**Advanced Modeling and Simulation
in Engineering Sciences**

RESEARCH ARTICLE

Open Access

3D B-Rep meshing for real-time data-based geometric parametric analysis

Tristan Maquart[1,2,3*] , Thomas Elguedj[1,2*], Anthony Gravouil[1,2] and Michel Rochette[3]

*Correspondence:
tristan.maquart@hotmail.fr;
thomas.elguedj@insa-lyon.fr
[1]Univ Lyon, INSA-Lyon, CNRS
UMR5259, LaMCoS, 69621 Lyon,
France
Full list of author information is
available at the end of the article

Abstract

This paper presents an effective framework to automatically construct 3D quadrilateral meshes of complicated geometry and arbitrary topology adapted for parametric studies. The input is a triangulation of the solid 3D model's boundary provided from B-Rep CAD models or scanned geometry. The triangulated mesh is decomposed into a set of cuboids in two steps: pants decomposition and cuboid decomposition. This workflow includes an integration of a geometry-feature-aware pants-to-cuboids decomposition algorithm. This set of cuboids perfectly replicates the input surface topology. Using aligned global parameterization, patches are re-positioned on the surface in a way to achieve low overall distortion, and alignment to principal curvature directions and sharp features. Based on the cuboid decomposition and global parameterization, a 3D quadrilateral mesh is extracted. For different parametric instances with the same topology but different geometries, the MEG-IsoQuad method allows to have the same representation: isotopological meshes holding the same connectivity where each point on a mesh has an analogous one into all other meshes. Faithful 3D numerical charts of parametric geometries are then built using standard data-based techniques. Geometries are then evaluated in real-time. The efficiency and the robustness of the proposed approach are illustrated through a few parametric examples.

Keywords: 3D quadrilateral meshes, Parametric geometry, Global parameterization, Data-based models

Introduction

Since few years, data-based models and reduced order modeling [10,11,47] have been particularly studied in many physics applications. Such approaches allows to determine complex solutions in real-time. By real-time we mean a computational cost drastically reduced in comparison with a classical approach. In most reduced order approaches, an offline step is first needed in order to build the database model. For parametric studies, this means constructing and running the model for given sets of parameters wisely chosen. In the online phase, executed in near real-time, this database is used to construct the solution of a new set of parameters. Common reduced order modeling techniques classically used in computational mechanics often require structured meshes to be efficient by avoiding inaccurate projection steps between all meshes. This is obviously encountered

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 2 of 28

when geometric parameters are considered. One way to circumvent this issue is to build isotopological meshes of all the geometric instances. Previous work has been done to build the foundations of our approach [2,3]. Moreover, research on the same problematic has been recently published [40–42]. For different geometric instances with the same topological properties (e.g. the same shape), being able to construct isotopological meshes of all these instances makes it very easy to compare these meshes. Consequently data-based approaches with geometric parameters are mainly guided by topological information throughout the offline step. Generating isotopological quad meshes therefore seems as the right way to solve this problem.

In numerous applications in domains like computational mechanics, aerodynamics, crash simulations and fluid-structure interactions, 3D quadrilateral meshes are recommended due to their highly regular structure. Generation of high quality quadrilateral meshes from B-Rep triangulated meshes has been recently extensively studied. All developed methods seek to construct such meshes with local and global specified properties. Most common desired properties are element quality, alignment with principal curvature directions or boundaries, global topology structure and respect of sharp features. Most recent techniques for quadrangulation use a global parameterization of the input triangulated surface [6,27,45,48]. Global parameterization is of particular interest for parameterizing the whole surface and handling discontinuities along seams. Constructing such global parameterization is usually done using a reliable field that serves as a guidance to drive the parameterization gradient. Cross fields can be used for that purpose, and in addition allow to capture principal curvature directions, boundaries and sharp features while understanding the topology. Different techniques exist for 2D or 3D quadrilateral meshes purposes [5,15,30,49]. Quadrilateral layouting (i.e., partitioning the triangulated mesh into generalized squares) is particularly efficient to build structured quadrilateral meshes [8,9].

In the present work, we seek to decompose objects defined by their triangulated boundary into geometry-aware quadrilateral layouts understanding geometric features as well as possible. This tedious task is first done using pants decomposition which was widely studied [12,21,34,53]. Secondly, we perform cuboid decomposition [32,35,36] on each pants patch that allow us to obtain a decomposition in quadrilateral patches. We present a novel decomposition method that splits automatically the surface into geometry-feature-aware patches. We use an optimization process of quadrilateral patches embedded on surfaces. An optimal global parameterization gives the final surface parameterization. Topological properties of the designed cross field is entirely provided from the layout. Our main contribution is a new way to design such quadrilateral layouts to ensure an optimized resulting quadrilateral patch decomposition of the surface suitable for isotopological meshing purposes.

With this generic setup, our method can be applied in automatic scenarios involving specific cuboid sketches per pant patch. These kind of automatic processes are really useful to compute the offline phase needed for parametric studies using reduced order modeling and data-based models.

The paper is organized as follows. We first extend the decomposition pipeline thought to understand the geometry in "Smart model decomposition" section. Then using global parameterization aligned with a cross field capturing geometric information, we extract feature-aligned quadrilateral meshes in "Direction field generation" and " Aligned global

Fig. 1 Parametric analysis problematic of different meshes with the same topology
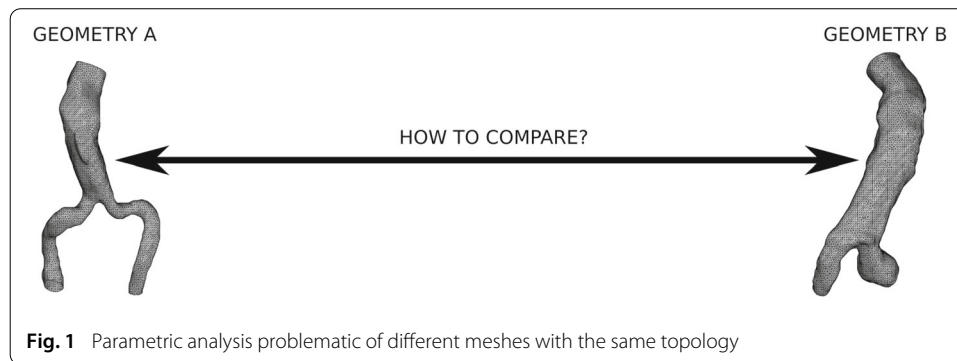
parameterization computation from geometric direction field" sections. Handling iso-topological datas throughout the 3D quadrilateral meshing process, geometric snapshots are used to build a reduced order model in a full autonomous integrated algorithm as presented in the various examples of "Application of the MEG-IsoQuad method togeometric parametric analysis" section.

Problem statement

In this article, we want to compute 3D quadrilateral meshes whose properties belong to a very restricted class. The goal is to be able to compare different shapes, i.e., different geometries of the same topology class. This kind of meshes are required in most parametric applications such as statistical shape analysis and reduced order modeling [19,24]. Figure 1 shows a graphical illustration of the problem. The most desirable properties of the targeted meshes are:
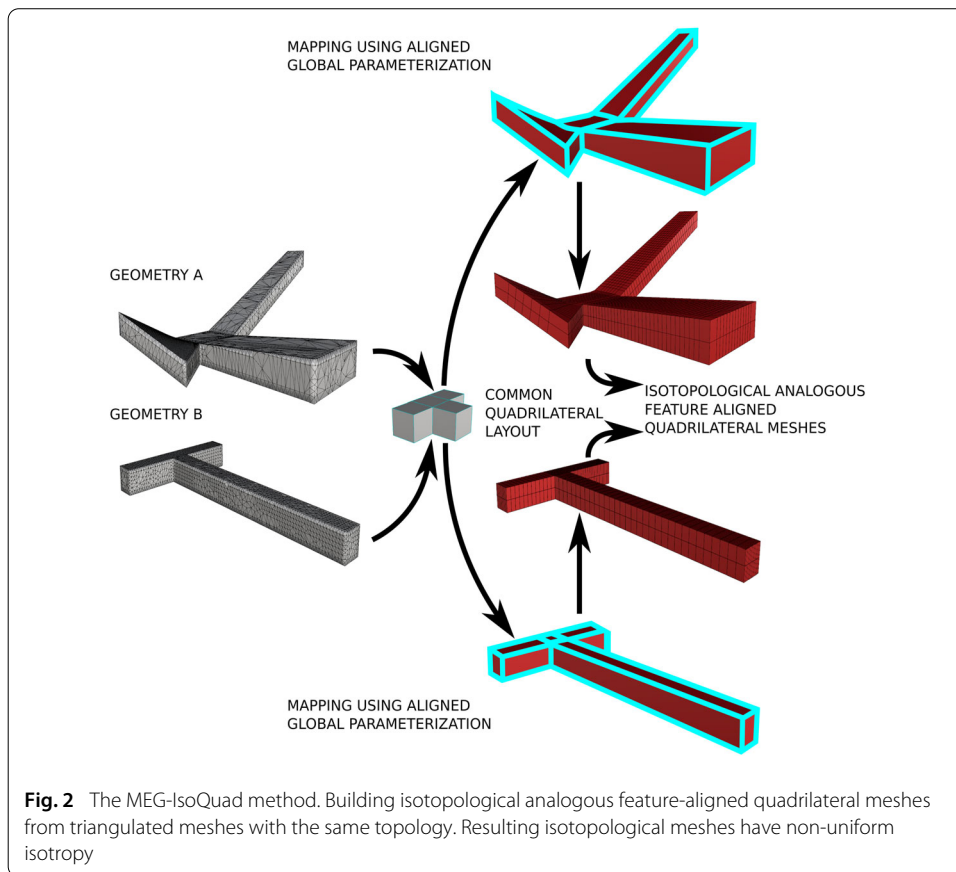
- Minimize the number of singularities, defined to be the nodes with a valency different from 4.
- Alignment with features, curvature directions and boundaries.
- Determine high quality elements, as close to a square as possible.
- Constrain the number of elements, connectivity and features locations to enable relevant comparison of all studied meshes.

Proposed approach

We give a workflow to compute high quality 3D quadrilateral meshes with specific properties needed in parametric analysis. Our goal is to generate such meshes for objects with complex geometry and arbitrary topology provided from boundary representation or scanned techniques. We want our framework to be as robust as possible.

We present an integrated pipeline partitioning the input triangulated surface to relevant domains until having comparable meshes for parametric studies. The first contribution consists in understanding the geometry while decomposing the mesh. The second contribution consists in computing an optimized aligned global parameterization using a very coarse quadrilateral layout helping us to define the required quadrilateral mesh.

Reduced order modeling is a tedious task that requires isotopological meshes in most cases. All geometric instances of the population must satisfy such mesh properties to be compared and studied. The issue of having isotopological meshes is addressed by

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 4 of 28

Fig. 2 The MEG-IsoQuad method. Building isotopological analogous feature-aligned quadrilateral meshes from triangulated meshes with the same topology. Resulting isotopological meshes have non-uniform isotropy

generating the same parameterization for all members of the given population. Figure 2 provides a simple understanding of our approach: the MEG-IsoQuad method.

Building isotopological analogous quadrilateral meshes: the MEG-IsoQuad method

In this section we present a method thought to build 3D quadrilateral instances from triangulated meshes suitable for data-based models with geometric parameters. This method allows us to identify the main shape variations. Quadrilateral mesh topology needs to be the same among all snapshots. We also strive to place vertices in a geometry-aware manner, i.e., all isotopological meshes have analogous points in the whole mesh. These constraints can be satisfied using our framework based on topology and geometry aspects.

Topology and parameterization prerequisites

Following prerequisites introduce briefly basic notions of geometry and topology that are needed to present our work. The interested reader unfamiliar with those concepts can find relevant detailed information in the cited references and in the corresponding literature.

Topology context and related work Topology is the study of properties like continuity, connectedness and boundaries of a space that are preserved under continuous deformations, such as bending and stretching, but not tearing and gluing. A *homeomorphism* is an *isomorphism* that admits a continuous function between two topological equivalent spaces that has a continuous inverse function. We are interested on transformations that

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 5 of 28

preserve all the topological properties of a given space. *Homeomorphic* spaces admit a *homeomorphism* between them, thus topological spaces are equivalent. See Hatcher [22] for more details.
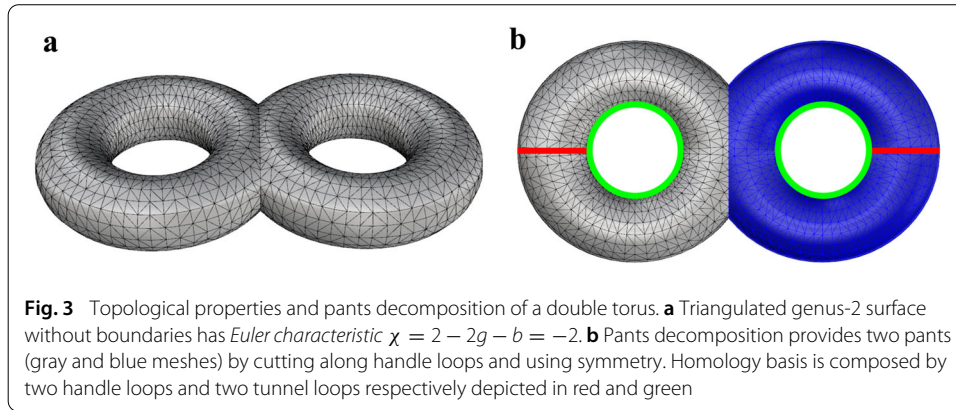
A surface M is a **2-manifold**, i.e., a topological space in which each point has a neighborhood *homeomorphic* to either the plane $\mathbb{R}^2$ or the closed half plane $\mathbb{R}^2_+$. Points with closed half-plane neighborhood are defined as the boundary ∂M of the surface M. In the following we investigate only connected, orientable, compact surfaces with boundaries. **Genus** of a connected and orientable surface M (i.e., a *2-manifold* embedded in $\mathbb{R}^3$) is the maximum number of non-intersecting closed curves which can be drawn on it without disconnecting the surface. With these definitions, taking into account a genus-g surface M possibly with b boundary components, we can now define the topological invariant. This invariant used to classify surfaces is called **Euler characteristic**: $\chi(M) = 2 - 2g - b$. If M is a triangulated surface with vertices V, edges E and faces F, we can define the characteristic as $\chi(M) = \dim(V) - \dim(E) + \dim(F)$. An example of *Euler characteristic* is provided in Fig. 3a. We note that surfaces with different *Euler characteristic* cannot be *homeomorphic*.

We define a **homology basis** for M to be any set of $2g$ *cycles* whose *homology classes* generate the first *homology group* H_1 [17]. Cutting along these cycles yields a genus-0 surface. The set of all *handle* and *tunnel* loops form a *homology basis*. Suppose a closed surface $M \subset \mathbb{R}^3$ separates $\mathbb{R}^3$ into a bounded space $\mathbb{I}$ and an unbounded space $\mathbb{O}$. *Handle* and *tunnel* loops on M can be defined as follows. A loop a_i is a *tunnel* if it spans a disk in the unbounded space $\mathbb{O}$. A loop b_i is a *handle* if it spans a disk in the bounded space $\mathbb{I}$.

Pants decomposition was studied in Hatcher et al.[23], subsequent work has been done to find the optimal segmentation of a given surface into relevant *pants* patches [12] using the shortest homology basis [26]. Geometry-aware *pants decomposition* has been also investigated by Zhang and Li [53]. Li et al. [34] developed a *pants decomposition* framework for computing maps between surfaces with arbitrary topologies. An example of pants decomposition using a homology basis can be seen in Fig. 3b. More recently, Hajij et al. [21] tried to segment surfaces into *pants* using a *morse* function.

Let $M_{g,b}$ be a surface of genus g with b boundary components. A *pants decomposition* of $M_{g,b}$ is a collection of pairwise disjoint simple cycles that splits the surface into *pants* patches. Each *pants* patch is a genus-0 surface (topological sphere) with 3 boundaries. We assume that M is a surface with negative *Euler characteristic*, i.e., M is none of the surfaces $M_{0,0}$ (topological sphere), $M_{0,1}$ (topological disk), $M_{0,2}$ (topological cylinder) and $M_{1,0}$ (topological torus). In this case *pants decompositions* of M do exist, and each *pants decomposition* consists of $3g + b - 3$ curves and divides M into $2g + b - 2$ *pants* patches.

A **direction field** defined on a surface M is a tangent unit vector field: at each point of the surface, there exists a direction u such that $\|\mathbf{u}\| = 1$ and $\mathbf{u}.\mathbf{n} = 0$, where $\mathbf{n}$ is the normal of M. A *N-symmetry direction field* $\mathcal{U}$ is a multivalued *direction field*: at each point of the surface M, there exists a *N-symmetry direction* $\mathbf{u}$ which is a set of N directions $\{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_{N-1}, \mathbf{u}_N\}$ preserved by rotations of $2\pi/N$ around the normal $\mathbf{n}$ of M. These fields are subject to the well known **Poincaré-Hopf theorem** for a *2-dimensional manifold* M. If M has boundaries, the vector field must be pointing in the outward normal direction along them (for higher symmetries, see the **boundary number theorem** [49]).

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 6 of 28

Fig. 3 Topological properties and pants decomposition of a double torus. **a** Triangulated genus-2 surface without boundaries has *Euler characteristic* $\chi = 2 - 2g - b = -2$. **b** Pants decomposition provides two pants (gray and blue meshes) by cutting along handle loops and using symmetry. Homology basis is composed by two handle loops and two tunnel loops respectively depicted in red and green

Used mapping techniques

Different approaches of mapping exist. In our case, we are interested in mappings or **parameterizations** which map a surface M embedded in $\mathbb{R}^3$ to a canonical domain D in $\mathbb{R}^2$. Throughout the paper, triangulated disk-like surfaces M with vertices V, edges E and faces F are considered for parameterization. For instance, techniques to map a multiply connected surface already exist [52]. We use discrete *harmonic* mapping to solve parameterizations on disks. *Harmonic* mappings have attributes derived from conformal parameterization, but there is no guarantee on angles. To proceed we construct a *harmonic* function $f : M \longrightarrow \mathbb{R}$ such that $\Delta f = 0$. Harmonic maps minimize *Dirichlet* energy:

$$E_D(f) = \frac{1}{2} \int_M \|\nabla f\|^2 dM. \tag{1}$$

The surface boundary ∂M is first mapped to the boundary of the parametric domain and then the parameterization for the interior vertices is obtained by solving the linear system:

$$\Delta_w f(v_i) = \sum_{j \in N_i} w_{ij}(f(v_j) - f(v_i)). \tag{2}$$

where $v_i, v_j \in V$, N_i is the neighborhood of v_i, and w_{ij} is the scalar weight assigned to the oriented edge $e_{ij}(v_i, v_j)$. Different parameterization methods assign different weights w_{ij}. The first definition of weight was introduced by Tutte [51]. In the parameter space, each vertex is placed at the barycenter of its neighbors. Recently, Saboret et al. [50] have implemented a CGAL package handling some of the state-of-the-art surface parameterization methods such as least squares conformal maps, discrete conformal map, discrete authalic parameterization, Floater mean value coordinates or Tutte barycentric mapping. In this context, we use mean value coordinates introduced by Floater [18]. According to the **Rado-Kneser-Choquet theorem**, if weights are positive and the boundary ∂M is mapped homeomorphically in a convex square parametric space, the mapping driven by the harmonic function f has to be bijective. This allows us to determine bijective mappings in order to avoid fold-overs in the smart model decomposition part, see e.g. "Smart model decomposition" section.

Smart model decomposition

In order to build valid quadrilateral layout to determine the surface parameterization, we proceed in two steps: geometry-aware pants decomposition ("Geometry-aware

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 7 of 28

pants decomposition" section) and feature-aware cuboid decomposition ("Feature-aware cuboid decomposition" section). Pants decomposition and cuboid decomposition are constructed to respect the topology of the input triangulated surface and consider features. Cuboid decomposition approximates very roughly the geometry while faithfully replicating its topology taking into account sharp edges and vertices. Due to its patch regular structure, it can serve as parametric domain needed for pure 3D quadrilateral mesh computation.

Geometry-aware pants decomposition

Closely related literature approaches and algorithmic prerequisites We focus on the decomposition of a triangulated surface into a set of pants patches $\{T_i\}$. Pants decomposition provides a canonical decomposition scheme for common surfaces. $\chi = -1$ for a pants patch, therefore a pant is the simplest topology after the sphere, disk, cylinder and torus. Many algorithms take as input handle or tunnel loops for genus-g surfaces to segment into pants patches [34]. The handle loops required for automatic pants decomposition are computed using the technique presented by K. Dey et al. [25]. An improvement was also developed by the same group [16]. In the following, we reuse the pants decomposition algorithm of Al-Akhras et al. [4] in order to integrate a most recent method.

New extensions According to previous work presented by Al-Akhras [3], we introduce an extension to enumerate the entire space of topological pants decomposition possibilities at each step. In other words, instead of picking 2 boundaries in an arbitrary manner, we suggest to take all couples of 2 boundaries among all non-repeating and commutative combinations when determining a pants patch relative to the considered step. Besides, giving some feature points, our new algorithm is able to determine a pant by slicing the mesh along a cycle passing across these locations.

User inputs User manual inputs are reduced to feature points selection in order to guide the decomposition passing through these points of interest. For other used geometric criteria, they are suggested to the user.

Feature-aware cuboid decomposition algorithm Given the homology basis formed by handle and tunnel loops, we can take a subset H composed of g simple pairwise disjoint handle loops $\{h_1, h_2, ..., h_{g-1}, h_g\}$. Slicing surface M with b boundary components along its g handle loops will lead to a genus-0 surface with $2g + b$ boundary components denoted as $W = \{w_1, w_2, ..., w_{2g+b-1}, w_{2g+b}\}$. We iteratively pick two boundaries w_i and w_j from W (see algorithm 1) taking into account combinations and compute a new simple cycle w_{ij} to bound them, i.e., w_{ij} is homotopic to $w_i \circ w_j$. The three cycles w_i, w_j and w_{ij} bound a pants patch T_k. We remove this pants patch T_k from M. The remaining patch is still genus-0 but its boundary number reduces by 1: the two cycles w_i and w_j are removed, and one new cycle w_{ij} is inserted. This is iteratively performed until $|W| = 3$. This idea is formulated in Algorithm 1, and the operation that traces a cycle w_{ij} homotopic to cycle $w_i \circ w_j$ is formulated in Algorithms 2 and 3.

Choice of the geometric criterion Shortest length of cycles on weighted triangulated meshes are computed using a Dijkstra's algorithm based on weighted tree graphs. A CGAL package exists to compute these paths [29]. Different geometric criteria can be used to guide the pants decomposition. Geometric criterion can be adapted to the mesh M. In algorithm 2 and also in Algorithm 1 to sort loops in L, this can be changed for areas

---

Algorithm 1 Geometry-aware pants decomposition

Input 1: Triangulated genus-g surface M with b boundary components.
Input 2: g geometrically relevant handle loops of M.
Input 3: Global geometric criterion for L.
Output: Set of $-\chi(M)$ pants patches $T = \{T_1, ..., T_{-\chi(M)}\}$, with $M = \cup T_i$.

01:   $k = 1$.
02:   Slice M along all its handle loops and get a surface M_k with $2g + b$ boundaries.
03:   Put all boundaries of M_k in a set $W = \{w_1, ..., w_{2g+b}\}$.
04:   **While** $|W| > 3$ **do**
05:       Build or reset an empty set of loops $L = \{0\}$.
06:       Compute N_c combinations: $\dim(N_c) = \frac{Size(W)!}{2!(Size(W)-2)!}$.
07:       **For** all couples $[w_i, w_j]$ in N_c:
08:          Compute a cycle w_{ij} homotopic to $w_i \circ w_j$.
09:          Add loop to L.
10:       **End For**
11:       Sort relevant loops in $L = \{l_1, ..., l_{\dim(N_c)}\}$ using a global geometric criterion.
12:       The optimal w_{ij} cycle is classified in L.
13:       $\{w_1, w_j, w_{ij}\}$ bound a pants patch T_k. Remove T_k from M_k: $M_k \leftarrow M_k \setminus T_k$.
14:       Remove w_i and w_j from W, and add w_{ij} into W.
15:       $k \leftarrow k + 1$.
16: **End While**

Algorithm 2 Homotopic cycle computation

Input 1: Genus-0 surface M with b boundary components $\{w_1, ..., w_b\}$.
Input 2: Geometric criterion.
Output: A cycle w_{ij} homotopic to cycle $w_i \circ w_j$.

01:   Compute the shortest path connecting w_i to w_j.
02:   Slice M along this path to get one new large boundary c_{ij}.
03:   Connect all other boundaries together using shortest paths.
04:   Slice M along these paths to get one new large boundary c_k.
05:   M becomes a topological cylinder.
06:   Compute the shortest path γ connecting c_{ij} and c_k.
07:   Slice M along the path γ.
08:   Every point $p_i \in \gamma$ ($n + 1$ points) is split into a pair $(p_i, \overline{p_i})$.
09:   Trace all shortest paths connecting points pairs $(p_i, \overline{p_i})$.
10:   Among these paths, the cycle w_{ij} satisfies the geometric criterion.

Algorithm 3 Feature homotopic cycle computation

Input 1: Genus-0 surface M with b boundary components $\{w_1, ..., w_b\}$.
Input 2: Set of n_p feature points $S = \{s_0, ..., s_{n_p}\}$.
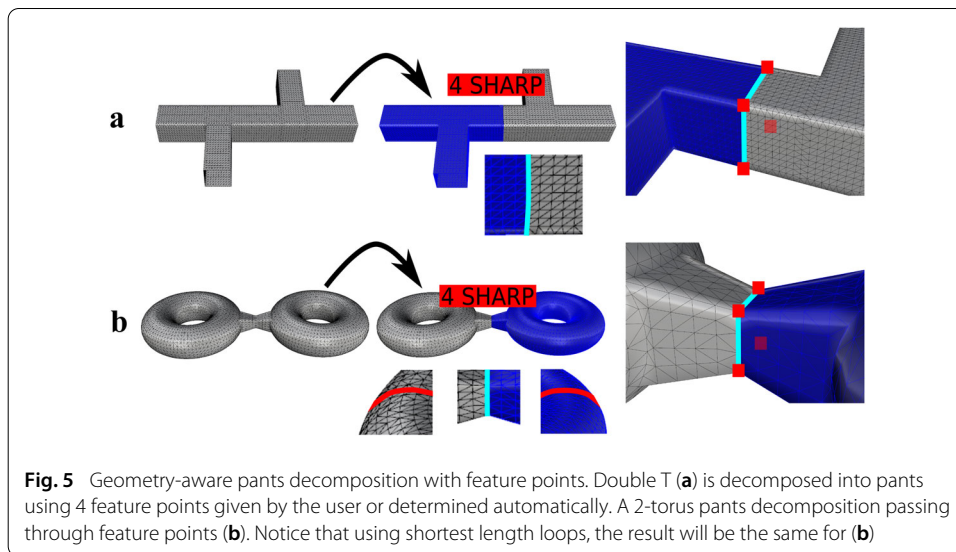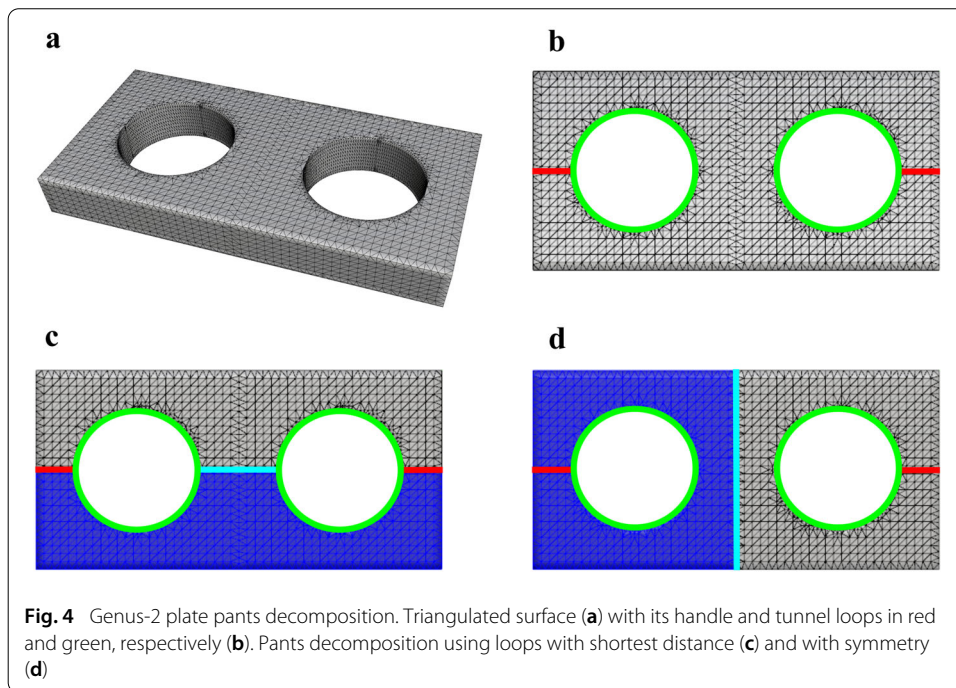**Output:** A cycle w_{ij} homotopic to cycle $w_i \circ w_j$.

01:   Compute the shortest path connecting w_i to w_j.
02:   Slice M along this path to get one new large boundary c_{ij}.
03:   Connect all other boundaries together using shortest paths.
04:   Slice M along these paths to get one new large boundary c_k.
05:   M becomes a topological cylinder.
06:   Compute the shortest path γ connecting c_{ij} and c_k.
07:   Slice M along the path γ.
08:   Every point $p_i \in \gamma$ ($n + 1$ points) is split into a pair $(p_i, \overline{p_i})$.
09:   $\{p_0, \overline{p_0}, p_n, \overline{p_n}\}$ are set to be the corners of a square harmonic mapping.
10:   Partial line inverse mapping passing through n_p points in S is performed.
11:   The cycle w_{ij} is reconstructed using line segments.

Fig. 4 Genus-2 plate pants decomposition. Triangulated surface (**a**) with its handle and tunnel loops in red and green, respectively (**b**). Pants decomposition using loops with shortest distance (**c**) and with symmetry (**d**)

Fig. 5 Geometry-aware pants decomposition with feature points. Double T (**a**) is decomposed into pants using 4 feature points given by the user or determined automatically. A 2-torus pants decomposition passing through feature points (**b**). Notice that using shortest length loops, the result will be the same for (**b**)

of minimum curvature [31], minimum length or symmetry. For instance in Fig. 4c pants decomposition is performed using loops with minimum length, whereas in Fig. 4d decomposition is made by symmetry. In Fig. 5, user selected feature points are given to guide the pants decomposition. All of these guidings yield a geometry-aware decomposition.

Discussion on robustness A common surface admits infinitely many pants decompositions. In general, not all pants decomposition results are suitable for the next step of our algorithm. By correctly defining feature points (in general a very small set of points) to lead the segmentation, the algorithm succeeds. If the pants decomposition is guided by the different geometric criteria presented above, we obtain satisfying results for all our test cases.

Feature-aware cuboid decomposition

Closely related literature approaches Pants patches provide a very simple topology and hence each pants patch can be treated separately. Li et al. [32] presented a method that generates cuboids per pants patch with a given user data input. Al-Akhras [3] improved it in order to perform the decomposition in a way that it does not require any input from the user. Nevertheless, developed techniques do not consider sharp features and deal with a simple cuboid decomposition scheme. They both try to generate corners and polyedges on each pants patch T_i and decompose it into a set of cuboids, each having 8 corners and 12 polyedges in order to construct a volumetric parameterization. In addition, these two previous approaches do not handle complicated cuboid configurations and generated arcs and nodes are not necessarily positioned onto features of the input surface mesh. Finally, prior automatic processes restrict the placement of the quadrilateral layout nodes in specific areas, e.g. using harmonic functions.

New extensions Our work presents an extension to handle more complicated geometry especially with sharp features. First, on each pants patch the number of quadrilateral patches $n_\mathbb{Q}$ is set to respect sharp vertices and edges: we aim to perform a feature-aware cuboid decomposition. Indeed, optimal number of quadrilateral patches depends on the features embedded into the considered pants patch. Secondly, irregular vertices of the layout with a valence $v \neq 4$ are placed onto relevant areas with high concentrated Gaussian curvature and/or with significant features.

Algorithmic prerequisites The space of valid quadrilateral layouts $\mathbb{Q}$ is defined in equation (3). A quadrilateral layout is composed by arcs and n_n nodes of valence v_i. Cuboid configurations $\mathbb{C}$ decompose a surface into a set of quadrilateral patches which are ready for hexahedral meshing due their cubic structure. In this paper, the cuboid decomposition is dedicated only to surface decomposition. Indeed, this work only focuses on structured quadrilateral meshes.
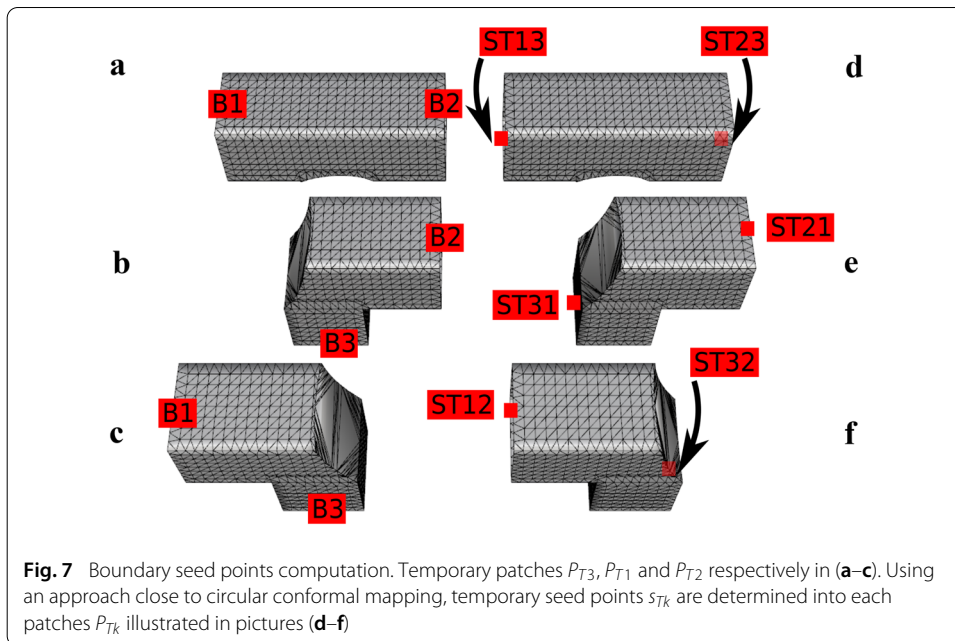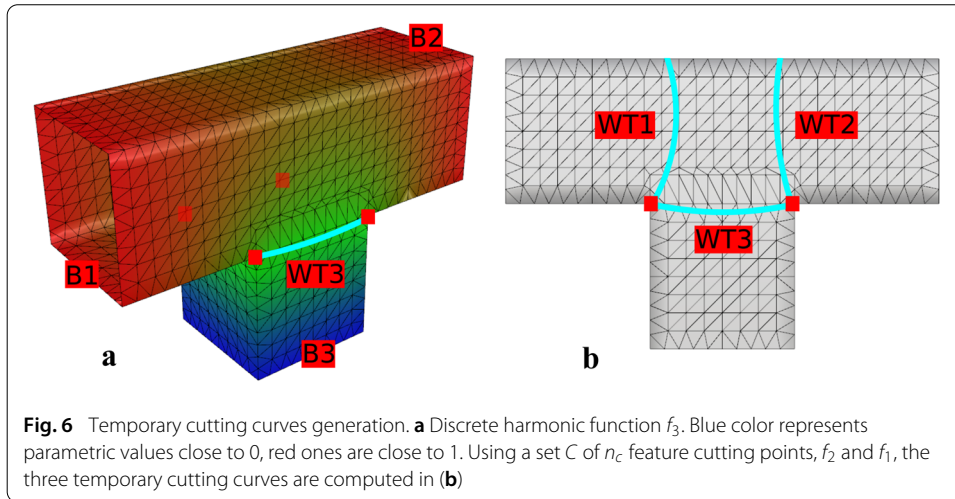
$$\forall v_i \in \mathbb{Q}, \sum_{i=1}^{n_n} \left(1 - \frac{v_i}{4}\right) = \chi(M)_{Closed} = 2 - 2g. \tag{3}$$

User inputs To reach the feature-aware cuboid decomposition goal, manual user inputs are necessary. Depending on the pant geometry and features to replicate in the cuboid decomposition, 2 inputs are mandatory. Using a cuboid configuration templates library, a segmentation layout is choosen by the user; thus involving the wish to suit the desired features of the pant into the decomposition. Then, the structure of the requested template requires the manual selection of its asscociated feature points. Nature of these features is automatically suggested and the user has just to specify their location.

Feature-aware cuboid decomposition algorithm The 3 boundaries of a given pants patch will be arbitrarily denoted by B_1, B_2 and B_3. We process these pants patches one by one in an arbitrary order. To guarantee cuboid corner alignment, when we determine one pants patch's result, we transfer its corners on the boundaries of the adjacent pants patches if they are not processed yet. Notice that just a general algorithm with the most relevant specifications is provided using a simple decomposition scheme of 4 cuboids per pant patch.
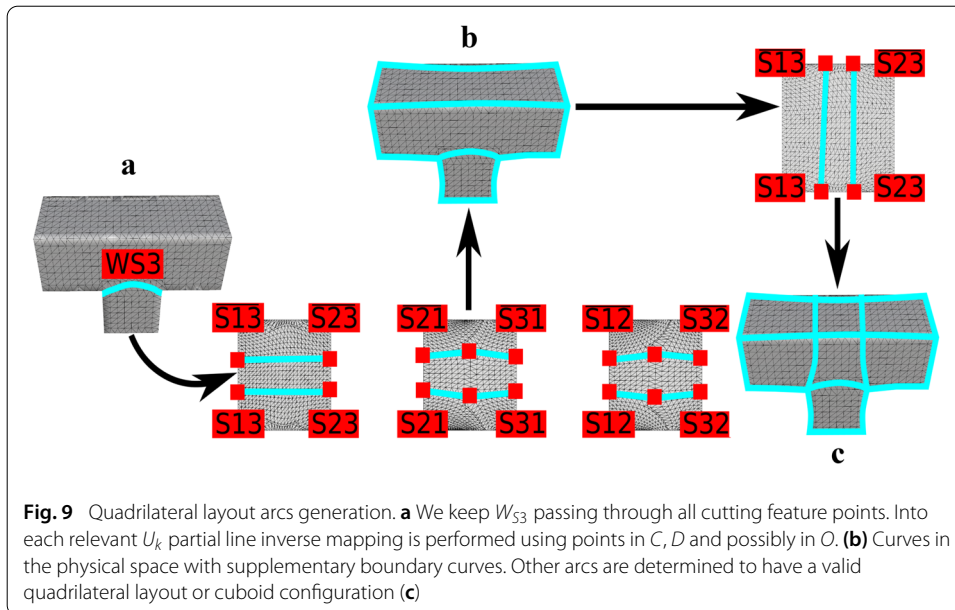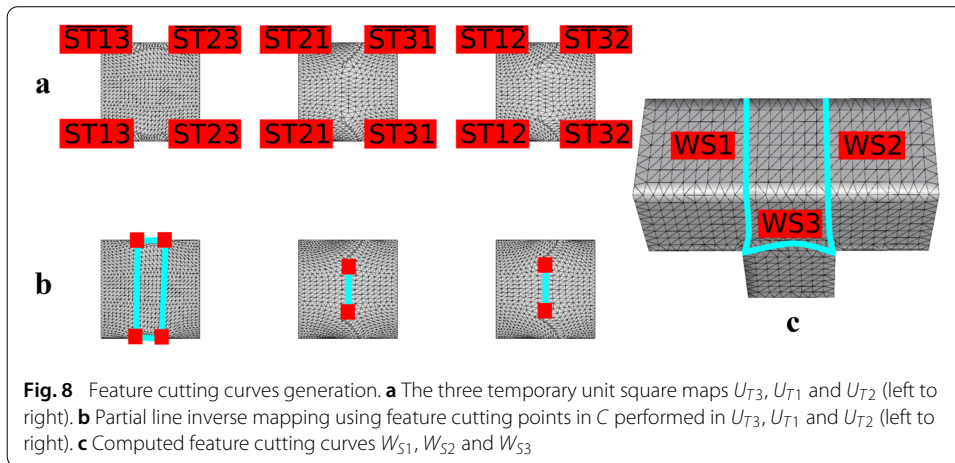
Step 1. We generate 3 feature cutting curves W_{S1}, W_{S2} and W_{S3}:

[A] We compute 3 discrete harmonic functions: f_1, f_2 and f_3. To compute f_i , we set $f_i = 0$ for vertices on the boundary B_i and $f_i = 1$ for vertices on the remaining two

Fig. 6 Temporary cutting curves generation. **a** Discrete harmonic function f_3. Blue color represents parametric values close to 0, red ones are close to 1. Using a set C of n_c feature cutting points, f_2 and f_1, the three temporary cutting curves are computed in (**b**)

Fig. 7 Boundary seed points computation. Temporary patches P_{T3}, P_{T1} and P_{T2} respectively in (**a**–**c**). Using an approach close to circular conformal mapping, temporary seed points s_{Tk} are determined into each patches P_{Tk} illustrated in pictures (**d**–**f**)

boundaries B_j and B_k . Then we solve $\Delta f_i = 0$ using mean value coordinates. Figure 6a.

[**B**] Each harmonic function f_i has one minimum component and two maximum components. Among a set C of n_c feature cutting points $C = \{c_0, c_1, ..., c_{n_c-1}, c_{n_c}\}$, let $\overline{f_i} = min[f_i(c_i)]$, $\forall c_i \in C$. Then a temporary non-feature cutting curve W_{Ti} is defined as the isoparametric curve of the function f_i for the value $\overline{f_i}$. Figure 6b.

[**C**] We remove a long branch, i.e., by cutting along its temporary cutting curve W_{Tk}. After filling the cutting hole, the resulting patch is a topological cylinder with 2 boundaries B_i and B_j. We denote this temporary patch by P_{Tk}. Figure 7a–c.

[**D**] Using previously computed temporary patches P_{Tk}, we first set $u = 0$ for vertices on B_i and $u = 1$ for vertices on B_j , then solve $\Delta u = 0$. Using an approach close to Zeng et al. [52], among all iso-v curves along ∇u starting from B_i and B_j, we automatically select two relevant curves who are intersecting the filled boundary B_k, one starting

Fig. 8 Feature cutting curves generation. **a** The three temporary unit square maps U_{T3}, U_{T1} and U_{T2} (left to right). **b** Partial line inverse mapping using feature cutting points in C performed in U_{T3}, U_{T1} and U_{T2} (left to right). **c** Computed feature cutting curves W_{S1}, W_{S2} and W_{S3}

Fig. 9 Quadrilateral layout arcs generation. **a** We keep W_{S3} passing through all cutting feature points. Into each relevant U_k partial line inverse mapping is performed using points in C, D and possibly in O. **(b)** Curves in the physical space with supplementary boundary curves. Other arcs are determined to have a valid quadrilateral layout or cuboid configuration (**c**)

from temporary seed point s_{Tik} on B_i and one starting from temporary seed point s_{Tjk} on B_j. Figure 7d–f.

[**E**] By performing this previous task on each temporary patch P_{Tk}, we obtain 6 temporary seed points s_{Tij}, s_{Tik}, s_{Tji}, s_{Tjk}, s_{Tki} and s_{Tkj}. Working on all patches P_{Tk} we plot an iso-v curve along ∇u from the seed vertex s_{Tik} on the boundary B_i to the seed vertex s_{Tjk} on boundary B_j. We slice P_{Tk} along this iso-curve and get two duplicated boundary paths. We finally set $v = 0$ and $v = 1$ on them respectively and solve $\Delta v = 0$ to obtain 3 oriented temporary unit-square maps denoted U_{Tk} with corners $\{s_{Tik}, s_{Tjk}, \overline{s_{Tik}}, \overline{s_{Tjk}}\}$. $\overline{s_{Tik}}$ and $\overline{s_{Tjk}}$ are duplicated points of s_{Tik} and s_{Tjk} respectively. Notice that triangles from filling are removed during U_{Tk} computation. Figure 8a.

[**F**] Points in C are mapped into each unit-square map U_{Tk} and automatically classified using parametric coordinates u and v. Then we perform partial line inverse mapping between each classified cutting points c_i related to relevant U_{Tk} and merge computed lines to obtain valid feature cutting curves W_{S1}, W_{S2} and W_{S3}. Figure 8b, c.

Step 2. We generate 3 feature-aware oriented unit-square maps denoted U_k:

[**A**] We remove a long branch by cutting along its feature cutting curve W_{Sk}. We proceed like **Step 1** [**C**] to obtain a feature-aware cutting patch P_k.

[**B**] Using the same workflow of **Step 1** [**D**] feature seed points s_{ik} and s_{jk} are determined.

[**C**] Using the same workflow of **Step 1** [**E**], we obtain 6 feature seed points $s_{ij}, s_{ik}, s_{ji}, s_{jk},$ s_{ki} and s_{kj}. Then 3 feature-aware oriented unit-square maps U_k are extracted.

**Step 3**. We generate all arcs of $\mathbb{Q}$:

[**A**] According to **Step 1** [**F**] feature cutting arcs W_{Sk} have already been computed. We keep W_{S3} passing through all cutting feature points. Figure 9a.

[**B**] Among a set D of n_d feature boundary points $D = \{d_0, d_1, ..., d_{n_d-1}, d_{n_d}\}$, we map these points into corresponding maps U_k. Using automatic classification with parametric coordinates u and v, we pair on each relevant U_k points in D possibly with feature cutting points in C. We perform partial line inverse mapping to obtain arcs of the quadrilateral layout. Boundary arcs are determined directly in the physical space. Figure 9b.

[**C**] Consider a non-empty set O of n_o common feature points $O = \{o_0, o_1, ..., o_{n_o-1}, o_{n_o}\}$. We proceed like **Step 3** [**B**] by pairing on each relevant U_k points in O possibly with C or D.

[**D**] Remaining arcs of the quadrilateral layout or cuboid configuration (see equation (3)) are computed using partial line inverse mapping into relevant U_k with points in D, O or in C. Other arcs and points are then obtained by line-line intersection analysis. Figure 9c.

With feature points in C (see **Step 3** [**C**]) or other feature points in D or O, a huge variety of shapes can be decomposed into cuboids and hence handling more complex geometry cases is possible. To go further, the pants-to-cuboids decomposition can be divided in two main parts: the biological and mechanical parts, see e.g. Fig. 10. For instance, specific biological quadrilateral layouts have been studied by Al-Akhras et al. [2]. Details for the mechanical workflow are explained in Fig. 11. Given a consistent pants decomposition, our algorithm is able to decompose a triangulated surface into quadrilateral patches suitable for surface parameterization. This previous task is done only by specifying relevant feature points relative to a given quadrilateral layout template.

Direction field generation

Introduced in "Topology and parameterization prerequisites" section, 4-symmetry directions fields (i.e., cross fields) are widely used to determine an aligned global parameterization [6–8,45]. In other words, these fields are useful to guide the parameterization to determine quality quadrilateral surfaces. Designing a smooth cross field $\mathcal{C}$ is done with a given set of constraints. We categorize these constraints into two groups: topological and geometric constraints. Topological constraints are imposed singularities and numbers induced by the surface topology $\chi(M)$. Geometric constraints are intrinsically embedded on the surface. Thus we seek a cross field $\mathcal{C}$ which is smooth, aligned with the local geometry and topologically compatible. The quadrilateral layout $\mathbb{Q}$ or cuboid configuration $\mathbb{C}$ contains topological properties whereas the surface M hold the geometric information.
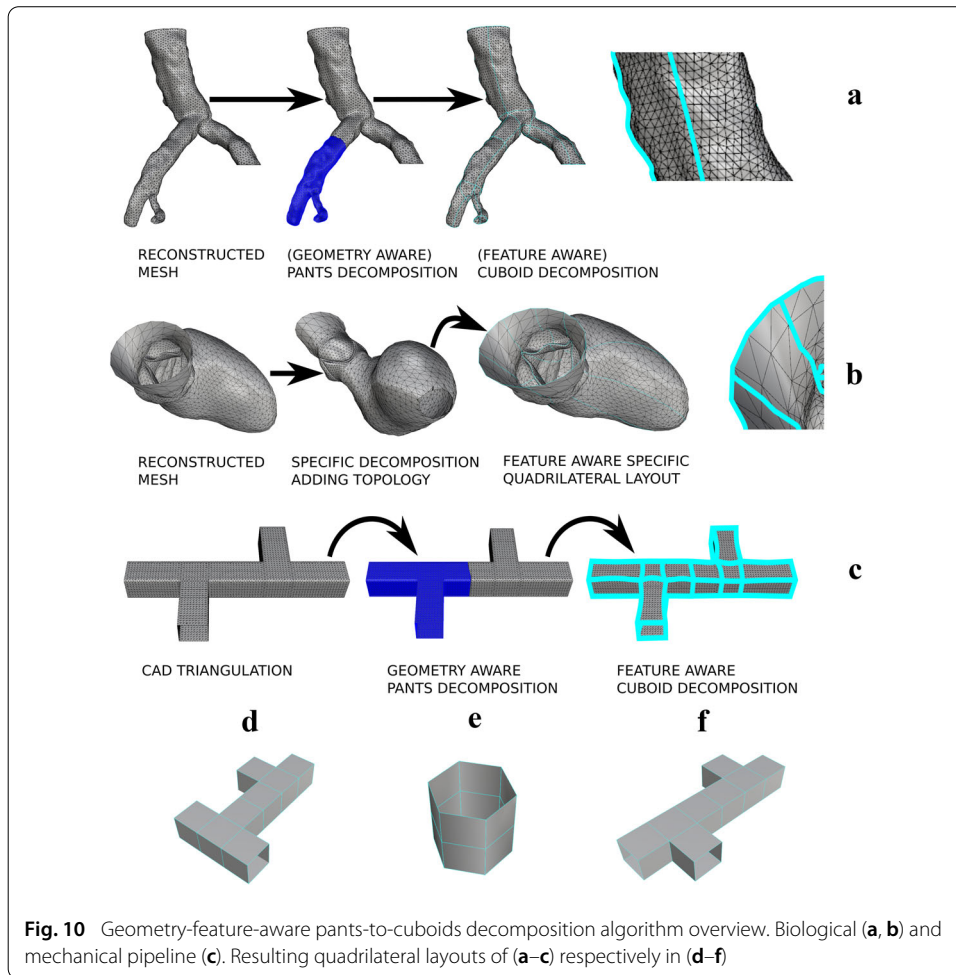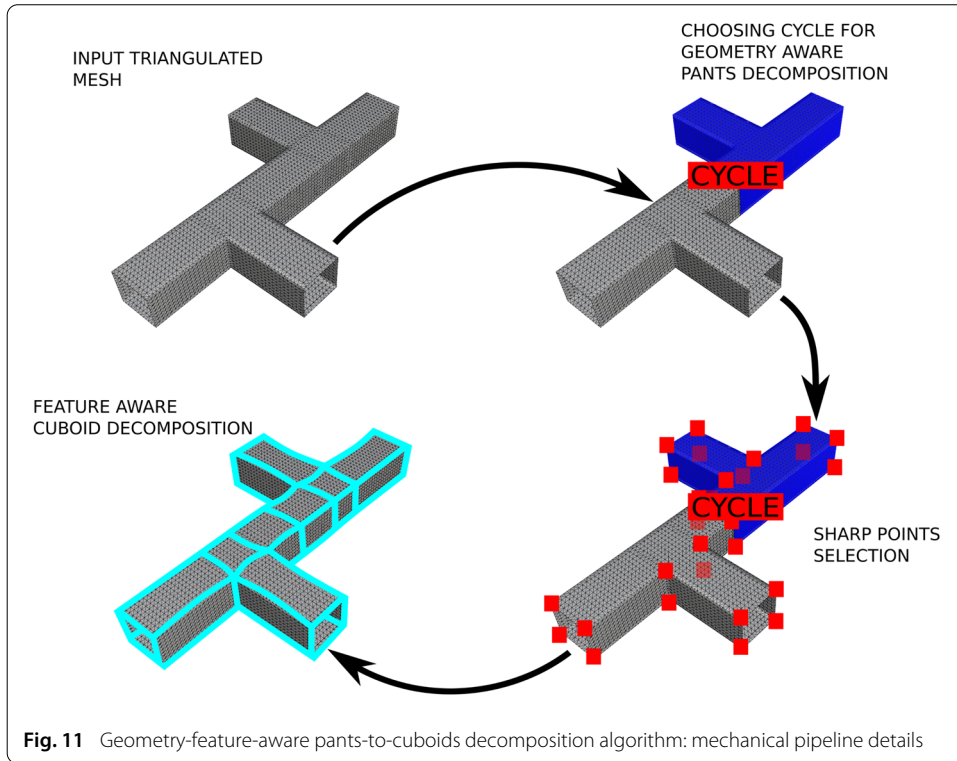
Fig. 10 Geometry-feature-aware pants-to-cuboids decomposition algorithm overview. Biological (**a**, **b**) and mechanical pipeline (**c**). Resulting quadrilateral layouts of (**a**–**c**) respectively in (**d**–**f**)

We consider a triangulation of the surface M, assumed to be a 2-dimensional manifold of genus-g with b boundary components. Directions will be stored at faces F to avoid definition of supplementary tangent planes. The first step of the discretization consists in finding a reference direction in each triangle. This is done by choosing a local orthonormal frame $(\mathbf{x}, \mathbf{y})$ attached on each face f. The vector $\mathbf{x}$ is an unitary vector along one of the oriented edges of face f, and $\mathbf{y} = \mathbf{n} \times \mathbf{x}$, where $\mathbf{n}$ is the normal of f. A direction $\mathbf{u}$ on f can be formulated in terms of polar coordinates. Due to the unit norm of such directions, it is completely parameterized by the polar angle α it forms with $\mathbf{x}$. We denote α the direction angle. By unfolding adjacent triangles isometrically to a plane along their common edge, angles can be expressed in a common coordinate frame. To specify the number of Nth turns the direction $\mathbf{u}_A$ undergoes to match with $\mathbf{u}_B$ when passing from A to B, the *period jump* [33] is adopted. Other works introduce an angle ω named *connection angle* to solve this ambiguity [14,15]. In the following, the *connection angle based discretization* is used for topological design and *period jump based discretization* is used for geometric design.

Topological cross field generation

The first discretization computes an angle ${}^c\alpha_i^j$ from face f_i expressed in adjacent face f_j which is in general equal to α_j^j, see e.g. equation (4). The angle $\kappa_{ij} \in [-\pi, \pi]$ represents

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 15 of 28

Fig. 11 Geometry-feature-aware pants-to-cuboids decomposition algorithm: mechanical pipeline details

the difference between a reference direction $\mathbf{x}_i$ in face f_i and $\mathbf{x}_j$ in face f_j. Angle ω_{ij} is the *connection angle* when walking on the dual edge linking face f_i to face f_j.

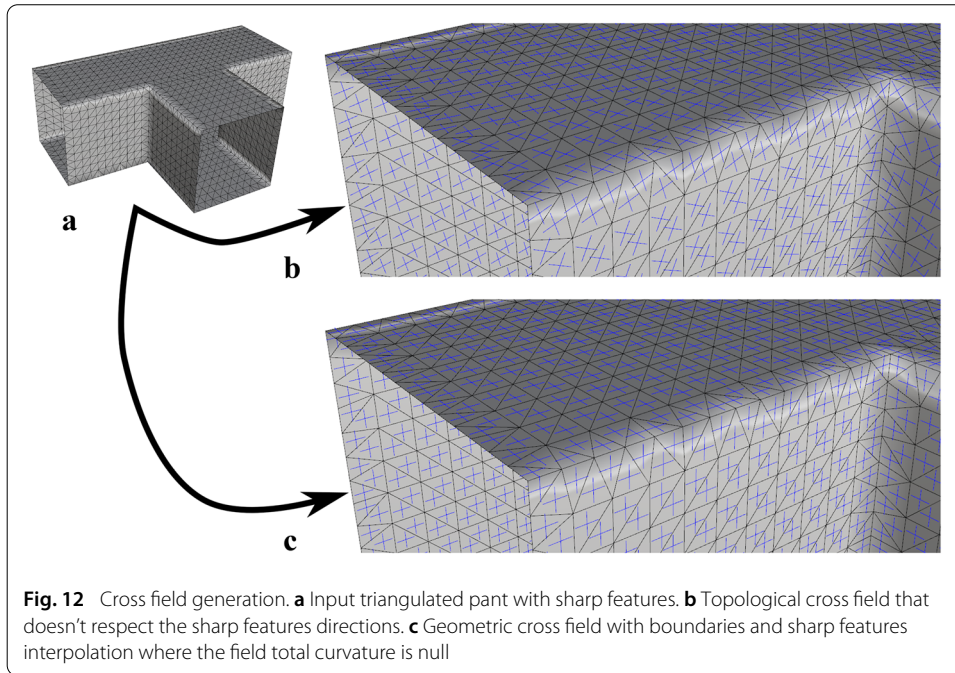$$^c\alpha_i^j = \alpha_i^i + \kappa_{ij} + \omega_{ij}. \tag{4}$$

We seek to fix all topological degrees of freedom to restrict the cross field topologically compatible to our quadrilateral layout $\mathbb{Q}$. Technically speaking, we target a smooth cross field $\mathcal{C}$ that is singular only at specified vertices: the position of irregular nodes of $\mathbb{Q}$. The approach of Campen and Kobbelt [8] is adopted to determine the cross field topological degrees of freedom from the input quadrilateral layout. Figure 12b provides a topological cross field calculated using the minimization of energy introduced by Crane et al. [15]. The generated field respect the Gauss-Bonnet theorem by distributing the Gaussian curvature in a smart manner [14].

Geometric cross field generation

The second discretization computes an angle $^p\alpha_i^j$ from face f_i expressed in adjacent face f_j which is in general different to α_j^j, see e.g. equation (5). Angles α_i^i and α_j^j are respectively expressed in their native faces. The integer $p_{ij} \in \mathbb{Z}$ is the *period jump* when passing from face f_i to face f_j. The integer N is the symmetry of the field, i.e., 4 for cross fields.

$$^p\alpha_i^j = \alpha_i^i + \kappa_{ij} + \frac{2\pi}{N}p_{ij}. \tag{5}$$

A smooth cross field that interpolates relevant principal curvature directions, sharp features and boundaries restricted to a given topology is targeted. We search the smoothest field taking into account constrained directions given by geometric features. The period

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.* (2021)8:8

Page 16 of 28

Fig. 12 Cross field generation. **a** Input triangulated pant with sharp features. **b** Topological cross field that doesn't respect the sharp features directions. **c** Geometric cross field with boundaries and sharp features interpolation where the field total curvature is null

jumps of the field are fixed. A technique introduced by Bommes et al. [6] and later used by Al-Akhras et al. [2] is adopted to compute such field. Figure 12c shows a geometric cross field that interpolates sharp features and boundaries.

Aligned global parameterization computation from geometric direction field

We now compute an *aligned global parameterization*, i.e., a map from the mesh M to a disk-like surface parameter domain $\Omega \in \mathbb{R}^2$. We assign a couple (u, v) of parameter values on each vertex of M. The parameterization should be locally aligned with the features of the mesh, this is done using the guiding geometric cross field previously computed. Such parameterization implies that the gradients ∇u and ∇v of the discrete scalar field must follow the cross field directions on each face.
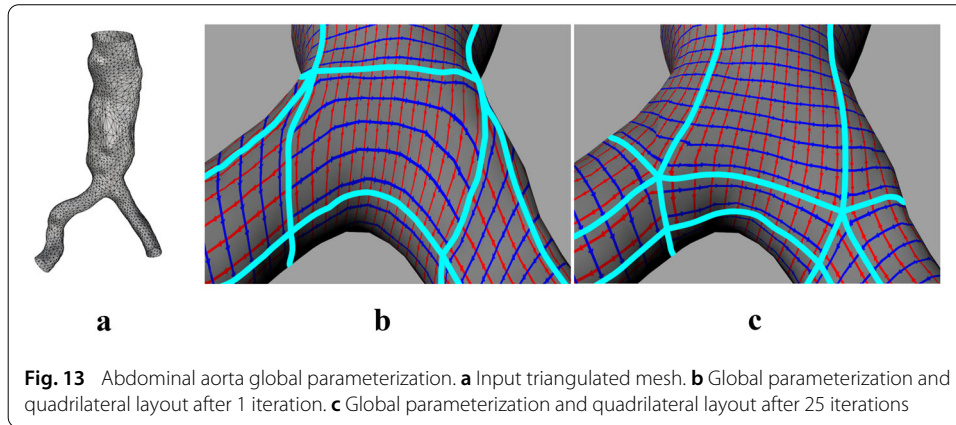
Seamless parameterization

A planar parameterization of a mesh M embedded in $\mathbb{R}^3$ into a parametric domain Ω embedded in $\mathbb{R}^2$ is in general done by computing a *cut graph*. A cut graph G is a connected graph formed by edges of M that splits the mesh into a disk-like surface mesh M^c. Seams are then defined as duplicated paths of G. Transitions across seams need to belong to a very restricted class. We search for rigid transformations with a rotation angle multiple of $\frac{\pi}{2}$. Moreover across each seam edge or vertex, the corresponding transition must be integral, i.e., relative to an integer. Thus we talk about integral seamless parameterization [27]. We target the cross field first and second directions $\mathbf{u}^{\mathcal{C}}$ and $\mathbf{v}^{\mathcal{C}}$ for the gradients of the parametric coordinates ∇u and ∇v. The parameterization is then computed as the solution of a constrained minimization problem:

$$\min_{u,v} \sum_{f \in F} [\|\nabla u - \mathbf{u}^{\mathcal{C}}\|^2 + \|\nabla v - \mathbf{v}^{\mathcal{C}}\|^2] A_f \text{ s.t. (7)}, \tag{6}$$

$$v_1' = R_{\frac{\pi}{2}}^{m_{ij}} v_1 + t_{ij} \& v_2' = R_{\frac{\pi}{2}}^{m_{ij}} v_2 + t_{ij}. \tag{7}$$

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.(2021)8:8*

Page 17 of 28

Fig. 13 Abdominal aorta global parameterization. **a** Input triangulated mesh. **b** Global parameterization and quadrilateral layout after 1 iteration. **c** Global parameterization and quadrilateral layout after 25 iterations

where A_f is the area of the considered face f. The corresponding final positions v_1' and v_2' are related to the cut edge e_{ij} from face i to face j with endpoints $v_1 \in \mathbb{R}^2$ and $v_2 \in \mathbb{R}^2$ in the parametric domain. Given a side of the cut by picking v_1 and v_2, the parameterization on the other side is determined by these previous transitions. The matching $m_{ij} \in \mathbb{Z}$ is defined between the two local charts which specify the rotation operator R. Rotations are made by angles multiple of $\frac{\pi}{2}$. The integer $t_{ij} \in \mathbb{Z}^2$ is a translation across cut edge e_{ij}. Aligned global parameterization can also integrate sharp edges and boundary constraints to ensure that a common relevant edge is mapped to an isoparametric curve.

Node connection constraints

Once a valid quadrilateral layout and consistent geometric cross field are provided, we wish to restrict each arc in a way that two incident nodes lie on a common isoparametric curve. With isoparametric we mean that either the u or v parameter is constant along the curve when taking transitions into account. The minimization problem (6) is then constrained using node connection constraints. These constraints are derived from the quadrilateral layout $\mathbb{Q}$. Typically, each arc must lie on a common isoparametric curve taking seams transitions into account [46]. We are now able to compute a suitable global parameterization of the mesh M, given the associated quadrilateral layout $\mathbb{Q}$.

Quadrilateral layout embedding optimization

Depending on geometry and position of $\mathbb{Q}$'s nodes, a better parameterization can be found using node relocation. Due to the nature of the quadrilateral layout, fixed nodes rise to large distortions or even local non-injectivities. Mathematically speaking we are going to optimize equation (6). The nodes are re-positioned based on the gradient of the parameterization's objective functional with respect to their positions. This is done iteratively until a global embedding quality is reached. We follow the method developed by Campen and Kobbelt [8] to perform such optimization. Figure 13 shows an abdominal aorta scanned mesh with its global parameterization optimization using an arbitrary valid quadrilateral layout. Observe the final global parameterization quality and nodes relocation. The final parameterization is then used to construct feature-aligned quadrilateral meshes. Due to patch structure of the quadrilateral layout, such quadrilateral meshes can be computed patch by patch according to connectivity constraints.

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 18 of 28

Isotopological analogous quadrilateral meshes construction from aligned global parameterization

Let us reformulate the previous defined problematic in section 1.2 in a more technical manner. Given a set of input triangulated meshes we strive to find 3D quadrilateral meshes which respect the four following properties:

- Pure quadrilaterals with low distortion.
- Feature aligned.
- Isotopological with analogous points into other geometric instances.
- Non-uniform isotropy.

Pure quadrilaterals with low distortion With the definition of a quadrilateral layout replicating perfectly the topology of the input triangulated surface mesh, it is possible to define a pure quadrilateral mesh. Low distortion is achieved by minimizing the alignment of the global parameterization with the guiding field.

Feature aligned Sharp edges, vertices and principal curvature directions are inportant features of meshes. Alignment to these features is required on all instances to fully conserve significant features. Thus the input quadrilateral layout must contain all the sharp edges and vertices to ensure right feature alignment into all input meshes.

Isotopological with analogous points into other geometric instances A sufficient condition to have *isotopological* meshes is to hold the same connectivity. A representative quadrilateral mesh of the population with a correct sampling is taken as reference to set the mesh connectivity. It should be noted here that it is not a sufficient condition to compare meshes because of orientation. This is why we want each point of the computed mesh to have an *analogous* point into all other geometric instances of the considered set. It is achieved globally by patch connectivity and locally by classifying sharp features and applying patch parameterization.

Non-uniform isotropy Mesh non-uniform isotropy provides the only degrees of freedom to have isotopological meshes from different geometric instances. Non-uniform *isotropy* is intrinsically set because of patch connectivity and discretization.

Application of the MEG-IsoQuad method to geometric parametric analysis
Preliminaries
Reduced order models

As mentioned in the introduction, Reduced Order Models (ROMs) enable real-time analysis due to their very low computational cost during evaluation. In the context of geometric parametric studies, their construction requires accumulating a certain number of geometries depending each on a set of parameters. This set of geometric instances are called snapshots. For ROM construction we need solution vectors with the same dimension associated to all snapshots in order to build matrices required to perform a Singular Value Decomposition (SVD). Isotopological and analogous meshes are thus appropriate.

Algorithmic details

The software *Rhinoceros 5* [43] is used for visualization and *RhinoCommon.dll* [44] for handling various Non-Uniform Rational B-Spline (NURBS) objects. Our algorithm is entirely incorporated into a *Rhinoceros 5* Plug-In implemented in VB.NET. C++ processes

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 19 of 28

Table 1 Range of parameters used for the T-shape part geometric reduced order model

Pant part range of parameters	Minimum	Maximum
X1 on boundary 1	− 40	40
Y1 on boundary 1	− 40	40
Z1 on boundary 1	− 35	35
X2 on boundary 2	− 40	40
Y2 on boundary 2	− 40	40
Z2 on boundary 2	− 35	35
X3 on boundary 3	− 40	40
Y3 on boundary 3	− 40	40
Z3 on boundary 3	− 35	35

are called from the Plug-In. Taking a B-Rep object provided from a file or constructed from the *Rhinoceros 5* graphical interface, a geometric reduced order model can be built using our fully integrated workflow. We use for that purpose the ROM builder proprietary software developed by ANSYS [1]. For instance, this builder has been used to prevent excessive compression of buttock's soft tissues by bony structures [39] in real-time for paraplegic persons. All computations have been performed on an Intel® Core™ i7-6820HQ CPU @ 2.70Ghz with 16Go RAM.
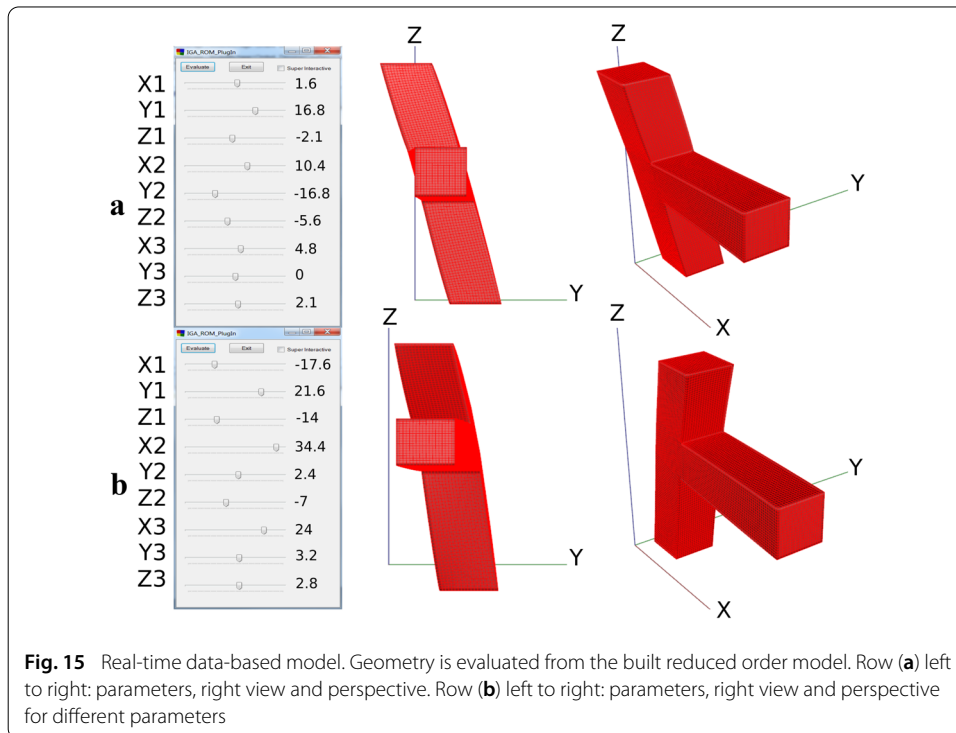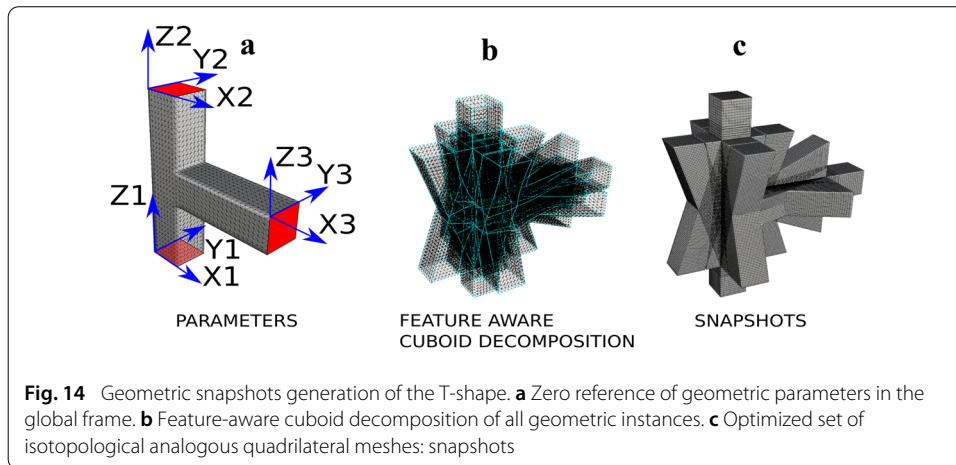
Mechanical parametric geometry

T-shape part

We first introduce the T-shaped part for a rapid understanding of our approach. Using this very simple shape, we define the number and the range of geometric parameters for an arbitrary study, see e.g. Table 1. These meshes come from a standard CAD software and each input triangulated mesh has an attached set of 9 geometric parameters based on the zero reference (Fig. 14a). Then a feature-aware cuboid decomposition is performed involving a pant mesh as topological input (Fig. 14b). Thereafter an optimized aligned global parameterization is computed with the quadrilateral layout and hence gives us the desired quadrilateral mesh with required geometric and connectivity properties. We proceed the workflow on all 37 input geometric instances to generate all isotopological analogous meshes (Fig. 14c).

To avoid large amount of triangulated meshes as input in order to compute an accurate ROM with 9 geometric parameters, we operate a very simple sparse grid sampling of the sets [37,38] (we took 4 snapshots per parameters, involving 37 meshes). ROM is then built using the isotopological snapshots. Once the reduced order model is built, we can generate in real-time all desired geometries handling our ROM evaluation interface (Fig. 15).

Casting part

This casting part is a classical one that can be found in mechanical systems for coupling shafts together. Starting with a feature-aware pants decomposition, we apply a specific cuboid configuration on each pant (Fig. 16c, d). Then the worflow is very similar to the T-shaped model presented in the previous paragraph. The sampling of the 6 input parameters (Fig. 16a, b) was made involving an optimal space filling technique so as to generate 65 triangulated meshes covering as best as possible the $\mathbb{R}^6$ space. Parameters

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 20 of 28

Fig. 14 Geometric snapshots generation of the T-shape. **a** Zero reference of geometric parameters in the global frame. **b** Feature-aware cuboid decomposition of all geometric instances. **c** Optimized set of isotopological analogous quadrilateral meshes: snapshots

Fig. 15 Real-time data-based model. Geometry is evaluated from the built reduced order model. Row (**a**) left to right: parameters, right view and perspective. Row (**b**) left to right: parameters, right view and perspective for different parameters
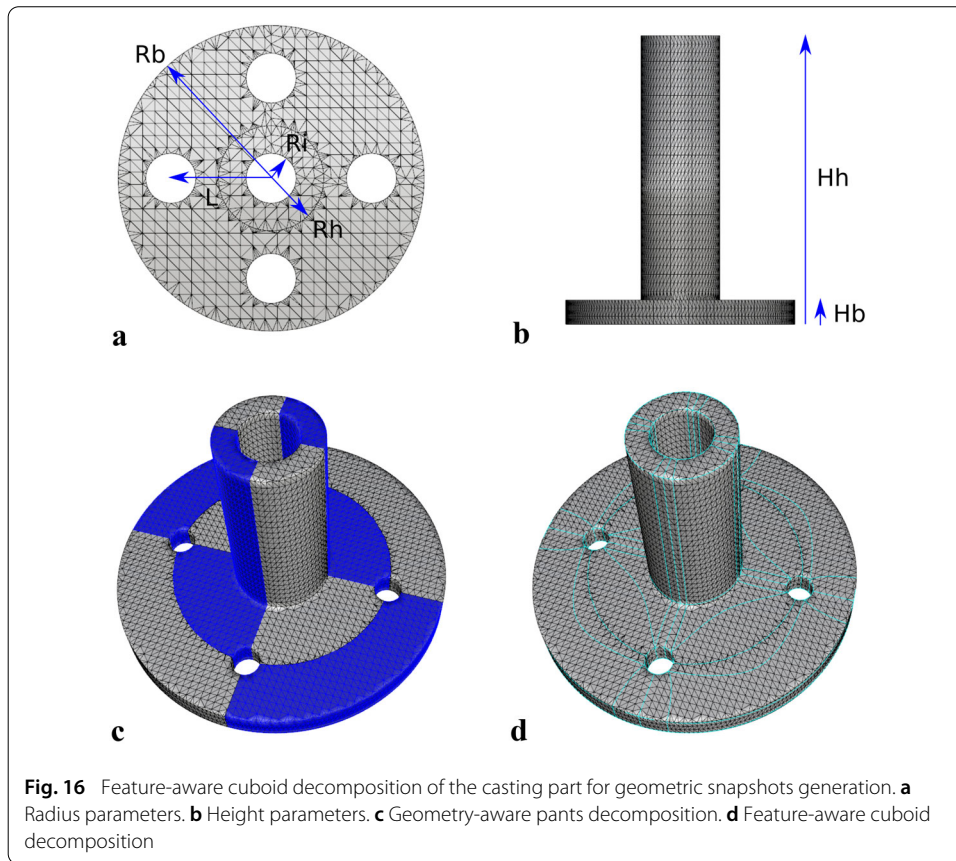
ranges have been defined so as to replicate a real industrial case with large and short variations, see e.g. Table 2. We generate in real-time all desired geometries handling the same ROM evaluation interface (Fig. 17) adapted to the casting part study.

Biological parametric geometry
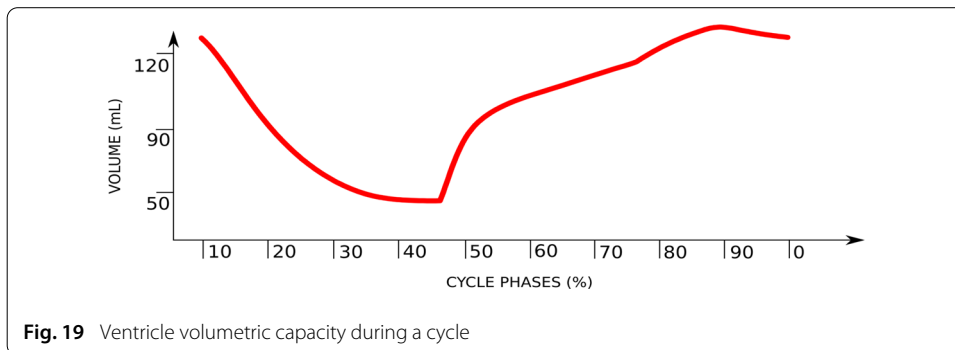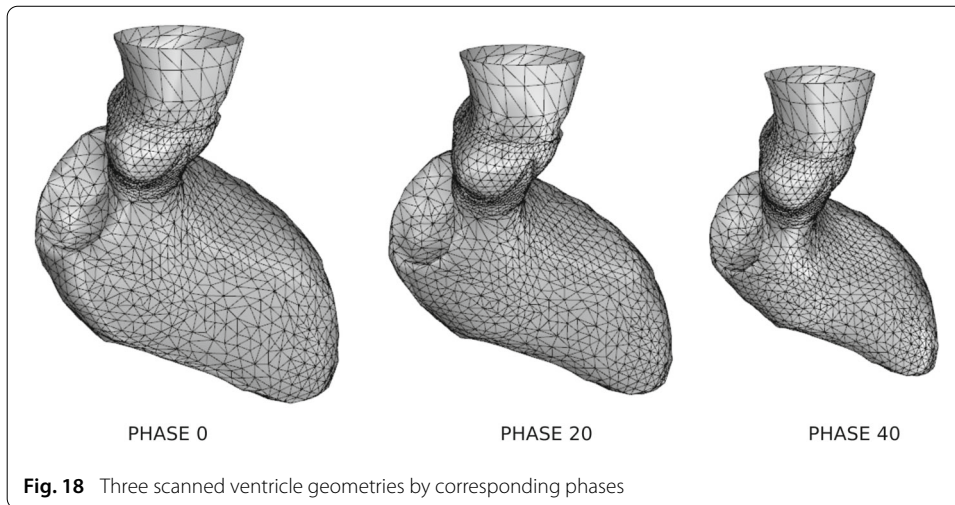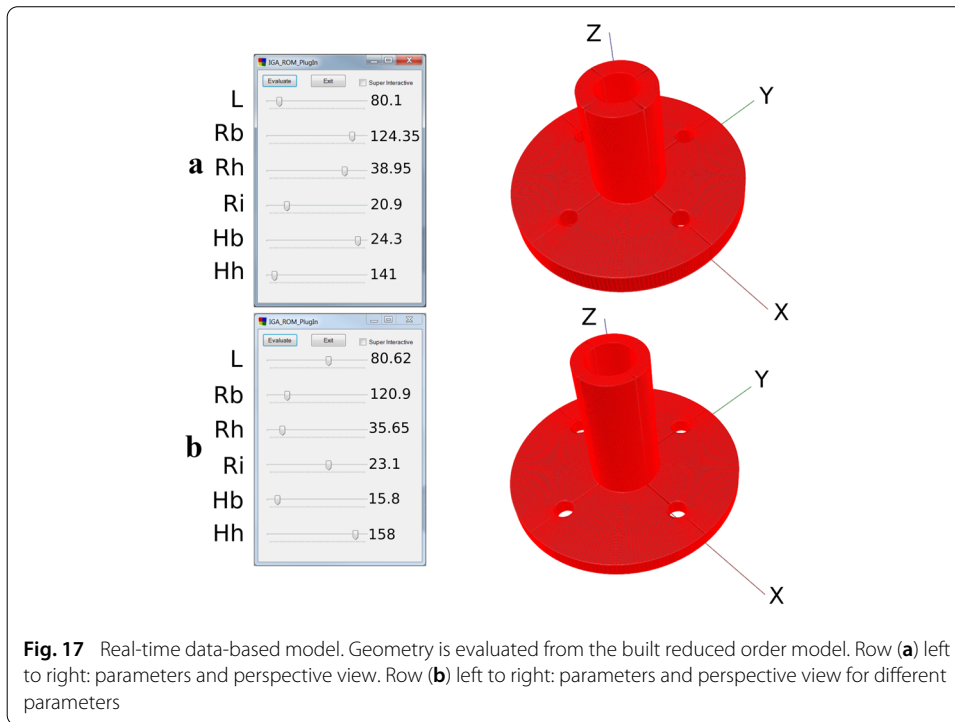
Heart ventricle

Ventricle geometric reduction is dedicated to a specific biological application. Scanned data from medical imaging related to one patient is provided during one cardiac cycle uniformly sampled by 10 meshes. Meshes are called "phases" and they start from 0 to 90 (Fig. 18). Ventricule volumetric capacity during a cardiac cycle can be seen in Fig. 19. Using a specific pipeline well-suited for these particular meshes, a quadrilateral layout
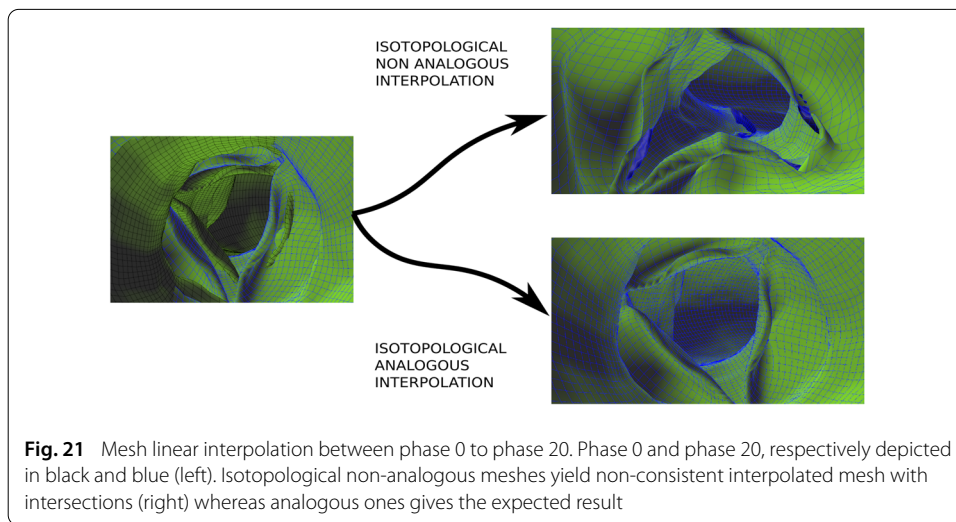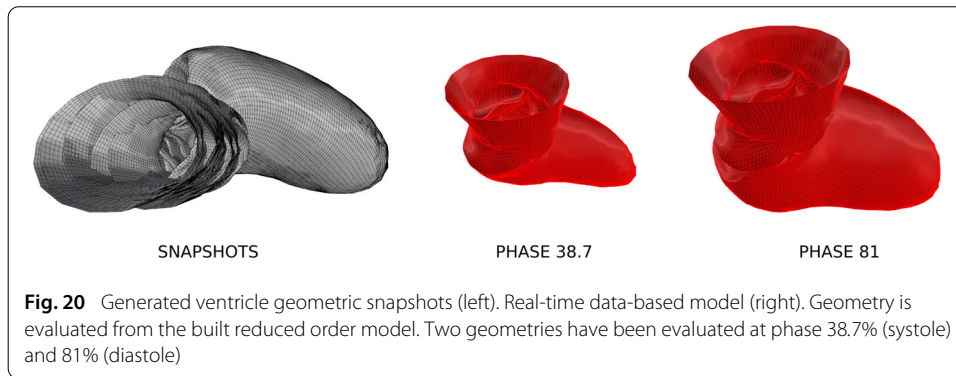
Fig. 16 Feature-aware cuboid decomposition of the casting part for geometric snapshots generation. **a** Radius parameters. **b** Height parameters. **c** Geometry-aware pants decomposition. **d** Feature-aware cuboid decomposition

Table 2 Range of parameters used for the casting part geometric reduced order model

Casting part range of parameters	Minimum	Maximum
Length L	80	81
Radius Rb	120	125
Radius Rh	35	40
Radius Ri	20	25
Height Hb	15	25
Height Hh	140	160

is computed (Fig. 10e). To be more precise, we modify the topology of scanned meshes from disk to cylinder by adding a little hole at the bottom of the ventricule. Afterwards an optimized aligned global parameterization is extracted using leaflet (i.e., the leaflet of the aortic valve) principal curvature direction interpolation for the geometric cross field. A reduced order model is built using the phase (i.e., time) parameter and isotopological analogous snapshots. For instance two geometries have been evaluated at phase 38.7% (systole, i.e., contraction phase) and 81% (diastole, i.e., expansion phase) in Fig. 20. Mesh linear interpolation can be simply performed if quadrilateral meshes have the required properties. An example of consistent interpolation is given in Fig. 21.

Fig. 17 Real-time data-based model. Geometry is evaluated from the built reduced order model. Row (**a**) left to right: parameters and perspective view. Row (**b**) left to right: parameters and perspective view for different parameters

Fig. 18 Three scanned ventricle geometries by corresponding phases

Fig. 19 Ventricle volumetric capacity during a cycle

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.(2021)8:8*

Page 23 of 28

Fig. 20 Generated ventricle geometric snapshots (left). Real-time data-based model (right). Geometry is evaluated from the built reduced order model. Two geometries have been evaluated at phase 38.7% (systole) and 81% (diastole)

Fig. 21 Mesh linear interpolation between phase 0 to phase 20. Phase 0 and phase 20, respectively depicted in black and blue (left). Isotopological non-analogous meshes yield non-consistent interpolated mesh with intersections (right) whereas analogous ones gives the expected result

Table 3 Run times and statistics: average number of faces in the input triangulated meshes, patches in the quadrilateral layout, average computation time per snapshot, number of snapshots in the reduced order model, reduced order model building time, number of modes and evaluation time of the geometry

ROM	avg. Faces	Patches	avg. Time (min)	Snapshots	ROM (s)	Modes	Evaluation (s)
T-Shape part	4406	18	1.6	37	11	12	1.5
Casting part	21680	168	15.6	65	19	15	1.8
Heart ventricle	6179	12	2.8	10	7	10	1.1

Results discussion

Run times and statistics

Any quadrilateral mesh included in the numerical charts range of parameters can be evaluated in seconds. Table 3 shows some statistics related to the whole workflow. B-Rep parametric geometries are turned into structured quadrilateral meshes in minutes. Then, the isotopological analogous constraints are applied to the population of meshes in order to obtain the same representation. Thanks to the same data structure and analogous properties, an accurate reduced order model is built with a prescribed number of modes. For each study, evaluation time is constant because equivalent numerical operations are performed.

Accuracy of the MEG-IsoQuad Method and geometric parametric analysis

Fidelity of the method can be measured in two steps. The first focuses on the quality of the generated quadrilateral meshes with respect to the input triangulated B-Rep. We refer the readers to Campen and Kobbelt[8] and Bommes et al.[6] to have more details on quality espects of computed quadrilaterals. In Fig. 22(a, left), we show that the quadrilateral mesh replicates the B-Rep with fidelity. Viewed contrasted shapes are relative to numerical error. Sharp features are conserved during surface conversion. Notice that for a shape with less localized curvature, the structured output mesh is still satisfactory for a correct sampling of quadrilateral elements. A geometry is evaluated in an unfavorable case and compared with a validation snapshot in Fig. 22(a, middle) which was not used for the reduced order model construction. The studied case is located in a corner of the 9-dimensional sparse grid, far from known information. Despite this difficulty, the evaluated geometry looks quite good with only 12 modes and 37 snapshots to cover this high-dimensional space. Unwanted curvature effects are related to the lack of input geometries and dependent geometric parameters. Accuracy can be significantly improved by adding snapshots to the reduced order model and modes in the truncated basis. Standard kriging responses surfaces will be thus more faithful. Error is given in Fig. 22(a, right).

Secondly, we investigate the truncated basis dimension focalized on a quarter of the casting part. For that purpose, a snapshot is taken and compared with several evaluated geometries approximated by an increased number of modes. In Fig. 22(b, left) 5 modes are used to represent the geometry until 25 in Fig. 22(b, right). Higher is the number of modes, better are the estimated shapes. While global distance error decreases, the error becomes more and more localized.
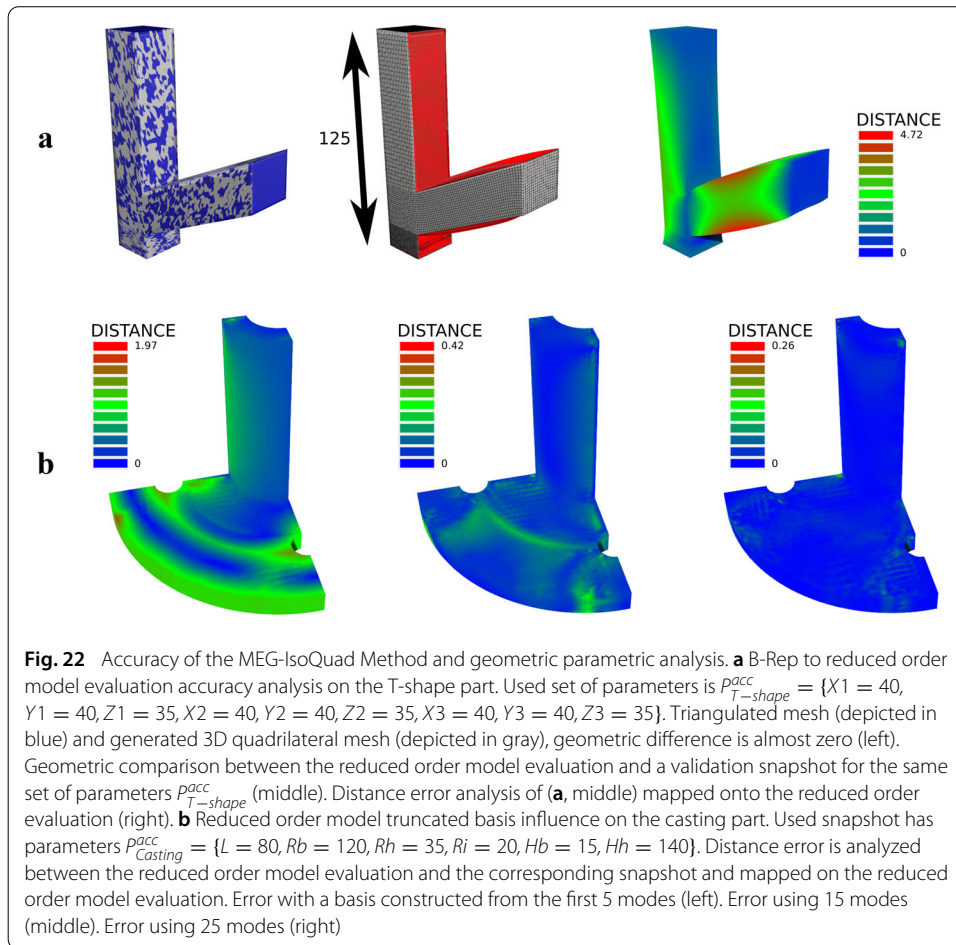
Comparison and contribution for reduced order modeling with geometric parameters

Most of current approaches to build reduced order models with geometric parameters are based on the 2 following methods: mesh morphing and immersed methods. Remark that, for elementary geometric parameters manual techniques are legitimate, see e.g. Lu et al.[37]. To our best knowledge, no recent methods using surface segmentation and global parameterization tools exist to construct accurate geometric numerical charts. Our method employs techniques from the graphics community and adapt them to the needs of reduced order modeling applications with geometric parameters. MEG-IsoQuad technique preserves input mesh features and build isotopological mesh structures directly usable in various SVD algorithms.

Firstable, mesh morphing is for instance operated to study biological shapes. Femur based Principal Component Analysis (PCA) requires mesh projection and hence morphing tools to compare instances with the same data structure, see e.g. Hraiech [24] and Grassi et al. [20]. Morphing is associated with distorted and overlapped elements and as a consequence this method is not robust for large variation of parameters.

On the other hand, researchers employ immersed methods to deal with the same mesh connectivity but different geometries. However, these approaches [13,28] also suffer from some drawbacks such as a clear definition of the integration rule for the cut finite elements.

Fig. 22 Accuracy of the MEG-IsoQuad Method and geometric parametric analysis. **a** B-Rep to reduced order model evaluation accuracy analysis on the T-shape part. Used set of parameters is $P^{acc}_{T-shape} = \{X1 = 40, Y1 = 40, Z1 = 35, X2 = 40, Y2 = 40, Z2 = 35, X3 = 40, Y3 = 40, Z3 = 35\}$. Triangulated mesh (depicted in blue) and generated 3D quadrilateral mesh (depicted in gray), geometric difference is almost zero (left). Geometric comparison between the reduced order model evaluation and a validation snapshot for the same set of parameters $P^{acc}_{T-shape}$ (middle). Distance error analysis of (**a**, middle) mapped onto the reduced order evaluation (right). **b** Reduced order model truncated basis influence on the casting part. Used snapshot has parameters $P^{acc}_{Casting} = \{L = 80, Rb = 120, Rh = 35, Ri = 20, Hb = 15, Hh = 140\}$. Distance error is analyzed between the reduced order model evaluation and the corresponding snapshot and mapped on the reduced order model evaluation. Error with a basis constructed from the first 5 modes (left). Error using 15 modes (middle). Error using 25 modes (right)

Limitations and guarantees

Limitations

Surface segmentation is highly dependent on the mesh discretization and also on the intrinsic geometry. Hence, the smart model decomposition step may have inputs which are problematic. For instance, pants decomposition is known to produce highly deformed segmentations or unwanted geometries [53]. To overcome partially this issue, we propose to use both topological tools and geometric criteria. Cuboid decomposition deals with consistent pants. The quadrilateral layout computed from the cuboid decomposition step is generally coherent but it may fail when exotic layouts are requested: this is the limitation of the constrained border parameterization to determine layout arcs.

Global parameterization step, including cross fields is entirely performed by techniques from the paper of Campen and Kobbelt [8]. We refer the readers to the limitation section of their work.

Guarantees

In spite of limitations expressed above, once quadrilateral meshes are generated, the reduced order modeling operation can not fail.

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 26 of 28

Conclusion

We have presented in this work a method for designing 3D quadrilateral snapshots adapted for data-based applications. Quadrilateral layouts are computed from a geometry-feature-aware pants-to-cuboids decomposition. These layouts replicate the input mesh topology and contain relevant isotopological information. Depending on the geometry, the pants cuboid configurations are chosen by the user to lie on the key features. Using aligned global parameterization, arcs and nodes of the suitable quadrilateral layout are optimized to achieve low overall patch distortion and alignment with sharp features. A key contribution of the present work is the automatic construction of a relevant quadrilateral layout per pant, setting automatically well-adapted cross field singularities in order to compute a good aligned global parameterization.

This allows us to extract a quadrilateral mesh satisfying sharp features, principal curvature directions and possibly boundary constraints. With such features, our method allows to produce isotopological meshes of all the geometric instances. They are created during the offline phase of ROM based parametric studies. Throughout design process of quadrilateral meshes, we have handled data of all geometric instances in a isotopological manner, no matter the input geometry. This helps us to define a set of isotopological analogous meshes using one mesh as reference. The previous sets were successfully used to construct accurate geometric parametric models. These data-based models are evaluated for any given parameter set in near real-time to give the specified geometry.

Declarations

Competing interests
All authors declare that they have no competing interests concerning this presented work.

Author details
[1]Univ Lyon, INSA-Lyon, CNRS UMR5259, LaMCoS, 69621 Lyon, France, [2] Present address: Laboratoire de Mécanique des Contacts et des Structures, 27 bis avenue Jean Capelle, 69621 Villeurbanne Cedex, France, [3]ANSYS, ANSYS France, 11 Avenue Albert Einstein, 69100 Villeurbanne, France.

References
1.   Reduced order model builder (non-official title) by ANSYS, Inc., filed 2019-01-22. U.S. patent application 16/253,635.

Maquart *et al. Adv. Model. and Simul. in Eng. Sci.*(2021)8:8

Page 27 of 28

2.   Al-Akhras H, Elguedj T, Gravouil A, Rochette M. Towards an automatic isogeometric analysis suitable trivariate models generation—application to geometric parametric analysis. Comput Methods Appl Mech Eng. 2017;316:623–45.

3.   Al-Akhras Hassan. *Automatic Isogeometric Analysis Suitable Trivariate Models Generation - Application to Reduced Order Modeling*. PhD thesis, INSA de LYON, École Doctorale MEGA ED 162, 2016.

4.   Al-Akhras Hassan, Elguedj Thomas, Gravouil Anthony, Rochette Michel. Isogeometric analysis-suitable trivariate nurbs models from standard b-rep models. Comput Methods Appl Mech Engrg. 2016;307:256–74.

5.   Beaufort Pierre-Alexandre, Lambrechts Jonathan, Henrotte François, Geuzaine Christophe, Remacle Jean-François. Computing cross fields a PDE approach based on the Ginzburg–Landau theory. Proc Eng. 2017;203:219–31.

6.   Bommes David, Zimmer Henrik, Kobbelt Leif. Mixed-integer quadrangulation. ACM Trans Graphics. 2009;28(3):77.

7.   Bommes David, Campen Marcel, Ebke Hans-Christian, Alliez Pierre, Kobbelt Leif. Integer-grid maps for reliable quad meshing. ACM Trans Graphics. 2013;32(4):98.

8.   Campen Marcel, Kobbelt Leif. Quad layout embedding via aligned parameterization. In: Computer graphics forum, vol 33. Wiley; 2014, p69–81.

9.   Campen Marcel, Bommes David, Kobbelt Leif. Dual loops meshing: quality quad layouts on manifolds. ACM Trans Graph. 2012;31(4):110:1-110:11.

10. Chinesta Francisco, Ammar Amine, Cueto Elías. Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models. Archiv Comput Methods Eng. 2010;17(4):327–50.

11. Chinesta Francisco, Ladeveze Pierre, Cueto Elías. A short review on model order reduction based on proper generalized decomposition. Archives of Computational Methods in Engineering. 2011;18(4):395.

12. Éric Colin de Verdière and Francis Lazarus. Optimal pants decompositions and shortest homotopic cycles on an orientable surface. Journal of the ACM, 54(4): Article No. 18, Jul 2007. A conference version appeared in Proc. Int. Symp. on Graph Drawing (GD). A preliminary version appeared in Abstr. Workshop on Computational Geometry (EuroCG): Europ; 2003. p. 2003.

13. Courard Amaury. *PGD-Abaques virtuels pour l'optimisation géométrique des structures*. PhD thesis, Université Paris-Saclay, 2016.

14. Keenan Crane. Discrete connections for geometry processing. Master's thesis, California Institute of Technology, 2010. http://resolver.caltech.edu/CaltechTHESIS:05282010-102307125.

15. Crane Keenan, Desbrun Mathieu, Schröder Peter. Trivial connections on discrete surfaces. Computer Graphics Forum (SGP). 2010;29(5):1525–33.

16. Dey Tamal K, Li Kuiyu, Sun Jian, Cohen-Steiner David. Computing geometry-aware handle and tunnel loops in 3d models. ACM Transactions on Graphics (TOG). 2008;27(3):45.

17. Erickson Jeff, Whittlesey Kim. Greedy optimal homotopy and homology generators. *16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1038–1046, 2005.

18. Floater Michael S. Mean value coordinates. Computer Aided Geometric Design. 2003;20:19–27.

19. Galland Florent, Gravouil Anthony, Malvesin E, Rochette Michel. A global model reduction approach for 3d fatigue crack growth with confined plasticity. Computer Methods in Applied Mechanics and Engineering. 2011;200(5):699–716.

20. Grassi Lorenzo, Schileo Enrico, Boichon Christelle, Viceconti Marco, Taddei Fulvia. Comprehensive evaluation of pca-based finite element modelling of the human femur. Medical Engineering and Physics. 2014;36(10):1246–52.

21. Hajij Mustafa, Dey Tamal K, Li Xin. Segmenting a surface mesh into pants using morse theory. arXiv.org, arXiv:1608.06368v2, 2016.

22. Allen Hatcher. Algebraic topology. http://www.math.cornell.edu/hatcher, 2001.

23. Hatcher Allen, Lochak Pierre, Schneps Leila. On the teichmüller tower of mapping class groups. Journal Für Die Reine Und Angewandte Mathematik. 2000;521:1–24.

24. Hraiech Najah. *Morphing de maillage et indexation de forme pour la modélisation du fémur humain*. PhD thesis, Rennes 1, 2010.

25. Dey Tamal K, Li Kuiyu, Sun Jian. On computing handle and tunnel loops. *International Conference on Cyberworlds*, pages 357–366, 2007.

26. Dey Tamal K, Sun Jian, Wang Yusu. Approximating cycles in a shortest basis of the first homology group from point data. *ACM 26th Annual Symposium on Computational Geometry*, pages 166–175, 2010.

27. Kälberer Felix, Nieser Matthias, Polthier Konrad. Quadcover-surface parameterization using branched coverings. In *Computer Graphics Forum*, volume 26, pages 375–384. Wiley Online Library, 2007.

28. Kamensky David, Hsu Ming-Chen, Yu Yue, Evans John A, Sacks Michael S, Hughes Thomas JR. Immersogeometric cardiovascular fluid–structure interaction analysis with divergence-conforming b-splines. Computer methods in applied mechanics and engineering. 2017;314:408–72.

29. Loriot Stephen Kiazyk, Sébastien, de Verdière Éric Colin. Triangulated surface mesh shortest paths. In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.9 edition, 2016.

30. Kowalski Nicolas, Ledoux Franck, Frey Pascal. A pde based approach to multidomain partitioning and quadrilateral meshing. In Xiangmin Jiao and Jean-Christophe Weill, editors, *Proceedings of the 21st International Meshing Roundtable*, pages 137–154, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg. ISBN 978-3-642-33573-0.

31. Lee Yunjin, Lee Seungyong, Shamir Ariel, Cohen-Or Daniel, Seidel Hans-Peter. Mesh scissoring with minima rule and part salience. Computer Aided Geometric Design. 2005;22(5):444–65.

32. Li Bo, Li Xin, Wang Kexiang, Qin Hong. Surface mesh to volumetric spline conversion with generalized poly-cubes. *IEEE Transactions on Visualization and Computer Graphics*, 99(PrePrints):1, 2013. ISSN 1077-2626. https://doi.org/10.1109/TVCG.2012.177.

33. Li Wan-Chiu, Vallet Bruno, Ray Nicolas, Levy Bruno. Representing higher-order singularities in vector fields on piecewise linear surfaces. IEEE Transactions on Visualization and Computer Graphics. 2006;12(5):2006.

34. Li Xin, Gu Xianfeng, Qin Hong. Surface mapping using consistent pants decomposition. IEEE Transactions on Visualization and Computer Graphics. 2009;15(4):558–71. https://doi.org/10.1109/TVCG.2008.200 (**ISSN 1077-2626**).

35. Lin Juncong, Jin Xiaogang, Fan Zhengwen, Wang Charlie CL. Automatic polycube-maps. *Advances in Geometric Modeling and Processing*, pages 3–16, 2008.

36. Liu Lei, Zhang Yongjie, Liu Yang, Wang Wenping. Feature-preserving t-mesh construction using skeleton-based polycubes. Computer-Aided Design. 2015;58:162–72.

37. Lu Y, Blal N, Gravouil A. Adaptive sparse grid based hopgd: Toward a nonintrusive strategy for constructing space-time welding computational vademecum. International Journal for Numerical Methods in Engineering. 2018;114(13):1438–61.

38. Lu Ye. *Construction d'abaques numériques dédiés aux études paramétriques du procédé de soudage par des méthodes de réduction de modèles espace-temps*. PhD thesis, Université de Lyon, 2017.

39. Luboz Vincent, Bailet Mathieu, Grivot Christelle Boichon, Rochette Michel, Diot Bruno, Bucki Marek, Payan Yohan. Personalized modeling for real-time pressure ulcer prevention in sitting posture. Journal of tissue viability. 2018;27(1):54–8.

40. Maquart T, Elguedj T, Gravouil A, Rochette M. A note on topological properties of volumes constructed from surfaces. *HAL:* https://hal.archives-ouvertes.fr/hal-02318094/, 2019.

41. Maquart Tristan. *Trivariate models generation from unstructured surface manifolds for isogeometric analysis - application to reduced order modeling with geometric parameters*. PhD thesis, Université de Lyon, INSA de LYON, École Doctorale MEGA ED162, 2019.

42. Maquart Tristan, Wenfeng Y, Elguedj T, Gravouil A, Rochette M. 3d volumetric isotopological meshing for finite element and isogeometric based reduced order modeling. Computer Methods in Applied Mechanics and Engineering. 2020;362:112809.

43. McNeel Robert. Rhinocéros 5 3d, 2017a. https://www.rhino3d.com/, 2017.

44. McNeel Robert. Rhinocommon dll documentation, 2017b. https://developer.rhino3d.com/api/, 2017.

45. Myles Ashish, Zorin Denis. Controlled-distortion constrained global parametrization. ACM Transactions on Graphics (TOG). 2013;32(4):105.

46. Myles Ashish, Pietroni Nico, Kovacs Denis, Zorin Denis. Feature-aligned t-meshes. In *ACM Transactions on Graphics (TOG)*, volume 29, page 117. ACM, 2010.

47. Niroomandi Siamak, Alfaro Icíar, Cueto Elías, Chinesta Francisco. Real-time deformable models of non-linear tissues by model reduction techniques. Computer methods and programs in biomedicine. 2008;91(3):223–31.

48. Ray Nicolas, Li Wan Chiu, Lévy Bruno, Sheffer Alla, Alliez Pierre. Periodic global parameterization. ACM Transactions on Graphics (TOG). 2006;25(4):1460–85.

49. Ray Nicolas, Vallet Bruno, Li Wan Chiu, Lévy Bruno. N-symmetry direction field design. ACM Transactions on Graphics (TOG). 2008;27(2):10.

50. Saboret Laurent, Alliez Pierre, Lévy Bruno. Triangulated surface mesh parameterization. In *CGAL User and Reference Manual*. CGAL Editorial Board, 4.9 edition, 2016.

51. Tutte William Thomas. How to draw a graph. Proc London Math Soc. 1963;13(3):743–68.

52. Zeng Wei, Yin Xiaotian, Zhang Min, Luo Feng, Gu Xianfeng. Generalized koebe's method for conformal mapping multiply connected domains. *2009 SIAM/ACM Joint Conference on Geometric and Physical Modeling*, SPM '09.:89–100, 2009.

53. Zhang K, Li X. Searching geometry-aware pants decomposition in different isotopy classes. Geometry, Imaging, and Computing. 2014;1(3):367–93.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.