INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2010; **00**:1–6 *Prepared using* nmeauth.cls *[Version: 2002/09/18 v2.02]*

High Quality Surface Remeshing Using Harmonic Maps. Part II: Surfaces with High Genus and of Large Aspect Ratio.

E. Marchandise[1], C. Carton de Wiart [1,4], W. G. Vos[3], C. Geuzaine[2] and J-F Remacle[1*]

[1] *Université catholique de Louvain, Institute of Mechanics, Materials and Civil Engineering (iMMC), Place du Levant 1, 1348 Louvain-la-Neuve, Belgium*
[2] *Université de Liège, Department of Electrical Engineering and Computer Science, Montefiore Institute B28, Grande Traverse 10, 4000 Liège, Belgium*
[3] *University Hospital Antwerp, Department of Pulmonology , Antwerp, Belgium*
[4] *Cenaero, Rue des Frères Wright 29, 6041 Gosselies, Belgium*

SUMMARY

This paper follows a previous one that was dealing with high quality surface remeshing using harmonic maps [1]. In [1], it has been demonstrated that harmonic parametrizations can be used as input for surface meshers to produce high quality triangulations. However, two important limitations were pointed out, namely surfaces with high genus and/or of large aspect ratio. This papers addresses those two issues. We first develop a multiscale version of the harmonic parametrization of [1] and then combine it with a multilevel partitioning algorithm to come up with an automatic remeshing algorithm that overcomes the above mentioned limitations of harmonic maps. The overall procedure is implemented in the open-source mesh generator Gmsh [2]. Copyright © 2010 John Wiley & Sons, Ltd.

KEY WORDS: surface remeshing; surface parametrization; STL file format; surface mapping; harmonic map; conformal map; finite elements

1. Introduction

Most of the surface meshing procedures require a parametrization of the surface. The surface mesh is generated in the parametric plane and is subsequently projected onto the 3D surface using the parametrization. Yet, there are situations when the only description of the surface is a triangulation. In the latter case, the geometric triangulation can usually not be used for a finite element analysis. One can either modify the geometric triangulation in 3D or use the triangulation for building a discrete parametrization.

The main driving force in the research on discrete parametrization techniques is Computer Graphics (CG) where discrete parametrizations are used among other things for texture mapping. Various approaches have been proposed in the CG literature and it is possible to

*Correspondence to: emilie.marchandise@uclouvain.be

Received
Revised

classify them as follows: linear, non-linear and hybrid methods. Linear algorithms are efficient, yet they usually do not guarantee the discrete mapping to be one-to-one. It is possible to combine linear algorithms with local checks to guarantee its bijectivity: we called that approach hybrid [1]. Non-linear algorithms are considered to be slow and will not be considered here.

The concept of discrete harmonic map has been described first by Eck [3]. Floater et al. [4] introduced a concept of convex combination map that guarantees that the discrete mapping is one-to-one. Sheffer and Strurler [5] presented a constrained minimization approach, the so-called angle based flattening (ABF), such that the variation between the set of angles of an original mesh and one of the 2D flattened version is minimized. In order to obtain a valid and flipping-free parametrization, several additional constrained algorithms are developed. More recently, they improved the performance of the ABF technique by using an advanced numerical approach and a hierarchical technique [6]. Much research has also been incorporated within the theory of differential geometry. For example, Levy et al [7] apply the Cauchy Riemann equations to compute a least square conformal map. This approach is quite similar to that of Desbrun [8] that minimize a combination of Dirichlet and distortion energy to compute a conformal map for interactive geometry remeshing. More recently, Mullen et al. [9] have also presented spectral computation of the conformal map [9].

Recently, discrete parametrizations methods have received some attention from non CG-specialists and in particular, in the domain of finite element mesh generation. Here, the target application is clearly surface meshing, especially for surfaces that are defined by a triangulation only or for cross-patch meshing. In [1], we have demonstrated that the use of such mappings as parametrizations allowed to generate quality finite element meshes. Yet, two important issues have not been completely addressed: reparametrization techniques fail when the surface has a large aspect ratio and/or when it has a high genus. Those issues are critical in the domain of mesh generation, maybe more than in computer graphics. This paper aims at addressing those two issues.

In [1], we have demonstrated that parametric coordinates computed using a discrete harmonic map become exponentially small for vertices located away from the boundaries of the surface to be remeshed. This makes any 2D remeshing procedure fail because the local coordinates become numerically undistinguishable. If this issue has already been tackled by Alliez et al. in the CG community [10], their proposed solution procedure can however not be used in the context of surface remeshing. In a few words, the idea of Alliez et al. is to partition the zero genus mesh of large geometrical aspect ratio into two parts. The partition is defined by first computing an harmonic map and by evaluating the area distortion map that indicates how the triangles have been shrunk or expanded during the parametrization. Then a vertical line is drawn in the parametric space that passes through the center of gravity of the distortion map and that defines the partition of the mesh. If the authors in [10] could partition this way some simple surfaces with moderate aspect ratio, we found that this area-distortion partitioning algorithm fails for larger geometrical aspect ratio for the same reason the remeshing procedure fails (numerically indistinguishable coordinates). Here, we thus extend the idea of Alliez and develop the concept of multiscale harmonic maps.

The second issue that we address deals with the parametrization of surfaces that are not homeomorphic to a unit disk, i.e., surfaces that do not have zero genus with at least one boundary. In the CG community, many authors use a partition scheme based on the concept of Voronoï diagrams [3] or inspired by Morse theory [7, 11]. The resulting mesh partitions are area-balanced patches that are disk-like. However, this approach results in a

large number of patches and hence a large number of interfaces between those patches. A large number of patches is however not desirable in the context of remeshing because it constrains the final mesh to have mesh edges on those interfaces. Other CG authors introduce seam generation techniques [7, 12] that generate cuts in the surface that are positioned in areas where they cause no texture artifacts. More recently, some authors suggested different methods to compute parametrizations that are globally smooth with singularities occurring at only a few extraordinary vertices [13, 14]. Even though the latter two techniques are attractive in the context of texture mapping, they are less efficient in terms of computational time than partitioning methods combined with a local parametrization method.

The solution strategies presented in this paper for the remeshing of surfaces with large geometrical aspect ratio and arbitrary genus are independent of the chosen type of linear harmonic maps. Among the harmonic maps, we have implemented the Laplacian harmonic map onto a unit disk [1], the convex combination map onto a unit disk [1, 4] and a finite element least square conformal map with open boundaries [9, 10]. The different linear harmonic maps along with the solution strategies presented in this paper are implemented in the open source mesh generator software Gmsh [2]. The different examples show that our method is of high interest for remeshing biomedical triangulations that have a very poor quality input triangulation, or for industrial CAD-based surfaces that contain too many tiny surfaces that are not appropriate for finite element computations.

2. Parametrization of a mesh with harmonic maps

Parametrizing a surface $\mathcal{S}$ is defining a map $\mathbf{u}(\mathbf{x})$

$$\mathbf{x} \in \mathcal{S} \subset \mathcal{R}^3 \mapsto \mathbf{u}(\mathbf{x}) \in \mathcal{S}^* \subset \mathcal{R}^2 \tag{1}$$

that transforms continuously a 3D surface $\mathcal{S}$ into a surface $\mathcal{S}^*$ embedded in $\mathcal{R}^2$ that has a well known parametrization. Such a continuous parametrization exists if the two surfaces $\mathcal{S}$ and $\mathcal{S}^*$ have the same topology, that is have the same genus $G(\mathcal{S})$ and the same number of boundaries $\partial\mathcal{S}_i$, $i = 1, ..., N_B$. The genus $G(\mathcal{S})$ of a surface is the number of handles in the surface[†].

In this work, we consider that the only available representation of a surface $\mathcal{S}$ is a conforming triangular mesh $\mathcal{T}$ in 3D, i.e. the union of a set of triangles that intersect only at common vertices or edges. Let us consider a triangulated surface that has n_V vertices, n_E edges and n_T triangles. The genus $G(\mathcal{T})$ is then given through the Euler-Poincaré formula:

$$G(\mathcal{T}) = \frac{-n_V + n_E - n_T + 2 - N_B}{2}, \tag{2}$$

where N_B is the number of boundaries of the triangulation.

As stated in the introduction, harmonic maps have been chosen for the parametrization because they are easy to compute. As an example, the discrete Laplacian harmonic map can

[†]For example, a sphere has a genus $G = 0$ and $N_B = 0$, a disk has $G = 0$ but $N_B = 1$ and a torus has $G = 1$ and $N_B = 0$

be computed as follows:

$$\left\{\begin{array}{llll} \Delta_\epsilon u = 0, & \Delta_\epsilon v = 0 & \text{in} & \mathcal{T}, \\ u = u_D, & v = v_D & \text{on} & \partial\mathcal{T}_1, \\ \partial_n u = 0, & \partial_n v = 0 & \text{on} & \partial\mathcal{T}\backslash\partial\mathcal{T}_1 \end{array}\right. , \tag{3}$$

where Δ_ϵ denotes the discrete Laplacian operator than can be easily computed with piecewise linear finite elements, and $\mathbf{u}_D(\mathbf{x})$ is the value for the Dirichlet boundary conditions. Figure 1 shows for example such a parametrization for a triangulation of a cylinder. The triangulation has two boundaries $N_B = 2$ and the lowest one is mapped onto the circle of the unit disk by imposing $u_D(\mathbf{x}) = \cos(2\pi l(\mathbf{x})/L)$ and $v_D(\mathbf{x}) = \sin(2\pi l(\mathbf{x})/L)$, with l denoting the curvilinear abscissa of a point along the boundary $\partial\mathcal{T}_1$ of total length L (red arrow in Fig. 1).

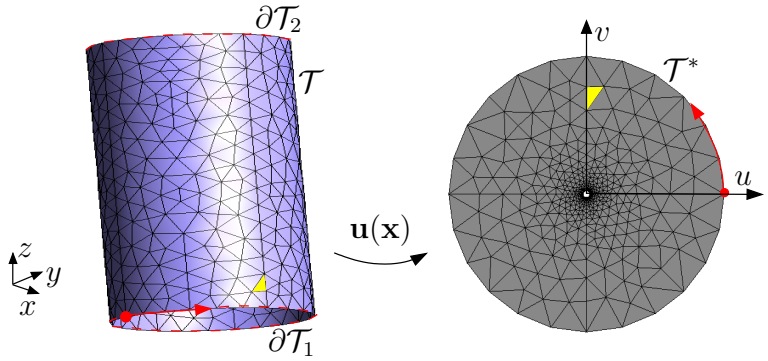


Figure 1. Parametrization. A piecewise linear map creates a correspondence between a 3D surface mesh $\mathcal{T}$and a 2D mesh $\mathcal{T}^*$ of same topology ($G = 0, N_B = 2, \eta = 2$), mapping each triangle from $\mathcal{R}^3$ to $\mathcal{R}^2$.

When going from continuous harmonic maps to linear discrete harmonic maps (3), three different issues can arise. The first issue concerns undistinguishable mapping coordinates. As shown in Fig.1 and as explained in [1] the solution of the mapping becomes exponentially small for vertices located away from $\partial\mathcal{T}_1$. As a consequence, local coordinates u and v of those distant vertices might numerically become indistinguishable. By deriving an analytical solution of Laplacian harmonic maps for cylinders, we showed in [1] that the geometrical aspect ratio of the surface η should be reasonable to be numerically able to distinguish the coordinates.

$$\eta = \frac{H}{D} < \eta_{\max}, \tag{4}$$

where H is the maximal geodesic distance between a mesh vertex and a boundary vertex of $\partial\mathcal{T}_1$ and D is the equivalent diameter of the boundary $\partial\mathcal{T}_1$. As the distance on the 3d mesh is not straightforward to compute, an upper and lower bound for η is computed. The upper bound for η is computed by using the analytical expression for cylinders: $\eta = 2\pi A/L^2_{\partial\mathcal{T}_1}$, where A is the area of the 3D surface mesh and $L_{\partial\mathcal{T}_1}$ is the arc length of the boundary $\partial\mathcal{T}_1$. The lower bound is estimated by choosing for H the maximal size of the oriented bounding box of the 3D surface mesh and for D the maximal size of the oriented bounding box for the boundary curve $\partial\mathcal{T}_1$. The oriented bounding boxes are computed with the fast Oriented

bounding box HYBBRID optimization algorithm presented in [15] which combines the genetic and Nelder-Mead algorithms [16].

As we showed in [1] that $\eta = 4$ corresponds to an area of mapped triangles of about $r_i^2 = 10^{-10}$ (see Eq. (23) and Fig. 10c in [1]), we choose $\eta_{\max} = 4$ as upper limit for the geometrical aspect ratio of the 3D surface mesh. A second issue is about triangle flipping. As the discrete harmonic map has no guarantee to be one-to-one, we suggested in [1] a *local cavity check algorithm* that locally modifies mesh cavities in which flipping occurs. In the case the algorithm fails, we suggest to switch to a guaranteed one-to-one convex combination mapping introduced by Floater [17, 18]. This convex combination is not used as default mapping since the metric tensor associated with this mapping is much more distorted than the one obtained with the harmonic mapping and hence the resulting new mapped mesh is of lower quality. Finally, the last issue concerns triangle flipping that might occur when computing linear conformal maps with open boundaries. Indeed, a linear algorithm cannot guarantee that no triangle folding will occur. To remain efficient in the context of surface remeshing, we use an idea similar to the one suggested by Sheffer et al. [5] that checks the presence of edge folding.

In this section, we have put to the fore in the context of discrete harmonic mapping three limitations, namely limitations on the genus G, the number of boundaries N_B and the geometrical aspect ratio η. Those three criteria can be summarized as follows:

$$\begin{aligned} &i) \quad G = 0; \\ &ii) \quad N_B \geq 1; \\ &iii) \quad \eta < \eta_{\max}. \end{aligned} \tag{5}$$

In the next section, we will first consider a disk-like surface ($G = 0, N_B \geq 1$) and present a novel max-cut partitioning algorithm for those family of surfaces. We will then present an automatic approach for arbitrary genus surfaces that combines this novel algorithm with a multilevel partitioning algorithm.

3. Multiscale Laplace partitioning method

The multiscale algorithm that we present here is an extension a method proposed in [10] that aims at building area-balanced maps. Recall briefly the idea of the max-cut partitioning algorithm of Alliez and his co-authors. Consider the cylindrical surface of Figure 1. In the parametric plane (u, v), elements far from the boundary $\partial\mathcal{T}_1$ have areas that rapidly tend to zero: the map is not area-balanced. The method proposed in [10] is to build a split line in the parametric plane. If we assume that every vertex $\mathbf{u}_i(u_i, v_i),\ i = 1, \ldots, N$ of the triangulation has a unit weight, it is possible to compute both the center of gravity $\mathbf{u}^c = \sum_i \mathbf{u}_i/N$ and the axes of inertia of the set of vertices as the eigenvectors of

$$\mathbf{J} = \begin{bmatrix} \sum_i (u_i^c - u_i)^2 & \sum_i (u_i^c - u_i)(v_i^c - v_i) \\ \sum_i (u_i^c - u_i)(v_i^c - v_i) & \sum_i (v_i^c - v_i)^2 \end{bmatrix}.$$

The split line considered by Alliez et al. passes through the center of gravity and has the direction of the principal axis of inertia. It splits the surface in two parts and produces a quality partitioning of the surface. This "max-cut" partitioning algorithm enables to partition

a non-closed triangulated surface mesh that has a large geometrical aspect ratio (think of arteries, bunny ears, lungs) into two parts. The two mesh partitions can then be subsequently remeshed by computing two different parametrizations. Yet, the method relies on the fact that it is possible to classify triangles in 2D with respect to the partition line. When the aspect ratio is too high, the nodes that are located far away from the Dirichlet boundary have coordinates that cannot be distinguished: it may be impossible to determine if a node is either on the left or on the right of the partition line. Moreover, because of the finite precision of linear system solvers, the mesh in the parameter space may become invalid.

Consider a cylinder of height H and diameter D with an aspect ratio $\eta = H/D = 25$. Applying Alliez's method with a geometry with that kind of aspect ratio leads to the generation of a mesh that is incorrect in the parameter space. Figure 2 shows the result of the harmonic map parametrization into the unit circle. Triangles get smaller and smaller when they get close to the center of the circle. Figure 2 shows all triangles that have areas smaller than 10^{-15}. Those small triangles correspond roughly to 30% of the total amount of triangles of the mesh. The center of gravity $\mathbf{u}^c$ is situated too high so that the split does not partition the mesh in an optimal way, as it is depicted on Figure 2.

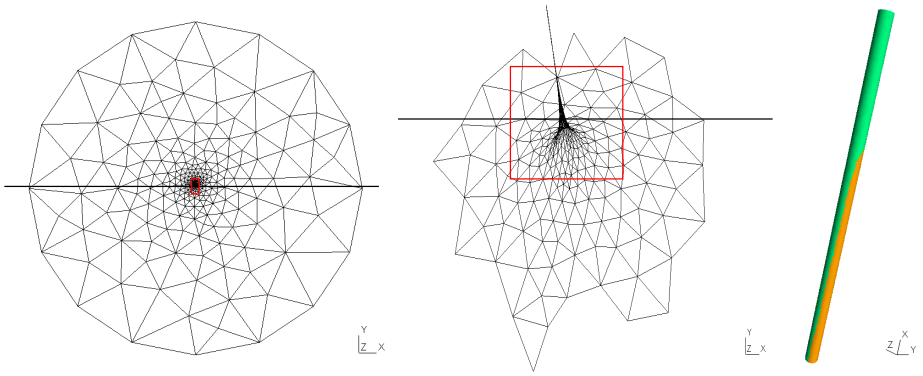


Figure 2. Partitioning a cylinder of aspect ratio $\eta = 25$. Left figure shows the mesh in the parameter space with the computed split line. Center figure shows a zoom in the parameter space with all triangles of size smaller than 10^{-15}. Right figure shows the (incorrect) partitioned mesh that relies on the split line.

The behavior of Alliez's method can become erratic for more complex geometries. Consider the example of Figure 3. Here we consider the geometry of an aorta with an aspect ratio $\eta \simeq 17$[‡]. The surface has $N_B = 13$ boundaries. The image of the partition line is far from being smooth in the real space and the resulting partition is not usable.

Here, we propose a multiscale partitioning algorithm that is a generalization of the partitioning algorithm of Alliez et al. Consider a surface with a high aspect ratio. The elements of the parametric plane that have too small areas form m clustered regions of the plane. Figure

[‡]the geometry has been downloaded from the simbios web site https://simtk.org/frs/download.php?file_id=662

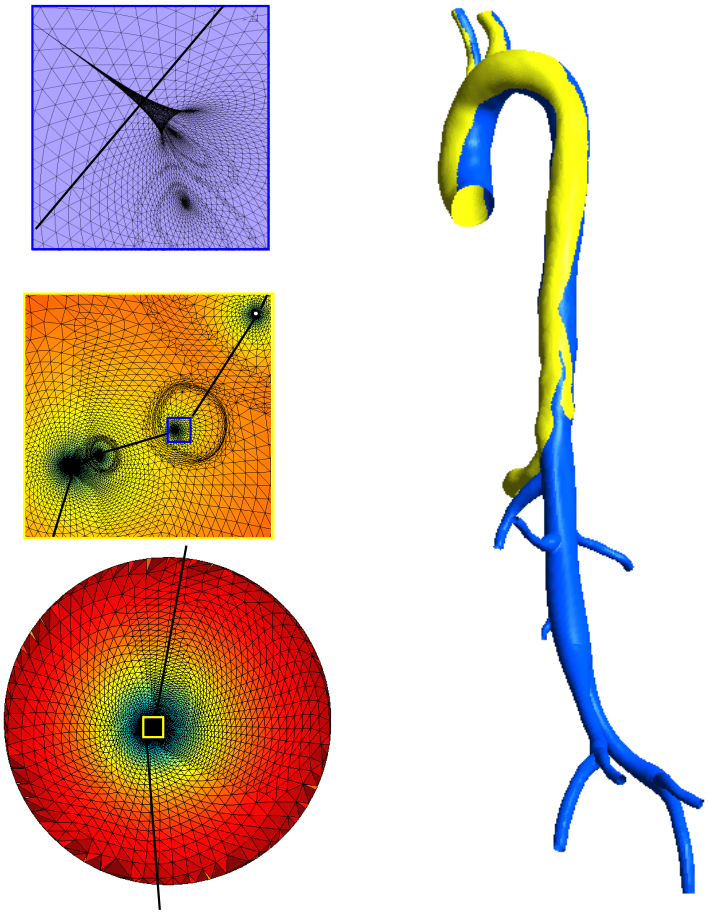


Figure 3. Partitioning an aorta of aspect ratio $\eta = 17$ with $N_B = 13$ boundaries using the area balanced partitioning method of Alliez. Top left figure shows the mesh in the parametric space. The color indicates the area distortion map, that is, the ratio between the area of each triangle in the 2D space and the triangles area in the 3D space. The black line shows the partition line that should split the mesh into two area balanced mesh partitions. However as can be seen on the left middle figure, the mapping could not be computed correctly due to numerical round-offs errors and triangles have been flipped. The resulting mesh partition is presented in the right figure.

4 shows an illustration of the first step of the multiscale partitioning algorithm for a mesh of an aortic artery.

Each of those m subregions can be translated to its center of gravity and rescaled so that its bounding box is of size 1. A new map per subregion can therefore be computed. The latter procedure can be applied recursively to those m subregions until every element of the map has a computable (i.e. sufficiently large) area. At the end, a split line is defined through all levels of the recursion, leading to a procedure that splits the surface into two parts that can

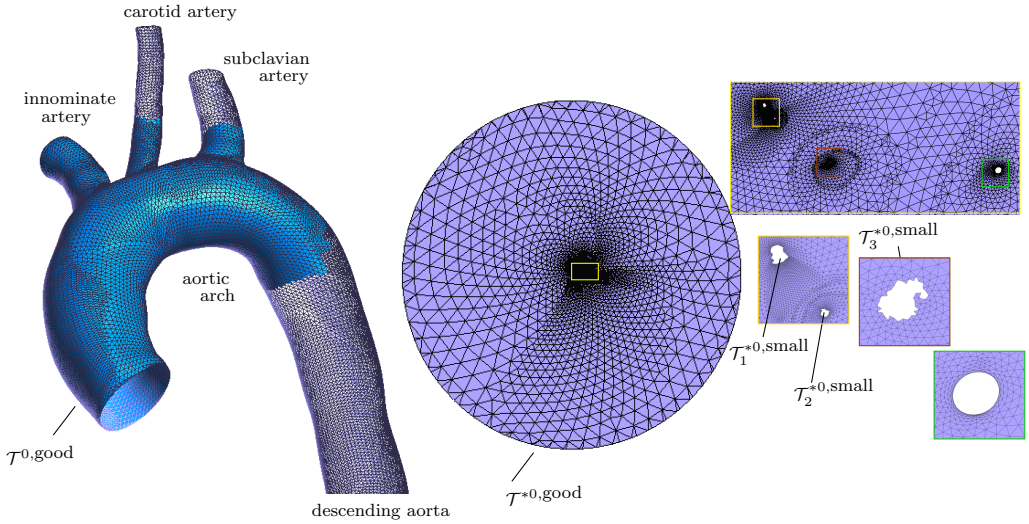


Figure 4. First step of the multiscale partitioning algorithm for the geometry of an aorta. The triangles that have a too small area in the parametric plane define here two clustered regions. Those regions are then withdrawn from the parametrization. The three small white regions visible in the parametric space $\mathcal{T}_i^{0,\text{small}}$, $i = 1, ..., 3$ correspond respectively in the 3D space to the subclavian artery, the carotid artery, and the descending aorta.

be easily parametrized i.e. for which no element is far from the unique boundary.

Our multiscale partitioning method is based on the computation for n levels of discrete Laplacian harmonic maps (denoted $\mathbf{u}^{ij}(\mathbf{x})$, $i = 0, ..., n$) and reads as follows (see Fig. 5):

1. At Level 0. Parametrize the surface triangulation $\mathcal{T}$ with a Laplace harmonic map $\mathbf{u}^{00}(\mathbf{x})$ and a Dirichlet boundary condition that maps one of the boundaries of the mesh $\partial\mathcal{T}_1$ onto the unit circle (red line on Fig.5). The parametrization is computed as the solution of the PDE's equations (3) that are discretized with linear finite elements on the initial STL triangulation (see paper [1] for more details about the harmonic mapping computation).

2. The triangles that have an acceptable size are tagged in the parametric space as $\mathcal{T}^{*0,\text{good}}$ (corresponding to $\mathcal{T}^{0,\text{good}}$ in the real space). See for example the blue part of the geometry on Fig. 5. The triangles that have a mapped area smaller than $\mathcal{A}_{2D}^{\min} = 1.e^{-10}$ are withdrawn from level 0 and are put in sets of connected small triangles $\mathcal{T}_j^{*0,\text{small}}, j = 1, ..., m$. In the case of Fig. 5, there are three sets ($m = 3$) of connected small triangles for level 0[§].

3. At Level $1j$. For each connected triangulation $\mathcal{T}_j^{0,\text{small}}$, define the parametrization $\mathbf{u}^{1j}(\mathbf{x})$ with a Laplace harmonic map and a Dirichlet boundary condition that maps the

[§]Note that the pathological sets of triangles which contain only very few triangles are not taken into account. The triangles belonging to those sets are then tagged as $\mathcal{T}^{*0,\text{good}}$.

boundary $\mathcal{T}^{0,\text{good}} \cap \mathcal{T}_j^{0,\text{small}}$ onto the polygon defined in the 2D parametric space at level 0: $\mathbf{u}_D = \alpha \mathbf{u}^0(\mathbf{x})$. Note that this polygon is scaled with a parameter α that is such that the equivalent diameter for $\mathcal{T}_j^{*0,\text{small}}$ has a value of 1.

4. For the levels $i = 2, ..., n$, repeat step 2 and 3 by replacing level 0 and 1 respectively by level $i-1$ and i until the set of small elements at level i ($\mathcal{T}_j^{*i,\text{small}}$) is empty.

5. Define in the parametric space a splitting line that connects recursively the center of gravities of the small mapped connected triangles $\mathcal{T}_j^{*i,\text{small}}$ to the left and right of the unit disk. For the last level, we take either the barycenter of the elements $\mathcal{T}_j^{*i,\text{good}}$ or the centers of the closed loops inside the 2D mapping (see for example mapping 11 and 30 in Fig. 5).

It must be noted that the multiscale procedure defined here somehow aims at computing one single harmonic mapping with a machine precision that exceeds the double precision. Some of the sub-problems consist in mapping 3D sub-triangulations onto non convex regions: this, in principle, does not guarantee those sub-mappings to be one-to-one. Yet, boundary conditions of the sub-problems arise from upper recursion levels, in a way that the global multiscale problem aims at solving one single harmonic map from $\mathcal{S}$ onto a unit circle (which is convex). This proves that every sub-problem will be well posed and will provide sub-mappings that are one-to-one.

This new multiscale partitioning method has allowed us to partition rather complex surfaces, with very high aspect ratios and a large number of boundaries. Figure 6 presents two examples of the multiscale partitioning algorithm applied to biomedical geometries. A correct partitioning of the aorta of Figure 5 is presented.

We have also run our algorithm for a more complex case: the geometry of human airways of very large geometrical aspect ratio $\eta = 89$ (Figure 6, right). Segmentation of the lung models is performed on low dose CT images, voxel size of $0.5\,mm$, using the commercially available Mimics software package (Materialise, Belgium). A semi-automatic segmentation algorithm places all identifiable airway branches in a separate mask. The smooth 3D models are generated from the segmentation masks. These models include upper airways, all central airways, and the distal airways with a minimal diameter of $1-2\,mm$. This corresponds to terminal airways in the 5th-9th generation. The smoothed models are cut perpendicular to the airway centerline to provide well defined inlet and outlet surfaces. The resulting STL triangulation of the lung models satisfies the topological conditions (5) (i)-(ii)($G = 0, N_B = 147$) so that our multiscale approach can be applied. In what follows, a recursion depth of $n = 15$ has often been reached.

4. Automatic remeshing

In this section, a fast and automatic algorithm is developed that overcomes the three topological and geometrical limitations of harmonic maps (see conditions (5)).

Topological conditions (i) and (ii) require the surface to have a genus $G = 0$ and at least one boundary ($N_B \geq 1$). The way to go is to split the surface into partitions that have the right topology. If $G > 0$, then it can be shown that it is possible to define G independent cuts that divide the surface into subsurfaces that all have $G = 0$. There exist specialized *computational*

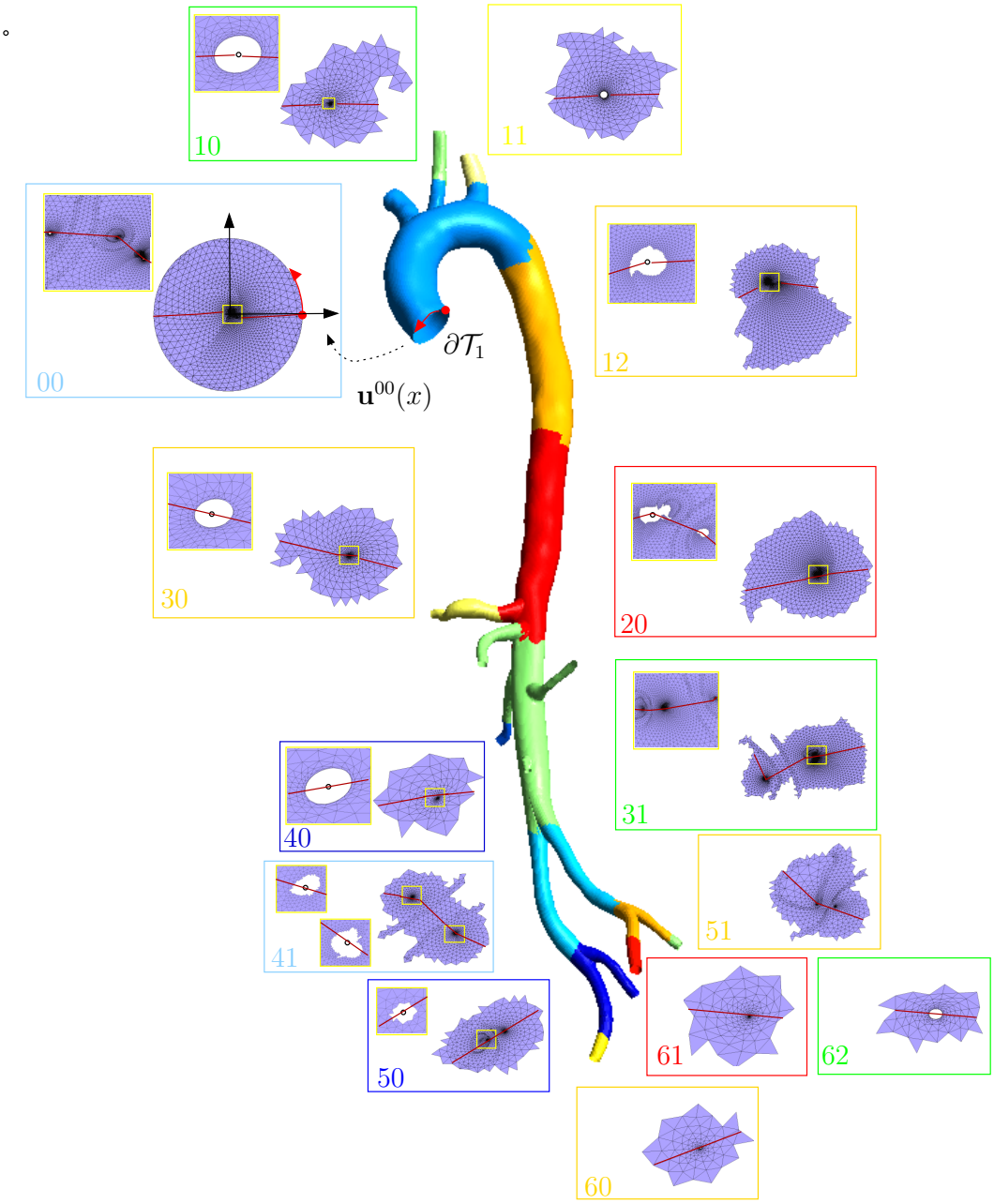


Figure 5. Multiscale Laplace partitioning method of an aorta($G = 0, N_B = 13, \eta = 17$). In this example there are $n = 6$ different levels on which harmonic maps are computed. The red line shows the partition line that recursively splits the mesh into two area balanced mesh partitions (see the resulting mesh partition in Fig.6).

a) b)

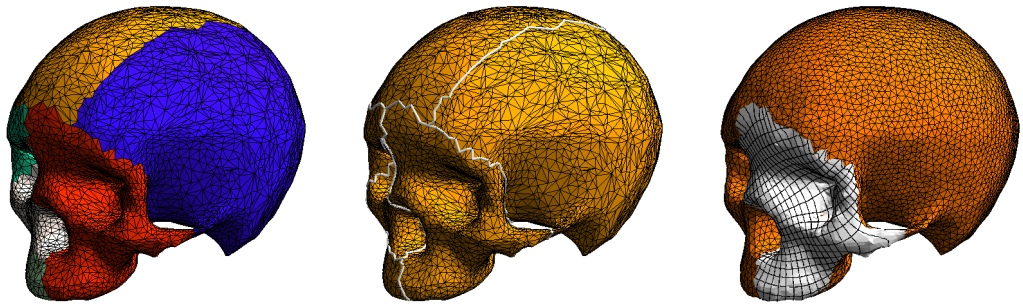
Figure 6. Examples of the multiscale Laplace partitioning method applied to geometries with large geometrical aspect ratio η: a) aorta ($\eta = 17$) and b) human airways ($\eta = 89$).

homology algorithms that aim at constructing such a minimum set of cuts [19]. Yet, those algorithm are usually slow (they act on a coarsened version of the triangulation, with the same topology and use a greedy algorithm to determine the cuts) and they do not always produce optimal cuts in term of their "shape". Here, we look at simpler solutions, even if more than G cuts are used. We use a multilevel edge partitioning software such as Chaco [20] or Metis [21] and partition recursively the triangulation into a minimal number of partitions that satisfy the topological and geometrical conditions. For the geometrical condition (iii) to be satisfied, we use the partitioning method based on the above presented multiscale harmonic map.

The automatic procedure for remeshing a triangulation $\mathcal{T}$ is illustrated in Figure 7 and reads as follows:

1. First, check conditions (i)-(ii) by computing the genus of the triangulation using (2) and the number of boundaries N_B. If the topological conditions are not satisfied, recursively split the mesh with the multilevel partitioning method until satisfied.

2. Next, check condition (iii) by computing approximately the geometrical aspect ratio η from the analytical expression for cylinders of height H and diameter D: $\eta = H/D \approx \pi A/L^2$, where A is the area of the triangulation and L is taken to be the maximal curvilinear length of all the closed boundaries of the triangulation ($L = 2\pi R$ for the



a) b) c)

Figure 7. Remeshing algorithm for a skull geometry. a) Initial triangulation ($G = 2, N_B = 0$) that is cut into different mesh partitions of zero genus, b) Remesh the lines at the interfaces between partitions, c) Compute a harmonic map for every partition and remesh the partition in the parametric space ($\mathbf{u}(\mathbf{x})$ coordinates visible for one partition).

cylinder). In case this condition is not satisfied, split the mesh into two parts with the multiscale laplacian partitioner defined in the previous section.

3. Compute an armonic mapping for every mesh partition.

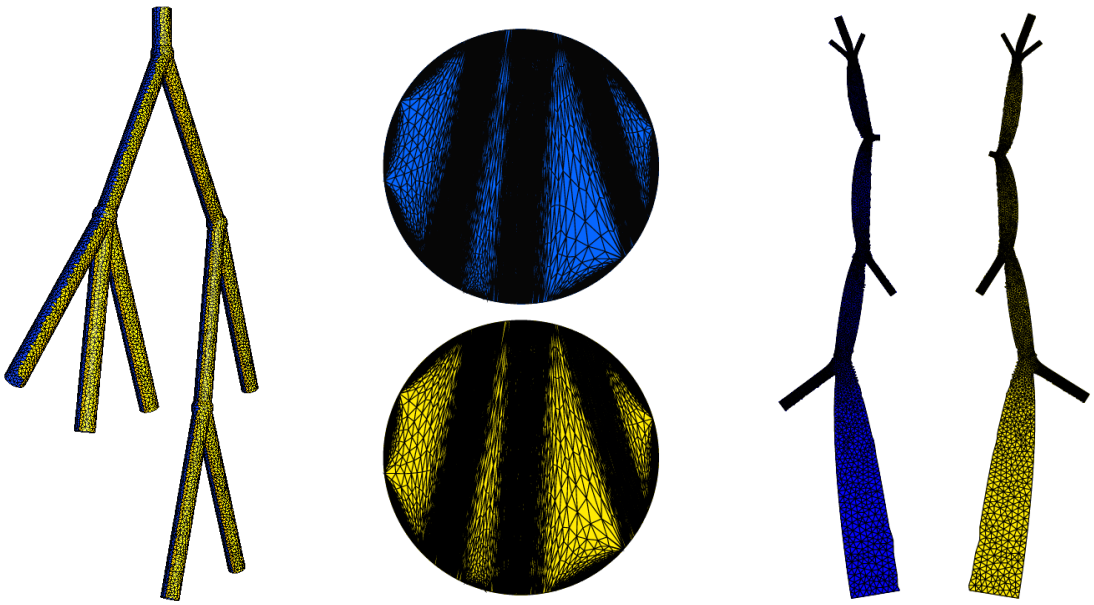
4. Remesh the lines that are the boundaries of the triangulation and the interfaces between the mesh partitions (see the interfaces between colored patches in 7a that are marked with thick white lines in Fig.7b). Those lines are defined as model edges and divided into N parts as follows: $N = \int_0^L ||\mathbf{x}_{,t}|| / \delta dt$, where δ is the prescribed mesh size field. The remeshed lines are embedded in the final mesh (see Fig. 7c).

5. Use standard surface meshers to remesh every partition in the parametric space and map the triangulation back to the original surface.

6. If a volume mesh is needed, generate a volume mesh from the new surface mesh using standard volume meshing techniques (frontal and Delaunay meshers are available in Gmsh).

The automatic procedure is implemented within the open source mesh generator Gmsh [2]. Examples of how to use it can be found on the Gmsh's wiki: `https://geuz.org/trac/gmsh`[¶].

[¶]Access the wiki with username *gmsh* and password *gmsh*



a) b) c)

Figure 8. Remeshing of an arterial tree ($G = 0, N_B = 7, \eta = 28$). The initial mesh is first split into two parts using the multiscale Laplacian partitioning method. Each of those two parts is then mapped in the parametric space by computing (a) a Laplacian harmonic map (b) and a conformal map (c).

5. Results

The first example on Fig.8 shows two different types of harmonic mappings (Laplacian and conformal) for an idealized arterial tree. The geometry has genus $G = 0$ and $N_B = 9$ boundaries, so hat the topological conditions are satisfied. However, the geometrical aspect ratio η is too high $\eta = 28$. Therefore, the input mesh has been partitioned into two parts with our multiscale Laplacian partitioning method. The mapped meshes are presented in Figs. 8b) and c). The mapped mesh obtained with the Laplacian harmonic map onto a unit disk presents lots of mesh distortion while the one obtained with the conformal mapping gives a less distorted parametrization and mesh metric. The remeshing being performed in the parametric space, it is important to keep in mind that the 2D meshing algorithms are more efficient with less distorted mesh metrics. Meanwhile our Gmsh's 2D meshing algorithms (Gmsh MeshAdapt, Delaunay and Frontal) are able to deal with a mesh distortion such as presented in Fig.8b.

The next example presented in Fig. 9 illustrates a uniform remeshing of a human pelvis that has genus $G = 9$ and that is a closed surface , i.e. $N_B = 0$. We compare the quality of the initial STL triangulation (obtained through a segmentation procedure) with the quality of the remeshed pelvis based on a laplacian harmonic map. The quality of the isotropic meshes is evaluated by computing the aspect ratio of every mesh triangle as follows [2]:

$$\kappa = \alpha \frac{\text{inscribed radius}}{\text{circumscribed radius}} = 4 \frac{\sin\hat{a}\sin\hat{b}\sin\hat{c}}{\sin\hat{a} + \sin\hat{b} + \sin\hat{c}}, \tag{6}$$

$\hat{a}, \hat{b}, \hat{c}$ being the three inner angles of the triangle. With this definition, the equilateral triangle has $\kappa = 1$ and degenerated (zero surface) triangles have $\kappa = 0$. For the example of the pelvis, it is observed that the mean $\bar{\kappa}$ and minimum quality κ_{min} of the new mesh are both very high: $\bar{\kappa} = 0.94, \kappa_{min} = 0.62$. Besides, the mean quality measure is found to be constant ($\pm 2\%$) for all examples and hence independent of the initial triangulation and the mesh density.

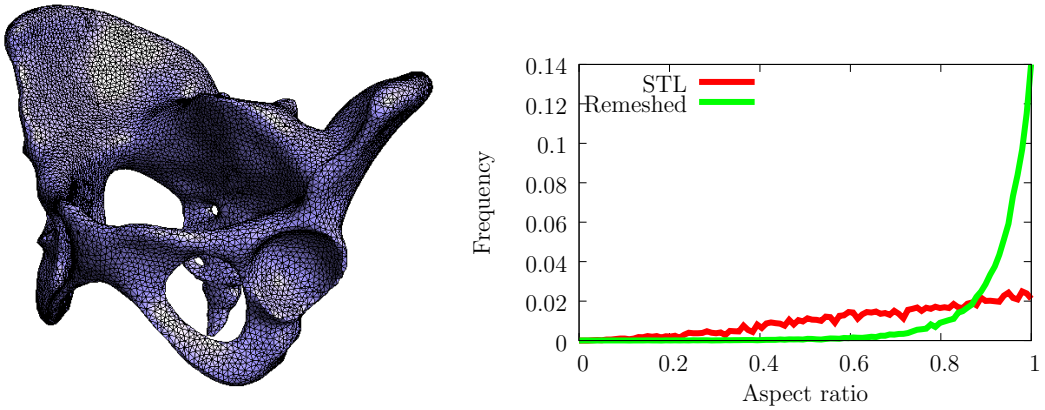


Figure 9. Remeshing of a human pelvis with a harmonic map ($G = 8, N_B = 0$). The left figure shows the remeshed surface and the right figure the quality histogram of both the initial STL file and the remeshed surface based on a Laplacian harmonic map.

The initial STL triangulation is made of 25.836 triangles and the remeshed human pelvis is composed of about the same number of elements (26.313 triangles). Figure 10 shows a pie chart diagram of the time spent in the different steps of the automatic remeshing procedure described in section 4. As can be seen in the diagram, most of the time is spent in the 2D meshing algorithms. The time needed for the parametrization computed with harmonic maps is only 5% of the total time, while the percentage of time needed for the partitioning (combination of the multiscale and multilevel partitioning methods) is less than 20% for this mesh. The total time for remeshing is $9s$ (on a MACBOOK PRO clocked at 2.4 GHz.) for this example. We also

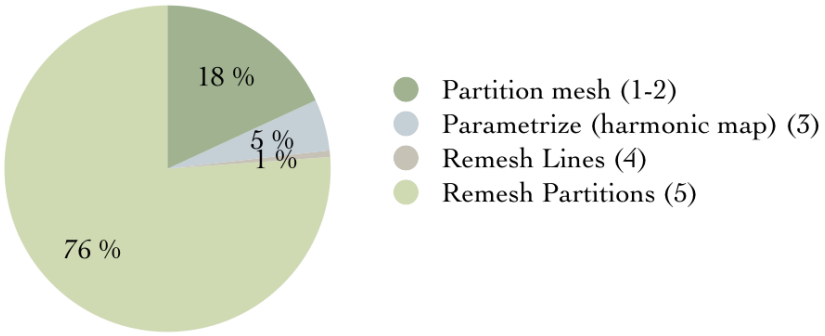


Figure 10. Time spent in the different steps of the automatic remeshing procedure described in section 4 fr th mesh model of the human pelvis $25k$ triangles. The total time for remeshing is $9s$.

Remeshing	Number of partitions	Partition time (s)	Parametrization time (s)	Total remeshing time (s)
LSCM Levy [7] (1.3Ghz)	23	30	95	–
Eck [3] (1.3Ghz)	88	-	-	33.5
ABF++ Zayer [22]	2	-	13	-
LinABF Zayer [22]	2	-	2	-
Presented method (2.4Ghz)				
* laplacian partitioner	2	16.7	1.4	25
* multilevel partitioner	10	7	1.4	14

Table I. Remeshing statistics and timings for the bunny mesh model of $70k$ triangles (new mesh of $25k$ triangles). Comparison (when available) of the presented method with other techniques presented in the Computer Graphics community.

compare the proposed method with two other remeshing packages presented in the literature. We first consider the well-known bunny mesh model of $70k$ triangles presented in Fig.11[‖]. The original mesh has $N_B = 5$ holes and its genus is $G = 0$. We compare in table I some statistics and timings of our algorithm with the least square conformal map (LSCM) of Levy et al. [7], with the multiresolution remeshing of Eck et al. [3] and with the angle based parametrization (ABF) of Zayer [22]. The table as well the timings for our remeshing procedure considering the two different partitioners.

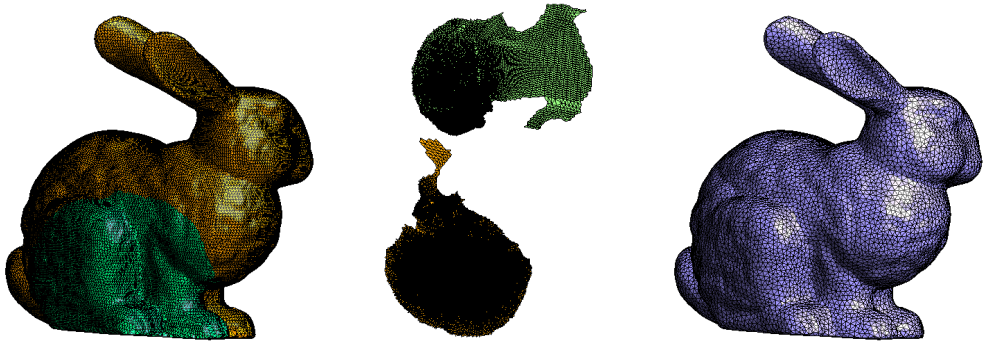


Figure 11. Remeshing of the bunny mesh model of $70k$ triangles ($G = 0, N_B = 5$). Left figure shows the two partitions, middle figure shows the conformal harmonic parametrization that has been computed for both mesh partitions and right figure shows the remeshed bunny with about $25k$ triangles.

Another standard test case that is used in the litterature is the feline mesh model of $50k$ vertices ($G = 0, N_B = 0$) presented in Fig. 12. Statistics on the remeshing procedure are presented in table II. The first remeshing package we compare with is a remeshing method

[‖]The model can be downloaded at the following web site:
`http://www.sonycsl.co.jp/person/nielsen/visualcomputing/programs/bunny-conformal.obj`

Mesh	Vertices	Min θ	Mean θ	L_2 Error	Remeshing time(s)
Original in [23, 24]	49 864	3.8°	40°	-	
Remeshed in [23, 24]	10 825	7.4°	48.3°	0.64%	74
Remeshed in [25]	256	–	–	–	900
Remeshed with presented method	12 946	11.6°	48.7°	0.53%	19

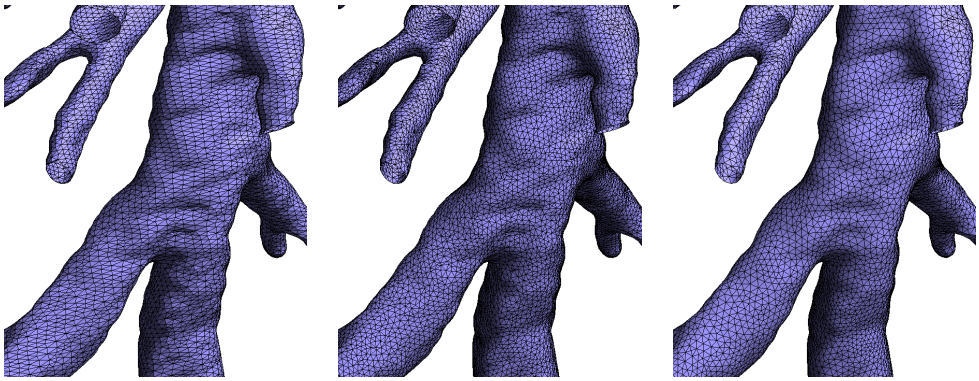
Table II. Remeshing statistics for the feline mesh model of $50k$ vertices ($95k$ triangles) with the remeshing time (new mesh of $25k$ triangles).

based on a local parametrization approach [23, 24] and the second approach is based on a global parametrization approach with an atlas of open base charts that covers the base mesh [25]. Table II shows the statistics of the re-meshes of the feline model along with the computational time for the remeshing procedure. The statistics are given in terms of the minimal angle of the triangles. For a high-quality mesh, the minimum of these values should be no less than $\theta = 10°$, and the average should be no less than $\theta = 45°$. We also present the geometrical remeshing error as the Hausdorff distance normalized by the bounding box diagonal, obtained using the Metro tool [26] and show the remeshing time. Results in table II show that our approach, besides from generating meshes with the highest quality with respect to all criteria, is more efficient in terms of computational time than the two other approaches.

Figure 12. Remeshing of the feline model ($G = 0, N_B = 0$) of $50k$ vertices with the presented method. Left figure shows the initial mesh that has been automatically partitionned into 13 patches and right figure shows the remeshed feline model.

Optimization of surface triangulations is very important in the context of biomedical geometries where the triangulations are constructed mostly from medical images through a segmentation procedure. We further illustrate the capabilities of our algorithm by showing

the remeshing of airways models presented in Figs.13 and 14. The presented algorithm is compared with the meshing algorithm of Mimics. Mimics uses a two step mesh adaptation strategy in order to optimize the initial STL triangulation: in a first step three iterations of remeshing improve skewness to a minimal value of 0.4. In this step the maximal edge length is set to $0.5\,mm$ and the maximal geometrical error to $0.01\,mm$. Hereafter a quality preserving triangle reduction is performed. Again three iterations are done with a maximal edge length set to $0.5\,mm$ and the maximal geometrical error set to $0.05\,mm$. This provides the final remeshed model with Mimics. In comparison, our technique relies on a multiscale partitioning of the airway models into two parts of moderate geometrical aspect ratio. Each of those two parts is then parametrized using a harmonic map and the remeshing is then performed in the parametric plane using standard 2D meshing algorithms (MeshAdapt, Delaunay or frontal). As can be seen in Figs.13 and 14, the quality of the meshes obtained with a remeshing procedure based on a harmonic map is much higher than with the mesh adaptation strategies (such as the one implemented in Mimics).



a) b) c)

Figure 13. Remeshing of human lungs: a) part of the initial STL triangulation, b) remeshed geometry with Mimics (after 2 steps) and c) remeshed lung based on the Harmonic mapping remeshing procedure.

The next example presented in Fig. 15 shows parts of the initial STL mesh of a complete aorta illustrated in Figs. 5 and 6 and the mesh after the remeshing procedure. The remeshing procedure aims at removing the very small elements present in the initial mesh at the junctions of the arteries. The remeshing has been performed by first partitioning the mesh using the multiscale partitioning method. The two resulting mesh partitions are then suitable for the computation of harmonic maps and the remeshing procedure is finally performed in the parametric space.

Remeshing with parametrizations based on harmonic maps can also be very interesting for computational mechanics. In many cases, the surfaces are designed using a CAD system composed of multiple patches. When their sizes are of the same order of magnitude as the mesh size, those patches can result in very distorted mesh elements and impact dramatically on the quality of the boundary layer mesh and, hence, the CFD solution. For those small patches, the remeshing based on cross-patch parametrization enables to remove the small elements present in the cad-based meshes.

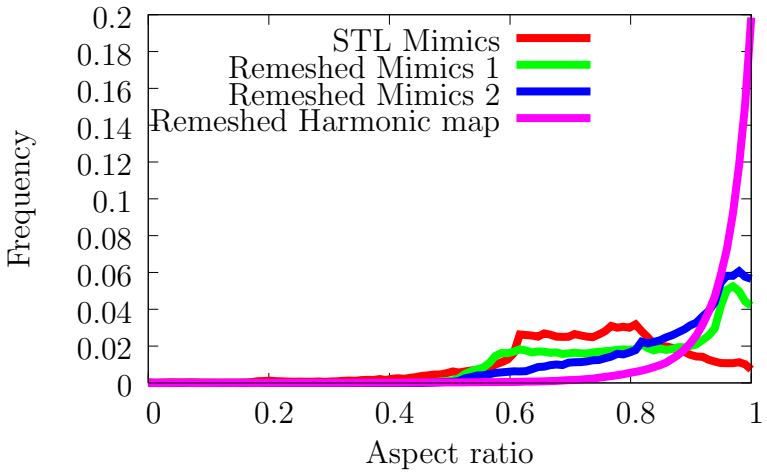


Figure 14. Remeshing of human lungs with the presented algorithm as compared with a commercial package such as Mimics.

Figure 16a shows a zoom of an initial mesh of a landing gear consisting of about 10^6 triangular mesh elements composed of 852 different patches. For this example, the small elements presented in the initial cad-based surface mesh (see for example Fig.17a) prevented us to build any CFD volume boundary layer mesh. Thanks to the cross-patch parametrization (Figs. 16b and 17b), we were able to reduce the number of patches to only 291 surface patches and build a suitable CFD boundary layer mesh of 12M nodes for that model (Fig. 19).

Figure 18 shows the the quality histograms of the two surface meshes: the cad-based mesh and remeshed mesh. We can see that the very small elements have been removed during the remeshing procedure. A preliminary computation has been done using the in-house flow solver developped at Cenaero [27]. The flow Reynolds number based on the diameter of the shock strut is $Re = 73000$ and the Mach number is $Ma = 0.166$. The flow is supposed to be fully turbulent. As the mesh is not fine enough to capture correctly all the turbulent scales, a Delayed Detached-Eddy Simulation [28] (DDSS) turbulence model is used. The DDES is a hybrid RANS (Reynolds-Averaged Navier Stokes)/LES (Large Eddy Simulation) method which uses a RANS modelling in the attached boundary layer and a LES model for the detached flow. The simulation ran on 200 CPU and the time needed to compute one convective time ($t_c = L/U_c$) is 3 hours. Fig. 20 shows the instantaneous pressure on the surface of the landing gear. An improvement of the volume mesh is in progress in order to obtain a better representation of the wake. The results will be used to measure, with an acoustic solver, the sound created by the landing gear. This case has been studied within the frame of the BANC (Benchmark problems for Airframe Noise Computations) Workshop.

The definition of the patches to be cross-parametrized is performed by the user but takes few time regarding the complexity of the CFD computations. For the industrial example of the landing gear, the definition of the new patches and the remeshing takes only 1 hour on 1 CPU compared to 3 hours on 200 CPU for the CFD simulation. Besides, as the remeshing method based on cross-patch parametrization is fast and simple, it can be easily integrated and automated in an industrial CFD computation chain.

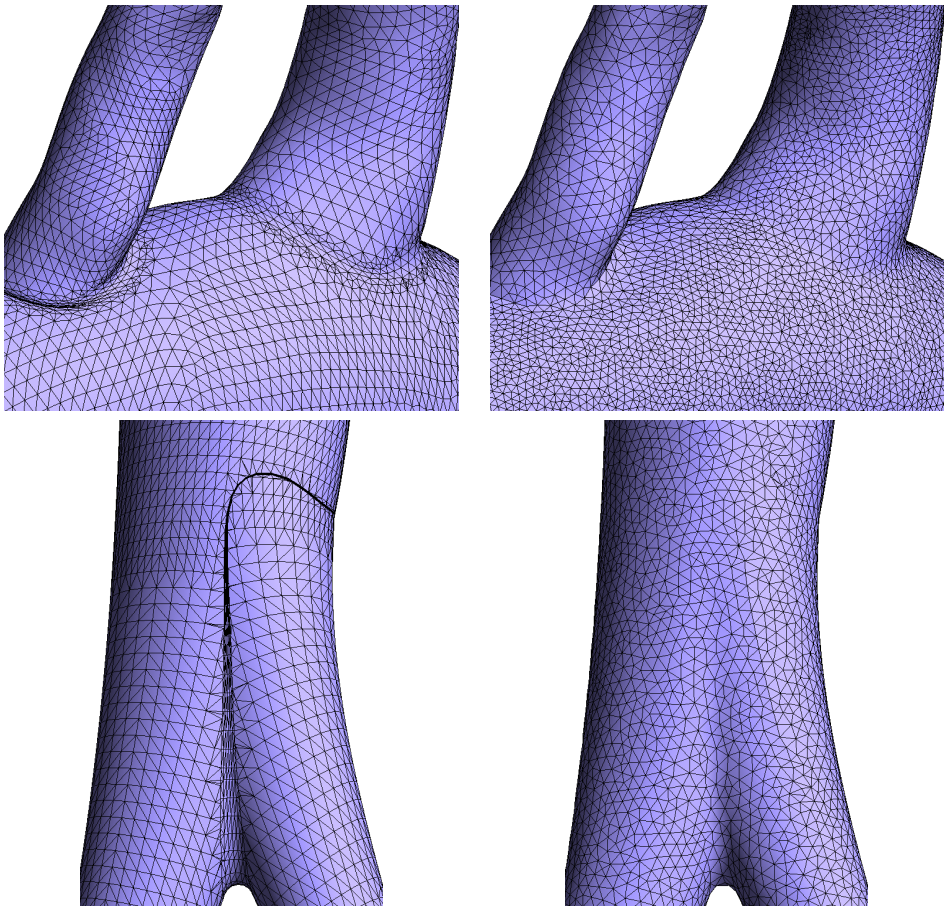
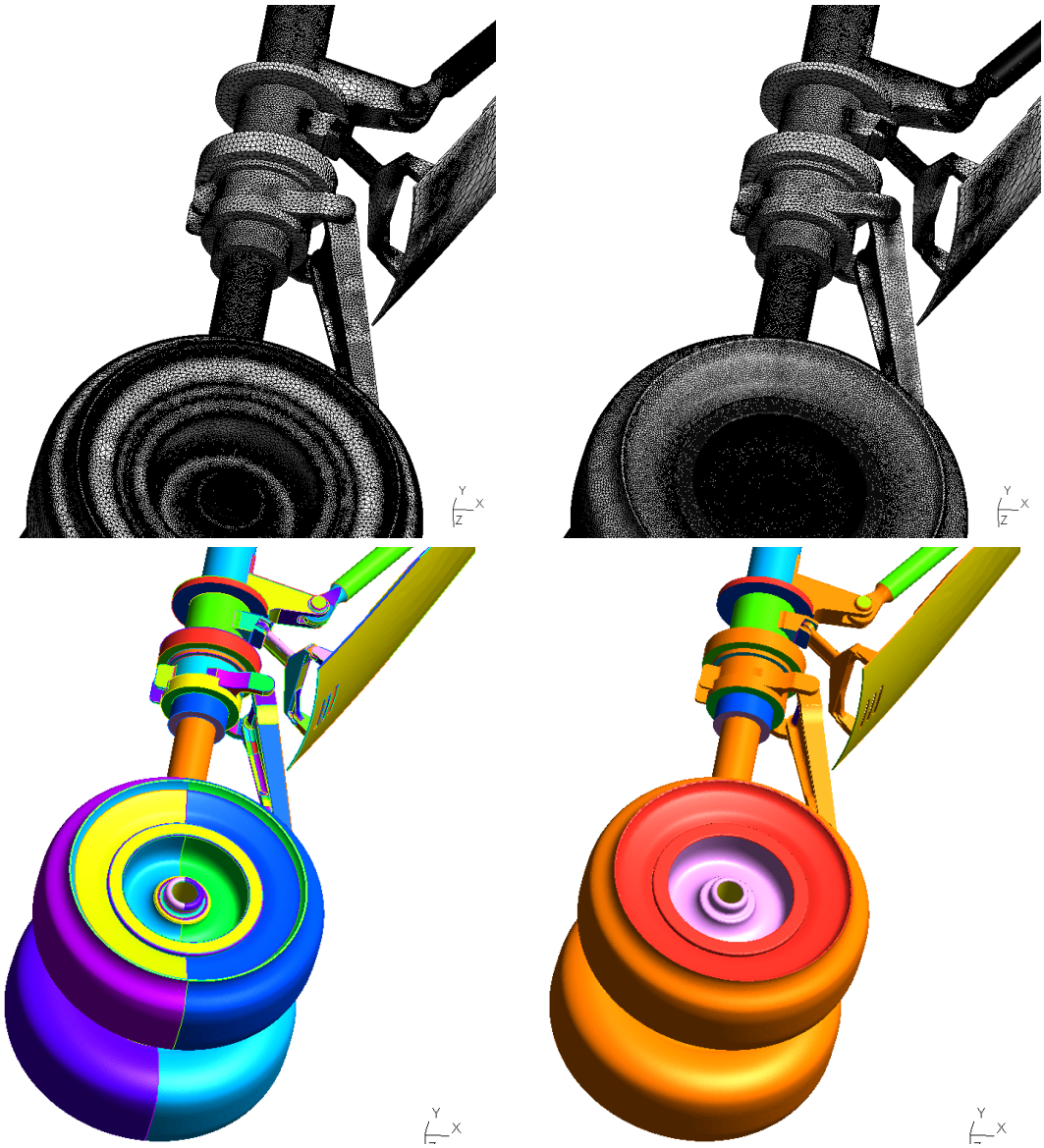


Figure 15. Remeshing of a the aortic artery presented in Fig. 5. We show parts of the mesh near the bifurcations before (left figures) and after (right figures) the remeshing procedure based on a Laplacian harmonic map.

6. Conclusion

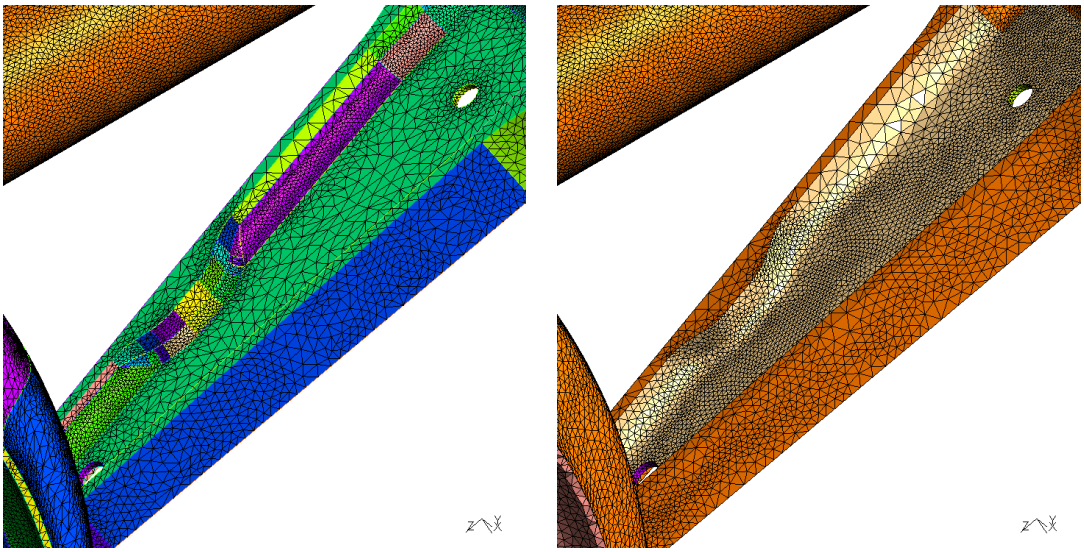
In this work, we have presented a fully automatic approach based on harmonic mappings for remeshing surfaces that overcomes the limitations of harmonic maps: namely limitations on the geometrical aspect ratio and on the genus of the surface. The approach is original as it combines an novel multiscale harmonic partitioning method with a multilevel partitioning algorithm to come up with an automatic remeshing algorithm that overcomes the limitations of harmonic maps. With the present approach, we are able to remesh any surface with any topological genus and with large geometrical aspect ratios. We showed that the remeshing procedure produces high-quality meshes and that it can be used for fairly complex biomedical and industrial applications.



a) b)

Figure 16. Cross-patch remeshing of a landing gear (with patches of $G = 0$, $N_B = 1$). The left figures a) show a surface mesh that is made of 582 geometrical patches and the right figures b) show the remeshed surface made of only 291 patches. The remeshed patches are computed using cross-patch parametrization.

ACKNOWLEDGEMENTS



a) b)

Figure 17. Remeshing of a landing gear. Left Figure shows a part of the initial mesh with many patches and right Figure shows the mesh with the reparametrized patches.

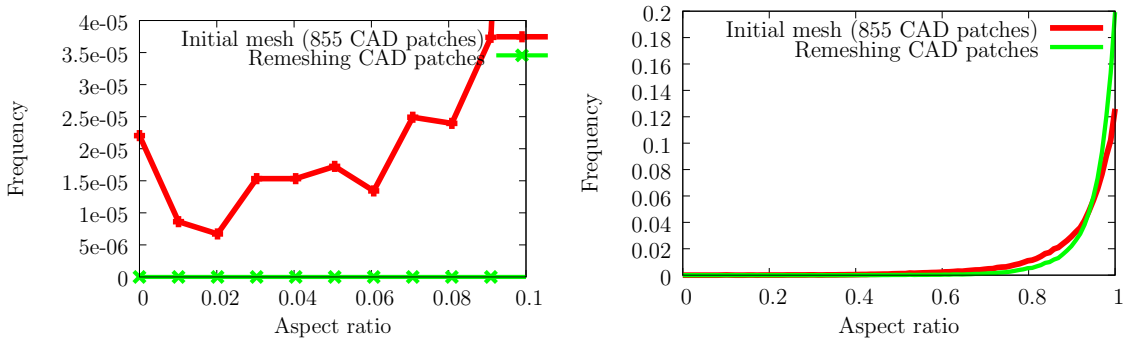


Figure 18. Quality histogram for the remeshing of a landing gear. The left figure shows a zoom for range of small aspect ratio $\kappa \in [0 : 0.1]$ and the right figure the whole range of aspect ratio $\kappa \in [0 : 1]$. As can be seen, the remeshing procedure has removed the small elements by merging different patches using a cross-patch parametrization.

The authors gratefully acknowledge the Walloon Region (EFCONIVO Project) for their financial support. The authors would also like to thank the people from FluidDA n.v. for providing the airway models and their feedback in the analysis of the results.

REFERENCES

1. Remacle JF, Geuzaine C, Compère G, Marchandise E. High quality surface remeshing using harmonic maps. *International Journal for Numerical Methods in Engineering* 2010; **83**(403-425).

Figure 19. Remeshing of a landing gear. Figure shows part of the 3D mesh boundary layer mesh around the landing gear.

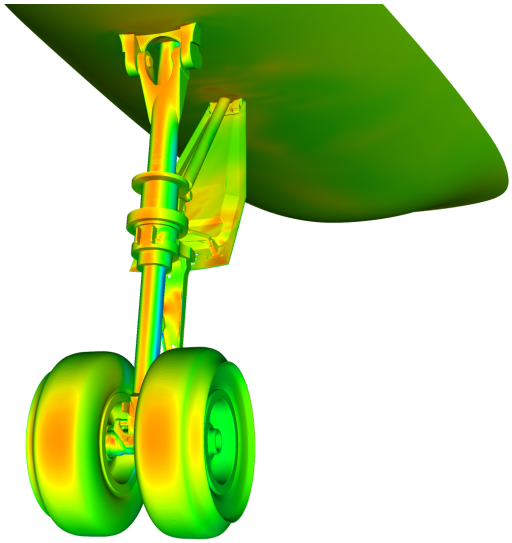


Figure 20. Instantaneous pressure field on the surface of the landing gear.

2. Geuzaine C, Remacle JF. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering* 2009; **79**(11):1309–1331.
3. Eck M, DeRose T, Duchamp T, Hoppe H, Lounsbery M, Stuetzle W. Multiresolution analysis of arbitrary meshes. *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive*

techniques, 1995; 173–182.
4. Floater MS. Parametrization and smooth approximation of surface triangulations. *Computer aided geometric design* 1997; **14**(231-250).
5. Sheffer A, de Sturler E. Parameterization of faceted surfaces for meshing using angle-based flattening. *Engineering with Computers* 2001; **17**(3):1435–5663.
6. Sheffer A, Lévy B, Lorraine I, Mogilnitsky M, Bogomyakov E. Abf++: fast and robust angle based flattening. *ACM Transactions on Graphics* 2005; **24**(311-330).
7. Levy B, Petitjean S, Ray N, Maillot J. Least squares conformal maps for automatic texture atlas generation. *Computer Graphics (Proceedings of SIGGRAPH 02)*, 2002; 362 – 371.
8. Alliez P, Meyer M, Desbrun M. Interactive geometry remeshing. *Computer graphics (Proceedings of the SIGGRAPH 02)* 2002; :347–354.
9. Mullen P, Tong Y, Alliez P, Desbrun M. Spectral conformal parameterization. *In ACM/EG Symposium of Geometry Processing*, 2008.
10. Alliez P, Meyer M, Desbrun M. Interactive geometry remeshing. *Computer graphics (Proceedings of the SIGGRAPH 02)* 2002; :347–354.
11. Shinagawa Y, Kunii T, Kergosien YL. Surface coding based on morse theory. *IEEE Computer Graphics and Applications* 1991; **11**(5):66–78.
12. Gu X, Gortler S, Hoppe H. Geometry images. *ACM SIGGRAPH*, 2002; 656–361.
13. Gu X, Yau S. Global conformal surface parameterization. *Proceedings of the first symposium on geometry processing*, 2003; 127–137.
14. Khodakovsky A, Litke N, Shröder P. Globally smooth parameterizations with low distortion. *ACM Transactions on Graphics* 2003; **22**(3):350–357.
15. Chang CT, Gorissen B, Melchior S. Fast computation of the minimal oriented bounding box on the rotation group so(3). *ACM Transactions on Graphics* 2009, to appear; .
16. Durand N, Alliot JM. A combined nelder-mead simplex and genetic algorithm. *GECCO'99: Proc. Genetic and Evol. Comp. Conf*, 1999; 1–7.
17. Floater MS. Parametric tilings and scattered data approximation. *International Journal of Shape Modeling* 1998; **4**:165–182.
18. Floater MS. One-to-one piecewise linear mappings over trinagulations. *Math. Comp* 2003; **72**(685-696).
19. Kaczynski T, Mischaikow K, Mrozek M. *Computational Homology*, *Applied Mathematical Sciences*, vol. 157. Springer, 2004.
20. Hendrickson B, Leland R. A multilevel algorithm for partitioning graphs. *In Proc. Supercomputing*, 1995.
21. Karypis G, Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs. *International Conference on Parallel Processing*, 1995; 113–122.
22. Zayer R, Lévy B, Seidel HP. Linear angle based parameterization. *ACM/EG Symposium on Geometry Processing conference proceedings*, 2007.
23. Surazhsky V, Alliez P, , Gotsman C. Isotropic remeshing of surfaces: a local parameterization approach. *in proceedings of 12th International Meshing Roundtable,*, Santa Fe, New Mexico, USA, 2003; 215–224.
24. Surazhsky V, Gotsman C. Explicit surface remeshing. *Eurographics Symposium on Geometry Processing*, vol. 17-28, L Kobbelt HHE P Schröder (ed.), 2003.
25. Guskov I. Manifold-based approach to semi-regular remeshing. *Graphical Models* 2007; **69**(1):1–18.
26. P Cignoni CR, 1998 RS. Metro : Measuring error on simplified surfaces. *Computer Graphics Forum* 1998; **17**(2):167–174.
27. Georges L, Winckelmans GS, Geuzaine P. Improving shock-free compressible rans solvers for les on unstructured meshes. *Journal of Computational and Applied Mathematics* 2008; **215**(2):419–428.
28. Spalart P, Deck S, Schur M, Squires K, Strelets M, Travin A. A new version of detached-eddy simulation, resistant to ambiguous grid densities. *Theoretical and Computational Fluid Dynamics* 2006; **11**(8):181–195.