Reconstruction of atrial excitation conduction velocities and implementation into the inverse problem of electrocardiography

D.-T. Rudolph¹, W.H.W. Schulze¹, D. Potyagaylo¹, M.W. Krueger¹, O. Dössel¹
¹KIT Karlsruhe, Institut für Biomedizinische Technik, Karlsruhe, Deutschland, publications@ibt.kit.edu

Abstract

An exact diagnosis of the heart’s electrophysiological behaviour is currently linked to invasive catheter interventions. A reconstruction of electrophysiological properties from non-invasive ECG data would be preferred. This requires solving the so-called inverse problem of electrocardiography which, however, is strongly underdetermined. Therefore, it needs a priori knowledge to be solvable in a reliable manner. This paper presents a method to calculate excitation conduction velocities (ECV) from activation times and an implementation into an inverse solver of activation times. The sole calculation of ECVs achieved a mean relative error of 0.0916 ± 0.0744. In addition, an inverse solver for activation times is tested with an adaption scheme for the related ECVs and without. The resulting mean relative error of the ECVs improved from 0.3228 ± 0.2768 to 0.2715 ± 0.2318 using the adaption of ECVs.

1 Introduction

Electrophysiological dysfunctions of the heart can be seen and diagnosed with an ECG. However, it does not reveal the exact location of the source of the dysfunction. Tedious invasive interventions are necessary to find the origin with probes inside the heart. The inverse problem of electrocardiography describes the attempt to derive e.g. transmembrane voltages in the course of a heart beat out of an electrocardiogram. Since the problem is heavily underdetermined and sensitive to noise, it is yet only solved to a certain degree of precision. Additional a priori knowledge is necessary to improve the solution. In this paper, the inverse problem is solved for activation times. Excitation conduction velocities (ECV) are reconstructed from activation times and compared to the true ECVs to form an additional boundary condition for the inverse problem of ECG. First the forward and the inverse problem of electrocardiography are described briefly. Then the algorithm to calculate ECVs is introduced. The method is applied on a simulated ECG of atrial activation propagation and the results are evaluated.

2 Methods

2.1 Simulation of an electrocardiogram

A simulation of the transmembrane voltages of an atrial beat is performed on a personalized heart model with a cellular automaton [1]. Three different excitation conduction velocities are defined:

- 0.4 m/s in the isthmus
- 1.0 m/s in the crista terminalis
- 0.75 m/s otherwise

The bidomain model is used to calculate the related extracellular potentials by treating the intra- and extracellular space separately, each with its own potential distribution and conductivity tensor. The relation between transmembrane voltages inside the heart’s myocardium and the body surface potential map (BSPM) is linear:

$$\Phi_{\text{BSPM}} = A \cdot X_{\text{TMV}},$$

with A being the lead field matrix. $\Phi_{\text{BSPM}}$ and $X_{\text{TMV}}$ are both matrices whose columns represent the time steps and whose lines represent one node or unipolar lead, respectively. Eq. 1 (the forward problem) is solved for $\Phi_{\text{BSPM}}$ using personalized FEM-models [2].

2.2 Reconstruction of activation times

The inverse solver then uses the resulting 80-lead ECG to reconstruct the activation times $\tau$ at each node within the atria. This is done on a reduced heart model which consists of $m = 2332$ nodes in an unstructured tetrahedral mesh. The lead field matrix A cannot be inverted straight forward, therefore it is not possible to simply multiply $A^{-1}$ to both sides of Eq. 1. In addition, the problem is heavily underdetermined. Therefore, a first assumption of $\tau$ is made and then optimized iteratively. In each iteration a cost term $\Psi$ is formulated:

$$\Psi(\bar{\tau}) = ||\Phi - A \cdot X(\bar{\tau})||^2 + \lambda^2 ||\Delta \bar{\tau}||^2 \rightarrow \text{min.}$$

Here, $\Delta$ is a discrete approximation of the volumetric Laplacian operator and $\lambda$ is a regularization parameter which balances the cost term. The implementation is described by Fischer et al. in [3].

2.3 Calculation of excitation conduction velocities

In each iteration step the computed activation times $\bar{\tau}$ are used to reconstruct the excitation conduction velocities at each node of the heart model. Each node is member of several tetrahedrons and has therefore several direct neighbours...
linked to it via the edges of the tetrahedrons. Since the Cartesian coordinates of the nodes are known, a velocity between the node and each of its neighbours can be calculated using the Euclidian distances:

\[
v_{kn} = \frac{\vec{s}_k - \vec{s}_n}{\vec{s}_k - \vec{s}_n} = \frac{\Delta \vec{s}_{kn}}{\Delta \vec{s}_{kn}},
\]

with \( k \) being the number of the current node and \( n \) the number of the current neighbour. No velocity \( v_{kn} \) is calculated if \( \Delta \vec{s}_{kn} = 0 \).

The minimum of the calculated velocities \( v_{kn} \) is taken as the best assumption for the velocity \( v_k \) at node \( k \). It is saved for every node in an array \( e \vec{c} v \):

\[
v_k = \min(v_{k1}, v_{k2}, \ldots, v_{kp})
\]

\[
e \vec{c} v = \begin{pmatrix}
  v_1 \\
  \vdots \\
  v_k \\
  \vdots \\
  v_m
\end{pmatrix}
\]

### 2.4 Quality assessment of the calculated excitation conduction velocities

Two values are calculated only once for each heart model, because their value depends on the arrangement of nodes in the tetrahedral mesh:

- A censor \( c \) is introduced for each node, which is either 0, if the node is in a tissue connecting right and left atrium, or 1 otherwise. This is done because the ECV on nodes in tissues connecting right and left atrium cannot be reconstructed reliably.
- A simulation with isotropic velocities is performed and the ECVs reconstructed. A relative error \( \epsilon_{\text{ref}} \) can be calculated. This calculation shows for each node with which reliability the algorithm can calculate the ECV. Therefore it provides a reference value for the expectable relative error of each node.

### 2.5 Implementation into the iteration scheme

In the course of one iteration step, the inverse solver calculates the change of \( \vec{\xi} \) for the next step. A distance \( \text{dist}_i \) is calculated by a parabolic approximation to determine the amount of change of \( \vec{\xi} \). A gradient descend method is used to determine the direction of change:

\[
\vec{\xi}_{i+1} = \vec{\xi}_i + \text{dist}_i \cdot \frac{\hat{\text{grad}}_i}{||\text{grad}_i||}
\]

where \( \hat{\text{grad}}_i = \frac{\partial \Psi(\vec{\xi}_i)}{\partial \vec{\xi}} \bigg|_{\vec{\xi}_i} \)?

This term should now be influenced by the result of a calculation of ECVs. The idea is that with correct ECVs no change in \( \vec{\xi} \), the node-specific component of \( e \vec{c} v \), is necessary. With incorrect ECVs, however, \( \vec{\xi} \) should be changed to full extent by the inverse solver. This leads to a new rule of estimation for \( \vec{\xi}_{i+1} \) that is derived from Eq. 6:

\[
\vec{\xi}_{i+1} = \vec{\xi}_i + \text{dist}_i \cdot \frac{\hat{\text{grad}}_i}{||\text{grad}_i||},
\]

3 Results

3.1 Results of a calculation of excitation conduction velocities

The reconstruction of ECVs of a simulation with isotropic velocities resulted in a mean relative error \( \epsilon_{\text{ref}} \) of 0.0916 ± 0.0744. This shows that the calculation of ECVs is working with small errors compared to the expected error of a reconstruction of activation times. Because of the mesh producing false connections between nodes in the area of the septum, the reconstructed ECVs were strongly erroneous here, as can be seen in Fig. 2. These were partly covered by the censor \( c \), which identified 92 nodes. Consequently, they were not taken into account (see Eq. 8).

To see if the algorithm can also reconstruct more complex conduction patterns, a reconstruction of the before mentioned simulation with three different velocities was performed. This resulted in a mean relative error of 0.0987 ± 0.1005, which is only an insignificant change for the worse compared to simple conduction patterns.

The computation time is hereby of minor concern. The neighbours to each node have to be computed only once, which takes about one minute on a standard 2.8 GHz CPU. The reconstruction of ECVs takes less than 10 seconds. The computation time increases linearly with the number of nodes. Since no particular effort was put into the optimization, a further decrease of the computation time may be possible.

3.2 Results of reconstructions of activation times

An inverse solver was run with different values of \( \lambda \) and different initial distances \( \text{dist}_0 \). Results of reconstructions with and without implemented \( \vec{w} \) were compared using the mean relative error of the ECVs in every iteration step, which was between 25% and 35%. It is remarkable, that the first assumption of \( \vec{\xi} \) produces an error of about 30%. In case of no implemented calculation of ECV, this is the best value
Figure 1 Comparison of the relative error of the ECVs over the number of iterations. $b = 100$, $\lambda = 10^{-5}$.

Figure 2 Results of a calculation of ECVs based on a simulation with an isotropic velocity of 0.75 m/s. The relative error (RE) is shown.

Figure 3 Comparison of the relative error of the ECVs over the number of iterations. $b = 100$, $\lambda = 10^{-6}$.

Figure 4 Relative error (RE) of the last iteration step interpolated on the heart model. $b = 100$, $\lambda = 10^{-5}$.

achieved during the reconstruction. Also, with a calculation of ECV implemented, the first assumption of $\mathbf{r}$ provides the best error rate for most parameter configurations. Only a few settings produced a significantly better result. Fig. 1 ($\lambda = 10^{-5}$) and Fig. 3 ($\lambda = 10^{-6}$) show two of these. The initial distance parameter $b$ was set to 100 for both reconstructions. The resulting relative error for $\lambda = 10^{-5}$ interpolated on the heart model is depicted in Fig. 4. With this setting, the mean relative error of the ECVs in the last iteration step was $0.2715 \pm 0.2318$ with the implementation of ECVs.

4 Conclusion

The algorithm to reconstruct ECVs introduced in this work performs satisfyingly. It is able to calculate ECVs out of activation times on an unstructured tetrahedral mesh. With a relative error of about 10% it has error rates comparable to those of previous works that used more sophisticated algorithms [4, 5]. It is easily implementable and provides an extended error evaluation.

ECVs have a visible impact on the reconstruction of activation times. With a proper setting of parameters, ECVs of the reconstructed activation times could be improved by about 5%. The implementation shown in this work influences the activation times directly. However, it performs only with partial improvement of the results. The presented algorithm for the calculation of ECVs may be used in the future to introduce upper ECV-boundaries on the solution of the cost term (Eq. 2).

In a practical application, the true ECVs for the calculation of $\hat{\mathbf{r}}_{rel}$ in Eq. 8 may be derived from scar imaging or single catheter measurements. Additionally, future work will have to be concerned with an extended parameter optimization.
5 References


