






Digital Twinning of Cardiac Electrophysiology Models From the Surface ECG: A Geodesic Backpropagation Approach

Thomas Grandits , Jan Verhülsdonk , Gundolf Haase , Alexander Efland , and Simone Pezzuto 

Abstract—The eikonal equation has become an indispensable tool for modeling cardiac electrical activation accurately and efficiently. In principle, by matching clinically recorded and eikonal-based electrocardiograms (ECGs), it is possible to build patient-specific models of cardiac electrophysiology in a purely non-invasive manner. Nonetheless, the fitting procedure remains a challenging task. The present study introduces a novel method, Geodesic-BP, to solve the inverse eikonal problem. Geodesic-BP is well-suited for GPU-accelerated machine learning frameworks, allowing us to optimize the parameters of the eikonal equation to reproduce a given ECG. We show that Geodesic-BP can reconstruct a simulated cardiac activation with high accuracy in a synthetic test case, even in the presence of modeling inaccuracies. Furthermore, we apply our algorithm to a publicly available dataset of a biventricular rabbit model, with promising results. Given the future shift towards personalized medicine, Geodesic-BP has the potential to help in future functionalizations of cardiac models meeting clinical time constraints while maintaining the physiological accuracy of state-of-the-art cardiac models.

Index Terms—Cardiac digital twin, eikonal model, ECG, electrophysiology, backpropagation, machine learning.

I. INTRODUCTION

CARDIAC digital twinning is a potential pillar of future's precision cardiology [1]. A digital twin is a computational replica of the patient's heart, from its anatomy and structure up to its functional response. Commonly, it is obtained by fitting the parameters of a generic computational model to the clinical data of the patient. Here, we focus on cardiac electrophysiology, where we aim at personalizing a cardiac conduction model based

Manuscript received 8 August 2023; revised 11 October 2023; accepted 29 October 2023. Date of publication 4 December 2023; date of current version 21 March 2024. This work was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Grants EXC-2047/1-390685813 and EXC2151-390873048. (Corresponding author: Simone Pezzuto.)

Thomas Grandits is with the Department of Mathematics and Scientific Computing, University of Graz, Austria, and also with NAWI Graz, Austria.

Jan Verhülsdonk and Alexander Efland are with the Institute for Applied Mathematics, University of Bonn, Germany.

Gundolf Haase is with the Department of Mathematics and Scientific Computing, University of Graz, Austria.

Simone Pezzuto is with the Laboratory of Mathematics for Biology and Medicine, Department of Mathematics, University of Trento, 38123 Trento, Italy (e-mail: simone.pezzuto@unitn.it).

Digital Object Identifier 10.1109/TBME.2023.3331876

on the eikonal equation, from non-invasive electrocardiographic (ECG) recordings [2], [3].

Eikonal-based propagation models are an excellent basis for cardiac electrophysiology digital twins. They provide a good balance between computational efficiency and physiological accuracy for cardiac activation maps [4], [5], [6]. When supplemented with the lead field approach, they can also provide accurate ECGs [3], [7]. Hence, the parameters of eikonal-based models could be fitted from non-invasive clinical data—surface potentials and cardiac imaging—in an efficient manner. This goal can be achieved using optimization methods based on stochastic sampling strategies [3], [8], Bayesian optimization [9], gradient-descent [10], or deep learning approaches [11].

The numerical efficiency of the optimization procedure relies on two factors: the computational cost of the forward problem, and the number of samples required to achieve convergence within a given tolerance. Gradient-based methods can address the latter, if gradient computation is efficient and robust. For the former, many efficient eikonal solvers exist, but not all of them are suitable for automatic differentiation. A major criticality is that they internally solve a local optimization problem [12], [13], one for each element of the grid and for possibly many iterations. This may represent a bottleneck in the computation of the gradient. A possible solution is to compute the gradient for the continuous problem, which can be shown is equivalent to the computation of geodesic paths [10], [14], and then discretize. The *optimize-then-discretize* strategy is however not ideal, because it may introduce numerical noise in the gradient and degrade the convergence rate.

In this paper, we introduce a novel inverse eikonal solver—named *Geodesic-BP*—and use it to efficiently solve the ECG-based digital twinning problem for ventricular models. To this end, we start by introducing a highly-parallelizable anisotropic eikonal solver for triangulations in arbitrary dimension. The key contribution is a novel approach for the solution of the local optimization problem and based on a linear convex non-smooth optimization method. Locally, the linear geodesic path within each simplicial element is approximated with a guaranteed bound, similar to many presented eikonal solvers [12]. This solver can be implemented using current machine learning (ML) libraries to efficiently calculate the solution of the eikonal equation. The forward solver is suitable for the computation of the gradient via the backpropagation algorithm, effectively enabling a *discretize-then-optimize* strategy

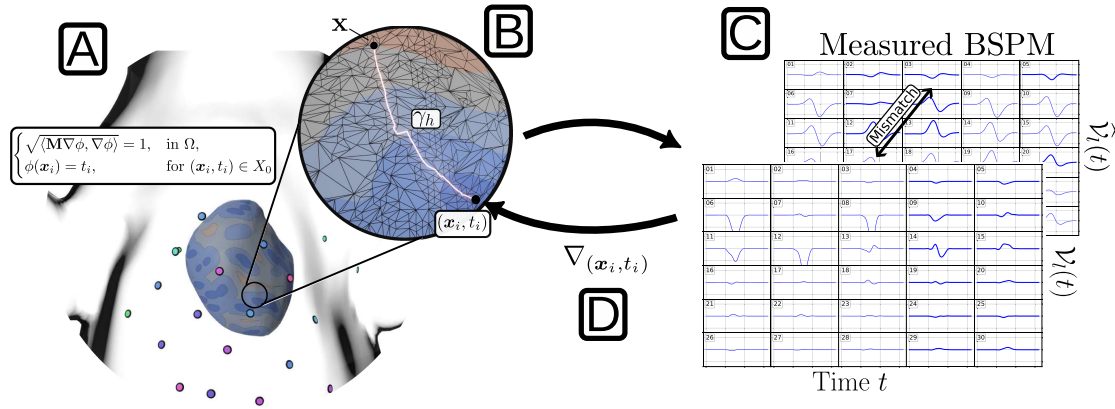


Fig. 1. Anisotropic eikonal equation is an efficient tool to compute the electrical activation of a heart (A), here shown on a public rabbit torso model. Many solvers utilize the relation of the eikonal equation to geodesic paths, which in the case of \mathcal{P}_1 -meshes become piecewise-linear paths γ_h between the initial conditions (\mathbf{x}_i, t_i) and points on the surfaces x (B). By applying a non-linear transformation explained in Section II-D, we can compute the measurable electrical activation on the torso surface and compute the mismatch against the measured potentials at sparse points (C). Implementing the approach in a differentiable manner using a backpropagation framework such as pytorch, we can efficiently calculate the gradient w.r.t. the initial conditions and use it to optimally choose (\mathbf{x}_i, t_i) to match the measured potentials (D).

(Geodesic-BP: Geodesic Back-Propagation). We will show that the backpropagation through such a solver is equivalent to piecewise-linear geodesic backtracking [10], [15], [16]. Finally, we will demonstrate the potential of these methods for the problem of estimating initial eikonal conditions to match a given body surface potential map (BSPM) in a synthetic 2D study, and real 3D torso model.

In summary, the major contributions are as follows:

- 1) A novel, highly-parallelizable anisotropic eikonal solver, well-suited for GPGPU architectures and ML libraries.
- 2) A backpropagation-based gradient computation, internally related to the computation of geodesics.
- 3) An efficient implementation of the method.
- 4) We demonstrate how to use Geodesic-BP for the identification of cardiac activation sequence from the non-invasive ECG recordings.
- 5) We show that the inverse procedure is robust to model perturbations.

An outline of Geodesic-BP is provided in Fig. 1.

II. MATERIAL AND METHODS

In this section, the modeling assumptions for the electrical propagation inside the heart are introduced along with an efficient method for computing this activation in a parallelizable way. Then, we elaborate on the computation of the electrical activation given measurements on the torso leads, which ultimately leads to the formulation of the inverse ECG problem used throughout all experiments.

A. Cardiac Activation

We model cardiac activation with the anisotropic eikonal equation (see [6], [7], [17]), which reads as follows

$$\begin{cases} \sqrt{\langle \mathbf{M}(\mathbf{x}) \nabla \phi(\mathbf{x}), \nabla \phi(\mathbf{x}) \rangle} = 1, & \text{in } \Omega, \\ \phi(\mathbf{x}_i) = t_i, & \text{for } (\mathbf{x}_i, t_i) \in X_0, \end{cases} \quad (1)$$

where $\Omega \subset \mathbb{R}^d$, $d \geq 2$, is a bounded domain, $\phi : \Omega \rightarrow \mathbb{R}$ is the activation map, $\langle \cdot, \cdot \rangle$ is the inner product in \mathbb{R}^d , and $\mathbf{M} : \Omega \rightarrow \mathcal{S}_d$ is a symmetric positive definite (s.p.d.) tensor field, where $\mathcal{S}_d \subset \mathbb{R}^{d \times d}$ denotes the space of d -times s.p.d. tensors, used to account for the orthotropic conduction velocity inside the heart [17]. The onset of activation is determined by the set of initial conditions: $X_0 = \{(\mathbf{x}_i, t_i)\}_{i=1}^K$, where $\mathbf{x}_i \in \Omega$ and t_i are respectively the location and the onset timing of the i -th activation site. For convenience, we additionally define the norm in the metric \mathbf{D} , i.e. $\|\mathbf{x}\|_{\mathbf{D}} = \sqrt{\langle \mathbf{D}\mathbf{x}, \mathbf{x} \rangle}$ for $\mathbf{D} \in \mathcal{S}_d$.

Next, we discretize the domain Ω (assumed polytopal) by a simplicial triangulation. We denote by $V = \{\mathbf{v}_i\}_{i=1}^{n_v}$ and $\mathcal{E} = \{E_j\}_{j=1}^{n_e}$ respectively the set of vertices and elements of the triangulation. Each element $E_j \in \mathcal{E}$ is a d -simplex (triangle in 2-D, tetrahedron in 3-D) with $d+1$ vertices. We also define $E_{j \setminus i} \subset \partial E_j$ as the $(d-1)$ -dimensional face of E_j opposite to vertex \mathbf{v}_i . Note that $E_{j \setminus i}$ is a $(d-1)$ -simplex (triangle in 3-D, segment in 2-D). For a general unstructured grid, we can approximate the activation time with a piecewise linear finite element function $\phi : \Omega \rightarrow \mathbb{R}$, see [18]. (For the sake of simplicity, we use the same symbol as for the continuous case.) For the nodal values we use the shorthand notation $\phi_i = \phi(\mathbf{v}_i)$. Following [12], [19], [20], we approximate $\mathbf{D}(\mathbf{x}) = \mathbf{M}^{-1}(\mathbf{x})$ as a piecewise-constant function and set $\mathbf{D}_j = \mathbf{D}(\mathbf{x})|_{E_j}$.

We numerically solve the anisotropic eikonal (1) by locally enforcing the following optimality condition for the nodal values:

$$\phi_i = \min_{E_j \in \omega_i} \phi_{j \setminus i}^*, \quad (2)$$

where $\omega_i = \{E \in \mathcal{E} : \mathbf{v}_i \text{ is a vertex of } E\}$ is the set of elements containing \mathbf{v}_i and $\phi_{j \setminus i}^*$ is the minimum of the following optimization problem:

$$\phi_{j \setminus i}^* = \min_{\mathbf{x} \in E_{j \setminus i}} \phi(\mathbf{x}) + \|\mathbf{v}_i - \mathbf{x}\|_{\mathbf{D}_j}. \quad (3)$$

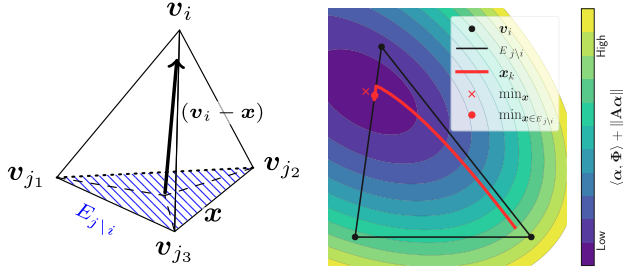


Fig. 2. Conceptual visualization of a linear upwind discretization (5) on the example of a single tetrahedron (left). Finding the minimum upwind origin $\mathbf{x} = \sum_n \alpha_n \mathbf{v}_{j_n}$ can be cast into a non-smooth linear optimization problem, solvable using the FISTA algorithm (right).

The optimality condition (2) is known as Hopf-Lax formula. It is possible to show that, under regularity assumptions on Ω , \mathbf{D} , and the triangulation, the numerical solution of (2) correctly converges to the unique viscosity solution of (1) when the triangulation is refined [21]. As explained later, this principle is at the basis of Geodesic-BP.

B. Solution of the Local Problem

In what follows, we show how to efficiently and in a parallelizable way solve (3). We observe that (2) and (3) consists of a series of constrained minimization problems, one for each element of the patch. Clearly, each local problem in (3) is convex, because the objective function is convex in \mathbf{x} (note that ϕ is linear on $E_{j \setminus i}$) and the constraint is a simplex, thus convex as well. Instead of relying on exact formulas, e.g., available in 2-D and 3-D from [20], we propose here a new dimension-independent algorithm for the solution of (3).

With no loss of generality, we suppose that in (3) the vertices of E_j are $\{\mathbf{v}_{j_1}, \dots, \mathbf{v}_{j_{d+1}}\}$ and ordered such that the last vertex $\mathbf{v}_{j_{d+1}}$ always corresponds to \mathbf{v}_i . Next, we consider the set of barycentric coordinates for the face $E_{j \setminus i}$:

$$C_d = \{\alpha \in \mathbb{R}^d : \langle \alpha, \mathbf{1} \rangle = 1 \wedge \alpha \geq 0\}, \quad (4)$$

such that each point \mathbf{x} in a face $E_{j \setminus i}$ can be uniquely written as $\mathbf{x} = \sum_{n=1}^d \alpha_n \mathbf{v}_{j_n}$. Similarly, the \mathcal{P}_1 function $\phi(\mathbf{x})$ on $E_{j \setminus i}$ is given by the linear combination of the vertex values: $\phi(\mathbf{x}) = \sum_{n=1}^d \alpha_n \phi_{j_n}$. Thus, (3) becomes

$$\begin{aligned} \phi_{j \setminus i}^* &= \min_{\alpha \in C_d} \sum_{n=1}^d \alpha_n \phi_{j_n} + \left\| \sum_{n=1}^d \mathbf{v}_{j_n} \alpha_n - \mathbf{v}_i \right\|_{\mathbf{D}_j} \\ &= \min_{\alpha \in C_d} \langle \alpha, \Phi_{j \setminus i} \rangle + \|\mathbf{A}_{j \setminus i} \alpha\|_2, \end{aligned} \quad (5)$$

where $\Phi_{j \setminus i} = (\phi_{j_1}, \dots, \phi_{j_d})^\top$ is the vector of nodal values of $\phi|_{E_j}$, except for ϕ_i , and

$$\mathbf{A}_{j \setminus i} = \mathbf{D}_j^{1/2} \left(\mathbf{v}_{j_1} \Big| \mathbf{v}_i \quad \dots \quad \mathbf{v}_{j_d} \Big| \mathbf{v}_i \right).$$

A graphical representation of the local optimization problem is given in Fig. 2. (5) now takes a common form of forward/backward-splitting algorithms, where the function to

Algorithm 1: Solution of the local problem in (5)

Data: \mathbf{A} , L , $\Phi = (\phi_1, \dots, \phi_d)^\top$
Result: α_{n_f}

- 1 $\hat{\alpha}_0 = \frac{1}{d} \mathbf{1}$
- 2 **for** $k = 1$ **to** n_f **do** #FISTA main loop
- 3 $h = \|\mathbf{A} \hat{\alpha}_{k-1}\|$
- 4 $\nabla h = h^{-1} \mathbf{A}^\top \mathbf{A} \hat{\alpha}_{k-1}$ #Eq. (6)
- 5 $\tilde{\alpha}_k = (\hat{\alpha}_{k-1} - L^{-1} \nabla h) - L^{-1} \Phi$
- 6 $\alpha_k = \text{proj}_{C_d}(\tilde{\alpha}_k)$ #See [23, Figure 1]
- 7 $\hat{\alpha}_k = \alpha_k + \beta_k (\alpha_k - \alpha_{k-1})$ #Acceleration

minimize is a linear combination of two convex functions with one potentially being non-smooth. Such problems can be minimized using the FISTA algorithm [22], which alternately optimizes using a proximal step into the non-smooth function and a gradient step in the direction of the smooth function. The smooth part of (5) was thus chosen as $h(\alpha) = \|\mathbf{A} \alpha\|_2$, and the non-smooth part as $g(\alpha) = \langle \alpha, \Phi \rangle + \delta_{C_d}(\alpha)$, where δ_C is the indicator function on the set C such that

$$\delta_C(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x} \in C, \\ \infty, & \text{otherwise.} \end{cases}$$

The gradient of the smooth part $h(\alpha)$ is given by

$$\nabla h(\alpha) = \frac{\mathbf{A}^\top \mathbf{A} \alpha}{h(\alpha)}, \quad (6)$$

which is well-defined since $h(\alpha) > 0$ for $\alpha \in C_d$ and non-collapsed elements. The proximal step in the non-smooth part $g(\alpha)$ is then

$$\begin{aligned} \text{prox}_{\frac{1}{L}g}(\tilde{\alpha}) &= \arg \min_{\alpha} g(\alpha) + \frac{L}{2} \|\alpha - \tilde{\alpha}\|^2 \\ &= \text{proj}_{C_d}(\tilde{\alpha} - L^{-1} \Phi), \end{aligned}$$

where L is a constant. The projection proj_{C_d} onto simplices is available in closed form [23, Fig. 1]. The ISTA algorithm consists of the following update step:

$$\alpha_{k+1} = \text{proj}_{C_d} \left((\alpha_k - L^{-1} \nabla h(\alpha_k)) - L^{-1} \Phi \right). \quad (7)$$

An optimal choice of L in the algorithm is the Lipschitz constant of ∇h , which we can bound as

$$L \leq \frac{\|\mathbf{A}^\top \mathbf{A}\|}{\underline{h}} \left(1 + \frac{\|\mathbf{A}\|}{\underline{h}} \right), \quad (8)$$

where $\|\cdot\|$ in (8) refers to the spectral norm of the operator and $\underline{h} = \frac{1}{d\sqrt{d}} \sigma_{\min}(\mathbf{A}) \leq \|\mathbf{A} \mathbf{x}\|_2 = h(\mathbf{x})$ is a lower bound of $h(\mathbf{x})$, for $\sigma_{\min}(\mathbf{A})$ being the smallest singular value of \mathbf{A} . Note that \mathbf{A} is constant in each element, so the local Lipschitz constants can be efficiently pre-computed.

We summarize the solution of the local problem in Algorithm 1, where we also include the acceleration on Line 7 with $\beta_k = \frac{k-1}{k+1}$, as per the FISTA method, see [22]. Note that Algorithm 1 works for arbitrary spatial dimension $d \in \mathbb{N}$. Since the local problem is convex, it has a unique minimum α^* . The

Algorithm 2: Geodesic-BP

Data: \mathcal{P}_1 mesh $M = (V, \mathcal{E})$, piecewise-constant metric \mathbf{D} , initial conditions $X_0 = \{(\mathbf{x}_i, t_i)\}_{i=1}^K$

Result: Geodesic distances ϕ_K

- 1 $\forall \mathbf{v} \in V : \phi_{-1}(\mathbf{v}) = \phi_0(\mathbf{v}) = \infty$
- 2 $\forall (\mathbf{x}_i, t_i) \in X_0 : \phi_{-1}(\mathbf{x}_i) = \phi_0(\mathbf{x}_i) = t_i$
#Compute all simplex upwind directions towards each vertex i inside element j
- 3 $\tilde{E} = \bigcup_{i=1}^{d+1} \bigcup_j E_{j \setminus i}$ Compute for each element $\mathbf{A}_{j \setminus i}, L_{j \setminus i}$ #see Eqs. (5) and (8)
- 4 **for** $k = 1$ **to** K **do**
- 5 $\phi_k := \phi_{k-1}$
#Compute all updates in parallel
- 6 **for parallel** $\tilde{E}_{j \setminus i} \in \tilde{E}$ **do**
- 7 #Collect the simplex vertices opposite of the vertex i
 $(\mathbf{v}_{j_1}, \dots, \mathbf{v}_{j_d}) = \tilde{E}_{j \setminus i}$
#Optional: Only compute updates when necessary
- 8 **if** $\exists \mathbf{v}_{j_n} \in \tilde{E}_{j \setminus i} : (\phi_{k-2}(\mathbf{v}_{j_n}) - \phi_{k-1}(\mathbf{v}_{j_n})) \geq \varepsilon$ **then**
- 9 $\Phi = (\phi_k(\mathbf{v}_{j_1}), \dots, \phi_k(\mathbf{v}_{j_d}))^\top$
- 10 $\alpha_{n_f} = \text{Algorithm 1}(\mathbf{A}_{j \setminus i}, L_{j \setminus i}, \Phi)$
- 11 **With enabled gradient computations**
#Eq. (5)
 $\phi_{j \setminus i} = \langle \alpha_{n_f}, \Phi \rangle + \|\mathbf{A}_{j \setminus i} \alpha_{n_f}\|$
- 12 **With enabled gradient computations**
- 13 **for parallel** $\mathbf{v}_i \in V$ **do**
- 14 #see Eq. (2)
 $\phi_k(\mathbf{v}_i) = \min_{E_j \in \omega_i} \phi_{j \setminus i}$
- 15 # ε -convergence criterion
- 16 **if** $\forall \mathbf{v} \in V : (\phi_{k-1}(\mathbf{v}) - \phi_k(\mathbf{v})) < \varepsilon$ **then**
- 17 $\phi_K = \phi_k$
- 18 **break**

rate of convergence is (see [22, Theorem 4.4])

$$\phi_{j \setminus i}^k - \phi_{j \setminus i}^* \leq \frac{2L \|\alpha_0 - \alpha^*\|^2}{(k+1)^2}. \quad (9)$$

Finally, we compute the global eikonal solution by iteratively applying Algorithm 1 to each element $E_j \in E$ and for all vertices $\mathbf{v}_i \in E_j$, and then take minimum: see Algorithm 2.

C. Geodesic-BP

To utilize the computed eikonal solution in an inverse approach, we want to compute the gradients of the eikonal solution w.r.t. some quantity of interest, e.g., the initial conditions X_0 . More precisely, we are interested in the gradient $\nabla_{(\mathbf{x}_i, t_i)} \phi(\mathbf{x})$ and its efficient computation. This can be done analytically for (1), or algorithmically from Algorithm 2. The former (see e.g. [10]) relies on a *optimize-then-discretize* strategy, whereas the latter employs a *discretize-then-optimize* approach. Following the latter, we compute $\nabla_{(\mathbf{x}_i, t_i)} \phi(\mathbf{v})$ for all vertices $\mathbf{v} \in V$

and observe that

$$\nabla_{(\mathbf{x}_i, t_i)} (\phi(\mathbf{x})) = \langle \alpha, \nabla_{(\mathbf{x}_i, t_i)} (\phi_{j_1}, \dots, \phi_{j_{d+1}})^\top \rangle, \quad (10)$$

for $\mathbf{x} \in E_j$. Note that we neglect the derivative of α w.r.t. X_0 as a consequence of the first variation of the geodesic distance [24, Chapter 10]. The nodal values ϕ_{j_n} in succession are then derived through (3), until we terminate at the simplex containing the initial condition $\mathbf{x}_i \in E_k$. By defining \mathbf{x}_i in a continuous fashion (as was done in [10]), such that for all vertices \mathbf{v}_{j_n} of the element E_j containing \mathbf{x}_i it holds that

$$\phi_{j_n} = t_i + \|\mathbf{v}_{j_n} - \mathbf{x}_i\|_{\mathbf{D}_j}, \quad (11)$$

the derivative readily follows:

$$\nabla_{\mathbf{x}_i} \phi_{j_n} = -\frac{\mathbf{D}_j (\mathbf{v}_{j_n} - \mathbf{x}_i)}{\|\mathbf{v}_{j_n} - \mathbf{x}_i\|_{\mathbf{D}_j}}, \quad \nabla_{t_i} \phi_{j_n} = 1,$$

which can be effectively calculated using automatic differentiation of (11), or subsequent nodes (10) which are connected by a geodesic path to (\mathbf{x}_i, t_i) . In particular, we use the back-propagation algorithm implemented in many ML libraries. More technical aspects are provided in Section II-E.

D. ECG Computation

We describe the torso with a domain $\Omega_0 \subset \mathbb{R}^d$, whereas the heart is denoted by Ω , that is assumed to be well contained in Ω_0 . The full heart-torso domain is $\Omega_T = \Omega_0 \cup \Omega$, and the torso surface is $\Sigma = \partial\Omega_T$. The electrodes are placed on the torso at fixed locations $\tilde{X}_e = \{\mathbf{x}_{e_i}\}_{i=1}^N \subset \Sigma$. Each lead $\mathcal{V}_{l_e}(t)$ of the ECG is defined as the zero-sum of the electric potential at multiple locations. Thanks to the lead-field formula, we have the following representation:

$$\mathcal{V}_{l_e}(t) = \int_{\Omega} \langle \mathbf{G}_i(\mathbf{x}) \nabla Z_{l_e}(\mathbf{x}), \nabla V_m(\mathbf{x}, t) \rangle d\mathbf{x}, \quad (12)$$

where $\mathbf{G}_i : \Omega \rightarrow \mathcal{S}_d$ is the intracellular conductivity tensor, $V_m(\mathbf{x}, t) : \Omega \times [0, T] \rightarrow \mathbb{R}$ is the transmembrane potential, and $Z_{l_e} : \Omega_T \rightarrow \mathbb{R}$ is the l_e -th lead field, solution of the problem:

$$\begin{cases} -\nabla \cdot ((\mathbf{G}_i + \mathbf{G}_e) \nabla Z_{l_e}) = 0, & \text{in } \Omega_T, \\ -\nabla Z_{l_e} \cdot \mathbf{n} = \delta_{\mathbf{x}_{l_e}} - \frac{1}{|X_W|} \sum_{\mathbf{y} \in X_W} \delta_{\mathbf{y}}, & \text{on } \Sigma. \end{cases} \quad (13)$$

The set X_W contains the electrodes at the left/right arm and left foot, used to form the Wilson's Central Terminal (WCT). (13) can be solved once, since the problem is time-independent. The bulk conductivity of the torso, $\mathbf{G}_i + \mathbf{G}_e$, also depends on the extracellular conductivity tensor $\mathbf{G}_e : \Omega \rightarrow \mathcal{S}_d$. The derivation of this problem and exact formulation of (13) can be found in further detail in [5], [10], [25].

Next, we assume that the transmembrane potential is traveling wave with activation times dictated by $\phi(\mathbf{x})$:

$$V_m(\mathbf{x}, t) = U(t - \phi(\mathbf{x})) = U(\xi). \quad (14)$$

Since we are only interested in the depolarization sequence, we use the following template action potential, see [7], [10]:

$$U(\xi) = K_0 + \frac{K_1 - K_0}{2} \tanh(2\xi/\tau), \quad (15)$$

where K_0 and K_1 are respectively the minimum and maximum transmembrane potential.

Additionally, we precompute the discretized integral ECG operator (12) by means of a 2nd order simplicial Gaussian quadrature scheme. We used the library `scikit-fem` [26] to assemble this operator for our \mathcal{P}_1 mesh, automatically interpolating and integrating V_m at the Gaussian quadrature points. This way, we can compute the leads as $\mathcal{V}_{l_e}(t) = (BU(t - \phi))_{l_e}$ where $B : \mathbb{R}^{n_v \times N}$ is the mentioned discretized, precomputed lead field integration operator of (12), $n_v = |V|$ are the number of vertices and N is the number of leads.

For optimizing the inverse ECG problem, we want to fit \mathcal{V}_{l_e} of (12) against a measured $\hat{\mathcal{V}}_{l_e}$. We consider for that purpose the minimization problem

$$\min_{(\mathbf{x}_i, t_i) \in X_0} \frac{1}{N|T|} \sum_{l_e} \int_T \left(\mathcal{V}_{l_e}(\phi_{X_0}, t) - \hat{\mathcal{V}}_{l_e}(t) \right)^2 dt, \quad (16)$$

where ϕ_{X_0} is the eikonal solution for given initial conditions and N defines the number of measured/computed leads. Note that with B computed from (12), we can easily conclude for $\xi = t - \phi$ that $\nabla_{(\mathbf{x}_i, t_i)} \mathcal{V}_{l_e}(t) = -B^\top \frac{\partial U(\xi)}{\partial \xi} \nabla_{(\mathbf{x}_i, t_i)} \phi$, where $\nabla_{(\mathbf{x}_i, t_i)} \phi$ can be computed using Algorithm 2 as discussed in Section II-C. Note that backpropagation through $\nabla_{(\mathbf{x}_i, t_i)} \mathcal{V}_{l_e}$ will yield the equivalent result. The resulting gradients were then used for a gradient-based optimization utilizing ADAM [27].

E. Implementation Aspects

There are a few practical remarks when implementing Algorithms 1 and 2 that are important to consider: When computing the local update in Algorithm 1, the number of iterations n_f is the dominating factor for performance. Since the local problem is strictly convex with a precomputed Lipschitz constant L_E , the exact error bounds (see (9)) towards the global minimum in (5) are known and could be used to compute n_f on an element basis. Note that L_j only depends on $\mathbf{A}_{j \setminus i}$, which is strongly related to the mesh geometry and may increase as element quality decreases.

Furthermore, The upper bound presented in (8) is not tight and might lead to slow convergence of the local FISTA updates in Algorithm 1. In practice, we thus recomputed \underline{h} by minimizing α^* through Algorithm 1 with $\Phi = \mathbf{0}$ and using the estimate $\underline{h} = h(\alpha^*)/1.1$.

When considering Algorithm 2, note that the parallelization denoted in line 6 is more descriptive for CPU-parallel implementations. In ML frameworks, in contrast, the parallelization comes from the vectorization of single implemented operations, which might require re-ordering of some operations. Still, all operations inside the loop starting from line 6 can be parallelized, where only a special parallelization is required for Line 15 (scatter operation). This makes it well-suited for vectorized implementations on a GPU in frameworks such as PyTorch [28].

The optional performance optimization denoted in line 8 already significantly cuts down on the required memory for backpropagation, only computing updates when the solutions of simplex elements have changed. Checking for an ε change in activations before the update can also avoid unwanted gradient

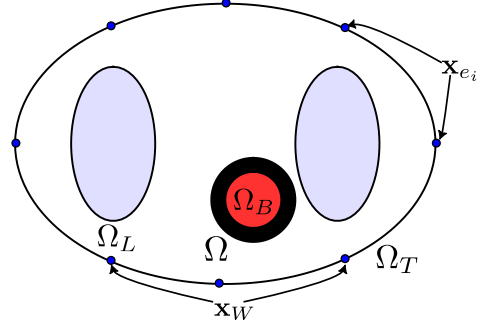


Fig. 3. 2D simulation setup. We consider a simple symmetrical left ventricular slice Ω to be located in a heterogeneous torso Ω_T , containing the blood pool Ω_B and lungs Ω_L , with different conductivities. 8 electrodes \mathbf{x}_{e_i} (blue circles) are circularly distributed around the torso, two of which are chosen as the WCT $\mathbf{x}_W \in X_W$.

eliminations in cases where the old and new values of ϕ take the same value.

Note that the local problem in Algorithm 1 could also be solved using the exact formulation derived for low-dimensional cases ($d \leq 4$) and fixed simplex dimensions as in previous works [12], [20]. In practice, however, the parallelization of this explicit solution is limited by the required condition to satisfy the constraint in (4). Such branching leads to unwanted performance losses in SIMD architectures when parallelizing over multiple elements.

Note that the number of activation sites is never modified by Algorithm 2. Some activation sites may however be overshadowed by the surrounding activation, thus effectively providing no contribution to the overall activation. When this happens, the gradient w.r.t. the overshadowed site is simply zero, thus the point is ignored by the algorithm. If an initial condition (\mathbf{x}, t_i) is overshadowed as described at the final iteration of the optimization, we often refer to it as inactive.

III. RESULTS

In this section, details of the dataset and results of applying our method for matching an ECG for both a synthetic 2-D and a real 3-D case are provided.

A. Simulation Study

1) Setup: To test and analyze our algorithm, we first start with a simple 2-D simulation study. We designed a simple setup consisting of an idealized torso with 8 electrodes distributed around the surface, with left and right arm electrodes composing Wilson's central terminal (WCT) (see Fig. 3). Seven lead fields were then computed each w.r.t. WCT.

The LV, meshed at a resolution of 0.9 mm (≈ 2700 DoFs), contained 4 initial condition tuples (\mathbf{x}_i, t_i) , located throughout half of the LV myocardium with timings between 2.5 ms and 10 ms. For computing the ground truth, we use another model with a 0.45 mm space resolution and perturbed conductivities in all sub-domains of the torso. As an initial guess, we consider 8 activation sites, all with 0 ms timing and evenly distributed

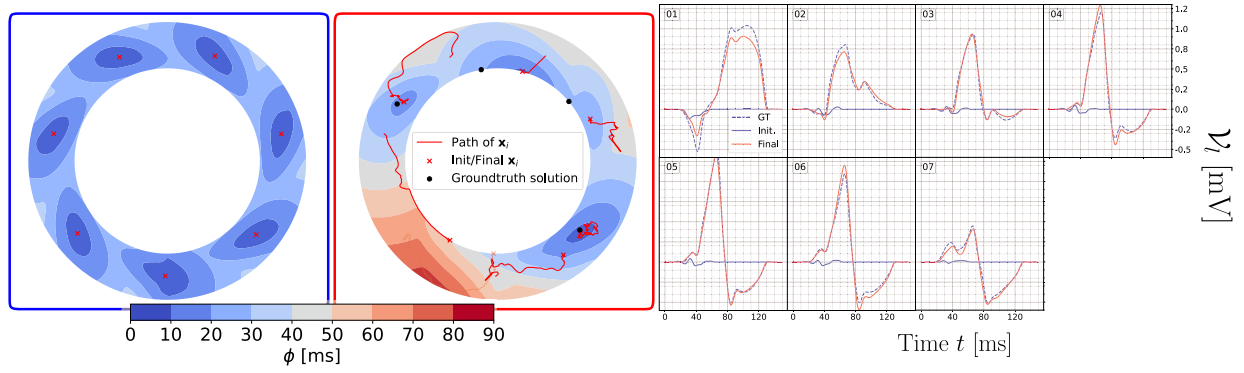


Fig. 4. Result of the ECG optimization on the idealized torso experiment using 7 leads. The initial model (left) uses a simple simultaneous activation, evenly spread throughout the myocardium. The optimization reduces the number of initial conditions and is able to match both ϕ and \mathcal{V}_i closely (middle and right). The transparent red lines in the middle plot show the optimization path of initial conditions (x_i, t_i), which are inactive in the final solution.

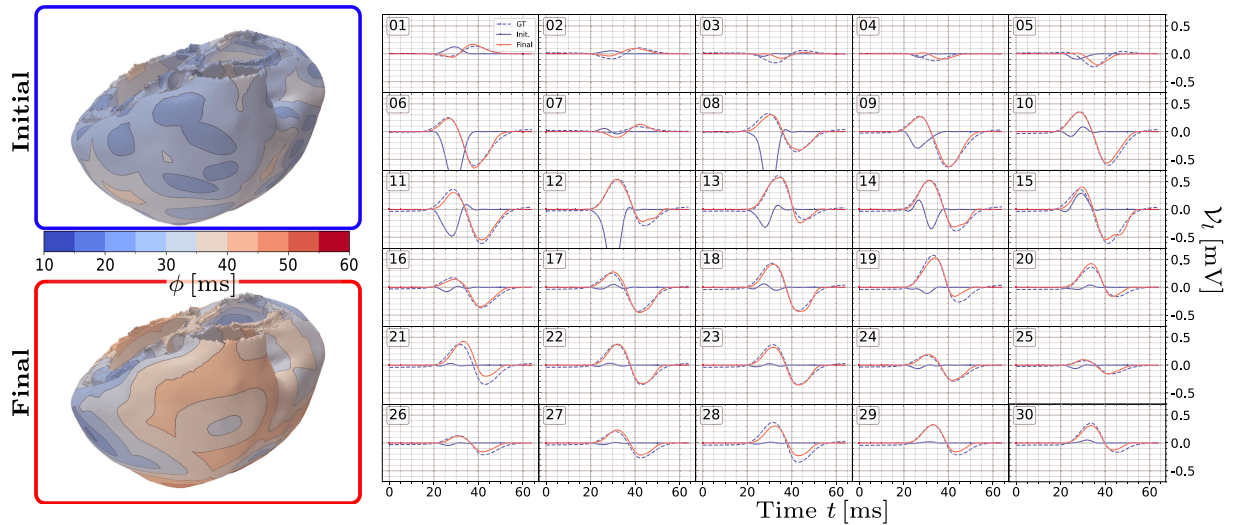


Fig. 5. Result of the ECG optimization on the rabbit heart model. We compare the initial and final model (blue/red) and show the computed activations $\phi(x)$ (left) and BPSM $\mathcal{V}_i(t)$ for each model (right). The computed ECGs (solid lines) are shown on top of the measured ECGs (dashed blue line). We show the first 30 of 31 available leads to save an extra row.

at the transmural center. The optimization was performed using ADAM [27] for 400 epochs using a learning rate of 0.5, taking ≈ 5 min on a AMD Ryzen 7 3800X 8-core processor.

2) Results: The results of the optimization against the initial and ground-truth solution and parameters are shown in Fig. 4. We can nearly perfectly fit the ECG and approximately recover the initial conditions. Overall, the total root mean squared error (RMSE) w.r.t. the eikonal solution ϕ is only 5.8 ms. We tested the algorithm also for a varying number of initial conditions and found that it is a hyperparameter of importance, but need not be chosen exactly: A cross-validation on the 2-D torso revealed that the worst match is achieved, by underestimating the number of initial conditions. In such a case, the achievable activations are not complex enough and the model is unable to faithfully replicate both the ECG and unobservable eikonal solution. Increasing the number of initial conditions beyond the ground truth's number is neither able to better match the

ECG by a significant margin, but will also not overfit and significantly increase the error on the eikonal solution. The most common cause of high errors on the eikonal solution is related to getting trapped in local minima when optimizing for the ECG, circumventable by batched optimization (not utilized in any of the shown experiments).

B. Body Potential Surface Map Reconstruction

1) Setup: The second anatomy is from a public rabbit ECGi model [29]. The anesthetized rabbit was equipped with a 32-lead ECG vest during normal sinus rhythm and the signals were recorded with a frequency of 2048 Hz. The ECG was recorded for 10 seconds, filtered, and averaged over all beats to produce a single beat. From this beat, we extracted the QRS duration to match the biventricular activation. The biventricular mesh consisted of 508975 DoFs and 2522501 tetrahedra, roughly

equivalent to a mean edge length of 0.82 mm in a human heart (assuming a size factor of 3 between rabbit and human hearts). Note that we also tested the optimization with a lower resolution of 143564 DoFs and 591975 tetrahedra, roughly equivalent to a mean edge length of 1.25 mm.

We use the electrophysiological parameters from [30], including cardiac fiber orientation, lead fields Z_l , and conduction velocity. In the conducted experiment, we chose $K_0 = -85$ mV, $K_1 = 30$ mV, $\tau_1 = 1$ ms in (15). The extracted QRS sequence was chosen during the time interval $t \in 100$ ms to 165 ms. We optimized for the initial conditions (x_i, t_i) of the model, i.e. X_0 in Algorithm 2. For this purpose, we randomly distributed 100 points on the biventricular surface ($x_i \in \partial\Omega$) and with all timings initialized to the approximate Q onset $t_i = 20$ ms.

2) Results: The experiment was run for 400 epochs using a learning rate of 0.5 on an Nvidia A100 GPU (40 GB), taking approximately 1.8 h (or 0.7 h on the reduced resolution). The computed results can be found on Zenodo [31]. The results for optimizing (16) with 400 epochs of ADAM using Algorithm 2 can be seen in Fig. 5. The initial and optimized activation are shown in the left upper and lower side of the image. It is shown together with the comparison of all ECGs (right). The initially randomly sampled points evenly cover the domain Ω , to allow for a fast excitation of the whole domain, though this leads to overexpressed deflections in many leads initially. The optimization significantly changes the activation of the heart and is able to achieve a promising match of the ECG in many leads. The loss in (16) converges after ≈ 250 epochs, with only minor variations in the parameters thereafter. Of the initial 100 points, only 91 are still active at the end of the optimization, meaning that the corresponding t_i is contained in the final solution and not earlier activated by closer initial conditions.

IV. DISCUSSION

We presented Geodesic-BP, an algorithm to efficiently solve the inverse ECG problem while being very efficient on a single GPU. We are able to fit the parameters of a patient-specific model only from the surface ECG and the segmented anatomy. The fitted model faithfully reproduced the recorded ECG in an animal experiment. In a synthetic case (Section III-A), Geodesic-BP is able to recover the ground-truth solution with high accuracy, even in the presence of perturbation in the forward model and in the lead field. The resolution of the considered 3-D heart model consists of $> 10^5$ DoFs, which is sufficiently accurate for a high-fidelity eikonal solution in a human heart, as it would correspond to roughly 0.8 mm edge length. At full resolution, all computations only take 16 s.

The considered cardiac parameter estimation problem is of high-relevance [1], [32], [33], [34], yet only a few works are able to efficiently solve this problem using solely non-invasive ECG measurements. The considered heart model and ECG originates from a real-world rabbit rather than a human model since it is (to the knowledge of the authors) the only publicly available full torso and heart model, captured with this high-level detail and including preprocessed electrophysiological measurements. Such animal models share many properties of human cardiac

electrophysiology [35]. Furthermore, the algorithm and equations involved translate 1:1 to human cardiac electrophysiology. Likewise, the methods involved are not restricted to biventricular models, and can be extended to atrial, or whole heart models under sinus rhythm. We also foresee the future possibility of functionalizing publicly available human heart and torso models, currently missing parameterization [36].

The runtime in the full rabbit heart model of 1.8 h for the computed 400 iterations is still expensive but it reduces to 0.7 h for a coarser resolution (see Section III-B2). Two main advantages of our novel method are the high number of parameters that can be simultaneously optimized, allowing for many initial sites in contrast to other works [2], [3], and having no spatial restrictions on the onset location [9]. The first advantage is a consequence of utilizing backpropagation to compute the gradient w.r.t. our parameters, resulting in the runtime not being dependent on the number of parameters and the convergence being dictated by properties of the underlying function to optimize [37]. Thus, compared to existing works based on derivative-free optimization with the same objective, our algorithm will scale much better in terms of spatially-varying parameters such as conduction velocities [38]. The latter advantage of arbitrary onset locations follows from the use of the volumetric description of the eikonal equation and lead fields, from which we can compute parameters and the local activation times not only on the heart surface, but also transmurally. In conjunction, both of these advantages allow us to model potentially more complex activations than previously were able by other works.

From an implementation point of view, Algorithm 2 can be readily applied to optimizing the conduction velocity tensor \mathbf{D} with little modification. Based on our previous research [16], [32], however, optimizing conduction velocities either requires strong prior information in the form of regularization, or multiple activation sequences to create meaningful results. The question how well such information can be reconstructed from the BSPM, and the space of possible solutions w.r.t. initial conditions both remain topics for future research.

One major limitation of Geodesic-BP is that it cannot handle re-entry phenomena such as tachycardia and fibrillation, because it relies on the eikonal equation with focal boundary conditions. However, the eikonal model can be reformulated to allow re-entry, see [39] and it could be used in an optimization problem where the initial condition is to be identified from the ECG. Finally, we also foresee many practical improvements for reducing the computational time of the local solver as outlined Section II-E.

V. CONCLUSION

This paper presents Geodesic-BP, an efficient method to compute piecewise linear geodesics, and demonstrated its performance for solving the inverse ECG problem in a limited parameter setting. By creating the link between backpropagation through a FEM-solver with geodesic backtracking, we were able to show that our previous FEM-based inverse eikonal models [15], [16] are linked to our recent work [10] working with geodesic backtracking. Such inverse procedures are current

research topics of high-relevance and efficient inverse eikonal models such as the present one may help in advancing patient-specific cardiac electrophysiology to practical real-world scenarios in the near future.

REFERENCES

- [1] J. Corral-Acero et al., "The 'digital twin' to enable the vision of precision cardiology," *Eur. Heart J.*, vol. 41, no. 48, pp. 4556–4564, Dec. 2020. [Online]. Available: <https://doi.org/10.1093/eurheartj/ehaa159>
- [2] S. Pezzuto et al., "Reconstruction of three-dimensional biventricular activation based on the 12-lead electrocardiogram via patient-specific modelling," *EP Europace*, vol. 23, no. 4, pp. 640–647, 2021.
- [3] K. Gillette et al., "A framework for the generation of digital twins of cardiac electrophysiology from clinical 12-leads ECGs," *Med. Image Anal.*, vol. 71, Jul. 2021, Art. no. 102080. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1361841521001262>
- [4] J. P. Keener, "An eikonal-curvature equation for action potential propagation in myocardium," *J. Math. Biol.*, vol. 29, no. 7, pp. 629–651, 1991.
- [5] P. C. Franzone, L. F. Pavarino, and S. Scacchi, *Mathematical Cardiac Electrophysiology*, vol. 13. Berlin, Germany: Springer, 2014.
- [6] A. Neic et al., "Efficient computation of electrograms and ECGs in human whole heart simulations using a reaction-eikonal model," *J. Comput. Phys.*, vol. 346, pp. 191–211, Oct. 2017. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0021999117304655>
- [7] S. Pezzuto et al., "Evaluation of a rapid anisotropic model for ECG simulation," *Front. Physiol.*, vol. 8, 2017, Art. no. 265.
- [8] J. Camps et al., "Inference of ventricular activation properties from non-invasive electrocardiography," *Med. Image Anal.*, vol. 73, Oct. 2021, Art. no. 102143. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1361841521001894>
- [9] S. Pezzuto, P. Perdikaris, and F. S. Costabal, "Learning cardiac activation maps from 12-lead ECG with multi-fidelity Bayesian optimization on manifolds," *IFAC-PapersOnLine*, vol. 55, pp. 175–180, 2022.
- [10] T. Grandits et al., "GEASI: Geodesic-based earliest activation sites identification in cardiac models," *Int. J. Numer. Methods Biomed. Eng.*, vol. 37, no. 8, 2021, Art. no. e3505, eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/cnm.3505>. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cnm.3505>
- [11] L. Li et al., "Deep computational model for the inference of ventricular activation properties," in *Proc. Int. Workshop Stat. Atlases Comput. Models Heart*, 2022, pp. 369–380.
- [12] W. Jeong and R. Whitaker, "A fast iterative method for eikonal equations," *SIAM J. Sci. Comput.*, vol. 30, no. 5, pp. 2512–2534, Jan. 2008. [Online]. Available: <https://epubs.siam.org/doi/abs/10.1137/060670298>
- [13] T. Grandits, "A fast iterative method python package," *J. Open Source Softw.*, vol. 6, no. 66, Oct. 2021, Art. no. 3641. [Online]. Available: <https://joss.theoj.org/papers/10.21105/joss.03641>
- [14] J. Fehrenbach and L. Weynans, "Source and metric estimation in the eikonal equation using optimization on a manifold," *Inverse Problems Imag.*, vol. 17, no. 2, pp. 419–440, 2023.
- [15] T. Grandits et al., "An inverse eikonal method for identifying ventricular activation sequences from epicardial activation maps," *J. Comput. Phys.*, vol. 419, Oct. 2020, Art. no. 109700. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0021999120304745>
- [16] T. Grandits et al., "PIEMAP: Personalized inverse eikonal model from cardiac electro-anatomical maps," in *Proc. Int. Workshop Stat. Atlases Comput. Models Heart*, 2021, pp. 76–86.
- [17] P. C. Franzone and L. Guerri, "Spreading of excitation in 3-D models of the anisotropic cardiac tissue. I. validation of the eikonal model," *Math. Biosciences*, vol. 113, no. 2, pp. 145–209, Feb. 1993. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/002555649390001Q>
- [18] A. Quarteroni and A. Valli, *Numerical Approximation of Partial Differential Equations*, vol. 23. Berlin, Germany: Springer Science & Business Media, 2008.
- [19] Z. Fu et al., "A fast iterative method for solving the eikonal equation on triangulated surfaces," *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2468–2488, Jan. 2011. [Online]. Available: <https://epubs.siam.org/doi/abs/10.1137/100788951>
- [20] Z. Fu, R. Kirby, and R. Whitaker, "A fast iterative method for solving the eikonal equation on tetrahedral domains," *SIAM J. Sci. Comput.*, vol. 35, no. 5, pp. C473–C494, Jan. 2013. [Online]. Available: <https://epubs.siam.org/doi/abs/10.1137/120881956>
- [21] F. Bornemann and C. Rasch, "Finite-element discretization of static Hamilton-Jacobi equations based on a local variational principle," *Comput. Vis. Sci.*, vol. 9, no. 2, pp. 57–69, 2006. [Online]. Available: <https://doi.org/10.1007/s00791-006-0016-y>
- [22] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, Jan. 2009. [Online]. Available: <http://epubs.siam.org/doi/abs/10.1137/080716542>
- [23] J. Duchi et al., "Efficient projections onto the l_1 -ball for learning in high dimensions," in *Proc. 25th Int. Conf. Mach. Learn.*, 2008, pp. 272–279. [Online]. Available: <https://doi.org/10.1145/1390156.1390191>
- [24] B. O'Neill, *Semi-Riemannian Geometry With Applications to Relativity*. Cambridge, MA, USA: Academic Press, 1983.
- [25] J. P. Keener and J. Sneyd, *Mathematical Physiology*, vol. 1. Berlin, Germany: Springer, 1998.
- [26] T. Gustafsson and G. D. McBain, "scikit-fem: A python package for finite element assembly," *J. Open Source Softw.*, vol. 5, no. 52, pp. 2369–2373, 2020.
- [27] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, San Diego, CA, USA, 2015.
- [28] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html
- [29] R. Moss et al., "A volumetric model of rabbit heart and torso including ECG data and ventricular activation sequence," Mar. 2022, doi: [10.5281/zenodo.6340066](https://doi.org/10.5281/zenodo.6340066). [Online]. Available: <https://zenodo.org/record/6340066>
- [30] R. Moss et al., "A computational model of rabbit geometry and ECG: Optimizing ventricular activation sequence and APD distribution," *Plos One*, vol. 17, no. 6, Jun. 2022, Art. no. e0270559. [Online]. Available: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0270559>
- [31] T. Grandits et al., "Geodesic-BP dataset," Nov. 2023, doi: [10.5281/zenodo.8027712](https://doi.org/10.5281/zenodo.8027712). [Online]. Available: <https://zenodo.org/record/8027712>
- [32] C. Ruiz et al., "Physics-informed neural networks to learn cardiac fiber orientation from multiple electroanatomical maps," *Eng. Comput.*, vol. 38, pp. 3957–3973, Jul. 2022. doi: [10.1007/s00366-022-01709-3](https://doi.org/10.1007/s00366-022-01709-3).
- [33] S. Coveney et al., "Probabilistic interpolation of uncertain local activation times on human atrial manifolds," *IEEE Trans. Biomed. Eng.*, vol. 67, no. 1, pp. 99–109, Jan. 2020.
- [34] C. H. Roney et al., "A technique for measuring anisotropy in atrial conduction to estimate conduction velocity and atrial fibre direction," *Comput. Biol. Med.*, vol. 104, pp. 278–290, Jan. 2019. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0010482518303184>
- [35] K. E. Odening, I. Baczko, and M. Brunner, "Animals in cardiovascular research: Important role of rabbit models in cardiac electrophysiology," *Eur. Heart J.*, vol. 41, no. 21, Jun. 2020, Art. no. 2036. [Online]. Available: <https://doi.org/10.1093/eurheartj/ehaa251>
- [36] K. Aras et al., "Experimental data and geometric analysis repository—EDGAR," *J. Electrocardiol.*, vol. 48, no. 6, pp. 975–981, Nov. 2015. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0022073615002307>
- [37] A. Beck, *First-Order Methods in Optimization*. Philadelphia, PA, USA: SIAM, 2017.
- [38] A. R. Conn, K. Scheinberg, and L. N. Vicente, *Introduction to Derivative-Free Optimization*. Philadelphia, PA, USA: SIAM, 2009.
- [39] E. Pernod et al., "A multi-front eikonal model of cardiac electrophysiology for interactive simulation of radio-frequency ablation," *Comput. Graph.*, vol. 35, no. 2, pp. 431–440, 2011.