

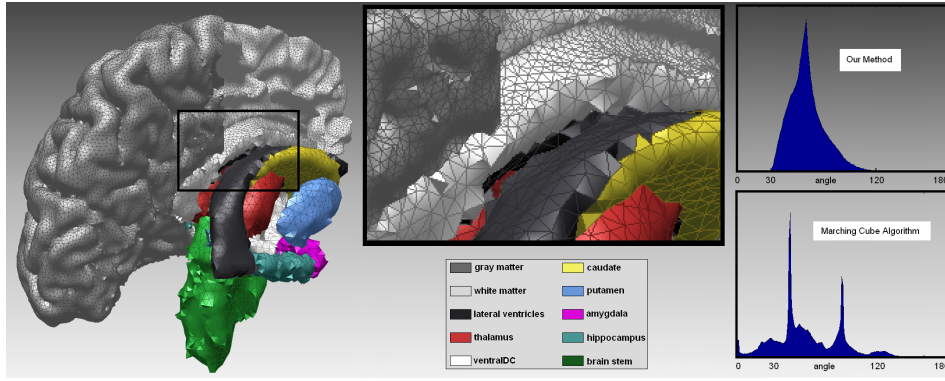
# High-Quality Consistent Meshing of Multi-Label Datasets

J.-P. Pons<sup>1</sup>, F. Ségonne<sup>1</sup>, J.-D. Boissonnat<sup>2</sup>, L. Rineau<sup>2</sup>, M. Yvinec<sup>2</sup>, and R. Keriven<sup>3</sup>

<sup>1</sup> WILLOW, ENPC / ENS / INRIA, Paris, France

<sup>2</sup> GEOMETRICA, INRIA, Sophia-Antipolis, France

<sup>3</sup> ODYSSEÉ, ENPC / ENS / INRIA, Paris, France



**Fig. 1. Surface meshes.** *Left:* Meshing the boundaries of brain tissues with a tissue-dependent resolution. We requested the cortical surface (colored in dark gray) to have a resolution twice higher than other tissues. As a result, output surface meshes of other anatomical structures have a coarse resolution except in *local* regions where they neighbor gray matter. This is apparent in the magnified view of the white matter mesh (in light gray), in which the gray matter interface has been partially removed for visualization purposes. *Right:* The quality of the obtained surface meshes is much higher than with the marching cubes algorithm.

**Abstract.** In this paper, we extend some recent provably correct Delaunay-based meshing algorithms to the case of multi-label partitions, so that they can be applied to the generation of high-quality geometric models from labeled medical datasets. Our approach enforces watertight surface meshes free of self-intersections, and outputs surface and volume meshes of the different tissues which are consistent with each other, including at multiple junctions. Moreover, the abstraction of the tissue partition into an oracle that, given a point in space, answers which tissue it belongs to, makes our approach applicable to virtually any combination of data sources. Finally, our approach offers extensive control over the size and shape of mesh elements, through customizable quality criteria on triangular facets and on tetrahedra, which can be tuned independently for the different anatomical structures. Our numerical experiments demonstrate the effectiveness and flexibility of our approach for generating high-quality surface and volume meshes from real multi-label medical datasets.

# 1 Introduction

## 1.1 Motivation

The generation of realistic geometric patient models from high-resolution medical images is of great significance in many clinical and research applications. An increasing number of numerical simulations of physical or physiological processes (e.g. electroencephalography (EEG) and magnetoencephalography (MEG) [?,?], image-guided neurosurgery [?,?], electromagnetic modeling [?,?], ...) require geometrically-accurate and topologically-correct models. The latter consist either of surface meshes representing boundaries between different anatomical parts, or of volume meshes.

However, due to the lack of reliable fully-automated tools for the unstructured discretization of medical datasets, simplistic geometric models are still of wide use. For example, in electromagnetic modeling, such as specific absorption rate studies, for which finite element methods (FEM) on unstructured grids conforming to anatomical structures would be desirable [?], most numerical simulations have been conducted using finite difference methods on rectilinear grids, although the poor definition of tissue boundaries (stair-casing effect) strongly limits their accuracy. Similarly, in the EEG/MEG source localization problem using the boundary element method (BEM), as pointed out in [?], popular simplistic head models consisting of nested tissue layers may yield a significantly lower accuracy than realistic models featuring multiple junctions.

In addition to accuracy, geometric patient models often have to fulfill several requirements in terms of smoothness, number, size and shape of mesh elements, sampling density, among others, in order to obtain acceptable results and make useful predictions, avoid instabilities in the simulations, or simply reduce the overall processing time. For instance, during image-guided neurosurgery, real-time constraints impose strong limitations on the complexity of the geometric brain model being dynamically registered onto the patient anatomy [?].

## 1.2 Previous Work

While different strategies can be used to obtain realistic geometric models from labeled medical datasets, few of them offer sufficient flexibility: handling of data coming from different sources, control over the density and quality of the mesh elements. Also, most existing approaches have been designed to extract surface meshes corresponding to boundaries between labeled anatomical structures, hence necessitating post-processing steps to generate volume meshes needed by finite element methods.

The most celebrated technique for producing surface meshes from sampled data is undoubtedly the *marching cubes* algorithm, introduced by Lorensen and Cline [?], and its variants (e.g. [?]). Given a scalar field sampled on a rectilinear grid, the marching cubes algorithm efficiently generates a triangular mesh of an isosurface by tessellating each cubic cell of the domain according to a case table constructed off-line.

Initially dedicated to binary classification, this method has been extended to non-binary volumes [?,?,?,?]. Unfortunately, these techniques produce unnecessarily large meshes (at least one triangle per boundary voxel) of very low quality (lots of skinny triangles). Frequently, the resulting meshes have to be regularized, optimized and decimated in order to obtain suitable representations, while simultaneously controlling the

approximation accuracy and preserving some topological properties, such as the absence of self-intersections, which turns out to be a difficult task.

Delaunay-based meshing is recognized as one of the most powerful techniques for generating surface and volume meshes with guaranteed quality. It offers easy control over the size and shape of mesh elements, for instance through a (possibly non-uniform) sizing field.

Our work builds on some recent provably correct Delaunay-based algorithms for meshing smooth surfaces [?] and volumes bounded by such surfaces [?]. These two algorithms are proven to terminate and to construct good-quality meshes, while offering bounds on the approximation accuracy of the original boundary and on the size of the output mesh. The refinement process is controlled by highly customizable quality criteria on triangular facets and on tetrahedra. A notable feature of the method of Boissonnat and Oudot [?] is that the surface needs only to be known through an oracle that, given a line segment, detects whether the segment intersects the surface and, in the affirmative, returns an intersection point. This makes the algorithm useful in a wide variety of contexts and for a large class of surfaces. Very recently, Oudot, Rineau and Yvinec [?] have proposed to combine the latter Delaunay-based surface mesher with a Delaunay refinement volume mesher [?]. This algorithm has the particularity of sampling the interior and the boundary of the object at the same time.

However, the case of multi-label partitions is not addressed in these works, therefore restricting their applicability to real datasets. The contribution of our paper is to reformulate the two above methods in the non-binary perspective.

### 1.3 Novelty of our Approach

Our work extends the method of Boissonnat and Oudot [?] and the method of Oudot, Rineau and Yvinec [?] to the case of multi-label partitions, so that they can be applied to the generation of high-quality geometric models from labeled medical datasets.

To that end, we define a *partition of Delaunay tetrahedra induced by a space subdivision*. It is related to the concept of *restricted Delaunay triangulation*, borrowed from computational geometry. In our approach, the subdivision of the domain of interest is approximated by a discrete partition of the Delaunay tetrahedralization of a point set: each Delaunay tetrahedron is labeled with one adequate tissue type. Output surface meshes are composed of the triangular facets adjacent to two tetrahedra having different labels. The point set is iteratively refined until it forms a “good” sample of the boundaries between the different anatomical structures, and, if a quality volume mesh is desired, a “good” sample of their interior.

Interestingly, our approach directly enforces watertight surface meshes free of self-intersections. Also, the consistency of surface meshes and volume meshes of the different tissues with each other is guaranteed by construction, including at multiple junctions.

Another notable feature of our approach is that the continuous partition need not to be represented explicitly. It is known only through a *labeling oracle* that, given a point in space, answers which tissue it belongs to. This makes our approach applicable to virtually any combination of data sources, including labeled 3D images, polyhedral surfaces, unstructured volume meshes, fuzzy membership functions, possibly having

different resolutions and different coordinate systems. The different data sources may even be inconsistent with each other due to noise or discretization artefacts. In this case, the labeling oracle has the responsibility of resolving the conflicts using some user-defined rules. As a result, our meshing algorithm is not affected by the heterogeneity and possible inconsistency of the input datasets.

Finally, another major advantage of our approach is that the customizable quality criteria on boundary facets and/or on tetrahedra can be tuned independently for the different anatomical structures.

The remainder of this paper is organized as follows. Section ?? gives some background on the basic computational geometry concepts needed in our approach: Voronoi diagrams, Delaunay triangulations and restricted Delaunay triangulations. Our method is described in Sect. ?. In Sect. ?, we report on some numerical experiments which demonstrate the effectiveness and flexibility of our approach for generating high-quality surface and volume meshes from real multi-label medical datasets.

## 2 Background

### 2.1 Voronoi Diagram and Delaunay Triangulation

Voronoi diagrams are versatile structures which encode proximity relationships between objects. They are particularly relevant to perform nearest neighbor search and motion planning (e.g. in robotics), and to model growth processes (e.g. crystal growth in materials science). Delaunay triangulations, which are geometrically dual to Voronoi diagrams, are a classical tool in the field of mesh generation and mesh processing due to its optimality properties.

Most of the following definitions are taken from [?]. We also refer the interested reader to some computational geometry textbooks [?,?].

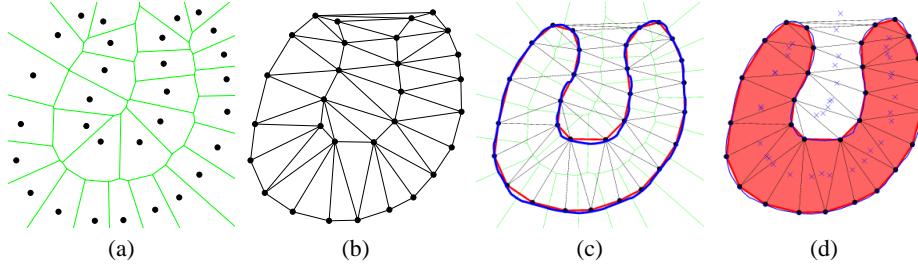
In the sequel, we call  $k$ -simplex the convex hull of  $k+1$  affinely independent points. For example, a 0-simplex is a point, a 1-simplex is a line segment, a 2-simplex is a triangle and a 3-simplex is a tetrahedron.

Let  $E = \{p_1, \dots, p_n\}$  be set of points in  $\mathbb{R}^d$ . Note that in this work, we are mainly interested in  $d = 3$ . The *Voronoi region*, or *Voronoi cell*, denoted by  $V(p_i)$ , associated to a point  $p_i$  is the region of space that is closer from  $p_i$  than from all other points in  $E$ :

$$V(p_i) = \{p \in \mathbb{R}^d : \forall j, \|p - p_i\| \leq \|p - p_j\|\} . \quad (1)$$

$V(p_i)$  is the intersection of  $n-1$  half-spaces bounded by the bisector planes of segments  $[p_i p_j]$ ,  $j \neq i$ .  $V(p_i)$  is therefore a convex polytope, possibly unbounded. The *Voronoi diagram* of  $E$ , denoted by  $\text{Vor}(E)$ , is the partition of space induced by the Voronoi cells  $V(p_i)$ .

See Fig. ??(a) for a two-dimensional example of a Voronoi diagram. In two dimensions, the edges shared by two Voronoi cells are called *Voronoi edges* and the points shared by three Voronoi cells are called *Voronoi vertices*. Similarly, in three dimensions, we term *Voronoi facets*, *edges* and *vertices* the geometric objects shared by one, two and three Voronoi cells, respectively. The Voronoi diagram is the collection of all these  $k$ -dimensional objects, with  $0 \leq k \leq d$ , which we call *Voronoi objects*. In particular, note that Voronoi cells  $V(p_i)$  correspond to  $d$ -dimensional Voronoi objects.



**Fig. 2.** (a) Voronoi diagram of a set of points in the plane. (b) Its dual Delaunay triangulation. (c) The Delaunay triangulation restricted to the blue curve is plotted with a thick red line. (d) The Delaunay triangulation restricted to the region bounded by the blue curve is composed of the filled red triangles, whose circumcenters (blue crosses) are inside the region.

The *Delaunay triangulation*  $\text{Del}(E)$  of  $E$  is defined as the geometric dual of the Voronoi diagram: there is an edge between two points  $p_i$  and  $p_j$  in the Delaunay triangulation if and only if their Voronoi cells  $V(p_i)$  and  $V(p_j)$  have a non-empty intersection. It yields a *triangulation* of  $E$ , that is to say a partition of the convex hull of  $E$  into  $d$ -dimensional simplices (i.e. into triangles in 2D, into tetrahedra in 3D and so on).

The fundamental property of the Delaunay triangulation is called the *empty circle* (resp. *empty sphere* in 3D) property: in 2D (resp. in 3D), a triangle (resp. tetrahedron) belongs to the Delaunay triangulation if and only if its circumcircle (resp. circumsphere) does not contain any other points of  $E$  in its interior.

The algorithmic complexity of the Delaunay triangulation of  $n$  points is  $\mathcal{O}(n \log n)$  in 2D, and  $\mathcal{O}(n^2)$  in 3D. Fortunately, as was recently proven in [?], the complexity in 3D drops to  $\mathcal{O}(n \log n)$  when the points are distributed on a smooth surface, which is the case of interest here.

## 2.2 Restricted Delaunay triangulation

Each  $k$ -simplex in the Delaunay triangulation is dual to a  $(d - k)$ -dimensional Voronoi object. In 3D, the dual of a Delaunay tetrahedron is the Voronoi vertex which coincides with the circumcenter of the tetrahedron, the dual of a Delaunay facet is a Voronoi edge, the dual of a Delaunay edge is a Voronoi facet, and the dual of a Delaunay vertex  $p_i$  is the Voronoi cell  $V(p_i)$ .

Given a subset  $\Omega \in \mathbb{R}^d$ , typically a manifold of dimension  $k \leq d$ , we call the *Delaunay triangulation of  $E$  restricted to  $\Omega$* , and we note  $\text{Del}|_{\Omega}(E)$  the subcomplex of  $\text{Del}(E)$  composed of the Delaunay simplices whose dual Voronoi objects intersect  $\Omega$ . For example, in 2D, as illustrated in Fig. ??(c), the Delaunay triangulation restricted to a curve  $C$  is composed of the Delaunay edges whose dual Voronoi edges intersect  $C$ . Similarly, as shown in Fig. ??(d), the Delaunay triangulation restricted to a region  $R$  is composed of the Delaunay triangles whose circumcenters are contained in  $R$ . The attentive reader may have noticed that in both cases the restricted Delaunay triangulation forms a good approximation of the object.

Actually, this is a general property of the restricted Delaunay triangulation. It can be shown that, under some assumptions, and especially if  $E$  is a “sufficiently dense” sam-

ple of  $\Omega$ , in some sense defined in [?],  $\text{Del}|_{\Omega}(E)$  is a good approximation of  $\Omega$ , both in a topological and in a geometric sense: as regards topology,  $\text{Del}|_{\Omega}(E)$  is homeomorphic to  $\Omega$ ; as regards geometry, the Hausdorff distance between  $\text{Del}|_{\Omega}(E)$  and  $\Omega$  can be made arbitrarily small; normals and curvatures of  $\Omega$  can be consistently approximated from  $\text{Del}|_{\Omega}(E)$ .

Based on these approximation properties, a family of provably correct algorithms for mesh generation and mesh reconstruction from point clouds have been designed in the last decade. We refer the reader to [?] and references therein for more details.

### 3 Methods

#### 3.1 Partition of Delaunay Tetrahedra

Let us consider  $\mathcal{P} = \{\Omega_0, \Omega_1, \dots, \Omega_n\}$  a partition of space into the background  $\Omega_0$  and  $n$  different tissues, i.e.

$$\mathbb{R}^3 = \sqcup_{i \in \{0, \dots, n\}} \Omega_i, \quad (2)$$

and let  $\Gamma$  denote the boundaries of the partition:

$$\Gamma = \cup_i \delta\Omega_i. \quad (3)$$

Given a set of points  $E$  in  $\mathbb{R}^3$ , we define the *partition of Delaunay tetrahedra induced by  $\mathcal{P}$* , denoted by  $\text{Del}|_{\mathcal{P}}(E)$ , as the partition of the tetrahedra of  $\text{Del}(E)$  depending on the region containing their circumcenter. In other words,

$$\text{Del}|_{\mathcal{P}}(E) = \{\text{Del}|_{\Omega_0}(E), \dots, \text{Del}|_{\Omega_n}(E)\},$$

where  $\text{Del}|_{\Omega_i}(E)$  is the set of tetrahedra of  $\text{Del}(E)$  whose circumcenters are contained in  $\Omega_i$ .

$\text{Del}|_{\mathcal{P}}(E)$  induces mutually-consistent surface meshes and volume meshes of the different tissues. In particular, the surface meshes are composed of the triangular facets adjacent to two tetrahedra assigned to different tissues (i.e. belonging to different parts of  $\text{Del}|_{\mathcal{P}}(E)$ ) and of the convex hull facets adjacent to non-background tetrahedra. These facets are called *boundary facets* in the sequel.

It can be proven that the resulting surface and volume meshes form a good approximation of the original partition  $\mathcal{P}$  as soon as  $E$  is a “sufficiently dense” sample of its boundaries, in some sense defined in [?]. The proof is omitted here due to space limitations. The outline of the proof is the same as in [?].

With this concept in hand, our meshing algorithm boils down to generating a point sample  $E$  which fulfills the above sampling condition as well as some additional user-defined quality criteria on boundary facets and tetrahedra.

#### 3.2 Incremental construction of the point sample

Our algorithm for generating  $E$  closely parallels the surface meshing algorithm of Boissonnat and Oudot [?]. The algorithm starts with a small initial point sample  $E_0$  of  $\Gamma$  and, at each iteration, it inserts a new point of  $\Gamma$  into  $E$  and updates  $\text{Del}|_{\mathcal{P}}(E)$ . Each point inserted into  $E$  is the intersection between  $\Gamma$  and the dual of a boundary facet

(that is to say, a ray or a segment of the Voronoi diagram of  $E$ ). We call *surface Delaunay ball* of a boundary facet the ball circumscribing the boundary facet and centered on the intersection point. Note that such an intersection always exists, by construction. In case there are several intersections, any of them can be chosen, without compromising the good continuation of the algorithm. The algorithm stops when there are no bad boundary facets left.

The surface is known only through a *labeling oracle* that, given a point in space, answers which tissue it belongs to. This oracle can be formulated as a *labeling function*  $L_{\mathcal{P}} : \mathbb{R}^3 \rightarrow \{0, \dots, n\}$  associated to the partition  $\mathcal{P}$ , such that  $L_{\mathcal{P}}(p) = i$  if and only if  $p \in \Omega_i$ . Intersections of a segment or a line with  $\Gamma$  can be computed to the desired accuracy using a dichotomic search on  $L_{\mathcal{P}}$ .

When high-quality volume meshes are needed in addition to quality surface meshes, this procedure is complemented with the refinement of bad tetrahedra by insertion of their circumcenter in  $E$ . This technique, pioneered by Ruppert [?], is known as *Delaunay refinement*. Very recently, Oudot, Rineau and Yvinec [?,?] have described how to properly combine these two meshing levels (facet refinement and tetrahedron refinement) in the binary case.

Under these considerations, the overview of our algorithm is given below:

```

while there is a bad boundary facet or a bad tetrahedron do
  if there is a bad boundary facet then
    let  $f$  be the worst boundary facet
    let  $p$  be an intersection between  $\Gamma$  and the dual of  $f$ 
    insert  $p$  in  $E$ 
  else {there is a bad tetrahedron}
    let  $c$  be the circumcenter of the worst tetrahedron
    if there is a boundary facet  $f$  whose surface Delaunay ball contains  $c$  then
      let  $p$  be the center of the surface Delaunay ball
      insert  $p$  in  $E$ 
    else
      insert  $c$  in  $E$ 
    end if
  end if
  update  $\text{Del}_{|\mathcal{P}}(E)$ 
end while

```

### 3.3 Quality Criteria

In the above algorithm, the determination of “good” and “bad” boundary facets and tetrahedra is devoted to some user-defined criteria, that are typically a combination of thresholds on the following elementary quality measures:

- **Boundary facets:** *aspect ratio* (minimum angle), *size* (circumradius of surface Delaunay ball), *curvature* (distance between facet circumcenter and center of surface Delaunay ball), *edge length*, ...
- **Tetrahedra:** *aspect ratio* (ratio between tetrahedron circumradius and shortest edge length), *size* (tetrahedron circumradius), *edge length*, *minimum angle*, ...

Note that these thresholds are possibly non-uniform over space, which allows to locally increase accuracy in a region of interest, while keeping the total size of the geometric model, and hence the computational and memory cost, sustainable. This capability is illustrated in Experiment 2 in Sect. ??.

Another important source of flexibility of our approach is that the quality criteria can be tuned independently for the different anatomical structures. Thus, a boundary facet must be tested against the criteria of its two adjacent tissues. It is classified as a good facet if it fulfills both criteria. For example, as illustrated in Experiment 3 in Sect. ??, if different resolutions are required for the two tissues adjoining the facet, the higher resolution is enforced.

### 3.4 Sliver Removal

While Delaunay refinement techniques can be proven to generate tetrahedra with a good *radius-edge ratio* (ratio between tetrahedron circumradius and shortest edge length), they cannot guarantee against badly-shaped tetrahedra of a special type called *slivers*. A sliver is a tetrahedron whose four vertices lie close to a plane and whose projection to that plane is a quadrilateral with no short edge. Such tetrahedra have a good radius-edge ratio but a very poor *radius-radius ratio* (ratio between circumradius and radius of largest contained sphere). Unfortunately, the latter measure typically influences the numerical conditioning of finite element methods.

In order to remove slivers from our volume meshes, we use a post processing step called *sliver exudation* [?]. This step does not include any new vertex in the mesh, nor does it move any of them. Simply each vertex is assigned a weight and the Delaunay triangulation is turned into a Delaunay weighted triangulation. The weights are carefully computed in such a way that no vertex nor any boundary facet disappear from the mesh. Within these constraints, the weight of each vertex is chosen in turn to maximize the minimum dihedral angles of tetrahedra incident to that vertex. Although the guaranteed theoretical bound on radius-radius ratio is known to be miserably low, this algorithm is efficient in practice and generates almost sliver-free meshes.

### 3.5 Implementation Aspects

By using *CGAL* (Computational Geometry Algorithms Library, homepage: [www.cgal.org](http://www.cgal.org)) [?], we have been able to implement our approach with only 1000 lines of C++ code. CGAL defines all the needed geometric primitives and provides an excellent algorithm to compute the Delaunay triangulation in 3D: it is robust to degenerate configurations and floating-point error, thanks to the use of exact geometric predicates, while being able to process millions of points per minute on a standard workstation.

Moreover, the incremental surface meshing algorithm of Boissonnat and Oudot [?] is now available as a CGAL package. Although this algorithm is dedicated to restricted Delaunay triangulations, the code can be modified to compute the partition of Delaunay tetrahedra induced by a partition. Although this is not optimal as regards computation time, we have chosen this option for our prototype. Also, our implementation of sliver pumping could be heavily optimized.



## 4 Experimental Results

In order to illustrate the wide applicability and the high flexibility of the proposed algorithm, we generated several meshes from real labeled medical datasets, under various parameter settings. The parameters (number of tissues and refinement criteria) and the quantitative results (number of vertices, of boundary facets and of tetrahedra; computation time) of our different experiments are gathered in Table ??.

Experiment	1	2	3	4
# tissues	15	72	72	77
# vertices	11K	36K	112K	389K
# boundary facets	23K	79K	228K	536K
# tetrahedra	73K	231K	728K	2370K
Time (sec)	35	54	340	1363 + 5080 (sliver pumping)
Refinement criteria	minimum angle $> 30^\circ$			
	size $< 1\text{mm}$	non-uniform size	cortex: size $< 1\text{mm}$ others: size $< 2\text{mm}$ tets: radius-edge ratio $< 2$ cortex tets: size $< 1.5\text{mm}$ other tets: no size constraint	

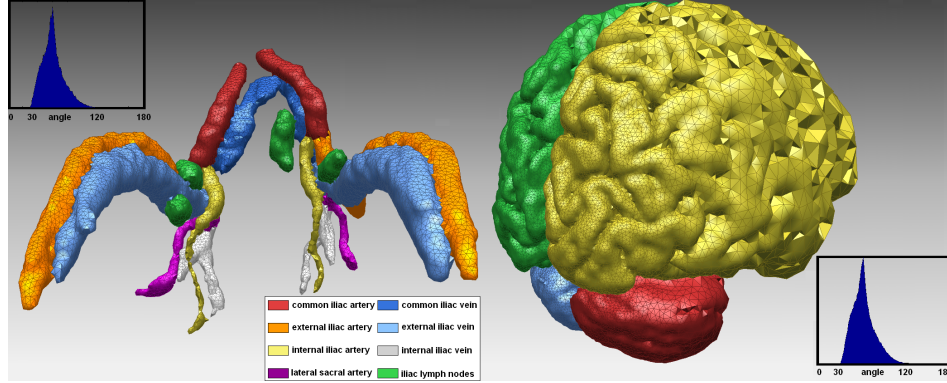
**Table 1.** Parameters and quantitative results of our different numerical experiments.

**Experiment 1: Uniform surface meshing.** In a first experiment (Fig. ??-Left), we meshed the interfaces of a labeled MR angiography of iliac veins and arteries ([?]) with a *uniform* boundary facet quality criterion: the output surface meshes are required not to contain any triangle greater than  $1\text{mm}$ . The resulting angle distribution illustrates the high-quality of the surface meshes that do not contain any badly-shaped triangles - no triangle forms an angle smaller than  $30^\circ$  (or larger than  $120^\circ$ ). Note also that the angle distribution peaks around  $60^\circ$ .

**Experiment 2: Non-uniform surface meshing.** In the three following experiments, we use some segmented brain data which were automatically generated from a T1-weighted magnetic resonance (MR) image using FreeSurfer’s Whole Brain Segmentation tool [?].

The second experiment demonstrates the ability of our algorithm to produce surface meshes with non-uniform resolution (Fig. ??-Right). The facet size criterion was chosen to be spatially-varying so that the occipital cortex would be more finely sampled than the frontal cortex. Although for clarity, only four surfaces are displayed (for each hemisphere, the cortical and cerebellum surfaces), a total number of 72 tissues were meshed using the aforementioned criteria.

**Experiment 3: Surface meshing with tissue-dependent resolution.** Figure ?? displays brain anatomical interfaces meshed with different sampling resolutions - different boundary facet quality criteria were assigned to different anatomical structures. The resulting meshes are consistent with each other (i.e. common anatomical interfaces share the same mesh representation) and interfaces are meshed with the finer density required for the neighboring anatomical structures.



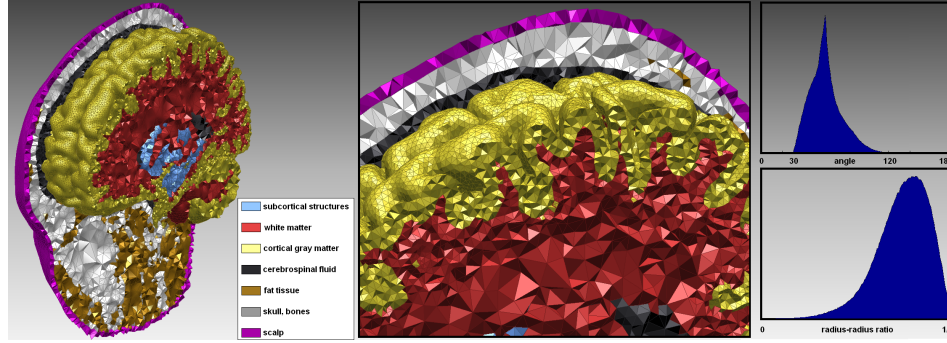
**Fig. 3. Surface meshes.** *Left:* Vascular mesh model, obtained by meshing uniformly 15 tissue interfaces. *Right:* Non-uniform surface meshing of brain tissues.

In this experiment, the cortical surface was required to have a finer resolution ( $\leq 1mm$ ) than all other structures ( $\leq 2mm$ ). Consequently, interfaces of anatomical structures other than cortical gray matter were meshed less densely, except for *local* regions where the structures were in contact with the cortex. This is clearly illustrated in the magnified region of Fig. ??, where the white matter interface is more densely meshed in regions that are in direct contact with gray matter.

In order to show the quality of the output surface meshes, we computed their angle distribution and compared it to the one of meshes extracted by a marching cubes algorithm [?]. Figure ??-*Right* shows both histograms. Contrarily to our method, which produces well-shaped triangles only, the MC algorithm yields meshes with lots of skinny triangles and whose angle distribution peaks around  $45^\circ$  and  $90^\circ$ .

**Experiment 4: Surface and volume meshing with tissue-dependent resolution.** The last experiment illustrates the ability of our method to generate high-quality consistent surface and volume meshes under elaborate tissue-dependent criteria. Similarly to the previous experiment, tissue-dependent boundary facet criteria constrained the cortical surface to be more finely meshed than other interfaces. In addition, we also meshed the anatomical volumes with tissue-dependent resolution. Tetrahedra in cortical gray matter were required to be smaller than  $1.5mm$ , while no size constraint was imposed on tetrahedra in other anatomical structures; on the other hand, the aspect ratio of all tetrahedra was required to be smaller than 2.

Figure ?? displays the resulting surface and volume meshes. Some cross-sections, obtained by cutting the models by different planes, reveal the high quality of tetrahedra. The surface meshes, which have tissue-dependent sampling densities, are consistent with the volume meshes. Note the size variation of white matter tetrahedra, which become very large as they get further from the cortical surface. The angle and radius-radius ratio distributions show that the obtained surface and volume meshes constitute high-quality geometric models of brain anatomical structures, well-adapted to numerical simulations.



**Fig. 4. Consistent tissue-dependent surface and volume meshes of head tissues.** *Left:* The cortical surface and volume meshes (in yellow) both have a high resolution. Other anatomical structures have a coarser resolution except in regions adjacent to the cortex. *Center:* The magnified view exemplifies some consequences of tissue-dependent volume resolution: the unconstrained white matter tetrahedra (in red) become smaller as they approach the cortical surface. *Right:* Angle and radius-radius ratio distributions of surface meshes and volume meshes, respectively.

## 5 Discussion and Conclusion

An increasing number of simulation tasks in clinical and research imaging necessitate more and more realistic geometric models (i.e. surface or volume meshes), whose elements (i.e. size, shape, number, ...) are often constrained by the numerical methods they are being designed for. The proposed approach has been designed to meet all these requirements, extensive control over the constructed meshes being achieved through two user-defined quality criteria on triangular facets and on tetrahedra (Sect. ??).

The produced models are watertight surface meshes, free of self-intersections and consistent with volume meshes of the different tissues. Contrarily to existing methods [?], the algorithm is not restricted to nested topologies and can handle multiple junctions. Our current implementation does not guarantee that each surface mesh constitutes a valid two-manifold when the resolution requested by the user is too low. A more elaborate refinement criterion automatically enforcing the manifold property is under development.

In conclusion, we have proposed a very flexible Delaunay-based technique for the generation of high-quality meshes from medical datasets. We plan to make our code available, and we hope that it will contribute to facilitate the creation of realistic geometric patient models.