

## GENERATION OF COMPUTATIONAL MESHES FROM MRI AND CT-SCAN DATA \*

ALBAN PAGÈS<sup>1</sup>, MAXIME SERMESANT<sup>2</sup> AND PASCAL FREY<sup>3</sup>

**Abstract.** There are fields of engineering where accurate and personalised data are required; biomedical applications are one such example. We describe here a general purpose method to create computational meshes based on the analysis and segmentation of raw medical imaging data. The various ingredients are not new: a segmentation method based on deformable contours and a surface and volume mesh adaptation method based on discrete metric specifications; but the challenge that motivated this paper is to put them together in an attempt to design an automatic, easy to use and efficient 3D code.

For non-engineering (like biomedical) applications, the user interface is often a key point. In this project, we put a great deal of emphasis on the automation of the whole procedure, making then possible to envisage large scale simulations with a minimal amount of user interaction. In particular, the user knowledge is required to help segmenting the image (the user is expected to have the know-how of the body anatomy), all meshing steps rely on fully automatic algorithms depending on a few parameters (for the external surface approximation).

One application example is presented and commented, for which the data preparation takes a few hours and results can be obtained overnight on a PC workstation.

**Résumé.** Dans certains domaines du calcul scientifique il est nécessaire de disposer de données précises et personnalisées; les applications biomédicales en sont un bon exemple. Nous présentons ici une méthode générique permettant de générer des maillages de calcul à partir de l'analyse et de la segmentation de données médicales discrètes. Si les divers composants ne sont pas nouveaux: une méthode de segmentation d'images basées sur des contours déformables et des techniques d'adaptation de maillage de surfaces et de volumes basées sur la notion de métrique, le challenge est ici d'assembler ces constituants de manière à obtenir un code de simulation efficace et facile à utiliser.

Pour des applications telles que biomédicales, l'interface utilisateur est un point clé. Dans ce projet, nous avons particulièrement soigné l'automatisation de la procédure, ce qui permet d'envisager des simulations de larges tailles pratiquement sans intervention de l'utilisateur. En particulier, les compétences de celui-ci (*i.e.*, ses connaissances en anatomie) sont utilisées pour la segmentation des images, les étapes de maillage étant entièrement automatisées et ne faisant intervenir qu'un ensemble réduit de paramètres (contrôle de l'approximation des surfaces notamment).

---

\* *We are grateful to the CEA-LIST for funding this work*

<sup>1</sup> CEA/LIST - Laboratoire Jacques Louis Lions

<sup>2</sup> King's College London

<sup>3</sup> Laboratoire Jacques Louis Lions

## 1. INTRODUCTION

Thanks to advances in computer technology and numerical analysis, it is now possible to solve Partial Differential Equations (PDEs) on large meshes representing physical, chemical or biological domains or even human organs in a reasonable computational time. Nevertheless, the actual limitation of numerical simulations is strongly correlated to the proper geometrical definition of the computational domain. This problem is relevant to biomedical simulations as the geometry differs largely from one patient to another. Nowadays, mesh generation procedures for biomedical data have not yet reached the same level of maturity and degree of automation as for more classical engineering applications. For instance, no geometric description (*i.e.*, a CAD model) of an organ is here at hand, and only a few algorithms can proceed discrete (voxel) data directly. Conversely, numerical simulations for physicians prevent them to widespread the practice because they require a lot of efforts to learn and to manage the various software tools. More precisely, cardiologists certainly do not want to become experts in mesh generation or image segmentation.

The problem considered here involves the coupling of image segmentation, mesh reconstruction and mesh adaptation procedures for biomedical (organ) domains. So the anatomical grey-level images provided by medical imaging devices are segmented using a set of morphological operations resulting in a series of closed contours corresponding to the various organs under investigation [13]. Then, a surface triangulation is created by means of a well known "Marching Cubes" algorithm [9]. In view of obtaining a computational mesh, this triangulation must be processed to eliminate all redundant information (thus reducing its complexity) as well as to improve the geometric approximation of the underlying surface [6]. A volume (tetrahedral) mesh is then generated based on this triangulation through a constrained Delaunay-based method [7].

The originality of this study lies in the efficient computer implementation of the various tools that allows to reduce the level of user interaction as much as possible; mesh generation procedures can be considered here as 'black boxes'. A classical Fitzhugh-Nagumo PDE model [4, 10] has been implemented here as a first attempt to perform numerical simulations on the computational meshes. The figures show what result can be obtained with a conventional computer equipment in a few hours. We claim that this approach provides a general purpose framework for constructing computational meshes prior to numerical simulations in biomedical applications.

## 2. PROBLEM STATEMENT

Polygonal meshes are commonly used in computer graphic applications and in numerical simulations. Indeed, they offer the advantage of describing as precisely as required the topology and the geometry of the object at hand. Moreover, polygonal meshes are also considered as a convenient data exchange format (more flexible than any Computer Aided Design file). With biomedical applications relying on human organs, the main difficulty comes from the large variety of possible configurations, thus making necessary to design generic simulation tools. Automation and reproducibility are the two main features that such tools must offer the clinical experts. These constraints also prevent the use of available 'tests cases' as the "Visual Human Project" (National Library of Medicine, USA) [1] and the "Human Modelling for safety" (INRETS, France) [14], except for validation purposes.

Computed tomography (CT) or Magnetic Resonance Imaging (MRI) data usually consists of series of 2d slices stacked to form a 3d discrete grey-level image. Each slice is composed of  $n \times m$  (usually  $256 \times 256$  or  $512 \times 512$ ) pixels, each pixel is associated to an intensity value in the discrete ranging  $[0, N]$ . The problem that we face is related to the proper characterisation of each organ under investigation. Ideally, an organ should be identified by a single intensity value and conversely. Practically however, the data are noisy and require high-level segmentation algorithms in order to identify each domain boundary properly.

The classification approach considered here has been developed at the INRIA Epidaure project (Sophia Antipolis, France) and has been successfully applied for the last few years in biomedical applications. It is based on 'snake' (deformable) contours to extract the geometry and topology of the organ. This technique is considered more robust than others, based for instance on neural networks or genetic algorithms, as it allows to account for the user knowledge of the topology of the organ. Starting from a seed, an initial contour is

progressively inflated until it matches the desired organ boundary. This process is then repeated in each slice using the know-how gained from the preliminary segmentation stage.

Once the various organs have been clearly identified and properly segmented, pixels values have been converted into a finite and reduced subset of the initial intensity values. Now a single value characterises each organ without ambiguity, allowing to process (to mesh) one such domain at the time.

The next stage consists in applying a well known "Marching Cubes" algorithm to reconstruct an initial surface triangulation [9]. The image dataset is considered as a set of disconnected voxels (each composed of eight pixels lying on two adjacent slices) that can be processed independantly. So each voxel is examined to determine whether or not it is intersected by the organ boundary. This simple question can be answered by checking the reduced intensity value at each of the eight pixels defining the voxel. If at least one pixel belongs to the domain boundary (or to the organ), the boundary intersects the voxel. The set of all configurations  $256 = 2^8$  can be reduced to a minimal subset of 14 elements. Each configuration is then triangulated in a unique manner, the union of all triangles provides the surface triangulation of the domain boundary. However, it is well known that such a method does not offer any theoretical guarantee about the domain topology, several authors have proposed post-processing ways to overcome this problem.

The surface triangulations produced by "Marching Cubes" approaches usually contain too many elements to be directly used for computational purposes and, more importantly, the element shape quality can be relatively poor. Both these features badly affect the efficiency, in terms of time and accuracy, of the numerical simulations. Thus, a surface mesh optimisation procedure must follow the surface reconstruction stage.

The initial surface mesh is first smoothed to remove the "staircase" artefacts that often occur after the reconstruction stage. Then it is analysed in order to extract the intrinsic properties (principal surface curvatures and directions) of the underlying surface. This allows to define a discrete metric tensor at each mesh vertex prescribing the desired size and stretching of the element. The decimation process is controlled by this metric map that accounts for the local surface variations. At completion of this procedure, a computational surface mesh is achieved.

The volume mesh generation consists in creating tetrahedral elements using a constrained Delaunay method [7]. The surface mesh is preserved during this stage, mesh elements are inserted into the domain to comply with the given metric map. A special attention is paid to the element shape quality for simulation purposes.

### 3. DATA SEGMENTATION

#### 3.1. Subdomains identification

Each organ on digital images corresponds to what radiologists call the density of the image for MRI and the number of Hounsfield units for CT scans. It is correlated to the proton concentration (MRI) or to the absorption coefficient (CT) of the organ. Consequently, materials with very close absorptions present very close colours on a CT image. This is especially the case for soft tissues. For the sake of efficiency, the initial grey level images are first mapped into 8 bytes images (256 intensity levels). This will allow to dramatically reduce the noise and thus the number of potential contours. More precisely, the pixels defining the contours will be considered as members of different classes of equivalence. Imaging data can present gradients of colours inside an organ that are not correlated to the gradient concentration or to the absorption coefficient. These gradients are well known by radiologists and are due to the acquisition method (Figure 1).

In our approach, each organ must be separately segmented and extracted. The segmentation of a single organ is realised in each slice through a "snake contours" method. The knowledge in human anatomy of a clinician is welcome, but only in the early stage (initialisation) of the procedure. The user is prompted to provide an initial guess for the geometry as well as for the topology of the organ or the region studied. The initial contour is then progressively deformed in order to make it fit to the real structure of the organ. As the latter is hard to characterise analytically, a stopping criterion related to contour variations and topological changes has to be introduced in the process. Obviously, a fully automatic segmentation procedure (*i.e.*, not requiring the user's intervention) would be preferable, but is not currently available.

The identification (classification) of an organ on the reduced color dataset is obtained by applying arithmetic operations on the intensity values associated with pixels. Using a 8-bytes coding, addition allows for permutations on colours. Indeed, due to the  $2^8$  modulo introduced by the discretisation on unsigned bytes, colors close to 255 will result by addition in colors close to 0. Moreover, dividing any color value by a larger value gives zero. Hence, proper divisions allow to merge subdomains into a zero-valued single one. A clever combination of addition-division operations leads to the identification of a subdomain; each pixel value inside (resp. outside) the domain being set to one (resp. zero).

Figure 1 shows the 2D graphic reconstruction of the thorax aorta from a 16 bits  $512 \times 512$  CT with axial cuts (left-hand side). The aorta valve cuspides that separate the ventricle and the aorta arch can be clearly distinguished around the heart junction. A gradient of colours can be noticed inside the aorta even if the only content is blood.

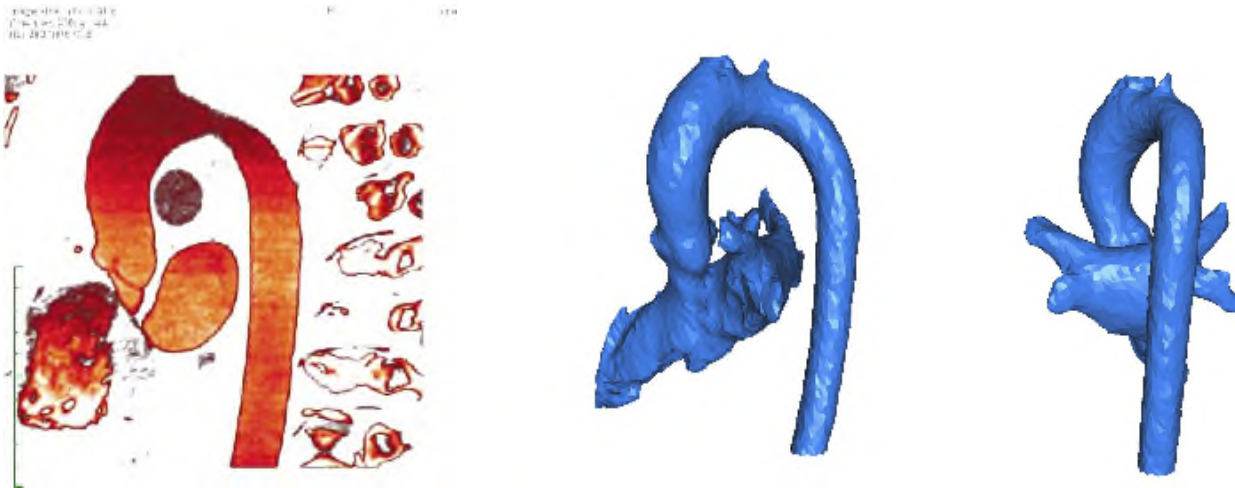


FIGURE 1. 2D graphic reconstruction of the thorax aorta from a 16 bits  $512 \times 512$  CT with axial cuts (left-hand side). Computational surface mesh of the aorta presenting the left auricle and ventricle (middle and right-hand side). The beginning of the brachio-cephalic trunk appears (top) as well as the extremity of the four pulmonary veins on the left auricle.

### 3.2. Subdomains cleaning

At completion of the classification stage, it is not uncommon to find domains presenting internal structures that are considered too small to be reasonably meshed (*i.e.*, with a minimal element size). For instance, contours defined by only a few pixels can be removed without jeopardising any further simulation. The desired level of accuracy of the mesh size must be chosen accordingly. Obviously, such a procedure cannot be applied blindly on the contours in each slice, as it may result in topological inconsistencies for the remaining part of the domain. However, it is desirable to remove sharp local variations in a contour, in an attempt to control the local curvature. Moreover, a tiny contour (its length measured in pixels being negligible) can be removed from one slice if it represents, for instance, the vanishing tip of a structure.

Mathematical morphology operations, namely erosion, dilatation or closing, are used to remove the thinnest structures. These operations consist in applying a convolution using a  $n \times n$  mask on each image pixel. A gaussian filtering stage is then carried out in order to regularise the domain boundaries. This procedure also improves (removes) the "staircases" artefacts due to the spatial discretisation and connectivity.

## 4. MESH GENERATION

### 4.1. Initial surface triangulation

The "Marching Cubes" algorithm has been developed in the mid eighties as an automatic tool for discrete implicit surface reconstruction [9]. The main feature of this technique lies in its ability to process a digital 3d image as a set of disconnected voxels, each of which associated to 8 pixels on two successive slices. An implicit surface is simply defined as an isovalue surface of the intensity function. Hence, checking the intensity values at the pixels and comparing it to the surface level is sufficient to decide whether or not the voxel is intersected by the implicit surface. The careful analysis of all possible configurations leads to identify a minimal set of 14 patterns. With each of these basic patterns is associated a unique triangulation of the surface within a voxel. Once all voxels have been processed, the union of all triangles produces the surface triangulation of the implicit surface.

Although this method presents nice and attractive features (the parallelisation is easy, the complexity is minimal), it may lead to non orientable or non manifold triangulations or triangulations having multiple connected components, even in case of regular bounded domains. Additional procedures have been designed to overcome these drawbacks.

However, the main problem with this approach is its inability to control precisely the element sizes and shapes. In any case, the element size is at most half the size of a voxel. The resulting surface triangulations usually contain a large number of elements (*i.e.*, a multiple of the number of intersected voxels) to describe a domain. Obviously, if large portions of a domain are almost planar, larger triangles could be used to represent these regions without altering the geometric accuracy of the discretisation. In addition, due to the discrete nature of the data, sharp or right angles may be introduced between adjacent triangles, thus modifying the underlying surface curvatures, even in case of smooth surfaces. Both problems need to and will be addressed.

### 4.2. Surface mesh adaptation

In the context of numerical simulations based on finite element/volume methods, it is well known that the quality of the numerical solutions is strongly related to the mesh quality [2]. Although a uniformly refined mesh (all elements having a minimal size  $h$ ) would certainly lead to an accurate solution, it seems more reasonable to find a better compromise between the number of elements and the desired accuracy. Hence, tuning the number of degrees of freedom without giving up on the numerical accuracy will substantially decrease the cpu time required to solve the large systems involved in the simulations. This is precisely the aim of mesh adaptation.

To generate a surface mesh, we proceed in two steps: a geometric triangulation is generated from the given initial triangulation and then a computational surface mesh is created. A geometric surface triangulation is a piecewise linear approximation of a surface corresponding to a bounded surface deviation error. The latter can be defined as the gap between the triangulation and the underlying analytical surface (not known in our case). More precisely, in a geometric surface triangulation, the deviations at orders 1 and 2 (with respect to limited Taylor expansions) are bounded by tolerance values [6].

Given an initial triangulation, we first proceed by smoothing the surface using a bilaplacian algorithm that behaves like a low-pass filter. This results in a triangulation where high curvature variations (assimilated to noise) have been largely reduced (2). Then geometric and topological tools are used to reduce the number of vertices and elements while preserving the geometric accuracy of the approximation. For instance, vertex collapsing, edge flipping, vertex relocation are used to perform this stage incrementally. The triangulation is iteratively modified until the desired geometric approximation is obtained. To make sure that the resulting triangulation does not deviate from the original one, the Hausdorff distance between the two sets of triangles is computed.

As no specific attention has been paid on element shape and size quality, the geometric surface triangulation cannot be considered as a computational mesh. In such a mesh, the element sizes are directly related to the local curvatures of the surface and the element shapes must be controlled to avoid degenerated (skinny) triangles. To achieve this control a discrete metric tensor is defined at the mesh vertices and will be used to prescribe

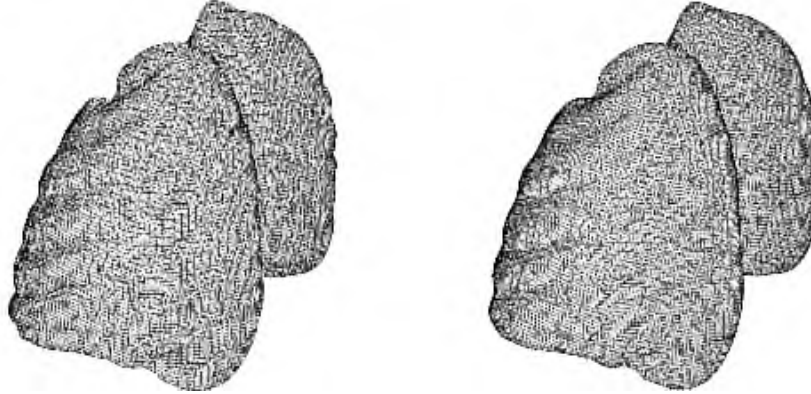


FIGURE 2. *Geometric surface meshes: initial and smoothed surface triangulation of lungs (left-hand side). Computational surface mesh of lungs (right-hand side). Each figure has the same number of vertices (37,904) and triangles (75,800). Notice the print of the ribs on the left side of the front lung.*

the size and stretching of the elements in the vicinity of each vertex. This metric specification allows to control the surface deviation as well as the element shape and size. To simplify, this metric map is normalized so as to prescribe unit size elements. With simplicial (triangular) elements, the control of the size and shape can be easily obtained by computing the edge length with respect to the metric (this requires modifying the classical dot product as the Euclidean space is now supplied with a Riemannian structure). The same topological and geometric tools can be used to modify iteratively the geometric triangulation to produce a computational surface mesh [6]. A slight modification of the metric tensor (gradation control) will also allow to control the number of elements in the resulting mesh.

### 4.3. Volume mesh generation

The generation of a volume mesh is achieved using a constrained Delaunay approach. The surface mesh is embedded in a bounding box that is meshed using a few tetrahedra. Each vertex of the surface triangulation is inserted in this initial mesh using the so-called *Delaunay kernel* [7]. At completion of this stage, the missing surface triangles (*i.e.*, that are not tetrahedra faces) are first recreated using edge and face flipping operations. Then internal vertices are created and inserted in the mesh using the same procedure. To this end, the discrete metric tensor defined at the previous stage is interpolated and the current mesh edges are analysed to decide whether they must be kept or splitted. The insertion procedure of a vertex  $P$  can be formally written as:  $\mathcal{T} = \mathcal{T} - \mathcal{C} + \mathcal{B}$ , where  $\mathcal{T}$  denotes the current triangulation,  $\mathcal{C}$  the cavity of  $P$  and  $\mathcal{B}$  is the remeshing of  $\mathcal{C}$  (a star-shaped polyhedron with respect to  $P$ ). Here, the construction of  $\mathcal{C}(P) = \{K \in \mathcal{T}, P \in \text{Ball}(K)\}$  ( $\text{Ball}(K)$  is the opencircumsphere of  $K$ ) is obtained using a proximity criterion.

For the same reasons as above, an optimisation stage aims at improving the overall quality of the elements, both in terms of shape and size. This last stage involves node relocation and face flipping operations.

## 5. A CASE STUDY

The summer school CEMRACS 2004 at the CIRM (Luminy) was a unique opportunity to test the proposed approach on a test case. The problem we consider here is to compute the cardiac depolarisation cycle in the

context of the detection of the right ventricular dysplasia arhythmogenic (RVDA) [8] using a finite element method.

### 5.1. Numerical modelling

ARVD is a cardiomyopathy characterized by fatty and fibrous infiltration of the right ventricular myocardium. It induces a localized slowdown conduction area. Solving the inverse problem of electrocardiology (*i.e.* determine the heart depolarisation sequence from thorax measurements), will certainly highlight this slowdown conduction area. Modelling the heart depolarisation and thorax potential generated by the depolarisation (*i.e.* solving the direct problem of electrocardiology) is necessary before solving this inverse problem. Depolarisation of the heart cells is modelised by a set of two differential equations first introduced by FitzHug and Nagumo in the early sixties. The original formulation of the equation system is the following:

$$\begin{cases} \frac{du}{dt} = c_1 u(u - a)(1 - u) - c_2 v \\ \frac{dv}{dt} = c_3(u - c_4 v) \end{cases}$$

In this model the transmembrane potential overshoots in the repolarisation phase and becomes more negative than the resting potential which is not characteristic for an action potential. A modification of this original model was suggested by Rogers and McCulloch in 1994 [11], that allows to get rid of this unphysiological overshoot in repolarisation. In order to implement the depolarisation and repolarisation propagation on the heart, a laplacian is introduced in the first equation.:

$$\begin{cases} \frac{du}{dt} = \sigma \Delta u + c_1 u(u - a)(1 - u) - c_2 uv & \text{on } \Omega_1 \\ \frac{dv}{dt} = c_3(u - c_4 v) \end{cases}$$

and the boundary conditions:

$$\frac{du}{dn} = 0 \quad \text{on } \Gamma_1$$

In this equation,  $\sigma$  represents the conductivity of tissue,  $\Omega_1$  the whole heart,  $\Gamma_1$  the whole heart surface. Heart depolarisation can be modelised thanks to this set of PDE. Energy is provided to the system by initial conditions, corresponding to a sinusal node activation in a real heart. The thorax depolarisation is obtained by a simple Poisson equation solving between the heart and the thorax surface:

$$\text{div}(\sigma \cdot \text{grad}(u)) = 0 \quad \text{on } \Omega_2$$

and the boundary conditions:

$$\begin{cases} u = u_0 & \text{given on } \Gamma_2 \text{ by preview resolution} \\ \frac{du}{dn} = 0 & \text{on } \Gamma_3 \end{cases}$$

$\sigma$  represents the conductivity of tissue in the thorax,  $\Omega_2$  the whole thorax,  $\Gamma_2$  the epicardium surface of the heart and  $\Gamma_3$  the surface of the thorax.

### 5.2. Data segmentation

Two sets of medical images have been used in order to find all organs of interest. The first set (data courtesy of Dr. W.P. Segar, NC University, USA) is not exactly a discrete dataset; it consists in a series of splines that define all organ boundaries. It was designed to simulate thorax plasticity during breathing movements. A rational non uniform B-spline interpolation is performed so as to allow the geometrical model to get closer to the imaging data as well as to ensure that all organ is closed and bounded [12]. This dataset has been converted *a posteriori* into a series of 2d slices similar to conventional imaging data.

The second dataset corresponds to CT investigations. The grey level images were used to extract a specific organ: the aortic arch (figure 1).

To extract the various subdomains of interest, we chose to use the user-friendly software tool *Yav++* developed at the Epidaure project (Inria, Sophia-Antipolis) [3] that works directly with a set of slices. After having defined a closed contour (a seed) in the first section, the software package manages to automatically extract closed contours on all slices. At completion of this stage, a crude surface triangulation can be generated for each organ that has been segmented.

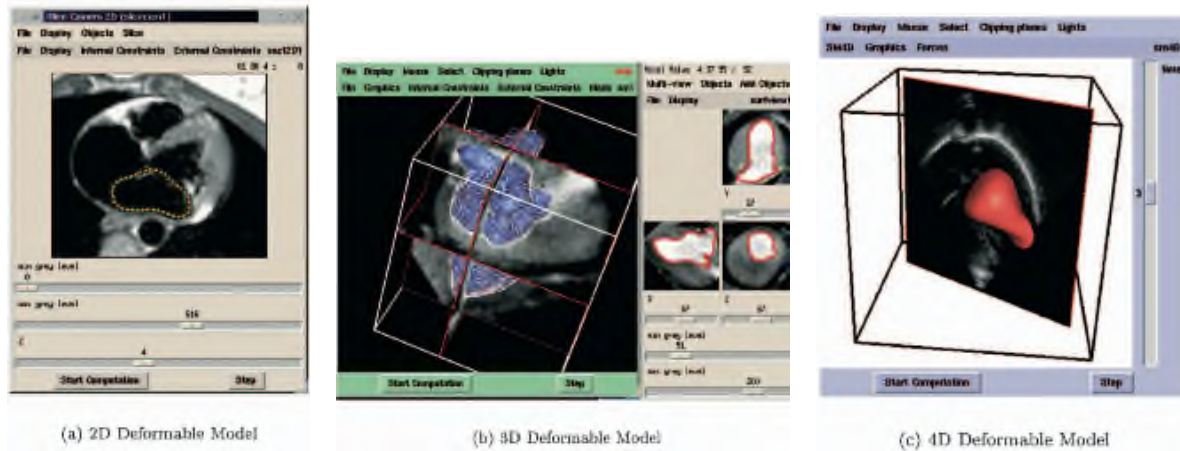


FIGURE 3. *Data segmentation using Yav++ (Epidaure project).*

This step triggered the writing of a piece of C++ code in order to merge all surface triangulations corresponding to various organs all together. As all segmentations have been performed independantly, a correction has been introduced in order to fix the problem of overlapping or intersecting triangulations. Finally, a mesh of the thorax has been achieved containing the skin, the ribs, the rachis, the lungs and the heart (Figure 5, left-hand side).

### 5.3. Mesh generation

As pointed out in the introduction, meshing tools are now widely available and can be seen as black boxes for most engineering applications. The challenge we had to face is to link in an automatic fashion various surface and volume tools together to create a computational mesh routine. To this end, we chose to use respectively *Yams* (LJLL, Paris) [5] and *GHS3d* (Inria, Rocquencourt) [7], described in the previous section. Both tools use the same mesh data format and are fully automatic adaptive procedures. The results are illustrated on Figures 2, 4, 5 and 6. The number of elements and the quality of the meshes are reported in Table 1. The quality function measures the isotropic aspect ratio of an element, namely:  $Q = \alpha \rho/h$ , where  $\rho$  is the inradius, and  $h$  the largest edge length,  $\alpha$  being a normalisation coefficient (to get a quality of 1 for a regular element).

Figure 6 (left-hand side) shows a frontal cut of the thorax. Each connected component has been indexed using a different colour. The mesh contains roughly 250,000 vertices, 100,000 triangles and 1.5 million tetrahedra.

### 5.4. Numerical simulation

Computation is realised on a PC workstation. Scheme of discretisation is half-implicite for FitzHug-Nagumo equations and implicite for Poisson equation. The resolution is performed with a conjugated gradient method. The computation time for a depolarisation cycle with a 1 ms step time is about a half day. It produces thorax and heart potential variations during a full cardiac cycle. Figure 6 (right-hand side) shows the visualisation of thorax potentials during a heart depolarisation computed using a FE method.



mesh	size	np	nt	ne	quality	quality range
heart	13.9 Mo	79,000	37,000	446,000	1.21	1.00 - 2.23
lungs	1.1 Mo	8,000	6,000	41,000	1.53	1.00 - 8.46
bone	3.3 Mo	23,000	32,000	90,000	2.45	1.00 - 47.92
aorta	1.4 Mo	10,000	8,400	47,000	1.40	1.00 - 8.53
skin	1.9 Mo	13,000	12,000	61,000	1.70	1.00 - 23.28
thorax	51.7 Mo	261,000	109,000	1,530,000	1.56	1.00 - 47.92

TABLE 1. Characteristics of resulting meshes,  $np$ ,  $nt$  and  $ne$  denote the number of vertices, triangles and tetrahedra, respectively.

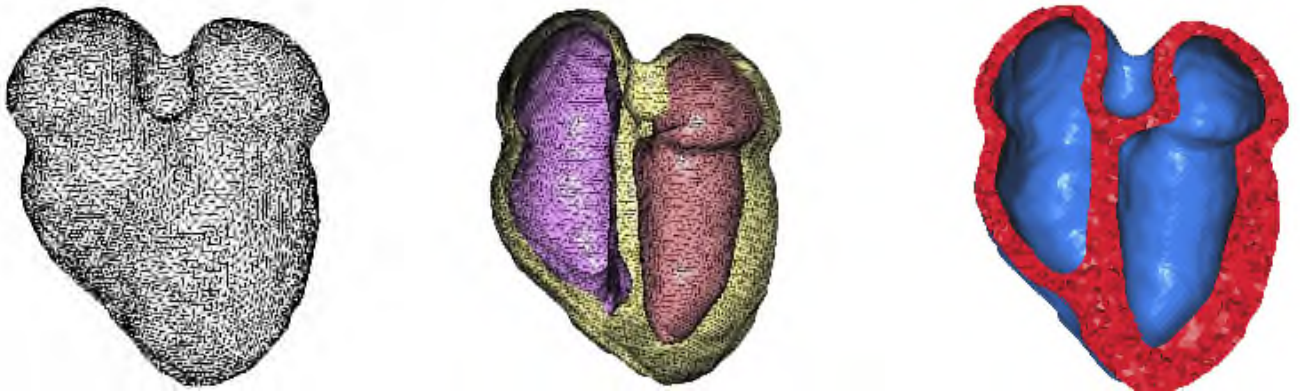


FIGURE 4. Computational surface mesh of the heart (left-hand side). Cut through the volume mesh of the heart (middle) showing the left and right auricles and both ventricles. The endocardium and epicardium meshes appear well differentiated. The final mesh contains roughly 50,000 vertices, 37,000 triangles and 260,000 tetrahedra (right-hand side).

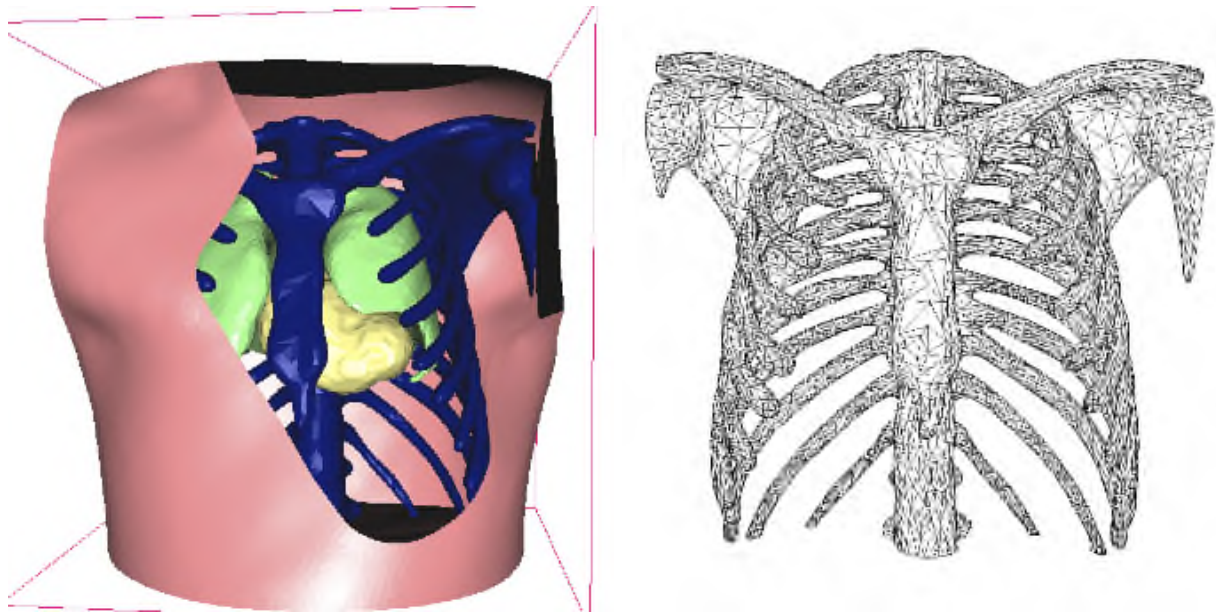


FIGURE 5. Resulting computational surface mesh of the thorax: lungs, heart and bone structures are visible (left-hand side). The structure of the bones is shown (right-hand side).

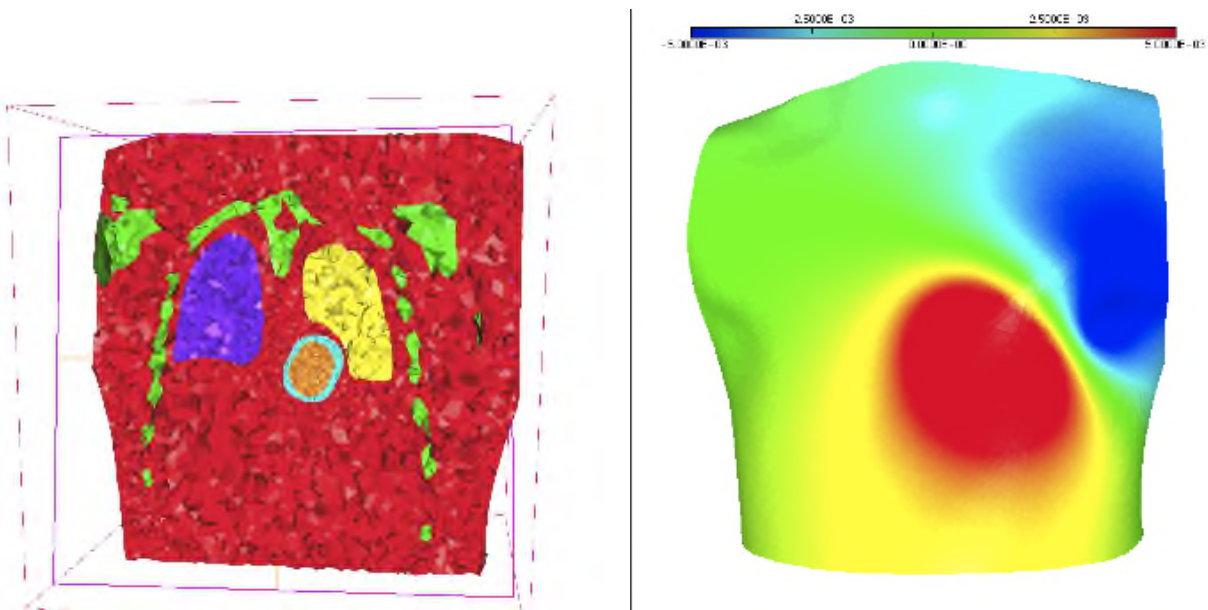


FIGURE 6. Frontal cut of the thorax (left-hand side) and visualisation of thorax potentials during a heart depolarisation (right-hand side).

## 6. CONCLUSION

In order to address tedious problems in biomedical applications, we have proposed an integrated approach chaining together some state of the art tools developed by the authors and collaborators. Three modules were linked and special developments were made for merging the meshes together. The preliminary results presented in this paper have demonstrated the feasibility and the efficiency of the approach especially with respect to the generation of adapted computational meshes. Data preparation never took more than a few hours and the performance are sufficient to allow runs on laptop and inexpensive computers.

However, this study has shown the limits of the current technology. More accurate (spatial and spectral resolution) digital images would be needed to envisage complex simulations. Actually, small anatomic structures like vessels are hard to capture using the segmentation tool and the risk is high that the topology of such organ is not preserved. New generation of scanning devices will most likely bridge this gap in the near future.

## REFERENCES

- [1] Banvard, Richard A., *The Visible Human Project Image Data Set From Inception to Completion and Beyond*, Proceedings CODATA 2002: Frontiers of Scientific and Technical Data, Track I-D-2: Medical and Health Data, Montr?al, Canada, October, 2002.
- [2] Ciarlet P.G., *The Finite Element Method for Elliptic Problems*, North-Holland, Amsterdam, 1978.
- [3] Delingette H., Bardinet E., Rey D., Lemarechal J-D., Montagnat J., Ourselin S., Roche A., Dormont D., Yelnik J., and Ayache N., *YAV++: a Software Platform for Medical Image Processing and Visualization*, In Workshop on Interactive Medical Image Visualization and Analysis satellite symposia of MICCAI, IMIVA'01, Utrecht, The Netherlands, October 2001.
- [4] FitzHugh R.: *Thresholds and plateaus in the Hodgking-Huxley nerve equation*. J Gen Physiol 43, 867-896 1960.
- [5] Frey P.J.: *YAMS, a fully automatic adaptive isotropic surface remeshing procedure*, INRIA - Rocquencourt Rapport Technique n?0252, November 8, 2001.
- [6] Frey P.J.: *Generation and adaptation of computational surface meshes from discrete anatomical data*, Int. J. numer. methods engng., 60, 1049-1074, 2004.
- [7] George P.L. et Borouchaki H., *Delaunay triangulation and meshing. Application to finite elements*, Hermès Science, Paris, 432 p., 1997.
- [8] Hulot J-S., Jouven X., Empana J-P., Frank R., and Fontaine G.: *Natural History and Risk Stratification of Arrhythmogenic Right Ventricular Dysplasia/Cardiomyopathy* Circulation, vol 110 : 1879 - 1884 Oct 2004
- [9] Lorensen, W. and Cline, H.: *Marching Cubes: a high resolution 3D surface reconstruction algorithm*, Computer Graphics (Proc. of SIGGRAPH), 21(4), 163-169, 1987.
- [10] Nagumo J., Animoto S., Yoshizawa S.: *An active Pulse transmission line simulating nerve axon*. Proc. Inst. Radio Engineers 50, 2061-2071, 1962.
- [11] Rogers JM, McCulloch AD.: *A Collocation Galerkin FEM of Cardiac Action Potential Propagation* IEEE Trans Biomed Eng vol 41 : 743-757 1994
- [12] Segars W. P., Lalush D. S., Tsui B.M.W.: *A Realistic Spline-Based Dynamic Heart Phantom*, IEEE Trans NuclSci, 46(3), 503-506, 1999.
- [13] Sermesant M., Forest C., Pennec X., Delingette H. and Ayache N., *Deformable biomechanical models: Application to 4D cardiac image analysis*, Medical Image Analysis, 7(4), 475-488, 2003.
- [14] Serre T., Bidal S., Durand F., Behr M., Basile F., Brunet C.: *3D Geometric acquisition of the human body in driving posture for modelling purposes*. Archives of physiology and biochemistry, 257me Congr?s de la Soci?t? de Biom?canique, Montr?al, Canada, 23-26 ao?t 2000, 108(2), 92, 2000.
- [15] [http://www.nlm.nih.gov/research/visible/visible\\_human.html](http://www.nlm.nih.gov/research/visible/visible_human.html)