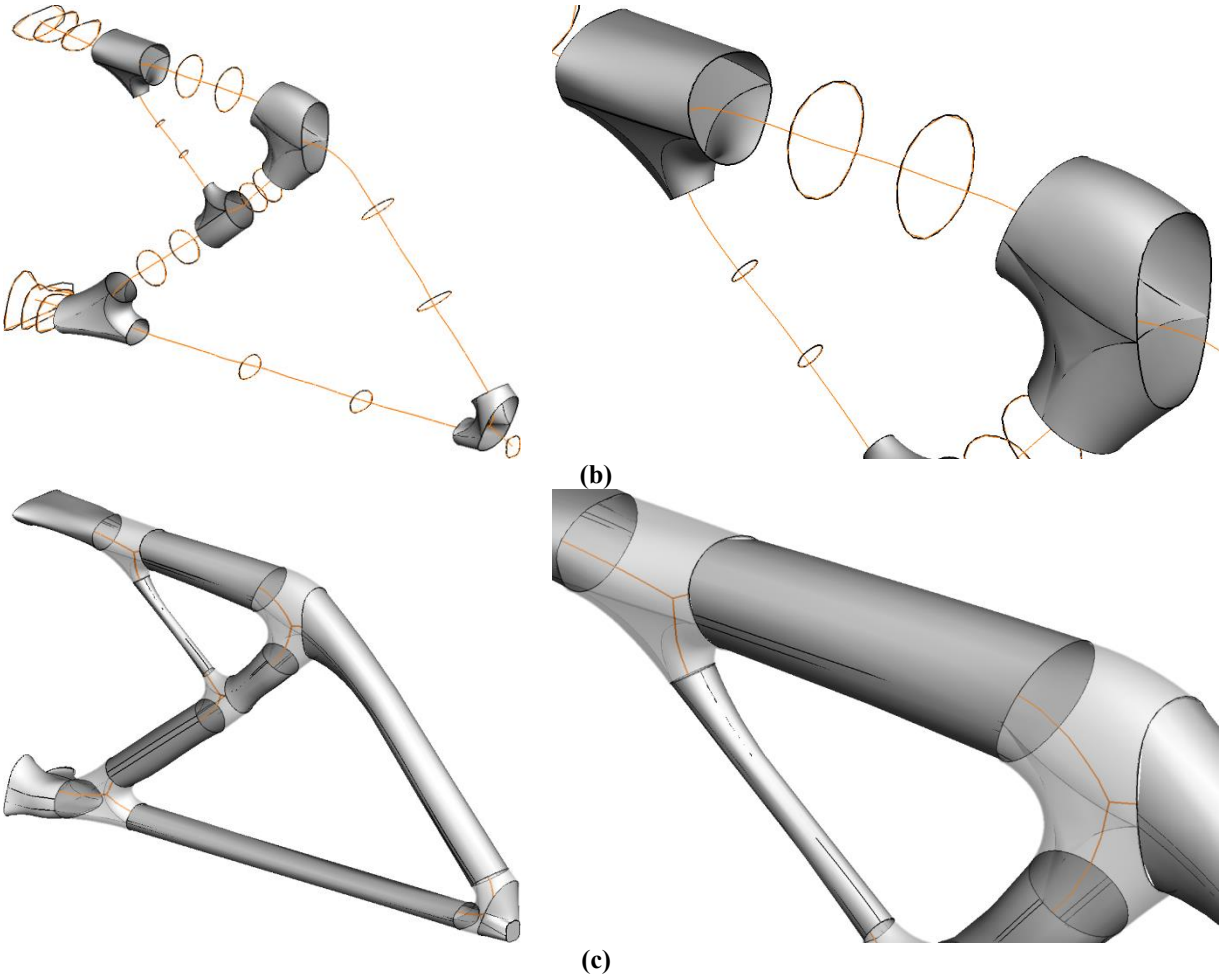
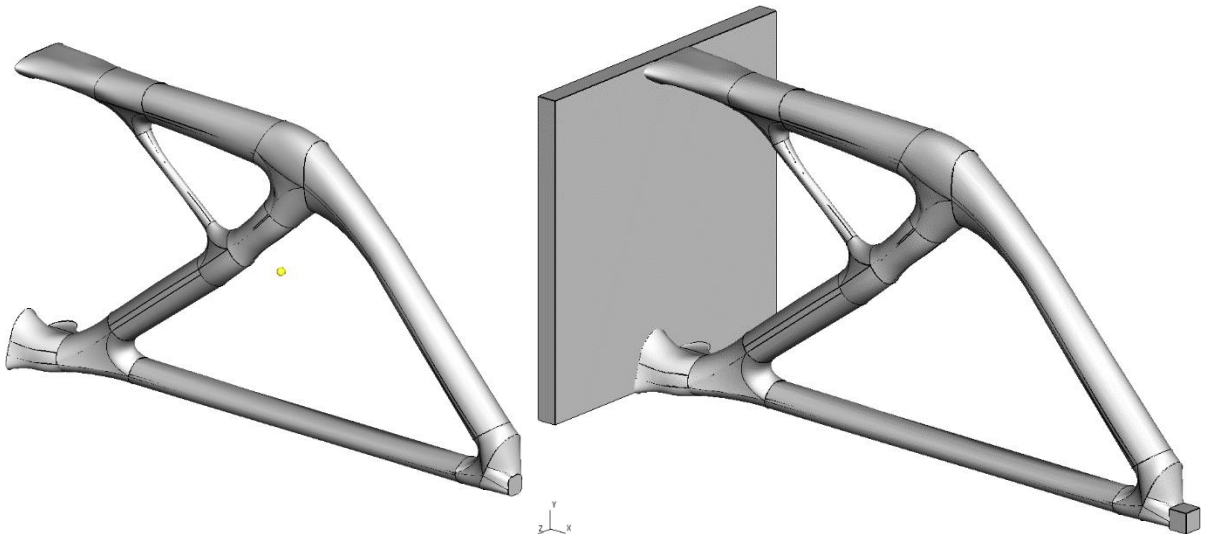


Fig. 14. Branches surfaces generation : (a) cross-sections interpolation points, (b) cubic B-spline curves interpolation and (c) branches lofting surfaces

4.6. Automatic construction of a solid model

The last step in the reconstruction algorithm consists in sewing all surfaces generated for junctions and branches to build a closed and watertight surface boundary of the reconstructed shape. Although all cross-sections and section segments are G_1 continuous, it is important to note that only G_0 continuity is ensured between surface patches. This closed and watertight surface model is then filled and converted into a solid model (see Fig. 15 (a)). This solid model represents the optimized design material, which is finally combined with that of the non-design subdomain using the union Boolean operator. Union between design and non-design solid models results in a reconstructed CAD model of the optimized part as shown in Fig. 15 (b).

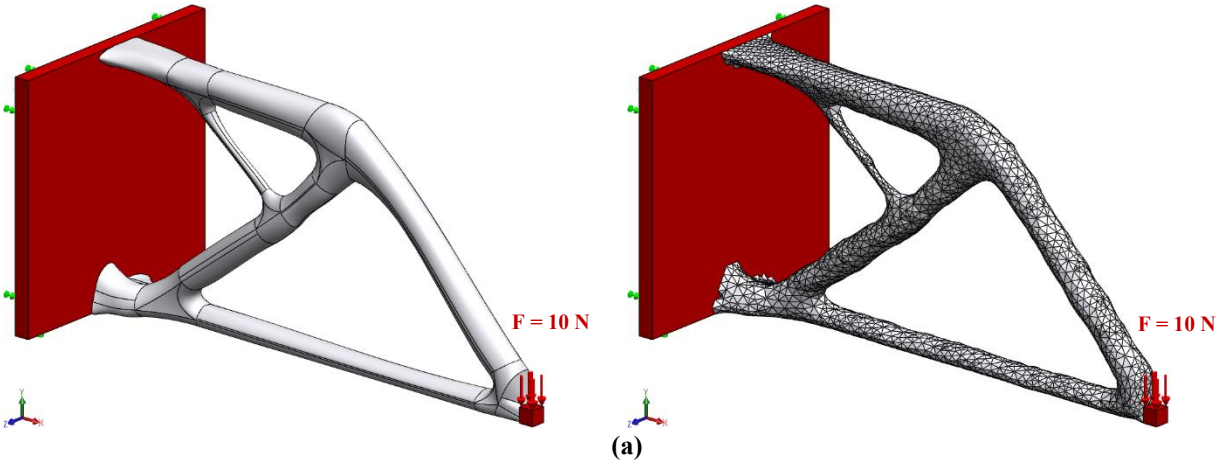


(a) (b)

Fig. 15. Reconstructed CAD model (a) without and (b) with non-design geometry

5. FEA results and validation

In order to validate this reconstructed CAD model, FEA is carried out on this reconstructed model and compared with FEA directly applied on the optimized shape generated by TO as a triangulation. Boundary conditions (BCs) applied in these two FEA models (see Fig. 16 (a)) are the same as BCs considered along the TO process itself with the SIMP method (see Fig. 1 (a)). This also applies for material data. An automatic h-refinement mesh adaptation scheme is used to reduce the total strain energy error to less than 3%. von-Mises stress (in Pa) and resultant displacement (in mm) distributions issued from both FEA models are respectively displayed in Fig. 16 (b) and (c) with the same color scale and with the same deformation factor to facilitate comparison between the two sets of FEA results.



(a)

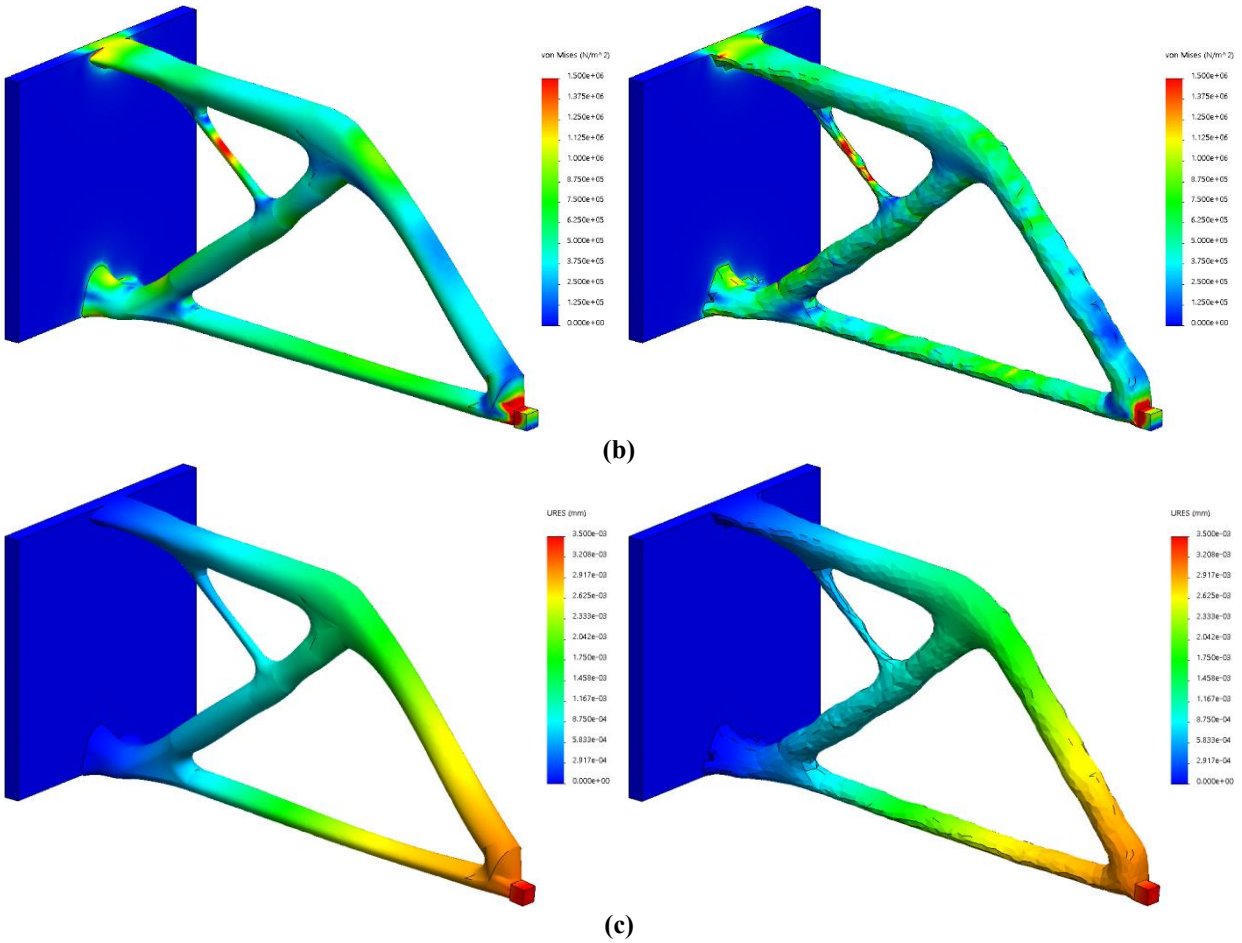


Fig. 16. For both the reconstructed CAD model (left) and the optimized shape (right) of the cantilever case: (a) BCs (b) von-Mises stress distribution (in Pa) and (c) resultant displacement distribution (in mm) on the deformed shape (deformation scale factor = 1000)

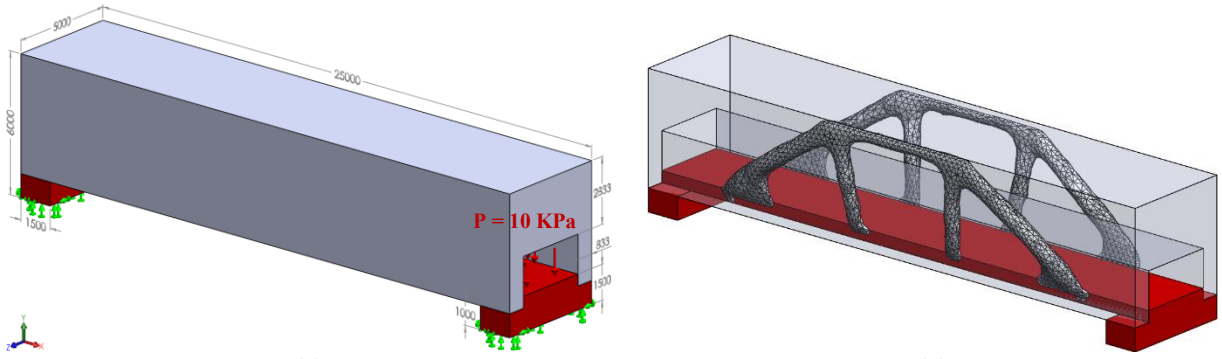
A deeper comparison between stress distributions for the two models is made using sets of stress probes that are located in the middle of all skeletal junctions. The idea is to measure and compare von-Mises stress values between the two FEA models in specific and representative locations. An average value of these von-Mises junction stresses is calculated for each model. Average junctions stresses for the reconstructed CAD model and the optimized shape are respectively 0.31 and 0.28 MPa, which makes a 11% relative difference. The difference between these two models for both the global compliance and the maximum resultant displacement is around 4%. There is only a 1% difference between design volumes of the two models. Globally, these results reveal a great agreement between the reconstructed CAD model and the optimized shape from which it is generated for this example (cantilever).

6. Case studies

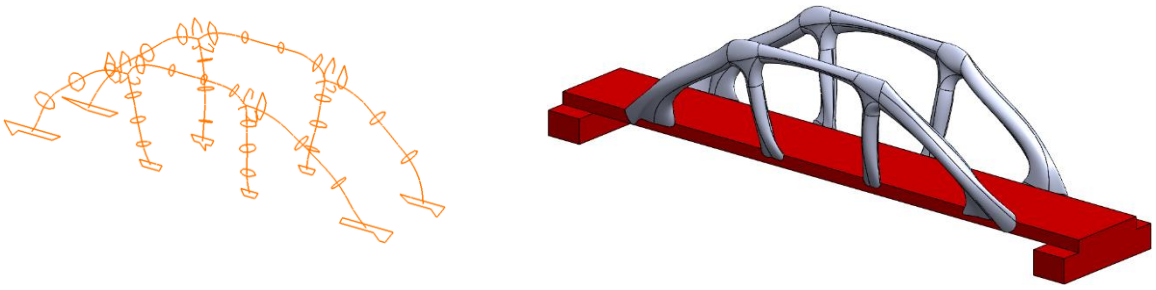
The reconstruction process described in previous sections is validated through several case studies that tend towards beam-like structures. These cases are processed with a research platform developed by our team, which is based on the Unified Topological Model [52], on Code_Aster [32] for implementing the SIMP topology optimization method and on OpenCascade [53] as geometric modeling kernel. This section presents results obtained on three cases : a bridge, a L-shaped bracket and a second cantilever beam. It is worth mentioning that for all case studies presented in this paper, the same material is used (Young modulus is 69 GPa and Poisson's ratio is 0,33) and, as mentioned in section 3.1, the same convergence criterion is applied for TO with the SIMP method ($\Delta_{\text{conv}} = 0.5\%$). Beam-like TO results are obtained using low volume fractions (3% or 4% depending on the case study considered). Like for the cantilever example presented in previous sections, FEA is performed on both models (reconstructed CAD and optimized shape) for validation. Table 1 lists parameters used for curve-skeletonization and geometry reconstruction, while Table 2 provides statistics for curve-skeletons obtained.

6.1. Bridge

The first example considered in this section is a bridge. The initial TO problem with initial design material (colored in gray), non-design material (colored in red), dimensions (in millimeters) and BCs applied are provided in Fig. 17 (a). TO analysis using the SIMP method is carried out. The volume fraction is $f = 4\%$ and the global average element size for mesh generation is $d_g = 275\ mm$. Convergence of SIMP optimization is reached after 23 iterations. By thresholding the final relative density distribution with $\rho_{th} = 0.25$ and smoothing the resulted rough triangulation as explained in section 3.2, we obtain a smooth triangulation that represents the optimized shape as shown in Fig. 17 (b). The result of curve-skeletonization applied to this smoothed boundary is illustrated by Fig. 17 (c) along with the calculated cross-sections. In this case, 4 cross-sections are generated for each branch using 9 interpolation points per section. Lofting operations through sets of these cross-sections lead to the final CAD model shown in Fig. 17 (d).



(a) **(b)**



(c) (d)

Fig. 17. Bridge (a) initial problem with dimensions (in mm) and BCs (b) smoothed boundary of the optimized shape given by TO (c) processed skeleton with cross-sections and (d) reconstructed CAD model

As presented in the previous section, FEA is carried out on both the reconstructed CAD model and the optimized shape for validation purposes. von-Mises stress and resultant displacement distributions for both models are respectively displayed in Fig. 18 (b) and (c) with the same color scale and the same deformation scale factor. Average junctions stresses computed for the two models are very close (3% difference). However, the reconstructed CAD solid model is slightly more flexible if compared to the optimized shape since maximum resultant displacements are 0.9 and 0.7 mm respectively.

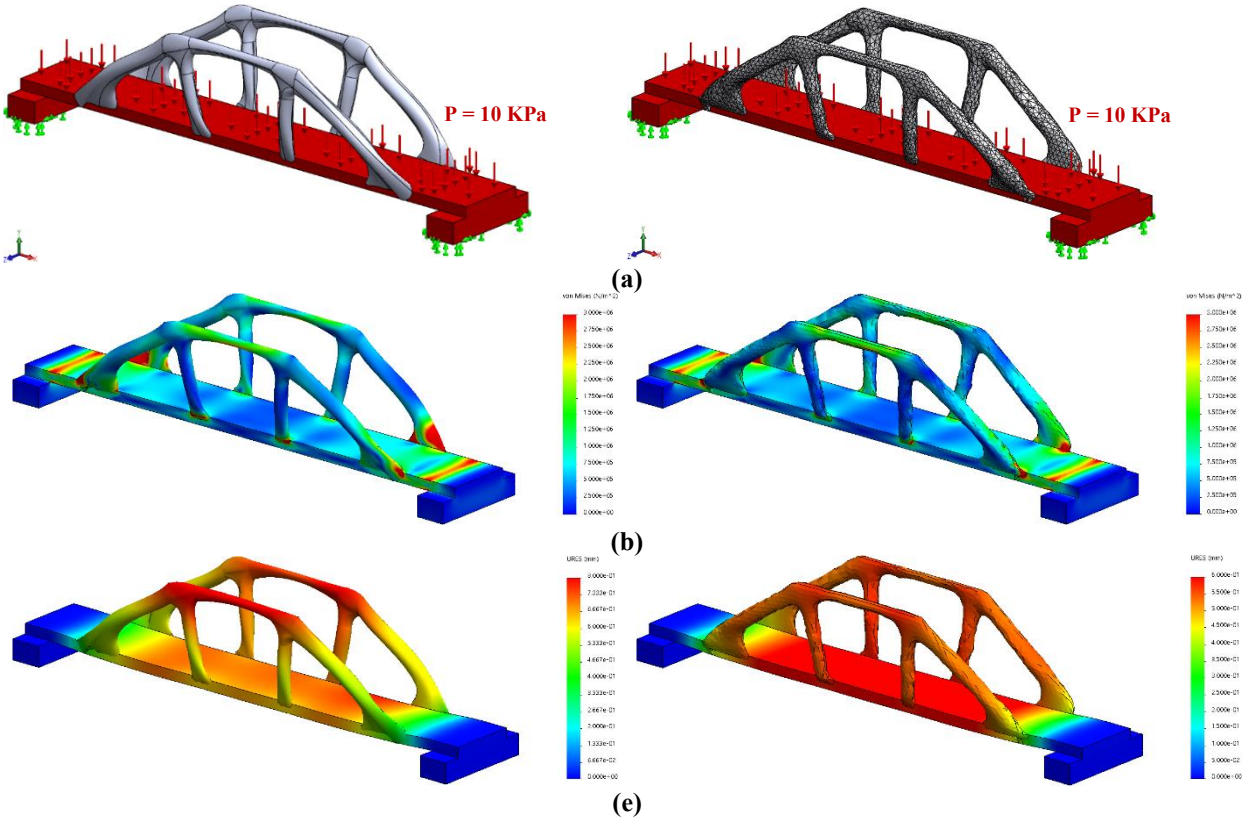
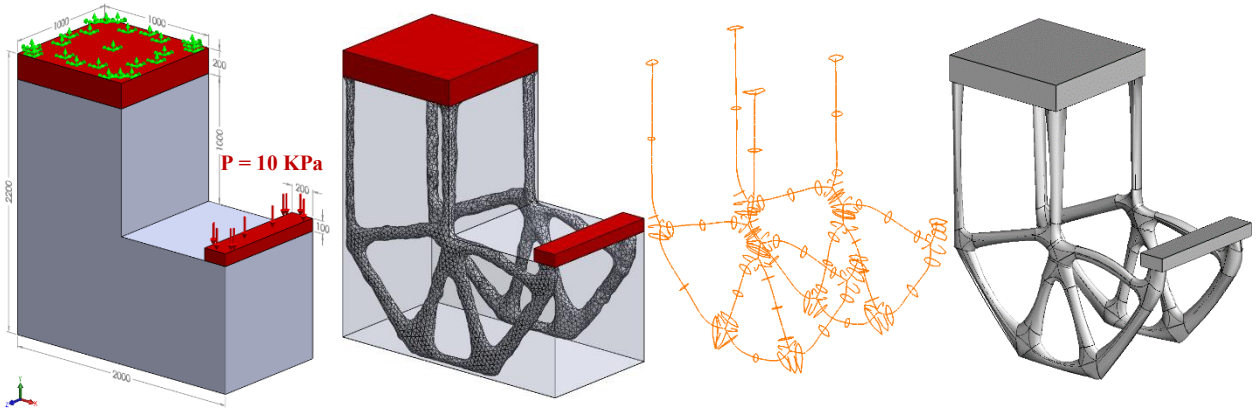


Fig. 18. For both the reconstructed CAD model (left) and the optimized shape (right) of the bridge case: (a) BCs (b) von-Mises stress distribution (in Pa) and (c) resultant displacement distribution (in mm) on the deformed shape (deformation scale factor = 300)

6.2. L-bracket

The second example in this section is an L-shaped bracket. Starting from the initial CAD model presented in Fig. 19 (a) on which design (colored in gray) and non-design (colored in red) subdomains are specified as well as BCs and material data, SIMP TO analysis is performed with a target volume fraction $f = 3\%$ and a global average element size for mesh generation $d_g = 40\,mm$. The final relative density distribution is obtained after 30 SIMP iterations. A rough triangulation representing the optimized shape is then extracted with respect to relative density threshold $\rho_{th} = 0.22$. Fig. 19 (b) provides the optimized shape after smoothing operations. The curve-skeleton extracted from the optimized shape after smoothing and processed as explained in section 3.3 is illustrated in Fig. 19 (c). This same figure also shows computed cross-sections for branches and junctions. In this case, 3 cross-sections are calculated for each branch using 8 interpolation points, while 7 interpolation points are used to compute each junction intermediate cross-section. By following all steps described in section 4, a CAD solid model is generated from TO results as demonstrated in Fig. 17 (d).



(a) **(b)** **(c)** **(d)**

Fig. 19. L-bracket (a) initial problem with dimensions (in mm) and BCs (b) TO processed result (c) processed skeleton with computed cross-sections and (d) reconstructed CAD model

In order to compare the reconstructed CAD model generated by our process (Fig. 19 (d)) to the optimized shape obtained by TO (Fig. 19 (b)), FEA is performed on these two models using the same material and BCs (Fig. 20 (a)). Results obtained from these FEA are almost the same for both models. Differences between results obtained with these two models in terms of global compliance, average junctions stress and maximum resultant displacement are 3%, 4% and 1% respectively. Fig. 20 (b) and (c) display von-Mises stress and resultant displacement distributions respectively using the same color scale and the same deformation scale factor.

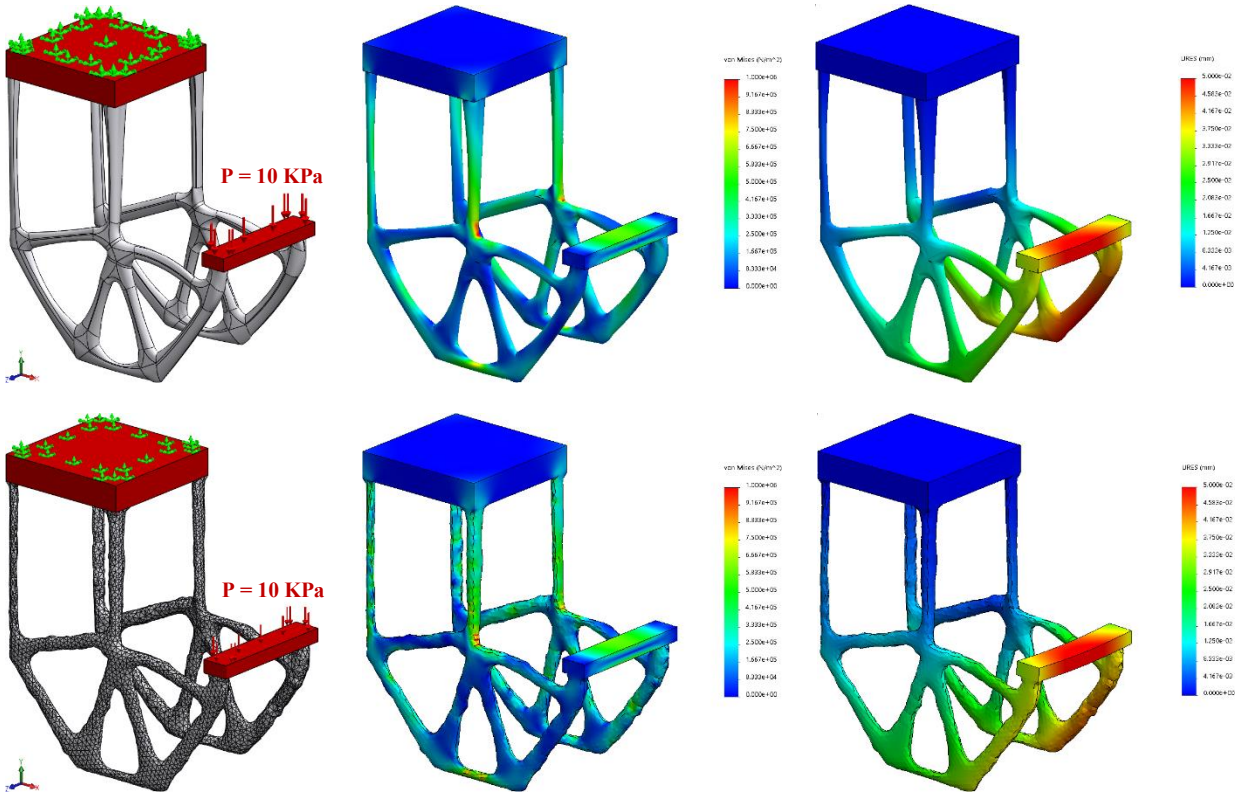


Fig. 20. For both the reconstructed CAD model (above) and the optimized shape (below) of the L-bracket case: (a) BCs (b) von-Mises stress distribution (in Pa) and (c) resultant displacement distribution (in mm) on the deformed shape (deformation scale factor = 1000)

6.3. Second cantilever

The last example considered in this work is a second version of the cantilever introduced in section 3. In this new case, the only modification, with respect to first version, is made on the non-design subdomain (see Fig. 21 (a)). As shown in Fig. 21 (b), a more complex optimized shape is derived from TO after 23 SIMP iterations. In this case, the objective volume fraction is $f = 4\%$ and the global average element size for mesh generation is $d_g = 1.6\,mm$. The relative density threshold used to extract a rough triangulation of the optimized shape is $\rho_{th} = 0.4$. The curve-skeleton generated from the smooth optimized shape along with computed cross-sections are provided in Fig. 21 (c). For this example, 4 cross-sections per branch are calculated using 6 interpolation points while intermediate cross-sections for junctions are calculated using 7 interpolation points. Lofting operations through these cross-section curves in addition with filling surfaces, lead to the final CAD model demonstrated in Fig. 21 (d).

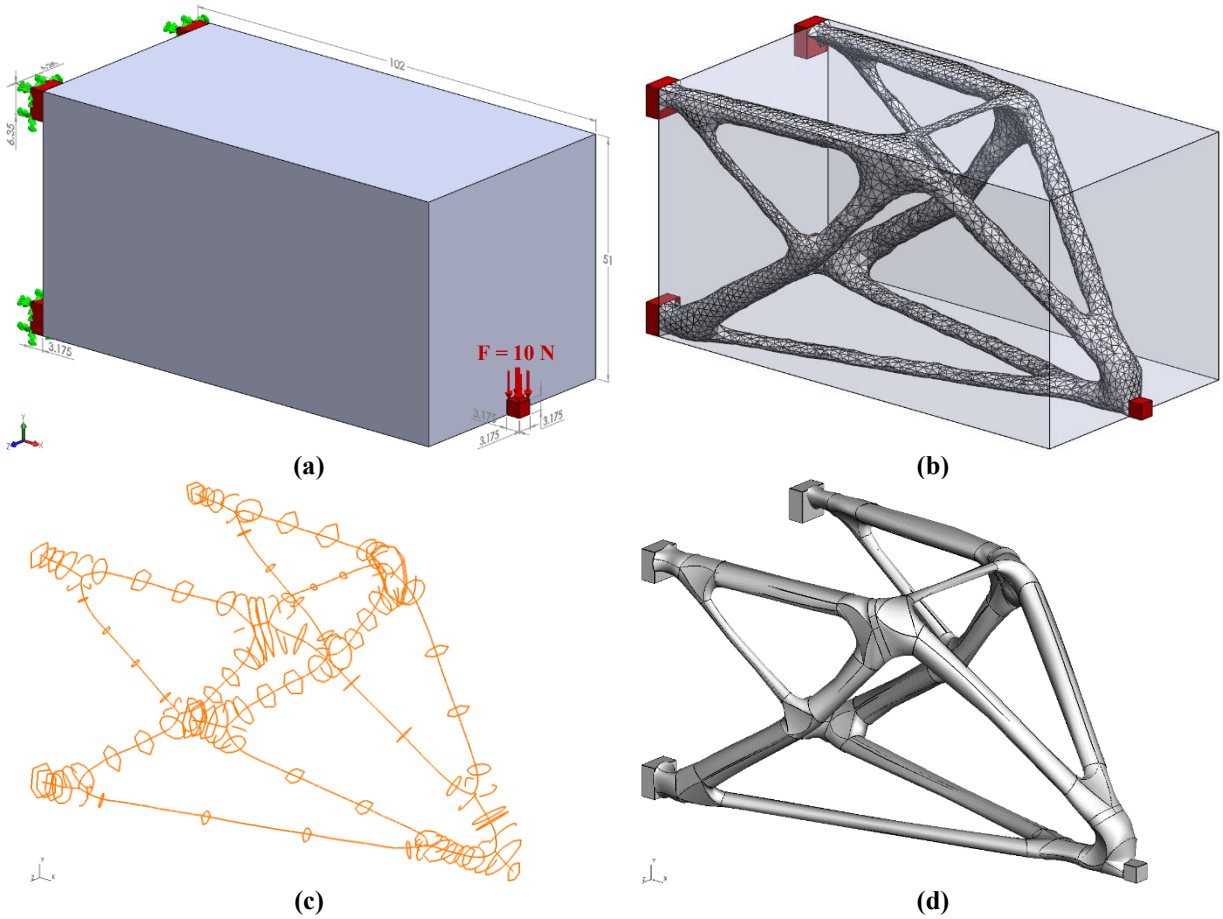
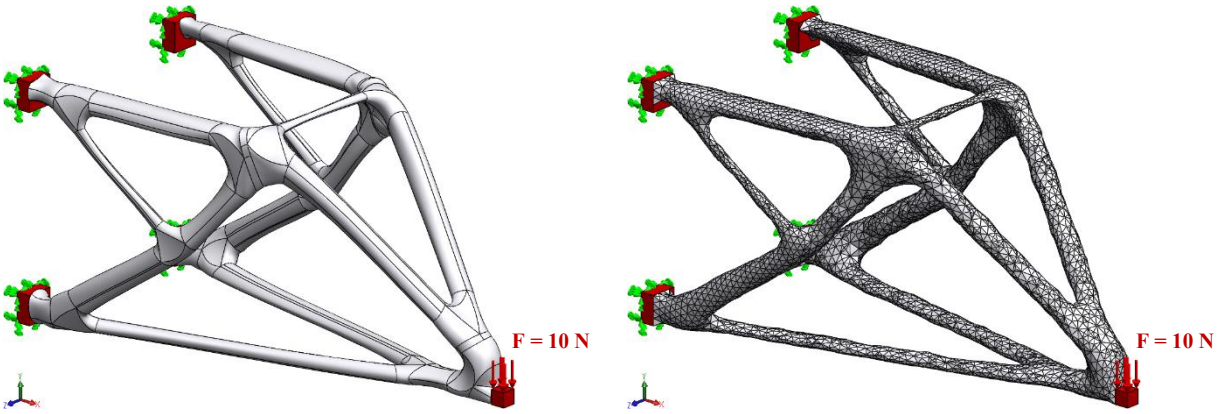


Fig. 21. Second cantilever (a) initial problem with dimensions (in mm) and BCs (b) TO processed result (c) skeleton with cross-sections and (d) reconstructed CAD model

Fig. 22 (b) and (c) provide von-Mises stress and resultant displacement distributions for the reconstructed CAD model (left) and the optimized shape (right) with the same color scale and the same deformation scale factor. Differences between these two models in terms of global compliance and maximum resultant displacement are 2% and 3% respectively. The reconstructed CAD model shows a higher average stress junctions if compared with average stress junctions computed for the optimized shape (0.22 MPa versus 0.19 MPa), but this difference remains acceptable.



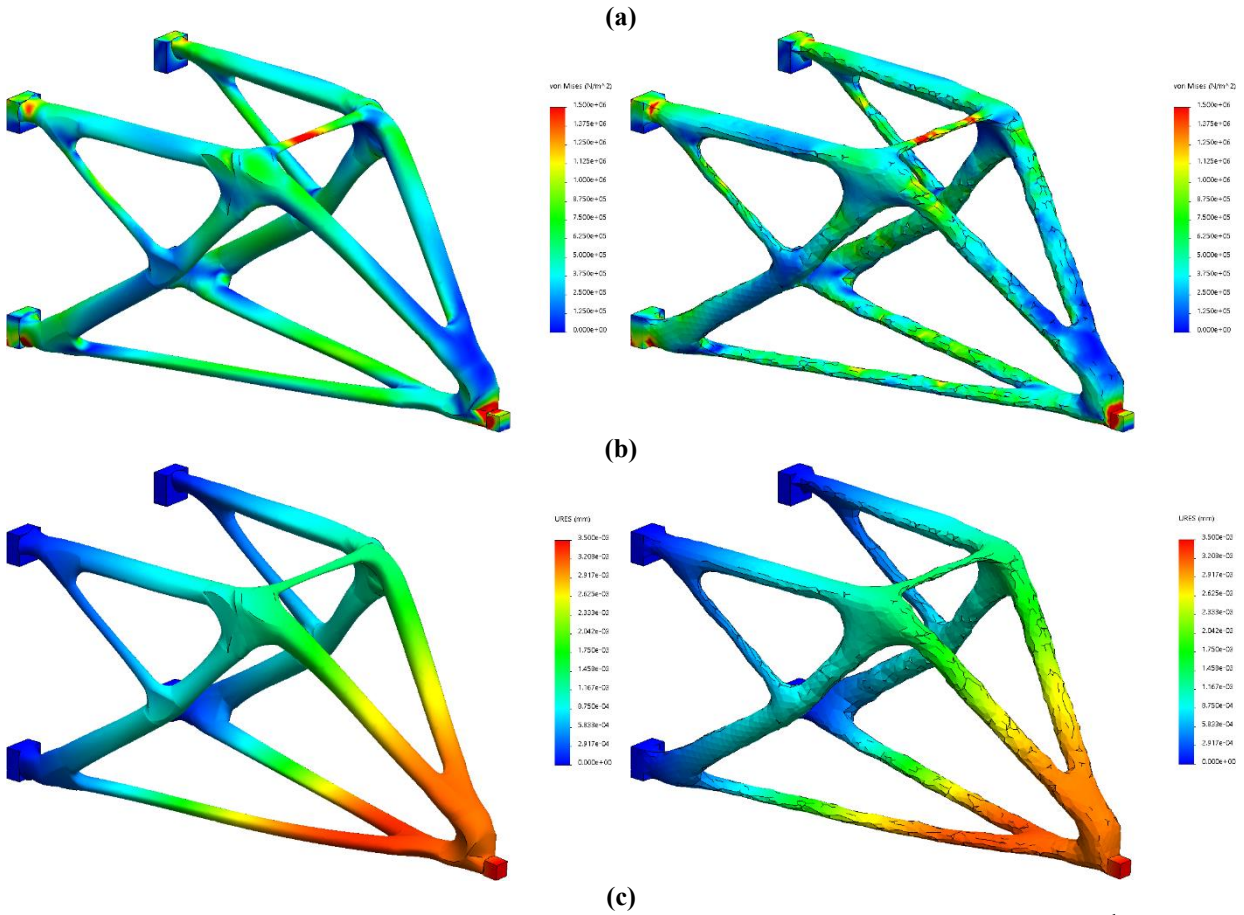


Fig. 22. For both the reconstructed CAD model (left) and the optimized shape (right) of the 2nd cantilever case: (a) BCs (b) von-Mises stress distribution (in Pa) and (c) resultant displacement distribution (in mm) on deformed shape (deformation scale factor = 3)

6.4. Summary of results

Table 3 presents a synthesis of main results obtained for all examples proposed in this paper. This table includes data and results for SIMP optimizations, optimized shape models and reconstructed CAD models. It shows a good agreement between results of reconstructed CAD models and results of corresponding optimized shapes with negligible differences. This consistency in results confirms effectiveness of the method presented in this paper towards automatically converting beam-like TO results into CAD models. For all cases presented, the optimized shape obtained from TO is closely approximated by the reconstructed CAD model generated from it. Indeed, differences between design volumes obtained for the two models are very low (2% or less for all cases except for the bridge which reaches 6%). It is worth mentioning that all examples presented in this paper, except the second cantilever beam, are taken from previous work made and published by our team [25].

Table 1. Skeletonization and reconstruction parameters

		Bridge	Cantilever	L Bracket	Cantilever 2
Skeleton	Parameter ω_L	1	1	1	1
	Parameter ω_H	0.8	0.8	0.3	0.5
	Parameter ω_M	1	1	0.3	0.6
Branch	Nb of cross-sections	4	4	3	4
	Nb of interpolation points / section	9	12	8	6
Junction	Junction distance factor	1.9	1.3	1.85	1.45
	Nb of interpolation points / section	5	6	7	7
	Projection range for intermediate cross-sections	90°	120°	80°	80°

Table 2. Curve-skeletons characteristics

	Bridge	Cantilever	L Bracket	Cantilever 2
Nb of branches	10	9	24	22
Nb of junction points	4	5	14	13
Nb of skeletal nodes	551	558	1401	1415
Nb of skeletal segments	549	559	1405	1419

Table 3. Summary of results

	Bridge	Cantilever	L Bracket	Cantilever 2
	SIMP optimization			
Element size d_g (mm)	275	1.6	40	1.6
Nb of tetrahedrons N	179167	382148	345441	367284
Objective volume fraction f	4%	3%	3%	4%
Nb of iterations for convergence	23	21	30	23
Final compliance $\widetilde{C}$ (J)	$4.36\ 10^{2}$	$7.95\ 10^{-5}$	$2.07\ 10^{-1}$	$1.65\ 10^{-4}$
	Optimized shape			
Relative density threshold ρ_{th}	0.25	0.25	0.22	0.4
Volume fraction f_{3D}	4.08%	3.16%	2.95%	3.99%
Compliance C_{3D} (J)	$3.72\ 10^{2}$	$3.30\ 10^{-5}$	$8.78\ 10^{-2}$	$3.32\ 10^{-5}$
Error Estimate (APE)	3.70%	2.29%	1.93%	2.56%
Maximum Displacement δ_{3D} (mm)	$6.64\ 10^{-1}$	$3.56\ 10^{-3}$	$5.36\ 10^{-2}$	$3.56\ 10^{-3}$
Junctions average stress σ_{3D} (MPa)	$3.00\ 10^{-1}$	$2.78\ 10^{-1}$	$1.01\ 10^{-1}$	$1.94\ 10^{-1}$
Ratio f_{3D}/f	102%	105%	98%	100%

	Reconstructed CAD model			
Volume fraction f_{CAD}	3.85%	3.12%	3.01%	3.93%
Compliance C_{CAD} (J)	4.14 10^{2}	3.42 10^{-5}	8.49 10^{-2}	3.40 10^{-5}
Error Estimate (APE)	3.78%	2.13%	2.00%	2.32%
Maximum Displacement δ_{CAD} (mm)	8.96 10^{-1}	3.73 10^{-3}	5.30 10^{-2}	3.66 10^{-3}
Junctions average stress σ_{CAD} (MPa)	3.08 10^{-1}	3.11 10^{-1}	1.06 10^{-1}	2.22 10^{-1}
Ratio f_{CAD}/f	104%	96%	100%	102%
Ratio f_{3D}/f_{CAD}	106%	101%	98%	101%
Ratio C_{3D}/C_{CAD}	90%	96%	103%	98%
Ratio δ_{3D}/δ_{CAD}	74%	96%	101%	97%
Ratio σ_{3D}/σ_{CAD}	97%	89%	96%	88%

7. Conclusions and perspectives

In this paper, we propose a fully automatic process for converting TO results, that tend towards beam-like structures, to CAD solid models. This work represents a significant contribution towards completely integrating TO as a common and powerful tool along the design process. Indeed, the proposed approach is fully automated without any user intervention, which is not the case, to the best of our knowledge, for almost all methods in the literature. Our process is mainly based on curve-skeleton extraction, which is used for cross-sections positioning and computation. Closed cubic B-spline curves are used to approximate cross-section contours. Lofting operations through series of these contours allow to create multi-section surfaces which are assembled to form, with the help of filling surfaces, a closed boundary. The conversion of this closed boundary into a solid CAD model completes the process. Effectiveness of our reconstruction process is demonstrated through four case studies where FEA is used to compare the reconstructed models to the optimized shapes from which they are derived. The generated CAD models can be easily edited and/or refined by shape optimization to meet further criteria. As can be seen in the flowchart of Fig. 3, volume fraction for SIMP TO could be automatically adjusted, according to FEA results, and then re-applying the whole process in order to satisfy allowable stress criteria. Our team is still working on this automatic stress-based optimization. However, the approach presented here shows limitations. At the time being, the approach is limited to beam-like TO result. It could successfully be applied to simple massive shapes, which represents a first step in the perspective of extending this approach to a wider variety of TO results. As mentioned in section 4.6, another limitation of the method proposed is that, although all cross-sections and section segments are constrained to be G_1 continuous, only G_0 continuity is ensured across connections between branches and junctions surfaces. Ensuring G_1 continuity everywhere is definitely feasible and would be a very interesting and useful improvement of the approach proposed here but it requires further work at

this point of our research. Applying the proposed process to results provided by other TO methods is also a forward-looking perspective for our team.

Acknowledgement

This study was carried out as part of a project supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and UQTR foundation.

References

1. Bendsøe, M.P. and N. Kikuchi, *Generating optimal topologies in structural design using a homogenization method.* Computer methods in applied mechanics and engineering, 1988. **71**(2): p. 197-224.

2. Bendsøe, M.P., *Optimal shape design as a material distribution problem.* Structural Optimization, 1989. **1**(4): p. 193-202.

3. Xie, Y.M. and G.P. Steven, *A simple evolutionary procedure for structural optimization.* Computers & structures, 1993. **49**(5): p. 885-896.

4. Querin, O.M., G.P. Steven, and Y.M. Xie, *Evolutionary structural optimisation (ESO) using a bidirectional algorithm.* Engineering Computations, 1998. **15**(8): p. 1031-1048.

5. Sethian, J.A. and A. Wiegmann, *Structural Boundary Design via Level Set and Immersed Interface Methods.* Journal of Computational Physics, 2000. **163**(2): p. 489-528.

6. Bourdin, B. and A. Chambolle. *The phase-field method in optimal design*. in *IUTAM Symposium on topological design optimization of structures, machines and materials*. 2006. Springer.

7. Deaton, J. and R. Grandhi, *A survey of structural and multidisciplinary continuum topology optimization: post 2000.* Structural and Multidisciplinary Optimization, 2014. **49**(1): p. 1-38.

8. Liu, J. and Y. Ma, *A survey of manufacturing oriented topology optimization methods.* Advances in Engineering Software, 2016. **100**: p. 161-175.

9. Cuillière, J.-C. and V. Francois, *Integration of CAD, FEA and Topology Optimization through a Unified Topological Model.* Computer-Aided Design & Applications, 2014. **11**(5): p. 493-508.

10. Marsan, A.L. and D. Dutta, *Construction of a surface model and layered manufacturing data from 3D homogenization output.* Journal of Mechanical Design, Transactions of the ASME, 1996. **118**(3): p. 412-418.

11. Chirehdast, M., *An integrated optimization environment for structural configuration design*. 1992, University of Michigan: U.S.A.

12. Tang, P.-S. and K.-H. Chang, *Integration of topology and shape optimization for design of structural components.* Structural and Multidisciplinary Optimization, 2001. **22**(1): p. 65-82.

13. Chang, K.H. and P.S. Tang, *Integration of design and manufacturing for structural shape optimization.* Advances in Engineering Software, 2001. **32**(7): p. 555-567.

14. Hsu, Y.-L., M.-S. Hsu, and C.-T. Chen, *Interpreting results from topology optimization using density contours.* Computers and Structures, 2001. **79**(10): p. 1049-1058.

15. Hsu, M.-H. and Y.-L. Hsu, *Interpreting three-dimensional structural topology optimization results.* Computers and Structures, 2005. **83**(4): p. 327-337.

16. Lin, C.Y. and L.S. Chao, *Automated image interpretation for integrated topology and shape optimization.* Structural and Multidisciplinary Optimization, 2000. **20**(2): p. 125-137.

17. Lin, C.Y. and Y.H. Chou, *Automated structural optimization system for integrated topology and shape optimization.* Journal of the Chinese Institute of Engineers, 2008. **31**(5): p. 745-756.

18. Chou, Y.-H. and C.-Y. Lin, *Improved image interpreting and modeling technique for automated structural optimization system.* Structural and Multidisciplinary Optimization, 2010. **40**(1-6): p. 215-226.

19. Larsen, S. and C.G. Jensen, *Converting topology optimization results into parametric CAD models.* Computer-Aided Design and Applications, 2009. **6**(3): p. 407-418.

20. Yi, G. and N. Kim, *Identifying boundaries of topology optimization results using basic parametric features.* Structural and Multidisciplinary Optimization, 2017. **55**(5): p. 1641-1654.

21. Koguchi, A. and N. Kikuchi, *A surface reconstruction algorithm for topology optimization.* Engineering with Computers, 2006. **22**(1): p. 1-10.

22. Park, J.M., et al., *Surface reconstruction from FE mesh model.* Journal of Computational Design and Engineering, 2019. **6**(2): p. 197-208.

23. Bremicker, M., et al., *Integrated Topology and Shape Optimization in Structural Design∗.* Journal of Structural Mechanics, 1991. **19**(4): p. 551-587.

24. Chirehdast, M., et al., *Structural configuration examples of an integrated optimal design process.* Journal Of Mechanical Design, 1994. **116**(4): p. 997-1004.

25. Cuillière, J.-C., V. François, and A. Nana, *Automatic construction of structural CAD models from 3D topology optimization.* Computer-Aided Design and Applications, 2017: p. 1-15.

26. Mandhyan, A., G. Srivastava, and S. Krishnamoorthi, *A novel method for prediction of truss geometry from topology optimization.* Engineering with Computers, 2017. **33**(1): p. 95-106.

27. Nana, A., J.-C. Cuillière, and V. Francois, *Automatic reconstruction of beam structures from 3D topology optimization results.* Computers and Structures, 2017. **189**: p. 62-82.

28. Yin, G., X. Xiao, and F. Cirak, *Topologically robust CAD model generation for structural optimisation.* Computer Methods in Applied Mechanics and Engineering, 2020. **369**: p. 113102.

29. Au, O.K.-C., et al. *Skeleton extraction by mesh contraction*. in *ACM Transactions on Graphics (TOG)*. 2008. ACM.

30. Cao, J., et al. *Point cloud skeletons via laplacian based contraction*. in *Shape Modeling International Conference (SMI), 2010*. 2010. IEEE.

31. Kresslein, J., et al., *Automated cross-sectional shape recovery of 3D branching structures from point cloud.* Journal of Computational Design and Engineering, 2018. **5**(3): p. 368-378.

32. code_aster. [cited 2017; Available from: https://www.code-aster.org/.

33. Cuillière, J.-C., V. Francois, and J.-M. Drouet, *Towards the integration of topology optimization into the CAD process.* Computer-Aided Design & Applications, 2014. **11**(2): p. 120-140.

34. Bendsøe, M.P. and O. Sigmund, *Topology optimization : theory, methods, and applications*. 2nd ed. corr. ed. 2003, Berlin, New York: Springer. xiv, 370 p.

35. Cuillière, J.-C., V. François, and J.-M. Drouet, *Automatic mesh generation and transformation for topology optimization methods.* Computer-Aided Design, 2013. **45**(12): p. 1489-1506.

36. Geuzaine, C. and J.F. Remacle, *Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities.* International Journal for Numerical Methods in Engineering, 2009. **79**(11): p. 1309-1331.

37. SolidWorks. [cited 2019; Available from: https://www.solidworks.com/.

38. Chen, C.-Y. and K.-Y. Cheng, *A direction-oriented sharpness dependent filter for 3D polygon meshes.* Computers & Graphics, 2008. **32**(2): p. 129-140.

39. Clark, B., N. Ray, and X. Jiao, *Surface mesh optimization, adaption, and untangling with high-order accuracy*. 2013. p. 385-402.

40. Taubin, G. *A signal processing approach to fair surface design*. in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. 1995. ACM.

41. Lu, X., et al., *An efficient approach for feature-preserving mesh denoising.* Optics and Lasers in Engineering, 2017. **90**: p. 186-195.

42. Saha, P.K., G. Borgefors, and G. Sanniti Di Baja, *A survey on skeletonization algorithms and their applications.* Pattern Recognition Letters, 2016. **76**: p. 3-12.

43. Cornea, N.D., D. Silver, and P. Min, *Curve-skeleton properties, applications, and algorithms*, in *IEEE Trans. Vis. Comput. Graph.* 2007. p. 530-548.

44. Ju, T., M.L. Baker, and W. Chiu, *Computing a family of skeletons of volumetric models for shape description.* Computer-Aided Design, 2007. **39**(5): p. 352-360.

45. Xie, W., R.P. Thompson, and R. Perucchio, *A topology-preserving parallel 3D thinning algorithm for extracting the curve skeleton.* Pattern Recognition, 2003. **36**(7): p. 1529-1544.

46. Hesselink, W.H. and J.B.T.M. Roerdink, *Euclidean Skeletons of Digital Image and Volume Data in Linear Time by the Integer Medial Axis Transform.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2008. **30**(12): p. 2204-2217.

47. Siddiqi, K., et al., *Hamilton-Jacobi Skeletons.* International Journal of Computer Vision, 2002. **48**(3): p. 215-231.

48. Reniers, D., J.J. van Wijk, and A. Telea, *Computing Multiscale Curve and Surface Skeletons of Genus 0 Shapes Using a Global Importance Measure.* Visualization and Computer Graphics, IEEE Transactions on, 2008. **14**(2).

49. Ogniewicz, R.L. and O. Kübler, *Hierarchic Voronoi skeletons.* Pattern Recognition, 1995. **28**(3): p. 343-359.

50. Tagliasacchi, A., et al. *Mean curvature skeletons*. in *Computer Graphics Forum*. 2012. Wiley Online Library.

51. Tagliasacchi, A., et al., *3D Skeletons: A State-of-the-Art Report.* Computer Graphics Forum, 2016. **35**(2): p. 573-597.

52. Cuillière, J.-C. and V. Francois, *Integration of CAD, FEA and Topology Optimization through a Unified Topological Model.* Computer-Aided Design and Applications, 2014. **11**: p. 493-508.

53. *OpenCascade*. Available from: www.opencascade.com.