

Universal Ventricular Coordinates: a New Way to Transfer Purkinje Networks Between Meshes

Julien Bouyssier^{1,2}, Jason Bayer^{1,2}, Edward Vigmond^{1,2}

¹IHU Liryc, Electrophysiology and Heart Modeling, fondation Bordeaux Université, Pessac-Bordeaux, France

²Univ Bordeaux, IMB, UMR 5251, Talence, France

Abstract

The representation of the Purkinje System (PS) in biventricular (BV) meshes is important for computer simulations of the cardiac electrical activity. Junctions between the PS and myocardium (PMJs) allow the transfer of electrical activity between this network and the myocardium. Unfortunately, having access to a PS mesh is complicated and when possible, transferring this network between different heart meshes is difficult with current mapping methods.

In this paper, we present Universal Ventricular Coordinates (UVCs), a generic coordinate system for BV models allowing the transfer of data between meshes. We describe how they allow easy transfer of a PS between meshes and illustrate the PS mapping process from a rabbit BV mesh to a canine BV mesh.

UVC computation time and the transfer process for PS mapping required minimal computer resources. The whole process can be done very quickly. The method preserves the topology of the created network. In conclusion UVCs is a new promising tool to transfer PS between meshes in order to improve computer simulations of cardiac electrical activity.

1. Introduction

The PS network is a structure of the heart that plays an important part of pump function. This network is a succession of cables that starts from the bundle of His. It runs through the septum, branching out to reach the endocardium and then spread into the myocardium (Fig. 5) with a different depth penetration according to the species. It is including PMJs that transfer the electrical pulses which control heart contractions and send the blood throughout the body. In order to perform numerical simulations comparable to experiments, we need to model and include a PS network mesh in BV models.

Obtaining information about PS network is challenging. It is generally not possible to acquire in vivo

images of the PS, and given its topological complexity, transferring PSs between meshes is difficult. Moreover, current transfer methods may require to manipulating a huge volume of data and may be time consuming and computationally demanding.

UVCs is a new way to transfer data between meshes without previous limitations. It is a global system of material coordinates describing positions on any 3D BV mesh [1]. At each node of a BV mesh, represented by three Cartesian coordinates, we can also assign associate UVCs. Computation of these coordinates is based on solving Laplace's equation with corresponding Dirichlet Boundary Conditions (DBC) chosen to be relevant for the description of the heart.

Data is mapped from cartesian to UVC space on a heart, and then mapped from UVC to cartesian space on a target heart. This mapping process can be used to transfer structural data like PS network between different BV heart meshes.

2. Definition

The UVC coordinate system has four coordinates. The first coordinate, v , identifies which ventricle the point is located in: Left Ventricle (LV) or Right Ventricle (RV). The three remaining coordinates represent spatial position of the point. The second UVC, \mathfrak{z} , describes how high in ventricles the point is, according the long axis of the ventricles. The apex corresponds to the lowest values and the base the highest. The third coordinate, ρ , indicates transmural depth between the endocardium and epicardium. Points of the endocardium surface have the lowest values and points of the epicardium surface the highest. The last coordinate, ϕ , is an angular coordinate that describes the rotation around the long axis of the considered ventricle.

We first illustrate the definition and computation of UVCs with coordinate \mathfrak{z} that represents height variations in ventricles. For a BV heart mesh, we identify the apex and base nodes. To define height variations, we consider

Laplace's equation in the volume mesh to be solved with two DBC: we assign 0 for apex nodes and 1 for base nodes. Solution of Laplace's equation with these DBC defines δ everywhere. The value is rescaled so that the change in δ is linear (Fig. 1).

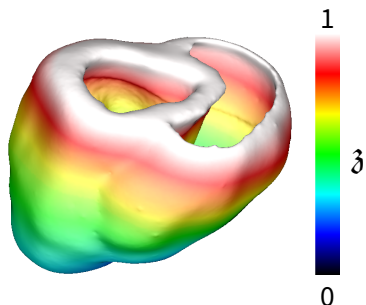


Figure 1. Illustration of δ on canine BV mesh.

The coordinates ρ and ϕ must be solved independently on the LV and RV. We separate them at their junction, by considering the septum of the BV mesh as a part of the LV. Laplace's equation is solved with three DBC: 0 and 1 respectively on LV and RV endocardium surfaces and 0.5 on the RV septum surface. Then each node in the volume mesh has a value between 0 and 1. We use this to set the coordinate v that indicates in which ventricle the considering point lies. Any point with a value ≤ 0.5 is defined as being in the LV and with $v = -1$. A point with a value > 0.5 is located into the RV and with $v = 1$ (Fig. 2).

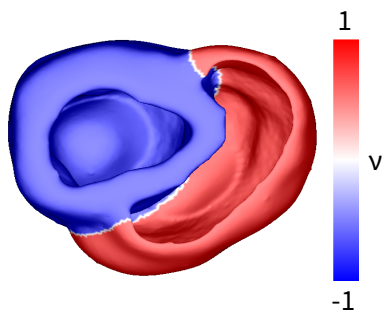


Figure 2. Illustration of v on canine BV mesh.

To compute ρ , that describes variations position of a point between endocardium and epicardium, we apply DBC equal to 0 and 1 for the LV endocardial and epicardial surfaces, respectively. DBC for RV are 0 on the non-septal endocardium surface and 1 on the epicardium surface. We solve the Laplace's equation separately over LV and RV with their respective DBC. It defines for each node of the volume mesh a value ρ between 0 and 1, which describes how far from the endocardium surface of one ventricle is the considering point. A point close to the endocardium or epicardium surface has a ρ value close to 0 or 1, respectively (Fig. 3).

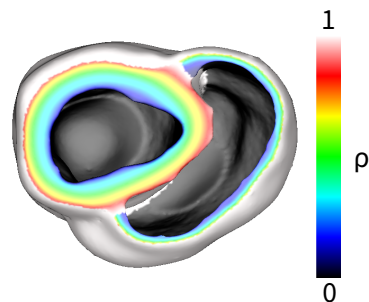


Figure 3. Illustration of ρ on canine BV mesh.

The coordinate ϕ represents the rotation of a point around the long axis of the considered ventricle. To ensure the coordinate will vary smoothly over the LV, we need to define several DBCs for the Laplace's equation: 0 at the middle of the septum surface, $+\pi/2.5$ and $+\pi/2$ at respectively inner and outer surfaces of the anterior junction between LV and RV, and $+\pi$ at the LV free wall. We assign the inverse for the posterior surface of the LV. For the RV, DBC are $+\pi/2$ and $-\pi/2$ at respectively anterior and posterior surfaces of the junction between LV and RV. We solve the Laplace's equation separately over LV and RV with their respective DBC. Then any node of the volume mesh has a value ϕ between $\pm\pi$, depending on anterior or posterior regions. With ϕ we can specify areas: for example nodes in the LV with $-\pi/2.5 < \phi < +\pi/2.5$ form the septum (Fig. 4).

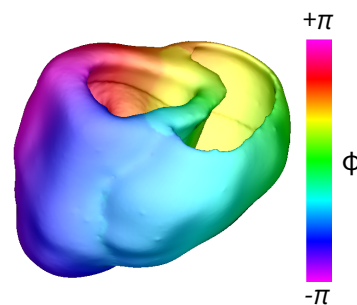


Figure 4. Illustration of ϕ on canine BV mesh.

Computation of UVCs is fast, on the order of several minutes. It took around 10 minutes, with a standard desktop computer, for the calculus of the UVCs presented in Figures 1 to 4 which contains 278788 nodes. The method is practical and straightforward: after providing input points from the BV mesh, an automatic process identifies different surfaces used to compute UVCs. Then Laplace's equations with the corresponding DBC are solved, using the CARP software [2]. The choice of the DBC values is important because they are independent of the mesh and its dimensions. The range of UVCs for any BV heart mesh is the same, explaining why it is a global

spatial framework. That means a point located in one mesh can be, with its UVCs, sought in another mesh, allowing the mapping of data between different BV heart meshes.

mesh. We use the first Python program one last time to construct connections between points of the canine PS mesh. We give as argument the canine PS Cartesian coordinates file and as input the file with all rabbit PS

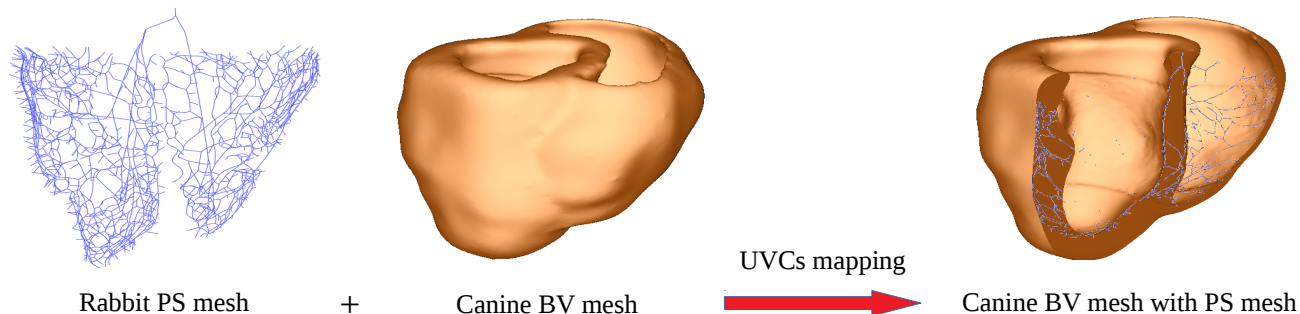


Figure 5. Illustration of the mapping process for PS transfer using UVCs.

3. Methods

We present and illustrate the PS mapping process using UVCs. The goal is to transfer a PS from an existing network of a rabbit BV mesh to a canine BV mesh (Fig.5). The PS can be defined in a heart mesh by a set of points connected by cables, forming a network. Cables are characterized by the spatial position of points in Cartesian coordinates. The PS mapping process consists of transferring coordinates of the rabbit network to the canine mesh using associated UVCs.

A prerequisite is to compute UVCs of both rabbit and canine BV volume meshes. The mapping process strictly speaking relies on two Python programs. With the first code we generate a file with Cartesian coordinates of the rabbit PS. They are extracted from a file containing all network data (connections, size, etc...), readable by the CARP software to perform numerical simulations. The second Python program can compute UVCs of a specific region of a BV mesh. For this we need to give as an argument a file containing Cartesian coordinates of the nodes of the region and as input the files of the volume mesh and its UVCs. We use this program to compute the UVCs for the rabbit PS by interpolating UVCs from the rabbit mesh points to the PS points. Now we have to convert the rabbit PS UVCs back to Cartesian coordinates onto canine mesh. We use again the second Python program to perform this operation. We give as argument the UVCs file of a specific region of a BV mesh and as input the files of the volume mesh and its UVCs. The program generates into a file Cartesian coordinates of the region, according with the volume mesh. More precisely, we search into the UVCs space elements of the canine mesh containing rabbit PS UVCs. After identifying them inverse distance weighted interpolations are performed to computed Cartesian coordinates of the PS for the canine

information. This operation generates the file of the PS network for the canine model. It can be included into the canine model to perform numerical studies with the CARP software.

4. Results

The PS mapping is presented in the first row of Figure 6 with left and middle pictures. The topology was verified to be correct. Because canine and rabbit heart geometries differ, the canine PS is different from the rabbit PS too (distance between ventricles for example), but the global topology is the same. The computation of UVCs for the rabbit mesh (547680 nodes) and the canine mesh (278788 nodes) took around 20 and 10 minutes respectively. Converting Cartesian coordinates of the PS rabbit mesh (6106 nodes) into UVCs took around 4 seconds. It took the same time to convert back UVCs coordinates of the PS mesh on Cartesian coordinates over the canine mesh, creating a PS network compound of 18162 nodes. Elements that were too long could result in propagation failure in computer simulation. Subdividing these elements resolved the issue. Extracting Cartesian coordinates from the PS data file and performing the backward operation is instantaneous.

Using UVCs we can easily perform transformations on the canine PS mesh. We removed some PMJs positioning too high in the PS network but not in septum area (Fig. 6, 2nd row, right picture). We can adjust PMJs depth penetration into myocardium simply by changing their ρ coordinates (Fig. 6, 2nd row left picture). In our example the Moderator Band (MB), which including Purkinje fiber and playing a significant role in myocardium depolarization, is not represented because its geometry and location vary a lot. One solution is to add MB into the canine PS mesh after the mapping process.

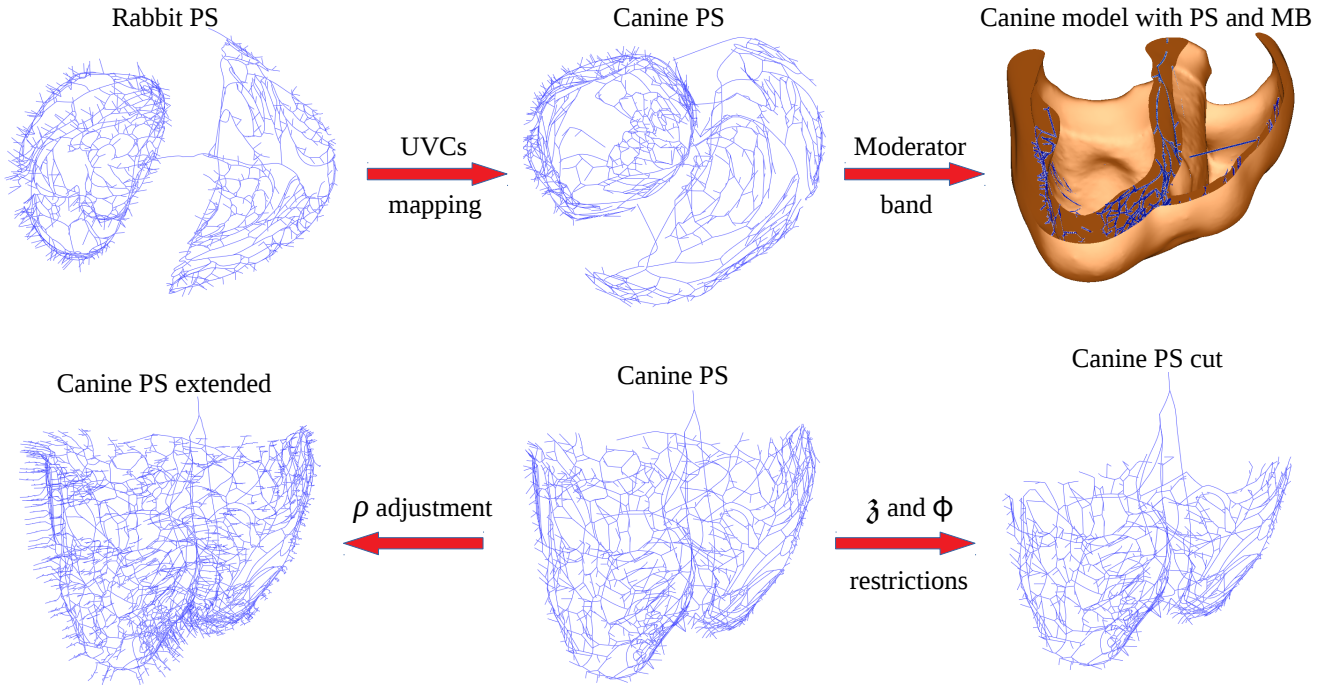


Figure 6. First row : a canine PS mesh (middle) resulting from rabbit PS mapping (left) and shown in the canine model after adding moderator band (right). Second row : illustration of penetration depth (left) and cutting (right) of PMJs for a canine PS mesh (middle).

We identify two candidate points, thanks to their UVCs, for attachment points of the MB and connect them by creating a new cable in the PS mesh (Fig. 6, 1st row, right picture).

5. Discussion

Defining UVCs of BV meshes is a fast process and almost automatic. The maximal computation time is around 20 minutes for the case presented. For bigger or finer meshes, more CPUs can be used to keep computation time in the same order of magnitude. UVCs are varying smoothly, they can be employed in functions that use positional arguments. This is what we did to be able to cut some branches of the PS canine mesh.

The PS transfer example is very fast, in the order of seconds. The mapping method keeps the global topology of the source network.

One limitation of the mapping process is to search into UVCs space. ζ and ρ have the same range in UVCs but not in corresponding Cartesian coordinates. Rescaling with parameters is needed to compute distance in UVCs space. Ability to identify elements containing transferred nodes, and so the mapping efficiency, rely on these parameters which are set empirically. Another difficulty is to avoid error induced by the mapping at the LV-RV junction because UVCs are discontinuous at this border.

6. Conclusion

Mapping a PS network between different BV meshes using UVCs is a promising tool. The method is efficient, fast and required low computer resources.

Acknowledgements

This work was supported by the ERACoSysMed PUSH CART project.

References

- [1] Bayer JD *et al.* Universal Ventricular Coordinates: A Generic Framework for Describing Position within Heart and Transferring Data. *Medical Image Analysis* 2018;45;83-93.
- [2] <https://carpentry.medunigraz.at/carputils/>

Address for correspondence:

Bouyssier Julien
 IHU Liryc – Hôpital Xavier Arnoz-Avenue du Haut Lévêque-
 33604 Pessac cedex
julien.bouyssier@ihu-liryc.fr