The 8th European Congress on Computational Methods in Applied Sciences and Engineering
ECCOMAS Congress 2022
5–9 June 2022, Oslo, Norway

A PRACTICAL ALGORITHM TO BUILD GEOMETRIC MODELS OF CARDIAC MUSCLE STRUCTURE

Mark Potse[1], Luca Cirrottola[2] and Algiane Froehly[3]

[1] IHU Liryc, Université de Bordeaux,
Campus Xavier-Arnozan, Avenue du Haut-Lévêque, 33600 Pessac, France,
e-mail: mark@potse.nl, www.potse.nl

[2] INRIA, Univ. Bordeaux, CNRS, Bordeaux INP, IMB, UMR 5251,
200 Avenue de la Vieille Tour, 33405 Talence cedex, France,
e-mail: luca.cirrottola@inria.fr, lcirrottola.github.io/

[3] Service d'expérimentation et de développement,
Inria Bordeaux – Sud-Ouest, Pau, France,
e-mail: algiane.froehly@inria.fr,
www.linkedin.com/in/algiane-froehly-3a247987/

Key words: Biological systems, Cardiac modeling, Mesh adaptation, Level-set Methods

Abstract. Cardiac muscle tissue has a unique, network-like structure. Three-dimensional models of this structure are needed for simulations of cardiac electrophysiology and mechanics. We developed an algorithm to produce such models artificially, using an implicit surface expressed on a tailored unstructured multi-domain mesh to define the cell membranes.

The algorithm first creates a random network of cell centers, observing angle and distance criteria inferred from real tissue. The space around the network edges is assigned to the cellular domains based on the nearest half-edge. The network is then immersed in a regular tetrahedral mesh which is refined to fit the domain boundaries and to offer sufficient density around the cell membrane. The refinements are alternated with mesh improvement operations to maintain an acceptable mesh quality. On the refined mesh a level-set function is expressed that defines the cell membrane. The remeshing code `Mmg3d` is then used to discretize the level set while retaining the domains, and to improve the quality of the final mesh.

A serial implementation of the algorithm was able to produce meshes of a few hundreds of cardiac cells in 15 minutes, but we are still facing difficulties in the remesher, likely resulting from the unusual complexity of these meshes. It was still possible, however, to correctly mesh a small network of cells that was designed to be replicated by successive mirroring. This allowed us to build models of upto $1\,\mathrm{cm}^3$ of tissue (10 million cells and 370 billion tetrahedra) that now serve in performance tests of a large-scale simulation code.

1 INTRODUCTION

Cardiac muscle tissue has a fiber-like structure consisting of elongated cells, connected mechanically to each other at their short ends. But unlike many other types of muscle, cardiac muscle fibers branch frequently, forming a network of interconnected cells. In addition, its intercellular links contain electrically conducting channels which serve to synchronize the electrical activation of the cells and thus their mechanical action. Structural damage to the cardiac muscle can break these links, and can lead to the appearance of tortuous routes where electrical activation can travel slowly and give rise to life-threatening cardiac arrhythmia [1, 2].

Numerical simulation is important to understand cardiac arrhythmia and to learn how to predict them [3, 4]. However, current research is almost entirely based on homogenization approaches [5, 6] which do not allow to represent the effects of damaged tissue very well [4]. To overcome this problem, several groups have recently directed their attention to models in which individual muscle cells are resolved, initially using simplified cell geometries [7, 8].

More realistic geometries could be obtained from imaging data. For example, using confocal microscopy it is possible to see details smaller than a micrometer and to visualize the distribution of different proteins that characterize the mechanical and electrical links between the cells. However, such imaging is limited in depth and in sample size, allowing at best to model a few hundred micrometers of tissue, that is, less than 10 complete cells [9].

Thus, for more macroscopic models we have to resort to synthetic geometries. To this end, we developed an algorithm that produces structures similar to cardiac tissue at a resolution in the order of micrometers. We chose to neglect details on a smaller scale. The method is based on an implicit surface expressed on a tailored unstructured multi-domain mesh to define the cell membranes as well as the connections between the cells, which are called intercalated disks.

2 METHODS

Our aim is to produce a three-dimensional tetrahedral mesh with interior surfaces fitting the cell membranes and the intercalated disks that separate the cells. The main steps of the algorithm, to be detailed in the following sections, are:

1. define a cell network in terms of *centers* and *links*, each divided halfway in two *halflinks*;
2. divide the volume among the halflinks and create an appropriate support for the level-set function;
3. define the level-set function itself;
4. discretize the level set; and
5. improve the resulting mesh.

The first three steps are performed with a dedicated code, while the discretization of the level set and mesh improvement are done with the `Mmg3d` software [10].

The volume subdivision helps to define the level-set function and also determines where the intercalated disks will be placed. To this end we create a tetrahedralization of the domain such that each tetrahedron belongs to the domain of a single halflink. Thus, edges may not cross

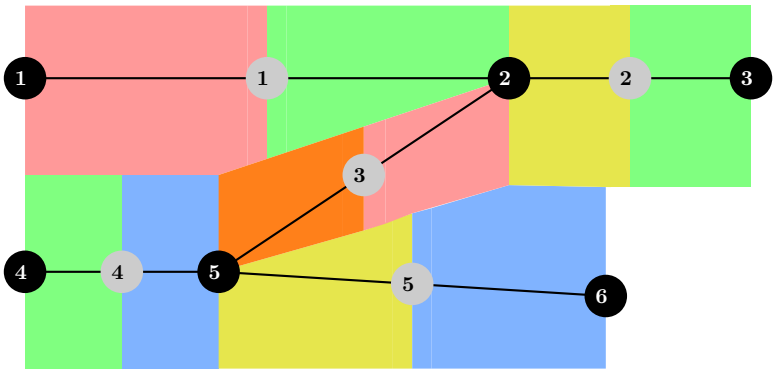

Figure 1: Decomposition of the domain in terms of halflink domains. Black dots represent centers, gray dots label the links, and colors identify the domains.

domain boundaries. When the domain boundaries are fully defined, we can be sure that there is a surface everywhere between the cells where an "outside" value of the level set function can be set. To define the intercalated disks we assign all domains that belong to a single cell the same reference value, and instruct `Mmg3d` to preserve the domains defined by these references when it discretizes the level set and improves the mesh.

A cell network and the corresponding space division in terms of halflink domains is illustrated in Figure 1. Cellular domains will be defined as the union of the halflink domains connected to each center.

2.1 Parameters

The principal algorithm parameters are the maximum cell radius $r_{\max}$, the minimum cell radius $r_{\min}$, the minimum cell separation s, and parameters governing the target density of centers, fiber orientation, and maximum angle between a link and the local fiber orientation. Here, the "fiber orientation" is the average orientation of the elongated cardiac muscle cells.

2.2 Cell network

The graph that forms the "skeleton" of the cell network is constructed by taking a random cloud of points, which we name "centers" as they loosely correspond to the centers of the cells. In order to form a graph that could be a realistic representation of a cardiac cell network, we remove centers that are too close to each other and create links between them according to heuristic arguments. We create a link between any pair of centers if it is not too long or too short, or has too large an angle with respect to the specified local fiber orientation. To avoid situations where one link cuts too large a hole in another, we must also make sure that no two halflinks not belonging to the same link or the same cell are anywhere closer to each other than $2r_{\min} + 2s$. Two links may touch at a center, but the remote halves of these links, which will be separated from each other by extracellular space, must be far enough from each other to ensure that each of them can have the minimum required radius. After the initial construction of the graph we count for each link how many other links are too close to either of its halves. Then we iteratively remove the most conflicting links until no more conflicts remain.

2.3 Dividing the volume

The next task is to define a mesh that can support the level-set function and at the same time define the halflink domains. We start with a simple repetitive tetrahedralization of the volume. Then we split all edges that cross a domain boundary. This process creates edges that may cross another domain boundary, at joints or corners, and must therefore be iterated until no more edges remain to be split. To this end, each vertex can be associated with multiple domains. When the vertices of a given edge each are associated with a domain that the other is not, the edge is split on the boundary between those two domains.

The edge splitting process does consecutively resolve boundaries, joints, and corners, but despite precautions it can generate tetrahedra with such low quality that mesh improvement after completion of the splits is no longer possible. Therefore we monitor mesh quality and when it decreases below a preset threshold we interrupt the splitting process to perform mesh improvement. This is done using edge collapses and movement of vertices. Care is taken not to collapse a vertex onto another vertex that is not associated with all the domains that the first vertex is.

Even though the mesh improvement process preserves vertices in important locations, the edge collapses move some edges such that they cross domain boundaries. Therefore the mesh improvement must always be followed by further tests and possibly edge splits. An example of the discretized domain boundaries is shown in Figure 2.

2.4 Level set definition

The location of the cell membrane is defined by a level set function, which is defined on the vertrices of the tetrahedral mesh, after the domain boundaries have been resolved. Let P be a vertex of the tetrahedralization and C the nearest point on any link. Then the function value at P is the smallest of

$$d_0 = r_{\max} - |\vec{PC}| \tag{1}$$

and

$$d_n = D_n - s(x) \tag{2}$$

where D_n is the distance from P to the nearest domain boundary. The distance between the cell membrane and the domain boundary, $s(x)$, is a function of position here. In most places it is equal to a fixed constant s_0, but when the two domains concerned belong to the same cell then we change it as a function of the position along the link to obtain a round corner (for example near the center of link 3 in Figure 3).

2.5 Level set discretization and mesh improvement

Discretization of the level set is done by the `Mmg3d` software, at the zero level of the function we defined. The tetrahedra that are given to `Mmg3d` each have a "reference" value that defines to which cell they correspond. A cell consists of the set of halflink domains that are associated with its center. Figure 3 illustrates the problem presented to `Mmg3d`.

Once the cell surface is defined, the final mesh can be improved to make it suitable for numerical simulations. This task is also left to the `Mmg3d` software.

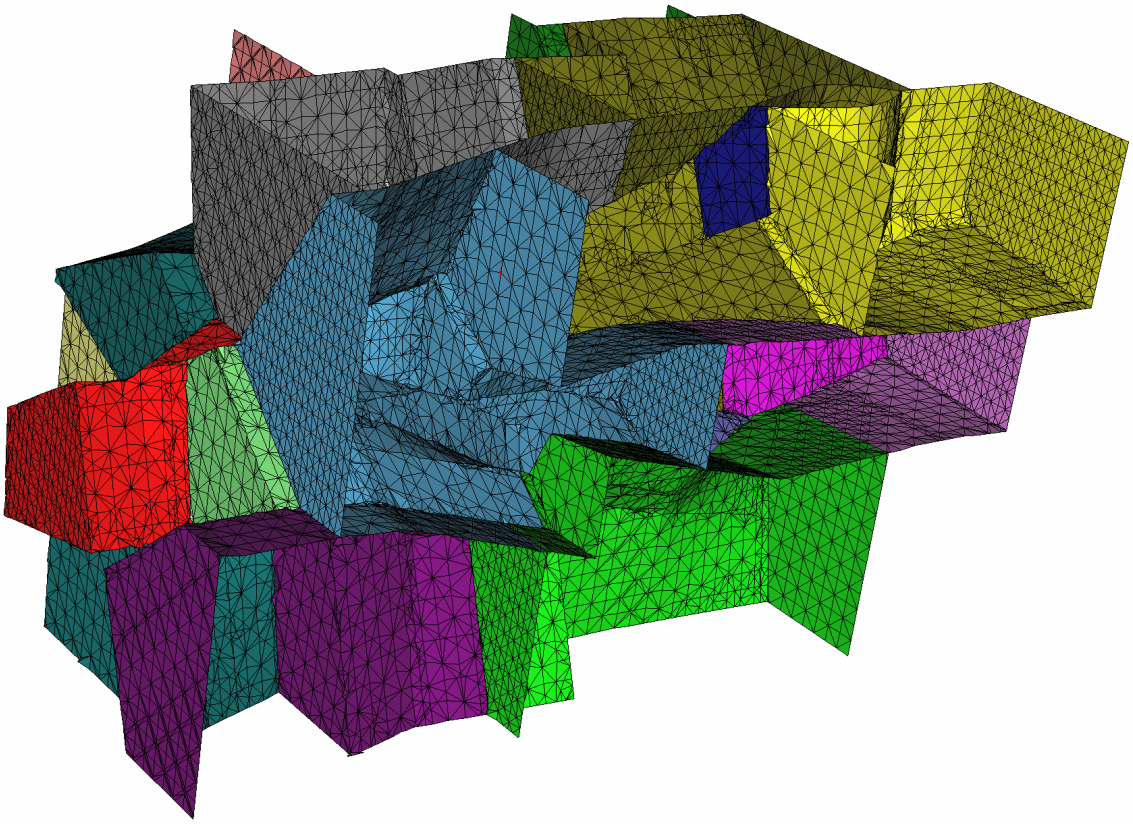

Figure 2: Domain boundaries in a relatively small example. Shown here are the cell domains, each consisting of one or more halflink domains.

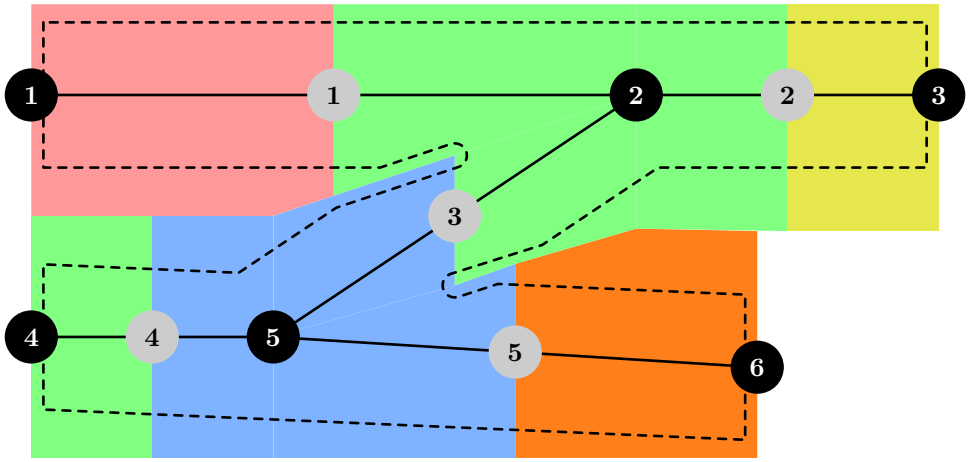

Figure 3: The problem presented to the remesher: each color represents a domain corresponding to a cell, and the dashed line represents the implicit surface to be discretized.

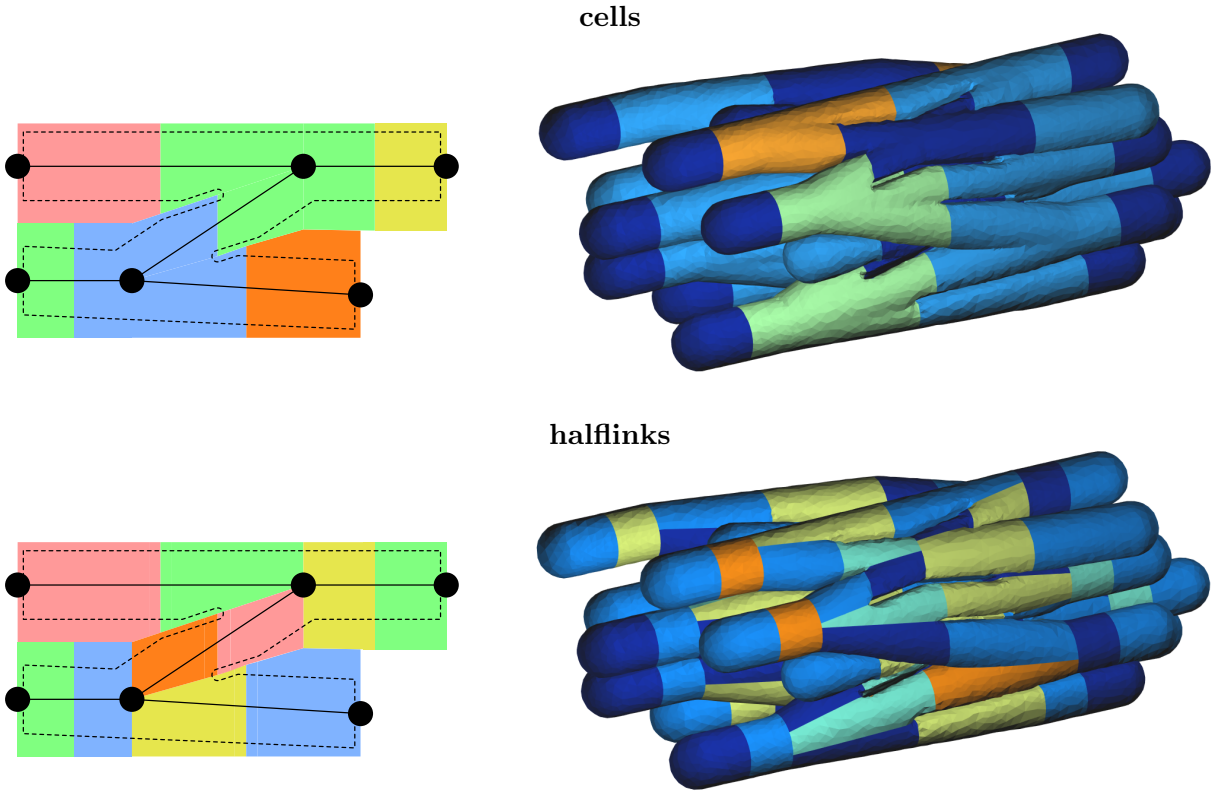

Figure 4: In the right panels, a small network of cells generated by our software is shown. The left panels show the underlying principles in two dimensions. In the top panels, each color labels a cell (colors can be used multiple times for non-connected cells). In the bottom panels the underlying halflink domains are shown. The rounded caps correspond to "cells" that consist of a single halflink at the ends of the network. These are removed for applications.

2.6 Code design

The software developed to create the initial mesh, domain boundaries, and level set, consists of about 5500 lines of C code and uses only standard libraries. Its algorithm for edge collapses follows the approach described by Dobrzynski [11]. Tailored data structures are used to store vertices, edges, and tetrahedra, with specialized properties such as the list of domains to which a vertex belongs. The data structure is somewhat redundant to facilitate access to different entities. Memory allocation is entirely dynamic and related entities link to each other with pointers. Garbage collection after deletion of mesh entities requires a traversal of a linked list for each kind of entity. No attempt was made to parallelize the code.

3 RESULTS

In Figure 4 a small generated cell network is shown. It was produced in a few minutes with no difficulties.

Figure 5 shows a larger example. In this case the `Mmg3d` code was unable to complete the mesh improvement process. We encountered this problem consistently when generating networks of

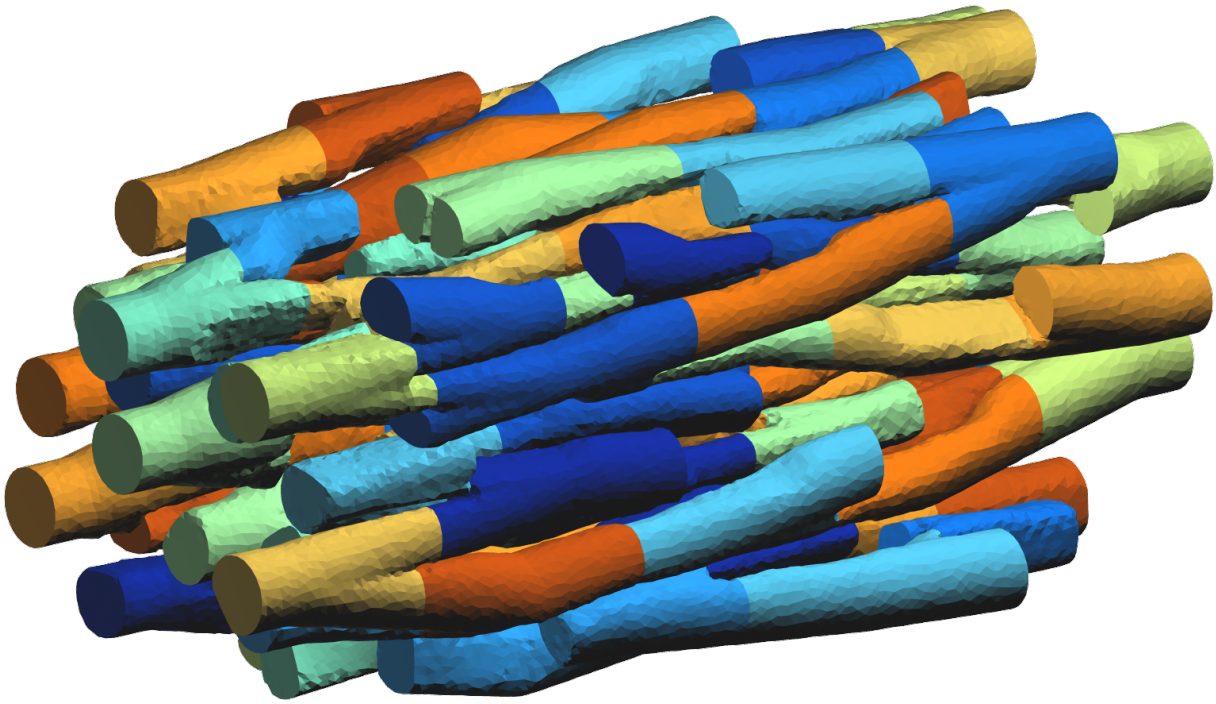

Figure 5: A larger generated cell network. Rough surfaces at some locations witness difficulties in the remesher in this case.

more than a few dozen cells, or when cells were tightly packed.

In order to create larger meshes nevertheless, we designed a small network with cell centers on all of its corners, such that it could be mirrored in each direction to obtain a repeatable unit. Using a small replication code we were then able to generate meshes as large as desired – limited only by the available disk space on the computer. We have tested this method up to $1\,\mathrm{cm}^3$ of tissue, which involved 10^7 cells, $3.7 \cdot 10^{11}$ tetrahedra, and 16 TB storage. A smaller example is shown in Figure 6.

4 DISCUSSION

We described an algorithm to produce artificial models of cardiac muscle tissue with micrometer resolution. To our knowledge, this is the first attempt to produce realistic models of this tissue artificially.

Our algorithm was developed in response to the needs of the European project MICRO-CARD (https://microcard.eu) which has the ambition to simulate cardiac electrophysiology on micrometer-resolution models of cardiac tissue with close to a billion cells. As such models cannot be created from observations now or in the foreseeable future, we must resort to artificial models such as developed here.

This work has, in the first place, shown how challenging the problem is. The meshes that we try to generate are extremely complex, consisting of dozens to hundreds of cells with about 10 µm diameter, separated from each other by (in reality) much less than 1 µm. Hundreds of tetrahedra per cell are necessary to represent the geometry, even though we simplified the cell

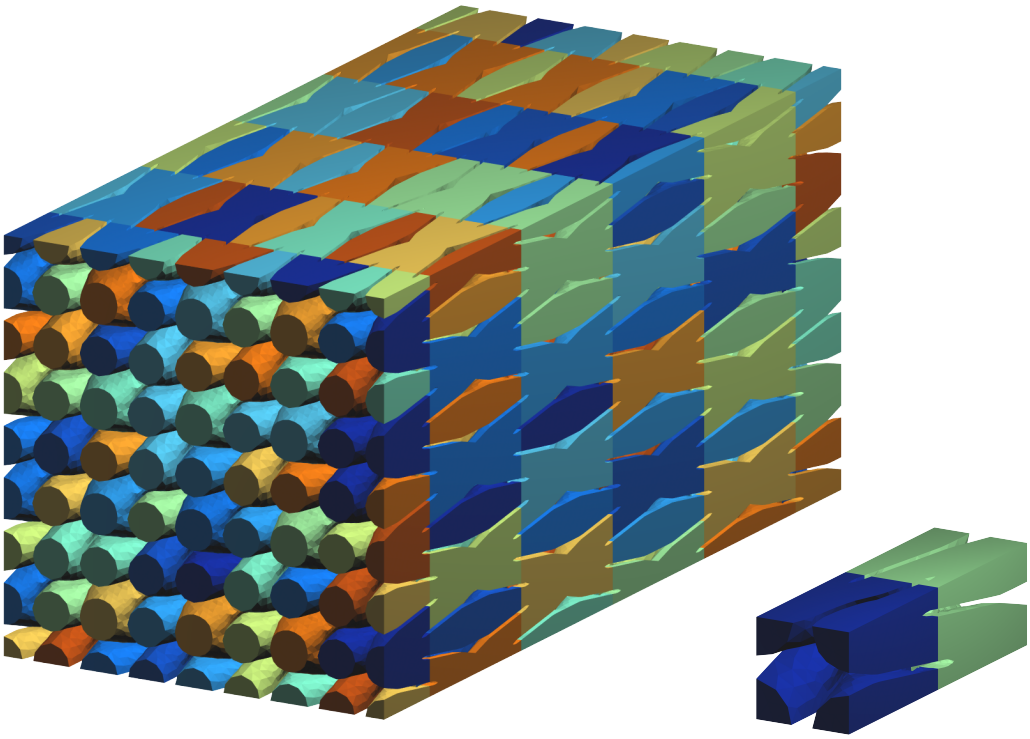

Figure 6: A mesh generated by repeating the small basic block shown on the right.

geometry already to the micrometer scale. Currently these meshes cannot be remeshed properly by the `Mmg3d` software. We suspect that a this is due to a software bug exacerbated by the complexity of these meshes. However, the mesh construction code that we developed is not free from problems either. For example, in many cases the edge splitting/mesh improvement iterations do not converge, and the implicit surface can have sharp ridges when cells are tightly packed. Nevertheless we find our current results encouraging and mandating further efforts, e.g. to ensure smoothness of the implicit surface.

More generally, we would like to highlight that the biological tissues studied here are very complex and difficult to mesh, due to the presence of numerous surface connections that intersect at small angles.

Despite its difficulties our algorithm is able to produce small correct models, and using replication of such a model we are currently able to generate technically realistic meshes that can be used for large-scale testing of code developed in the MICROCARD project.

5 ACKNOWLEDGEMENTS

This work was supported by the European High-Performance Computing Joint Undertaking EuroHPC under grant agreement No 955495 (MICROCARD) co-funded by the Horizon 2020 programme of the European Union (EU) and the French National Research Agency ANR.

This work was granted access to HPC resources of IDRIS under GENCI allocation 2021-A0110307379.

REFERENCES

[1] de Bakker J.M.T., Coronel R., Tasseron S., Wilde A.A.M., Opthof T., Janse M.J., van Capelle F.J.L., Becker A.E., Jambroes G. Ventricular tachycardia in the infarcted, Langendorff-perfused human heart: role of the arrangement of surviving cardiac fibers. *J. Am. Coll. Cardiol.*, 1990;**15**:1594–607.

[2] Coronel R., Casini S., Koopmann T.T., Wilms-Schopman F.J.G., Verkerk A.O., de Groot J.R., Bhuiyan Z., Bezzina C.R., Veldkamp M.W., Linnenbank A.C., van der Wal A.C., Tan H.L., Brugada P., Wilde A.A.M., de Bakker J.M.T. Right ventricular fibrosis and conduction delay in a patient with clinical signs of Brugada syndrome: a combined electrophysiologic, genetic, histopathologic and computational study. *Circulation*, 2005;**112**:2769–2777.

[3] Noble D., Rudy Y. Models of cardiac ventricular action potentials: iterative interaction between experiment and simulation. *Phil. Trans. Roy. Soc. London; Phys. Sc.*, 2001;**359**:1127–1142.

[4] Hoogendijk M.G., Potse M., Linnenbank A.C., Verkerk A.O., den Ruijter H.M., van Amersfoorth S.C.M., Klaver E.C., Beekman L., Bezzina C.R., Postema P.G., Tan H.L., Reimer A.G., van der Wal A.C., ten Harkel A.D.J., Dalinghaus M., Vinet A., Wilde A.A.M., de Bakker J.M.T., Coronel R. Mechanism of right precordial ST-segment elevation in structural heart disease: Excitation failure by current-to-load mismatch. *Heart Rhythm*, 2010;**7**:238–248.

[5] Davidović A., Coudière Y., Bourgault Y. Modelling the action potential propagation in a heart with structural heterogeneities: From high-resolution MRI to numerical simulations. *Int J Numer Meth Biomed Engng.*, 2020;(e3322).

[6] Bishop M.J., Plank G. Bidomain ECG simulations using an augmented monodomain model for the cardiac source. *IEEE Trans. Biomed. Eng.*, 2011;**58**:2297–2307.

[7] Tveito A., Jæger K.H., Kuchta M., Mardal K.A., Rognes M.E. A cell-based framework for numerical modeling of electrical conduction in cardiac tissue. *Front. Phys.*, 2017;**5**:48.

[8] Tveito A., Mardal K.A., Rognes M.E., editors. Modeling Excitable Tissue; The EMI Framework, volume 7 of Simula SpringerBriefs on Computing; Reports on Computational Physiology. Springer, 2021.

[9] Greiner J., Sankarankutty A.C., Seemann G., Seidel T., Sachse F.B. Confocal microscopy-based estimation of parameters for computational modeling of electrical conduction in the normal and infarcted heart. *Front. Physiol.*, 2018;**9**:239.

[10] Dapogny C., Dobrzynski C., Frey P. Three-dimensional adaptive domain remeshing, implicit domain meshing, and applications to free and moving boundary problems. *J. Comp. Phys.*, 2014;**262**:358–378.

[11] Dobrzynski C. Adaptation de Maillage anisotrope 3D et application à l'aéro-thermique des bâtiments. PhD thesis, Université Pierre et Marie Curie-Paris VI, 2005.