

RESEARCH ARTICLE

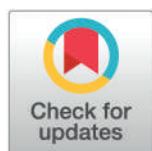
Learning cardiac activation and repolarization times with operator learning

Giovanni Ziarelli¹*, Edoardo Centofanti², Nicola Parolini³, Simone Scacchi¹, Marco Verani³, Luca F. Pavarino²

1 Dipartimento di Matematica, Università di Milano, Milano, Italy, **2** Dipartimento di Matematica, Università di Pavia, Pavia, Italy, **3** MOX Laboratory - Dipartimento di Matematica, Politecnico di Milano, Milano, Italy

* These authors contributed equally to this work.

* giovanni.ziarelli@unimi.it



OPEN ACCESS

Citation: Ziarelli G, Centofanti E, Parolini N, Scacchi S, Verani M, Pavarino LF (2026) Learning cardiac activation and repolarization times with operator learning. *PLoS Comput Biol* 22(1): e1013920. <https://doi.org/10.1371/journal.pcbi.1013920>

Editor: Haibo Ni, Nanjing University, CHINA

Received: May 15, 2025

Accepted: January 14, 2026

Published: January 27, 2026

Copyright: © 2026 Ziarelli et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data availability statement: Data employed for the numerical tests presented in this work are available at <https://zenodo.org/records/16913206>, whilst the software is available at <https://github.com/edoardo100/timeactml.git>.

Abstract

Solving partial or ordinary differential equation models in cardiac electrophysiology is a computationally demanding task, particularly when high-resolution meshes are required to capture the complex dynamics of the heart. Moreover, in clinical applications, it is essential to employ computational tools that provide only relevant information, ensuring clarity and ease of interpretation. In this work, we exploit two recently proposed operator learning approaches, namely Fourier Neural Operators (FNO) and Kernel Operator Learning (KOL), to learn the operator mapping the applied stimulus in the physical domain into the activation and repolarization time distributions. These data-driven methods are evaluated on synthetic 2D and 3D domains, as well as on a physiologically realistic left ventricle geometry. Notably, while the learned map between the applied current and activation time has its modeling counterpart in the Eikonal model, no equivalent partial differential equation (PDE) model is known for the map between the applied current and repolarization time. Our results demonstrate that both FNO and KOL approaches are robust to hyperparameter choices and computationally efficient compared to traditional PDE-based Monodomain models. These findings highlight the potential use of these surrogate operators to accelerate cardiac simulations and facilitate their clinical integration.

Author summary

Cardiac electrophysiology simulations are crucial for understanding how electrical signals propagate through the heart. However, solving the underlying mathematical models—typically partial differential equations—requires significant computational resources, especially when high-resolution anatomical detail is involved. This limits their real-time use in clinical settings. In this study, we explore two operator learning techniques, Fourier Neural Operators (FNO) and Kernel

Funding: EC, LP and SS acknowledge the CINECA award under the ISCRA initiative, for the availability of high-performance computing resources and support (projects DON2CARD and DDO2CARD). EC, NP, LP, SS, MV and GZ are members of INdAM-GNCS. EC and LFP have been supported by MUR (PRIN 202232A8AN_002 and PRIN P2022B38NR_001) funded by European Union - Next Generation EU. SS and GZ have been supported by MUR (PRIN 202232A8AN_003 and PRIN P2022B38NR_002) funded by European Union - Next Generation EU. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Operator Learning (KOL), as efficient alternatives to traditional solvers. These machine learning models learn how to predict activation and repolarization times—the key quantities that describe the electrical behavior of the heart—directly from the input electrical stimulus. We test these models on idealized 2D and 3D domains, as well as on a realistic human left ventricle geometry. Remarkably, our learned models achieve high accuracy while being orders of magnitude faster than traditional solvers. While activation times can be related to well-established mathematical models, repolarization times lack such a direct modeling framework—making our data-driven approach especially valuable. Our findings suggest that operator learning methods can make high-fidelity cardiac simulations more accessible for clinical applications by drastically reducing computation time while maintaining accuracy.

Introduction

Computational modeling of cardiac electrophysiology has become a fundamental tool for understanding heart function, diagnosing cardiac conditions, and developing therapeutic interventions [34,53]. Recent years have witnessed significant advances in both mathematical modeling, numerical techniques and computational capabilities, enabling increasingly sophisticated simulations of cardiac electrical activity [6,13,27,39]. Despite these advances, the computational complexity of high-fidelity cardiac models remains a substantial challenge, particularly for large-scale simulations, real-time applications, and clinical decision support systems. The Bidomain model [10,12,23,49,53] serves as the gold standard for describing the propagation of extra- and intracellular potentials in cardiac tissue, though its computational complexity presents significant challenges for large-scale simulations. In several works, in particular involving electromechanical coupling or interactions with fluid dynamics models of the human heart, researchers often turn to the more computationally efficient Monodomain model [12,23] as an alternative. The latter emerges as a simplified version of the Bidomain model where the intra- and extracellular conductivities are proportional. This simplification results in a model that maintains reasonable accuracy while significantly reducing the computational demand. The Monodomain model is constituted by a system of partial differential equations that describe the spatiotemporal evolution of the transmembrane potential and associated gating or recovery variables, which either represent the probability of ionic species flowing through the membrane or serve as recovery variables designed to reproduce observed phenomenological potentials (see, e.g., the two-variable models derived from the FitzHugh-Nagumo model [19]). A more computationally efficient approach is provided by Eikonal models [12,23], which focus on the evolution of cellular excitation wavefronts rather than the complete spatial and temporal reconstruction of ionic action potentials: these models are extremely computationally cheap, though they provide less details regarding the propagation of the electrophysiological (EP) signals.

Despite the extensive range of mathematical models and numerical schemes available for addressing EP problems, their computational burden remains a significant concern. Furthermore, the primary interest in solving these models often lies in extracting key informative quantities that can significantly assist clinicians, such as activation and repolarization times within the cardiac domain. The activation and repolarization times serve as critical markers in cardiac electrophysiology, providing essential information about the heart's electrical function: the activation time refers to the time when cardiac cells begin their depolarization process, whilst repolarization time denotes the time when cells return to their resting state. These markers are fundamental for understanding cardiac conduction patterns, identifying arrhythmogenic substrates [14], and evaluating the effects of drugs or interventions [3,56]. Traditional approaches for computing these times typically require solving the full Monodomain or Bidomain models, which can be computationally intensive depending on the geometry, and extracting the times from the resulting action potential waveforms. Moreover, while activation times can be evaluated more efficiently through the Eikonal model, repolarization times lack of a classical Eikonal-like counterpart.

The emergence of scientific machine learning offers new opportunities to address these computational challenges [9, 20,25,32,39,42,51,52]. In particular, one of its main branches, operator learning, aims to approximate unknown operators that map between potentially infinite-dimensional functional spaces. Given pairs of input/output functional data (u, f) , where $u \in \mathcal{U}$ and $f \in \mathcal{V}$ are functions defined on domains Ω and Ω' respectively, the goal is to learn an approximation of an operator $\mathcal{G} : \mathcal{U} \rightarrow \mathcal{V}$ using machine learning architectures. Among the various recently-proposed operator learning architectures (see, e.g. [21,30,54]), Fourier Neural Operators (FNOs) [28] have emerged as a powerful approach based on the Neural Operator paradigm [26], parameterizing the integral kernel layers within the architecture in the Fourier space and allowing efficient learning of mappings between function spaces with resolution independence. FNOs have shown comparable performances for equispaced domains with respect to the vanilla Deep Operator Networks [31]. Another promising approach is Kernel Operator Learning (KOL) [5], which builds on standard kernel regression arguments to approximate the mapping between function spaces. Compared to other neural operator methodologies, the key advantage of KOL approach lies in its non-iterative formulation; the operator is obtained by solving a (potentially large) symmetric and positive definite linear system, thereby eliminating the need for iterative training procedures typically required in neural network-based frameworks.

Using operator learning techniques to predict activation and repolarization times based on inputs such as tissue conductivity, fiber orientation, and stimulus location can help reduce computational bottlenecks in traditional cardiac modeling. These approaches will also improve efficiency, making computational tools more accessible for research and clinical practice. In this work, we take a step in this direction by learning the mapping between an initially applied current stimulus and the activation/repolarization times at each physical point in the considered 2D or 3D domains, as schematically represented in Fig 1. Specifically, we technically adapt FNO and KOL for the specific EP problem and compare the performances of FNO and KOL in terms of training time, test time, memory usage and accuracy in testing. In this way, we assess the potentialities of both strategies for retrieving fast and accurate simulations.

In summary, this study introduces novel applications of two promising operator learning schemes specifically tailored to cardiac electrophysiology problems and trained on *in-silico* data, with the potential to be extended to more realistic scenarios. A clinically relevant application of such surrogate models is in the context of inverse problems, such as reconstructing the site of origin of focal arrhythmias from observed activation maps. These inverse problems are typically solved by iteratively evaluating the forward model under different stimulation hypotheses, a process that is computationally expensive when using PDE-based models like the Monodomain or Bidomain formulations. Our operator learning approach aims to drastically accelerate this process by providing rapid approximations of activation/repolarization maps, thus enabling real-time or near real-time inference in future studies.

Furthermore, the proposed surrogate FNO and KOL models are also suited for training with real clinical data, even if it is out of the scope of this work. Indeed, activation times can be directly derived from extracellular potentials, which are routinely acquired in clinical practice: one possibility consists in computing the timing of the steepest negative slope (minimum variation of action potential) during the QRS complex [38,47,48]. Hence, endocardial or epicardial activation maps

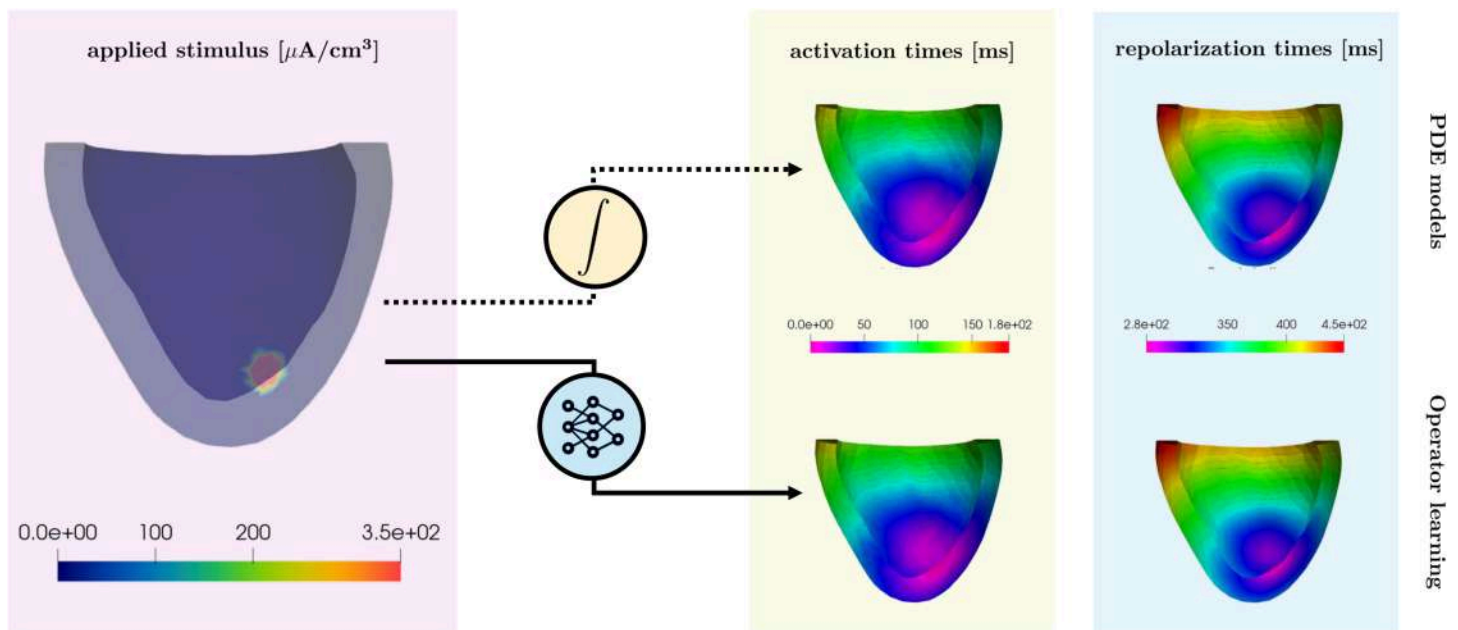


Fig 1. Schematic representation of the EP problem to address. In particular, we aim at reconstructing the activation/repolarization times of the cardiac tissue given the initial stimulus applied for 1 ms.

<https://doi.org/10.1371/journal.pcbi.1013920.g001>

are routinely obtained in electroanatomical mapping procedures and can serve as meaningful input data for inverse modeling. Accordingly, the proposed setting provides significant value for clinical applications, particularly in identifying ectopic arrhythmia origins and guiding digitally assisted ablation procedures.

Materials and methods

PDE models

Mathematical models of electrophysiology play a crucial role in understanding and simulating the electrical activity of cardiac tissue. These models describe the evolution of the transmembrane potential and ionic currents, capturing the fundamental mechanisms of excitation and propagation.

There exists a wide variety of EP models for representing membrane dynamics and calcium handling at the cellular level, differing in complexity and physiological detail depending on the specific application (see, e.g., [2,12,35,44]). On the other hand, spatial dynamics at the tissue scale are typically described by a few well-established macroscopic formulations, most notably the Monodomain and Bidomain [8,12] models. For more detailed modeling at the microscopic level, the Extracellular-Membrane-Intracellular (EMI) model [55] provides a cell-by-cell scale description.

The Monodomain model [57] is derived from the more challenging Bidomain model [1,6,8] when the intra- and the extracellular conductivity tensors, $\mathbf{D}_i$ and $\mathbf{D}_e$ respectively (measured in mS/cm), satisfy the following relationship [12]:

$$\mathbf{D}_e = \lambda \mathbf{D}_i \quad (1)$$

where $\lambda \in \mathbb{R}$ is a constant. The model reads as follows:

$$\left\{ \begin{array}{ll} \chi C_m \frac{\partial v}{\partial t} - \frac{\lambda}{1 + \lambda} \operatorname{div}(\mathbf{D}_i \nabla v) + I_{\text{ion}}(v, \mathbf{w}) = I_{\text{app}} & \text{in } \Omega \times (0, T), \\ \frac{\partial \mathbf{w}}{\partial t} - \mathbf{R}(v, \mathbf{w}) = 0 & \text{in } \Omega \times (0, T), \\ \frac{\partial \mathbf{c}}{\partial t} - \mathbf{C}(v, \mathbf{w}, \mathbf{c}) = 0 & \text{in } \Omega \times (0, T), \\ \mathbf{n}^\top \mathbf{D}_i \nabla v = 0 & \text{on } \partial\Omega \times (0, T), \\ v(\mathbf{x}, 0) = v_0(\mathbf{x}), \quad \mathbf{w}(\mathbf{x}, 0) = \mathbf{w}_0(\mathbf{x}) & \text{in } \Omega. \end{array} \right. \quad (2)$$

where v is the transmembrane potential (in mV), which represents the difference between the intra- and extracellular potentials, C_m and χ are the membrane capacitance per unit area (in $\mu\text{F}/\text{cm}^2$) and the membrane surface area per unit volume (in cm^{-1}) respectively, λ is the constant in (1), I_{ion} is a ionic current density representing the flow of ionic species through the cellular membrane and I_{app} is an applied stimulus in time, both measured in $\mu\text{A}/\text{cm}^3$. This latter term depends on v , as well as on the dimensionless gating or recovery variables of the ionic model, $\mathbf{w}$. These variables either describe the probability of ionic species flowing through the membrane or serve as recovery variables designed to reproduce an observed phenomenological potential, as seen in two-variable models derived from the FitzHugh-Nagumo [22] model. They are coupled to the reaction diffusion PDE through a system of differential equations describing their evolution in time as well as the dynamics of the ionic concentrations $\mathbf{c}$ (in mM), which are ruled by often nonlinear functions $\mathbf{R}(v, \mathbf{w})$ and $\mathbf{C}(v, \mathbf{w}, \mathbf{c})$. If there is no injection of current in the extracellular space, (2) can be considered as a good approximation of the Bidomain model.

An alternative approach for modelling the evolution of the cellular excitation wavefront through PDE systems is based on Eikonal models [12,13,23,24]. In this case, the unknown is the activation time of cardiac cells, *i.e.* the instant when the transmembrane potential first crosses a predefined threshold during an action potential, marking the onset of electrical excitation. Eikonal models seem particularly attractive from a computational perspective compared to Monodomain, since they involve a single steady-state PDE that does not require coupling with ODE systems. More importantly, unlike the transmembrane potential, the activation time lacks internal or boundary layers, eliminating the need for special mesh restrictions. However, they require to solve a nonlinear PDE, which may be a bottleneck for real-time clinical applications or large-scale simulations.

Finally, we remark that activation and repolarization times can be both derived by post-processing the solution of the Monodomain model. However, while activation times can be obtained by solving Eikonal equations, no Eikonal-based formulation has been proposed in the literature to directly extract repolarization times.

In the next section, we introduce the operator learning tools that will enable us to construct surrogate models for both activation and repolarization times.

Operator learning

Operator Learning (OL) aims to approximate an unknown operator

$$\mathcal{G} : \mathcal{A} \rightarrow \mathcal{U},$$

which maps functions from an input functional space (denoted by $\mathcal{A}$) into the corresponding output functions belonging to another functional space (denoted by $\mathcal{U}$). Given data pairs (a, u) , where $a \in \mathcal{A}$ and $u \in \mathcal{U}$ are functions defined on bounded domains $\Omega \subset \mathbb{R}^d$, $\Omega' \subset \mathbb{R}^{d'}$ and sampled over $\mathbb{R}^d$ and $\mathbb{R}^{d'}$ respectively, the goal is to learn an approximation of $\mathcal{G}$ by means of machine learning surrogate model. The problem can be formalized as follows [26,28]:

Problem 1. Let us consider $\{a_i, u_i\}_{i=1}^N$ samples in $\mathcal{A} \times \mathcal{U}$, such that

$$\mathcal{G}(a_i) = u_i, \quad \text{with } i = 1, 2, \dots, N. \quad (3)$$

We define the observation operators $\phi : \mathcal{A} \rightarrow \mathbb{R}^{d_a}$ and $\varphi : \mathcal{U} \rightarrow \mathbb{R}^{d_u}$ acting on the input and the output functions, respectively. These operators represent discretizations of the underlying function, which are commonly used in practical applications. The aim of operator learning is approximating the operator $\mathcal{G}$ through the observation of input/output pairs $\{\phi(a_i), \varphi(u_i)\}$.

In this work we consider a natural choice for the observation operators ϕ and φ , namely the pointwise evaluation at specific collocation points in Ω and Ω' , respectively. In particular, since in our setting $\Omega \equiv \Omega'$, we will also take the same collocation points for both ϕ and φ .

Furthermore, we work within a supervised learning framework, where, among all the possible operators ranging between $\mathcal{A}$ and $\mathcal{U}$, we aim at finding the one which minimizes the error on the observed (training) couples assuming tailored paradigms for the approximated operator, e.g. machine learning architectures. In the following, we briefly expand on the two methods employed for the reconstruction of activation and repolarization times, namely Fourier Neural Operators and Kernel Operator Learning.

Fourier neural operators

Fourier Neural Operators (FNOs) [28] belong to the broader class of operator learning frameworks known as Neural Operators (NOs) [26]. The NO paradigm defines a sequence of functions $a_0 \mapsto a_1 \mapsto \dots \mapsto a_T$, with each a_t having values in $\mathbb{R}^{d_{a_t}}$ for $t = 0, 1, \dots, T - 1$. As shown in Fig 2, the input $a \in \mathcal{A}$ is first lifted pointwise by a transformation P , resulting in $a_0(x) = P(a(x))$. The function is then propagated through a series of updates ($a_t \mapsto a_{t+1}$), and the final output is projected as $u(x) = Q(a_T(x))$ in order to belong to the same space of the function to reconstruct, where $Q : \mathbb{R}^{d_{a_T}} \rightarrow \mathbb{R}^{d_u}$ is another pointwise transformation. Each update step combines a non-local integral operator with a nonlinear activation, as follows:

$$a_{t+1}(x) = \sigma_{t+1} \left(W_t a_t(x) + \int_{\Omega_t} \kappa^{(t)}(x, y) a_t(y) d\nu_t(y) + b_t(x) \right) \quad \forall x \in \mathbb{R}^{d_{a_{t+1}}}. \quad (4)$$

FNOs arise by restricting the kernel to be translation-invariant, i.e., $\kappa^{(t)}(x, y) = \kappa^{(t)}(x - y)$, and leveraging the convolution theorem in the Fourier domain. Denoting by $\mathcal{F}$ the Fourier transform and by $\mathcal{F}^{-1}$ its inverse, the integral operator is thus computed as:

$$(K_t(a_t))(x) = \mathcal{F}^{-1} (R_{\theta_t} \cdot \mathcal{F}(a_t))(x), \quad \forall x \in \Omega_t \subset \mathbb{R}^{d_{a_t}}, \quad (5)$$

where R_{θ_t} denotes the truncated, learnable Fourier coefficients of the kernel. When the domain is discretized through evenly spaced points, Fast Fourier Transform (FFT) can be employed for computing efficiently (5). For further technical details, we refer the interested reader to Text A in S1 File.

Kernel operator learning

Kernel Operator Learning (KOL) is a recently formalized operator learning technique [5], based on standard kernel regression arguments. Following the diagram in Fig 3, retrieving the approximated operator $\tilde{\mathcal{G}}$ is equivalent to determining a vector-valued function $f^\dagger : \mathbb{R}^{n \cdot d_a} \rightarrow \mathbb{R}^{n \cdot d_u}$ ranging between discrete observations of the input to observations of the output. Endowing $\mathcal{A}$, $\mathcal{U}$ and the space in which we look for $f^\dagger$ with Reproducing Kernel Hilbert Space (RKHS) structures, the

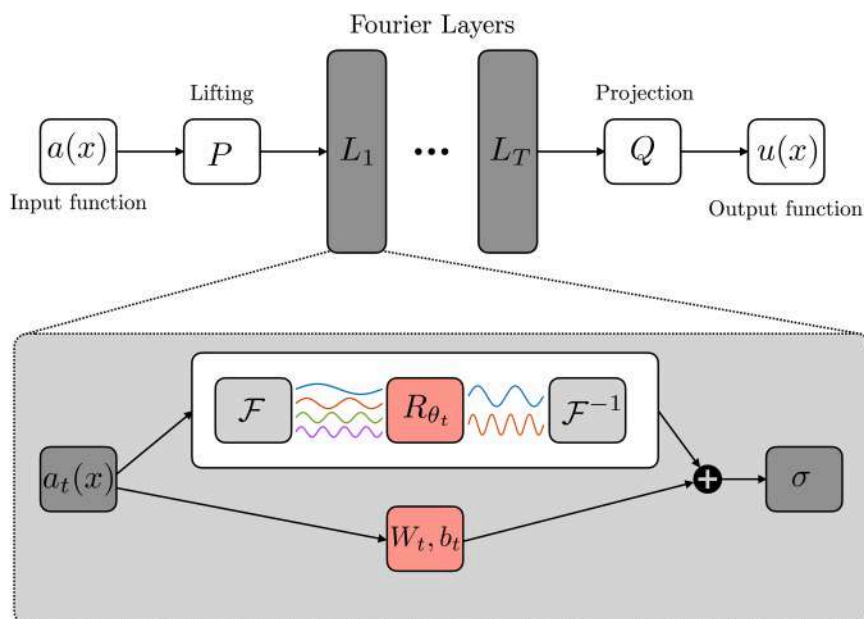


Fig 2. Schematic architecture of Fourier Neural Operators (FNO).

<https://doi.org/10.1371/journal.pcbi.1013920.g002>

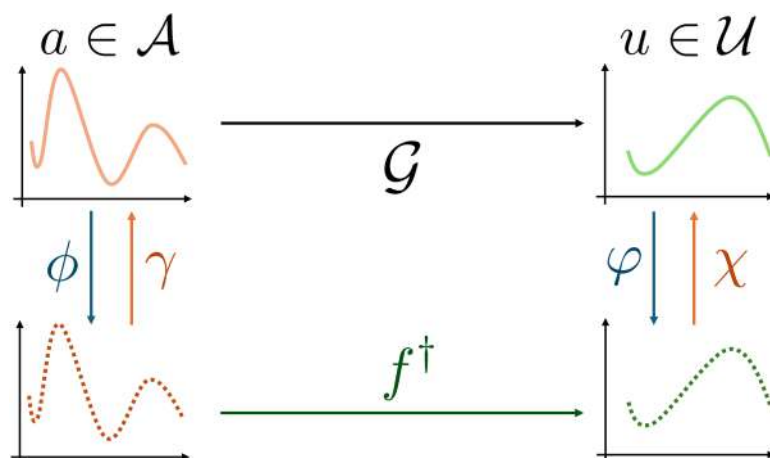


Fig 3. Kernel Operator Learning (KOL) diagram. Starting from the input function a , $\mathcal{A}$ collects observations of the function at different collocation points through ϕ . Then, the vector-valued $f^\dagger$ processes observations of the input into observations of the output $A = \phi(a)$. Finally, the reconstruction operator is applied to determine the output function $u = \chi(U)$.

<https://doi.org/10.1371/journal.pcbi.1013920.g003>

approximated operator can be written explicitly in closed form as

$$\tilde{\mathcal{G}}(a)(\mathbf{x}) = K(\mathbf{x}, \mathcal{X})K(\mathcal{X}, \mathcal{X})^{-1} \left(\sum_{j=1}^N S(\phi(a), A_j) \alpha_j \right), \quad (6)$$

where K is the kernel function induced by the RKHS structure of $\mathcal{U}$, the vector

$$\mathcal{X} = [\mathbf{x}_1^T, \mathbf{x}_2^T \dots \mathbf{x}_n^T]^T \in \mathbb{R}^{n \cdot d}$$

contains the collocation points, and $S : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$ is a properly chosen vector-valued kernel generating the space of $f^\dagger$ functions. Moreover, $K(\cdot, \mathcal{X}) : \Omega \rightarrow \mathbb{R}^n$ is a row vector such that $K(\mathbf{x}, \mathcal{X})_i = K(\mathbf{x}, \mathbf{x}_i)$, and $K(\mathcal{X}, \mathcal{X})$ is an $n \times n$ matrix such that $K(\mathcal{X}, \mathcal{X})_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$. Since we deal with pointwise observation functions, $K(\mathbf{x}, \mathcal{X})$ computes the evaluation of the linear interpolant of points in $\mathcal{X}$ at $\mathbf{x}$, whilst $K(\mathbf{x}_i, \mathbf{x}_j) = \delta_{ij}^j$. Parameters $\{\alpha_j\}_{j=1}^N \in \mathbb{R}^n$ are the kernel regression parameters over the input/output training pairs. We refer to [60] for the mathematical derivation of the explicit representation of KOL operator and more details.

From a computational standpoint, after selecting the scalar kernel $S(\cdot, \cdot)$ for the discrete vector spaces, the problem reduces to solving n linear systems of size N in order to determine the different components of each $\{\alpha_j\}_j$. To achieve this, we employ the Cholesky factorization of the matrix and solve the resulting systems using standard substitution methods. Moreover, a regularization term is introduced in the regression formulation, with a penalty parameter set to 10^{-10} .

A key factor influencing the approximation and generalization properties of KOL methods is the selection of the scalar kernel S . The optimal kernel function for kernel regression remains a subject of ongoing debate and is application-dependent, as it directly influences the accuracy of the trained architecture. Consequently, performing sensitivity analyses is essential to identify the kernel that maximizes accuracy for the specific application. For example, some novel approaches involve learning kernels by simulating data-driven dynamical systems, enhancing the scalability of Kernel Regression [36]. In general, any symmetric positive semi-definite function that maps two elements of the vector space to a positive value can serve as a kernel. In this study, we consider few choices for S : Radial Basis Functions (RBF), Neural Tangent Kernel (NTK) and the kernel generated by the Euclidean distance between centroids of the initially activated spots (IQ). For the mathematical definitions of those functions we refer to Text B in S1 File. We note that the latter proposed kernel is specifically tailored for input functions that represent cardiac stimuli: it is often necessary to appropriately adjust the kernel functions based on the specific application at hand. This kernel is particularly effective in reconstructing activation and repolarization maps, whose iso-contours vary consistently with the distance from the activation site.

Results and discussion

In this section we report the results of the numerical tests performed with FNO and KOL on three different test cases for our problem: a 2D squared domain (Subsection 2D case), a 3D slab (Subsection 3D slab) and a realistic ventricle unstructured mesh (Subsection 3D unstructured ventricle).

Dataset generation and computational details

To construct the dataset for training operator learning models, we generate input excitations and the corresponding solutions of the Monodomain model (2) discretized by Q1 finite elements, on quadrilateral grids for the 2D case and hexahedral grids for the 3D case (see Fig 4). The Monodomain model is coupled with either the Rogers-McCulloch ionic model [43] for the 2D case or the Ten Tusscher ionic model [50] for the 3D cases, in order to describe the transmembrane potential between the intra- and extracellular domains. Activation and repolarization times are computed as postprocessing of the Monodomain solutions. The applied excitation is defined as a fixed intensity pulse applied for 1ms over a random location of the domain. Regarding the diffusion parameters, we have considered $C_m = 1\mu F/cm^2$, $\chi = 1cm^{-1}$, and $\mathbf{D}_i = \sigma_1 \mathbf{n}_1 \mathbf{n}_1^T + \sigma_2 \mathbf{n}_2 \mathbf{n}_2^T + \sigma_3 \mathbf{n}_3 \mathbf{n}_3^T$, where the triplet $\{\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3\}$ represents the fiber orientation. The conductivities employed in the various scenarios are chosen according to the test cases outlined in [12].

All training and test samples exhibited conduction velocities (CVs) that align with the established physiological range reported in experimental and computational studies. Specifically, in the 2D and the 3D slab case we measure 0.06 cm/ms

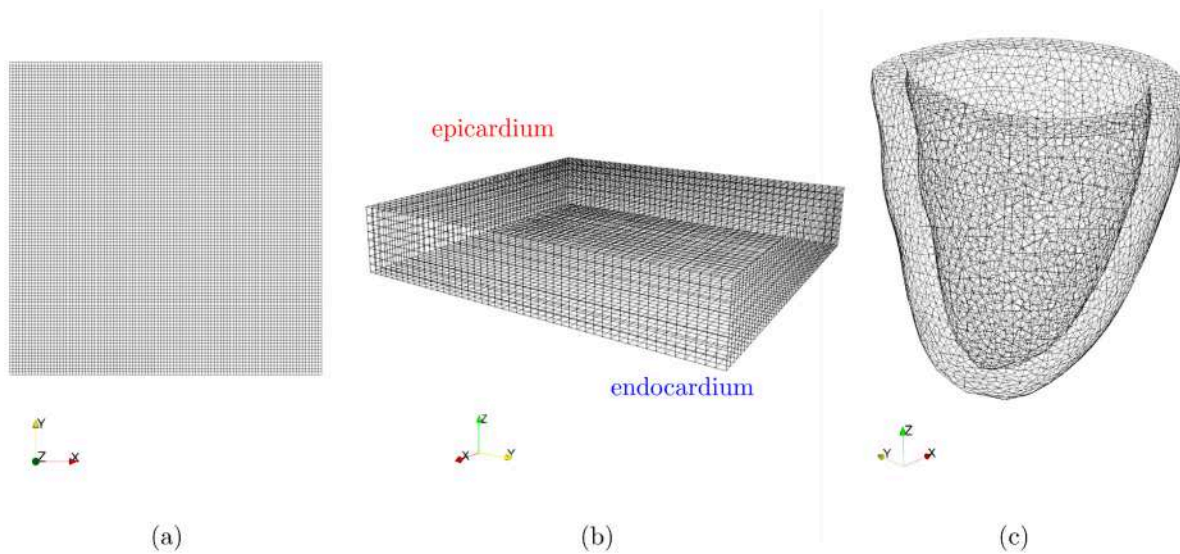


Fig 4. Grids adopted for the numerical simulations: (a) 2D grid (structured, 100×100 elements, physical area $1 \text{ cm} \times 1 \text{ cm} = 1 \text{ cm}^2$), (b) 3D slab (structured, $384 \times 384 \times 64$ elements, physical volume $3.84 \text{ cm} \times 3.84 \text{ cm} \times 0.64 \text{ cm} = 9.44 \text{ cm}^3$. The figure displays the coarser $48 \times 48 \times 8$ elements grid used to save the dataset employed to train the operator learning models.) and (c) 3D unstructured ventricle (about 35k nodes, physical volume 100 cm^3). For all the geometries, we have considered Q1 elements (regular squares/cubes for structured 2D and 3D cases, hexaedral elements for the 3D unstructured case).

<https://doi.org/10.1371/journal.pcbi.1013920.g004>

along-fiber CV and 0.03 cm/ms across, whilst in the 3D unstructured case we computed a CV of 0.05 cm/ms along-fiber CV and 0.02 cm/ms across using both a simulation on a cable model with the same conductivity parameters and the formula $\nabla T / |\nabla T|^2$, where T is the activation time expressed as a function of the position on the cardiac tissue [16]. The slightly slower propagation observed in the latter can be attributed to the coarser mesh resolution, which may lead to longer activation times as visible in Figs 18 and 19. Nonetheless, these values remain consistent with the physiological range reported for mammalian cardiac tissue in [11, 15].

For each training we consider an input dataset collecting electrical stimuli (named `iapps`) and the corresponding output maps of activation times (named `acti`) and repolarization times (named `repo`). The name of each dataset is followed by the number of samples it contains, where the repartition between training and test dataset is 80%/20%. The datasets' structure varies depending on the spatial dimension, as detailed below:

- **2D case:** we consider N different current stimuli in `iapps` in the form of $N \times 2$ matrices, collecting the (x, y) coordinates where each pulse is applied. For the basic case, activation and repolarization times are two matrices named of size $N \times n_{no}^2$, where n_{no} is the number of discretization points per dimension (101 in our case), and $\mathbf{n}_1 = \mathbf{n}_x$ and $\mathbf{n}_2 = \mathbf{n}_y$. In this scenario, we have also considered the case where the cardiac fibers are rotated by 45° counterclockwise around the z axis, with the rotation applied outward from the xy plane, and the corresponding output maps of activation and repolarization times are called `acti_rot` and `repo_rot`, respectively. The sizes of these datasets are the same of the basic case.
- **3D slab:** each stimulus is an $N \times n_x \times n_y \times n_z$ binary tensor, where ones indicate pulse application points. We consider $n_x = n_y = 49$ nodes in the x and y directions, $n_z = 9$ nodes in the z direction, extracted through a projection from a larger $385 \times 385 \times 65$ mesh. The activation and repolarization outputs have the same shape of the pulse. For the fiber direction we assume $\mathbf{n}_1 = \mathbf{n}_x$, $\mathbf{n}_2 = \mathbf{n}_y$ and $\mathbf{n}_3 = \mathbf{n}_z$.

- **3D unstructured ventricle:** the input stimuli are $N \times N_n \times 3$ matrices, where N_n is the number of nonzero nodes in the unstructured mesh of the ventricle. Similarly, activation and repolarization outputs are $N \times N_u \times 3$ tensors, with N_u being the total number of unstructured mesh nodes (about 35k DOFs, 30k elements extracted from a finer 1.98 mln DOFs mesh with 1.93 mln elements – in order to deal with activation and repolarization times lying in realistic human pathological ranges [7, 17]). In this case we consider the fiber orientation extracted by the physiological ventricle following the procedure described in [37]. This geometry is characterized by the following physical dimensions: maximum height of 7.23 cm, Left Ventricular End-Diastolic Diameter (LVEDD) of 5.85 cm, and Interventricular Septal Thickness in Diastole (IVSd) of 1.08 cm.

The generation of a single sample required approximately 8 minutes for the 2D case on an Intel Core i5 quad-core (2.7 GHz) processor, and 15–20 minutes for the 3D cases on a machine equipped with an NVIDIA Quadro RTX 5000 GPU. In Table 1 we report further details about the parameters employed for each case.

These structured and unstructured datasets provide diverse inputs for training operator learning architectures. In order to evaluate accuracy performance of the trained OL schemes we compute the generalization error as the discrete L^2 relative error, namely,

$$L_{err}^2 = \frac{1}{N_p} \sum_{n=1}^{N_p} \frac{\|\mathbf{y}_p^n - \mathbf{y}^n\|_2}{\|\mathbf{y}^n\|_2}, \quad (7)$$

where $\mathbf{y}^n$ is the vector containing the ground truth evaluations of activation/repolarization maps at the different points of the domain whilst $\mathbf{y}_p^n$ is the corresponding vector of predictions. For the 3D case representing a 3D ventricle we also compute the Pearson dissimilarity coefficient P is defined as $P = 1 - R$, with

$$R = \frac{\text{Cov}(Y, Y_p)}{\sigma_Y \sigma_{Y_p}}. \quad (8)$$

Here, $\text{Cov}(Y, Y_p)$ is the covariance between the ground truth of the tested samples and the predictions as flattened vectors (respectively Y and Y_p) and σ_{Y, Y_p} is the standard deviation relative to the test dataset or the predictions. This index

Table 1. Parameters of the high-fidelity solvers used for the 2D, 3D, and 3D unstructured cases. Here, h is the element diameter, dt the time step, T_{sti} the stimulation duration, χC_m the membrane capacitance per unit surface area, $\sigma_1, \sigma_2, \sigma_3$ the conductivities along the principal directions, and I_{ap} the amplitude of the applied stimulus current density and T_{sim} is the simulated physical time. The 2D case has been solved using MATLAB's direct solver, while the 3D cases have been solved using the Conjugate Gradient (CG) solver provided by the PETSc library and Hypr BoomerAMG as preconditioner [4, 59].

Case of study	2D	3D	3D unst.
DOFs	101×101	$385 \times 385 \times 65$ (proj. $49 \times 49 \times 9$)	1.98 mln (proj. 35k)
h [cm] (diameter)	0.01	0.01 (proj. 0.08)	0.05 (avg.) (proj. 0.18 (avg.))
dt [ms]	0.1	0.05	0.05
T_{sti} [ms]	1	1	1
T_{sim} [ms]	150	500	500
χC_m ($[\mu F/cm^2]$ or $[\mu F/cm^3]$)	1	1	1
σ_1 [mS/cm]	1.2E-03	1.2E-03	1.2E-03
σ_2 [mS/cm]	2.6E-04	2.6E-04	2.6E-04
σ_3 [mS/cm]	–	2.6E-04	2.6E-04
I_{ap} ($[\mu A/cm^2]$ or $[\mu A/cm^3]$)	200	350	250
Ionic Model	RM [43]	TT06 [50]	TT06 [50]
Solver	MATLAB backslash	CG+Hypr (PETSc)	CG+Hypr (PETSc)

<https://doi.org/10.1371/journal.pcbi.1013920.t001>

is commonly employed in machine learning to quantify the correlation between target outputs and their reconstructed counterparts.

Finally, we remark that in both 2D and 3D cases FNO architectures were trained on a workstation with an NVIDIA Quadro RTX 5000 GPU, while KOL was trained on an Intel i7 CPU. For a fair comparison from the end-user perspective, all inference timing analyses were conducted on a single device, a standard laptop equipped with an Apple M1 Pro chip (CPU only).

2D case

Figs 5 and 6 depict test samples reconstructed with FNO (top) and KOL (bottom) for activation and repolarization times. The FNO architecture employed comprises four Fourier layers with 16 Fourier modes along the x-axis and 4 modes along the y-axis, with a total number of approximately 532k trainable parameters. In Tables 2 and A in S1 File, results indicate that the use of a tailored learning rate reduction policy (`reduce- OnPlateau`), where the learning rate is reduced by a factor of 0.95 whenever the test loss is not decreasing, consistently outperforms training without such a policy. For instance, in the activation dataset `acti` with 3000 samples, the test error is reduced from 2.82×10^{-3} (no policy) to 2.66×10^{-3} when the policy is applied. In the FNO results the uncertainty bands arise from considering different trained architectures with various Kaiming normal initializations [18] of the trainable parameters.

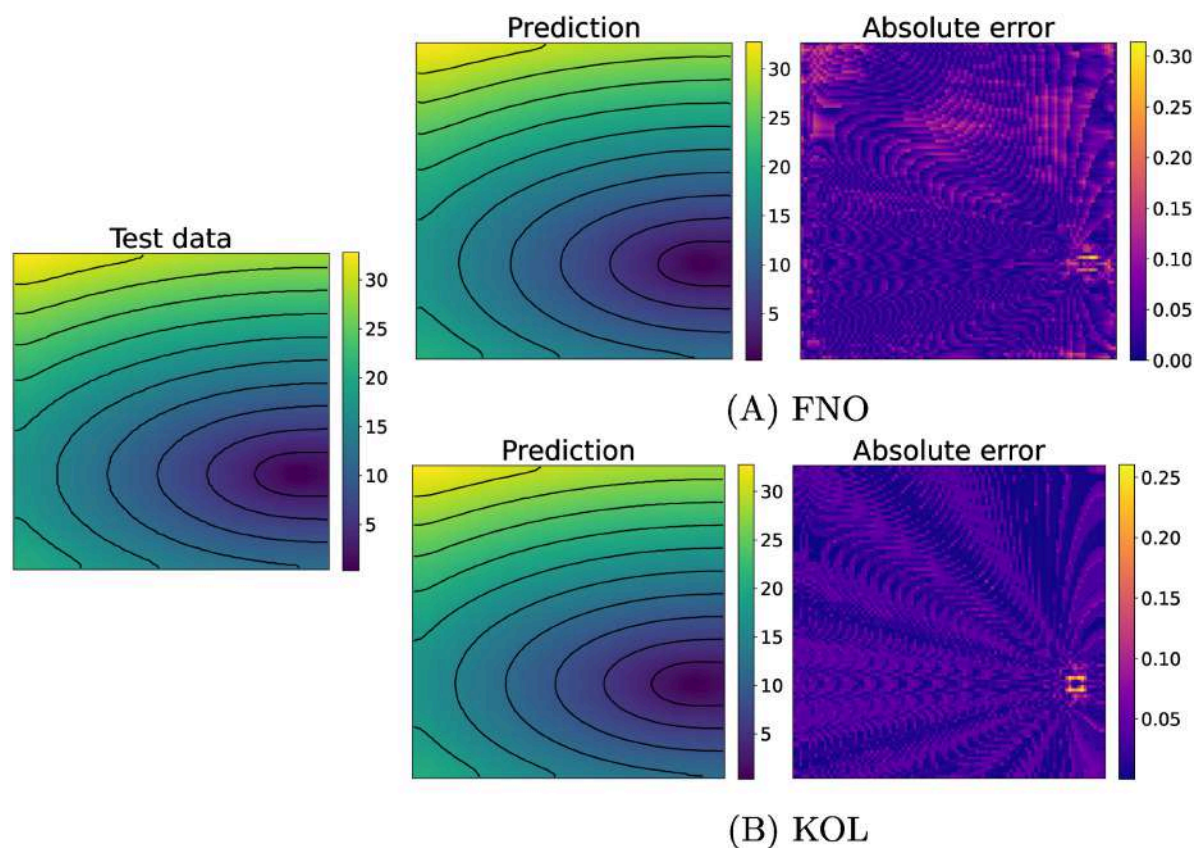


Fig 5. Comparison of FNO (A) and KOL (B) activation time predictions for the 2D case (acti with 2000 samples). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g005>

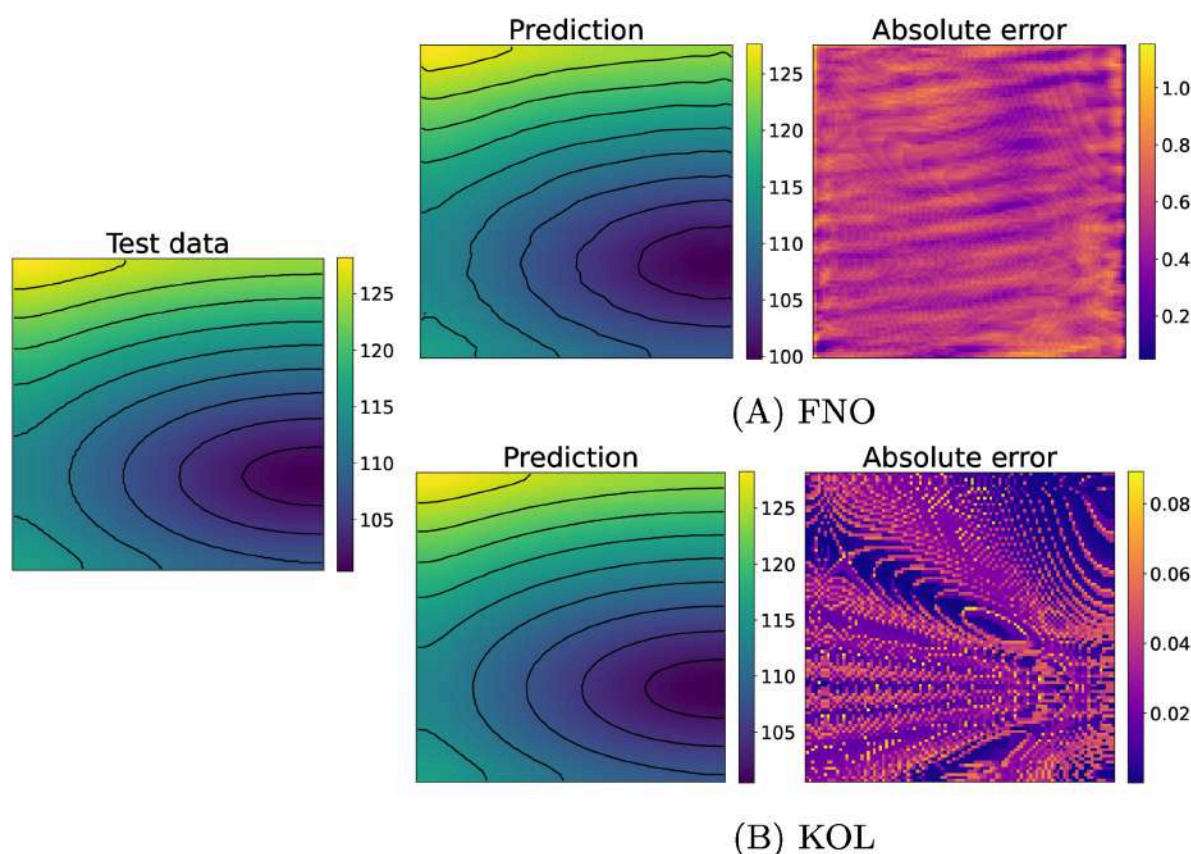


Fig 6. Comparison of FNO (A) and KOL (B) repolarization time predictions for the 2D case (repo with 2000 samples). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g006>

Table 2. Performance comparison of FNO (reduce-on-plateau learning rate policy) and KOL (iq4 kernel) methods on 2D datasets.

	Dataset (size)	Test Error	GPU Memory	Training Time	Test Time
FNO with lr policy <code>reduceOnPlateau</code>	acti (3000)	$2.66\text{E-}03 \pm 2.27\text{E-}04$	5.79GB	72 min	$5.7\text{E-}05$ sec
	acti_rot (2000)	$3.33\text{E-}03 \pm 3.13\text{E-}04$	5.62GB	48 min	$5.9\text{E-}05$ sec
	repo (3000)	$3.13\text{E-}03 \pm 2.33\text{E-}04$	5.79GB	72 min	$4.0\text{E-}05$ sec
	repo_rot (2000)	$3.53\text{E-}03 \pm 3.63\text{E-}04$	5.62GB	48 min	$5.4\text{E-}05$ sec
	Dataset (size)	Test Error	CPU Memory	Training Time	Test Time
KOL with iq4 kernel	acti (3000)	$9.52\text{E-}04$	1.15GB	10 min	$6.3\text{E-}04$ sec
	acti_rot (2000)	$9.34\text{E-}04$	0.91GB	8 min	$2.2\text{E-}04$ sec
	repo (3000)	$4.74\text{E-}04$	1.15GB	10 min	$7.1\text{E-}04$ sec
	repo_rot (2000)	$4.69\text{E-}04$	0.91GB	6 min	$2.9\text{E-}04$ sec

<https://doi.org/10.1371/journal.pcbi.1013920.t002>

The rotated fiber tests, where a 45° rotation of the myocytes is applied on the domain, generally show larger test errors compared to their non-rotated counterparts with the same number of training samples (cfr. Tables A and B in S1 File). This observation suggests that the FNO architecture might be sensitive to the structural alignment of input features. Moreover, we notice that increasing the dataset size improves the performance, as evident in both activation and repolarization datasets. For activation tests with 2000 and 3000 samples, the test error drops, highlighting the benefit of having more training data for learning the spatial features of the solution. In terms of computational efficiency, for FNO the GPU memory consumption remained stable at approximately 5.6GB for 2000-sample datasets and increased slightly to 5.79GB

for 3000-sample datasets. Training times scaled proportionally with dataset size, from 48 minutes for 2000 samples to 72 minutes for 3000 samples. We notice a general low absolute error, with some areas for the activation case propagating orthogonally with respect to level curves. This error distribution was observed also for the KOL case in Fig 5 (bottom, right), but a theoretical explanation of such a pattern remains unknown. A more uniform pattern is observed instead for the repolarization case.

We conducted a large number of numerical simulations using the same training and test datasets employed for FNO architectures to evaluate the performance of KOL (cfr. Tables 2, B and C in S1 File). The primary objective of this sensitivity analysis was to assess the impact of kernel selection, a critical factor that is highly problem-dependent and can significantly affect prediction quality.

From Tables B and C in S1 File, we observe that IQ kernel-based strategies yield generalization errors at least two orders of magnitude lower than those using NTK or RBF. This improved performance is attributed to the IQ kernel's efficiency in computing correlations of compact support indicator functions, which accurately represent activation regions. Notably, this advantage persists even when increasing the training set size for both activation and repolarization reconstructions. Additionally, KOL exhibits sensitivity to structural alignments of input features. Specifically, tests with rotated fibers achieve higher accuracy than unrotated counterparts. We observe that KOL significantly reduces training times with respect to FNO, decreasing from thousands of seconds to just few hundreds, as it requires solving a single symmetric, positive definite linear system rather than an iterative optimization process. However, as training size increases, the system's condition number rises which may impact on testing performances. Therefore, in this case, the use of tailored preconditioning strategies is crucial (cfr. [33,46]). Regarding computational efficiency, CPU memory consumption scales with training size but remains close to 1 GB. It is important to note that the accuracy and training time improvements offered by KOL must be considered in light of the non-negligible time required for kernel selection. Furthermore, we note that KOL, equipped with the chosen deterministic kernels, produces fully deterministic predictions: therefore, unlike FNO, it does not exhibit uncertainty bands due to hyperparameter initialization.

3D slab

Additional experiments on a 3D slab in $(0,1)^3$ have been conducted using the two operator learning approaches discussed. As will be shown, the performance of FNO and KOL remained robust despite the increased dimensionality of this test case. The architecture employed for FNO comprises four Fourier layers with 16 Fourier modes along the x-axis, 8 along the y-axis and 4 along the z-axis, resulting in approximately 8.4 million trainable parameters. Given the improved performance observed for the 2D case, all experiments used the `reduceOnPlateau` learning rate policy, where the learning rate decreases upon stagnation of the validation loss. Instead, we endow KOL with `iq4` kernel (cfr. Table E in S1 File) following the sensitivity analysis of the 2D case. In Table 3, the results indicate that increasing the dataset size significantly enhances prediction accuracy for both models. For the activation dataset, the FNO test error decreased from 4.15×10^{-2} to 3.27×10^{-2} when the sample size increased from 1000 to 2000. A similar trend is observed for repolarization, where the test error dropped from 8.46×10^{-3} to 6.91×10^{-3} . Conversely, KOL exhibits the opposite behavior: as the dataset size grows, the conditioning number of the SPD system increases, and, therefore test error increases (e.g. from 5.33×10^{-3} at 1000 dataset size to 6.19×10^{-3} at 2000 dataset size for repolarization). Nonetheless, test errors for KOL remain consistently lower than those of FNO in both activation and repolarization cases. However, as reported in Fig 7, most of the test data predictions for the `acti_2000` dataset are distributed under 1% of the relative L2 error, despite having around 3.5% of the data as outliers for the FNO case. Similar results are obtained for KOL in Fig 8. For the `repo_2000` dataset the reader can refer to Figs A and B in S1 File. In Fig 9 we have also reported the training and test loss decaying plot for FNO.

Compared to 2D tests, computational costs rise substantially in 3D. For FNO, GPU memory consumption increased from 13.66GB (1000 samples) to 14.09GB (2000 samples), whereas KOL required 1.67GB and 2.56GB, respectively.

Table 3. Performance comparison of FNO and KOL methods on 3D datasets.

	Dataset (size)	Test Error	GPU Memory	Training Time	Test Time
FNO with lr policy <code>reduceOnPlateau</code>	acti (1000)	$4.15\text{E-}02 \pm 2.02\text{E-}03$	13.66GB	94 min	4.6E-04 sec
	acti (2000)	$3.27\text{E-}02 \pm 5.86\text{E-}04$	14.09GB	188 min	2.4E-04 sec
	acti hetero (2000)	$2.75\text{E-}02 \pm 6.31\text{E-}04$	14.09GB	188 min	2.6E-04 sec
	repo (1000)	$8.46\text{E-}03 \pm 6.51\text{E-}04$	13.66GB	94 min	4.8E-04 sec
	repo (2000)	$6.91\text{E-}03 \pm 3.44\text{E-}04$	14.09GB	188 min	2.4E-04 sec
	repo hetero (2000)	$6.00\text{E-}03 \pm 8.76\text{E-}04$	14.09GB	188 min	3.0E-04 sec
	Dataset (size)	Test Error	CPU Memory	Training Time	Test Time
KOL with iq4 kernel	acti (1000)	1.67E-02	1.67GB	3 min	1.0E-03 sec
	acti (2000)	1.82E-02	2.56GB	7 min	1.4E-03 sec
	acti hetero (2000)	1.36E-02	2.56GB	12 min	1.4E-03 sec
	repo (1000)	5.33E-03	1.67GB	3 min	1.2E-03 sec
	repo (2000)	6.19E-03	2.56GB	7 min	2.0E-03 sec
	repo hetero (2000)	2.59E-03	2.56GB	12 min	2.0E-03 sec

<https://doi.org/10.1371/journal.pcbi.1013920.t003>

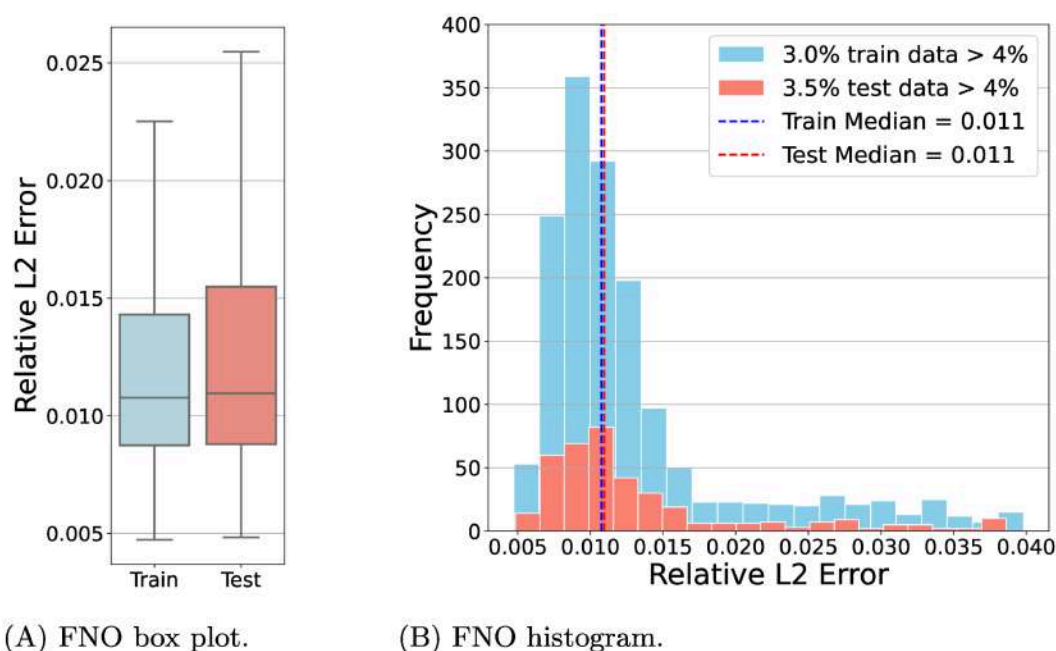
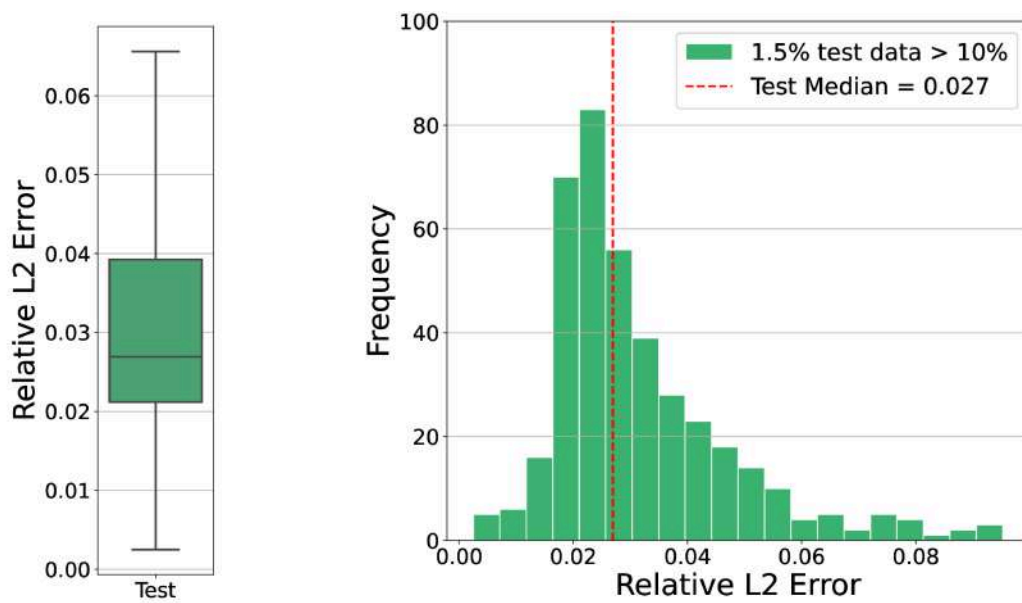


Fig 7. FNO box plot (A) and histogram (B) for the 3D dataset `acti_2000` relative to the best model trained. For the training set, 3% of the data have a relative L2 error greater than 4%, while for the test set, 3.5% of the data exceed this threshold.

<https://doi.org/10.1371/journal.pcbi.1013920.g007>

Training times scaled linearly for both models, with FNO requiring 94 minutes for 1000 samples and 188 minutes for 2000, while KOL completed training in 211 seconds and 427 seconds for the same dataset sizes. Figs 10, 11, 12 and 13 illustrate activation and repolarization time predictions for both models, along with the corresponding high fidelity solutions and absolute errors across three slices of the slab domain representing the endocardium, epicardium and the intermediate slice. The applied stimulus belongs to the epicardium surface for the reconstruction of activation times, whilst it belongs to an intermediate sheet between the endocardium and the middle of the slab for the repolarization case. In both cases, prediction error increases as far as the distance from applied stimuli increase. While FNO exhibits a nearly uniform absolute error distribution slightly higher than in the 2D case, KOL's errors tend to be concentrated near the activation region and propagate orthogonally to the level curves of the activation (or repolarization) times.



(A) KOL box plot.

(B) KOL histogram.

Fig 8. KOL box plot (A) and histogram (B) for 3D dataset *acti_2000* relative to the best model trained. Training results are not shown since we achieve machine precision. For the test set, 1.5% of the data have a relative L2 error greater than 10%.

<https://doi.org/10.1371/journal.pcbi.1013920.g008>

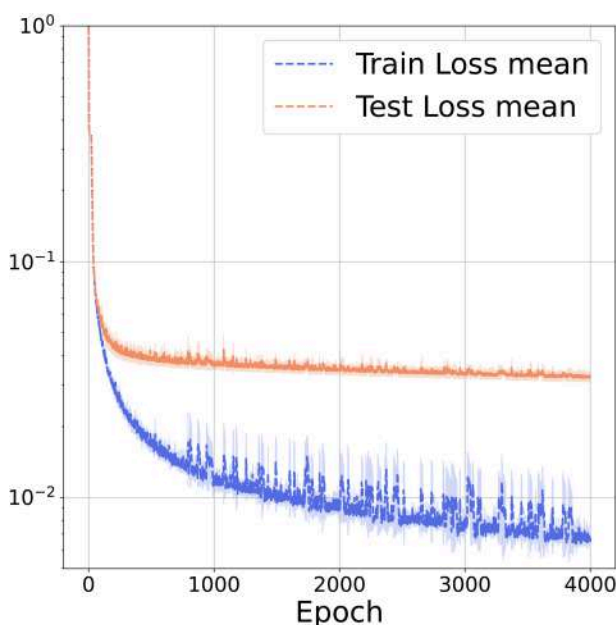


Fig 9. FNO loss plot for the 3D case (*acti_2000*) dataset). Mean train and test loss of three different randomly initialized models (dashed line). Standard deviation over the three models for each epoch is also reported (light shadow).

<https://doi.org/10.1371/journal.pcbi.1013920.g009>

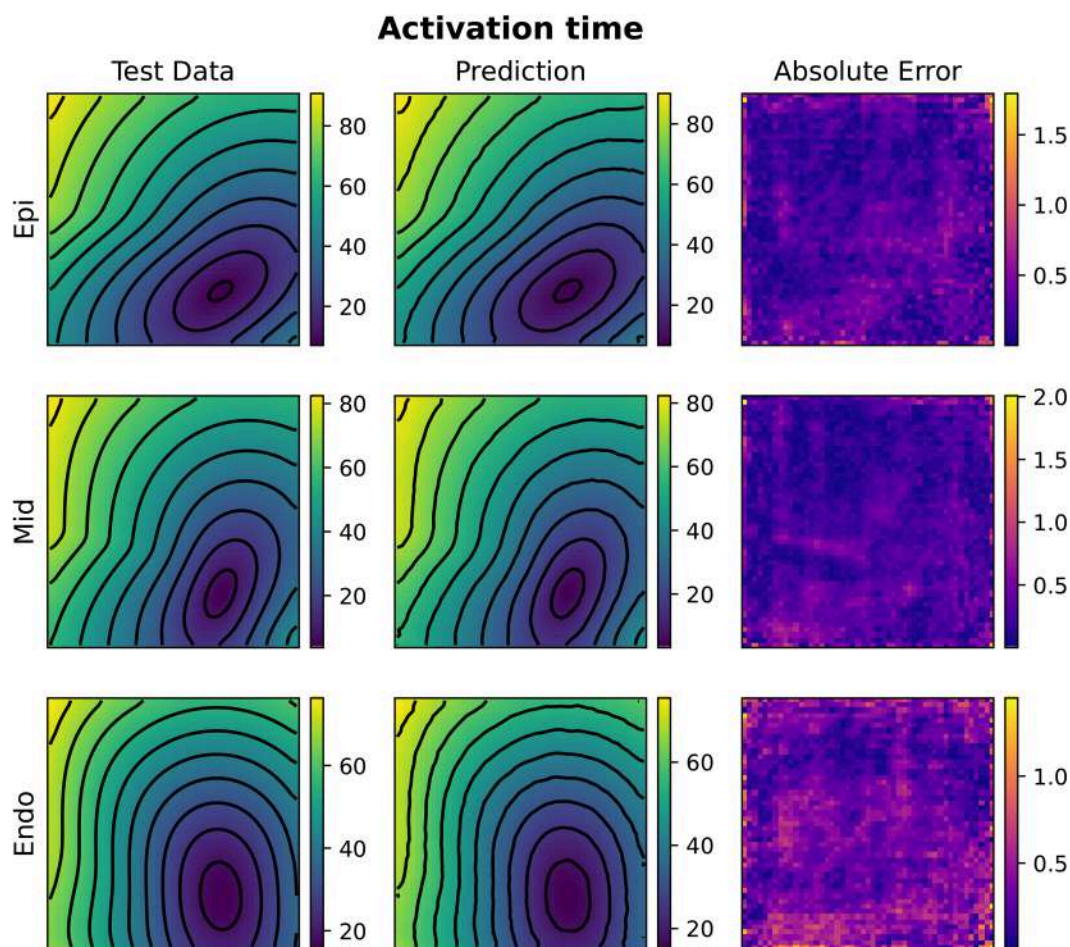


Fig 10. Example of FNO activation time prediction for the 3D case (acti with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g010>

Additionally, we investigate on spatial and cellular heterogeneity across the myocardium, such as differences between endocardial, epicardial, and midmyocardial cells, which represents a significant challenge in electrophysiological modeling. The following results indicate that the proposed learning framework is able to handle key forms of electrophysiological heterogeneity and can be extended to more complex tissue-specific scenarios.

We have conducted experiments in the 3D case incorporating spatial heterogeneity in action potential duration (APD) by varying the I_{Ks} conductance in the ten Tusscher ionic model [50]. The conductance parameter varies across nine myocardial layers to reflect physiological differences in cardiac tissue: the bottom three layers maintain the base conductance, the middle three layers exhibit a reduced conductance (70% of the base value), and the top three layers show an increased conductance (140% of the base value). This setup allows us to assess the robustness of the learning framework in the presence of realistic electrophysiological variability. As reported in Table 3, both the Fourier Neural Operator and Kernel Operator Learning models retained the expected test accuracy in this heterogeneous setting. In particular, the FNO achieved a relative test error of 2.69% for activation and 0.50% for repolarization, while KOL reported 1.36% and 0.26%, respectively. Figs 14–17 provide qualitative evidence of this performance, displaying accurate predictions across epicardial, midmyocardial, and endocardial layers.

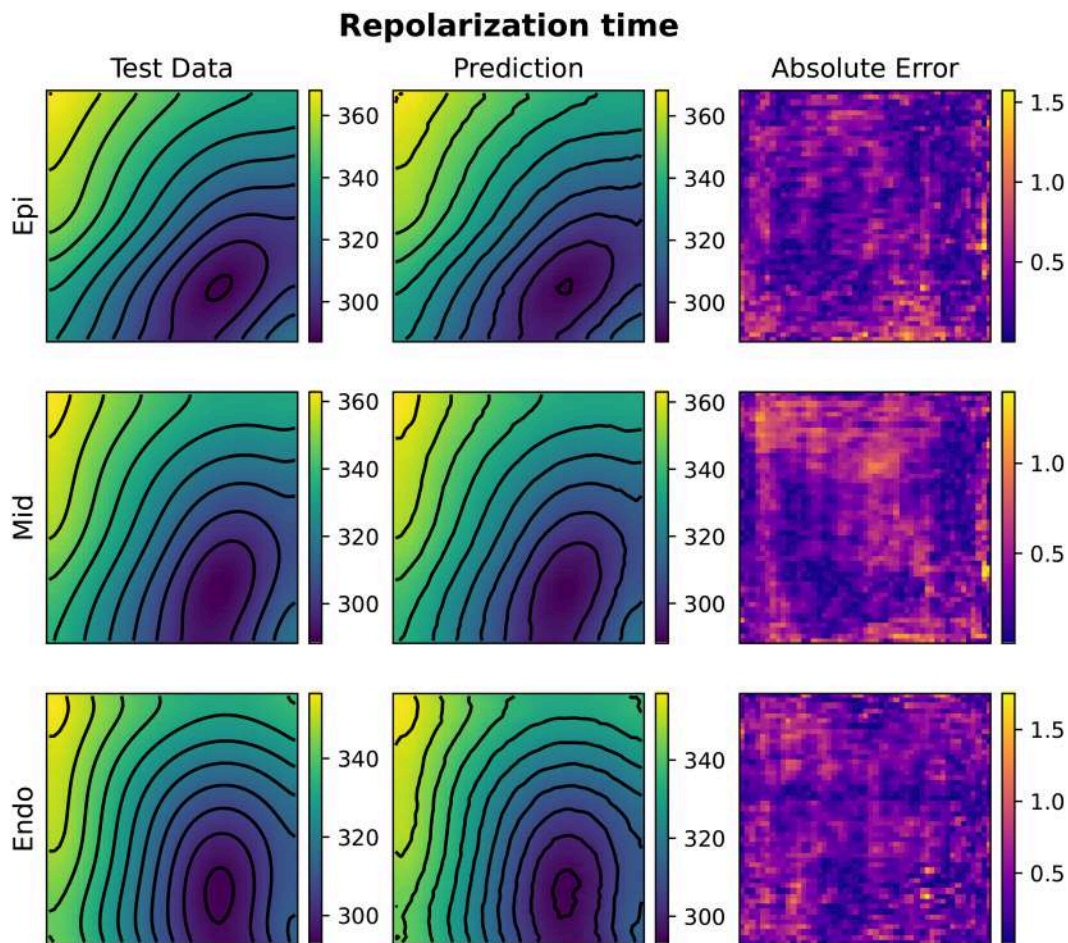


Fig 11. Example of FNO repolarization time prediction for the 3D case (repo with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g011>

3D unstructured ventricle

In the last test case considered, we solved the problem on an unstructured mesh representing a human ventricle, consisting of approximately 35k degrees of freedom. In this case, the FNO was adapted to handle irregular domains. In particular, we implemented a Non-Uniform Discrete Fourier Transform (NUDFT) extending to the 3D case the approach presented in [29]. On the other hand, KOL did not require any specific modifications to operate in this unstructured setting. Since the input is defined on an unstructured grid, we modified its representation accordingly. Specifically, each dataset consists of N samples, where each sample corresponds to a vector of size N_{nodes} with binary values: a value of 1 at a given node indicates the presence of an external stimulus, while 0 represents the absence of stimulation. Given that all stimuli have the same pulse intensity and duration, the input applied current was reformulated as a matrix of dimension $N \times N_{\text{stim}} \times 3$, where N_{stim} represents the maximum number of stimulated nodes across all samples, with padding applied where necessary. The numerical solutions for the activation and the repolarization times were structured as tensors of size $N \times N_{\text{nodes}} \times 3$, capturing the time evolution of the solution on the unstructured mesh.

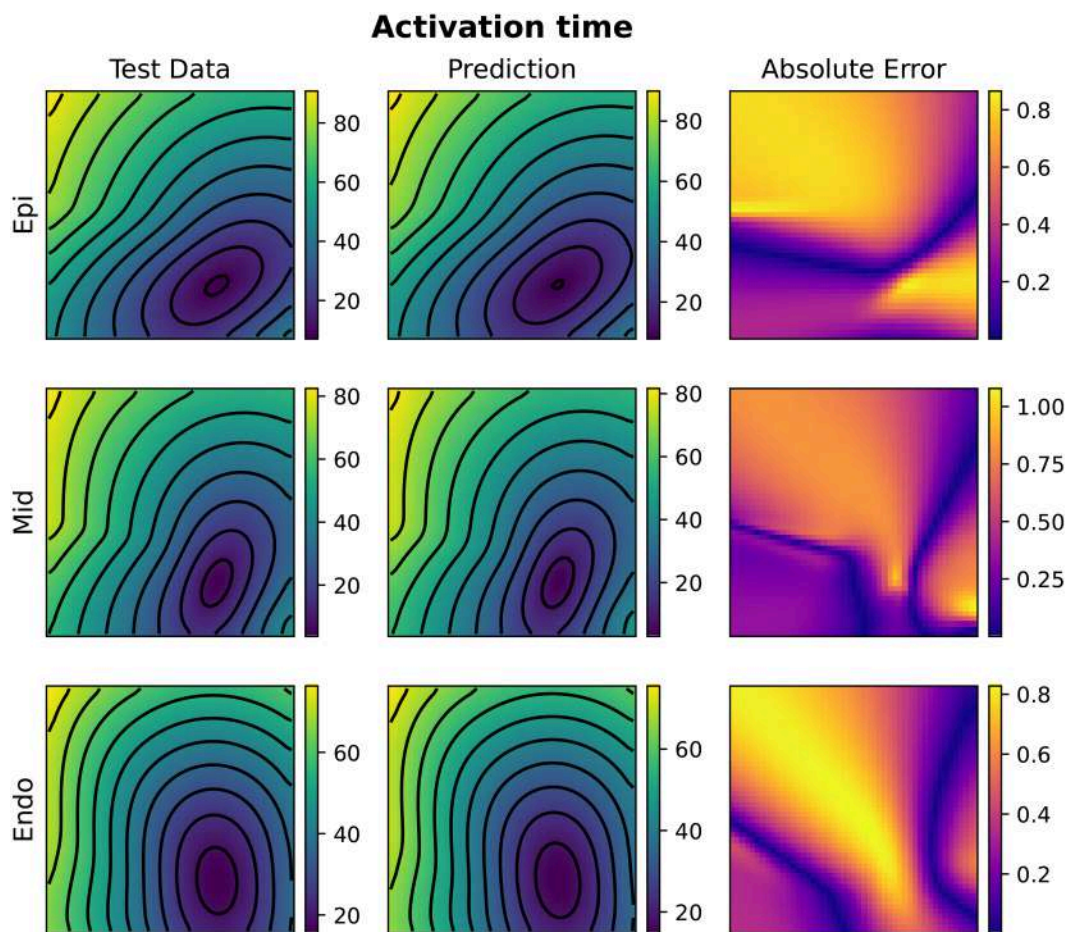


Fig 12. Example of KOL activation time prediction for the 3D case (acti with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g012>

The results in Table 4 provide insights into the performance of FNO and KOL on this complex domain. The FNO architecture employed consisted of Fourier layers with four Fourier modes in each spatial direction (x , y , and z), leading to trainable parameters ranging between 10.8M and 11.3M. We applied a `reduceOnPlateau` learning rate policy, as it consistently yielded better predictions in previous tests by dynamically adjusting the learning rate when the test loss plateaued. The architecture's depth and width also influenced performance. Notably, for $N = 2000$, increasing the model width from 2 to 16 substantially improved accuracy, reducing the test error from 1.38×10^{-1} to 5.86×10^{-2} (cf. Table D in S1 File). However, beyond a certain point, further increases in width did not yield significant improvements, with test errors fluctuating for widths above 32. Similarly, deeper architectures ($L > 1$) did not always lead to better performance, indicating that an optimal hyperparameters selection is crucial for balancing expressivity and generalization. In Table 4 we reported the most accurate performances of FNOs obtained considering $L = 1$, width=16 (acti) and $L = 3$, width=32 (repo).

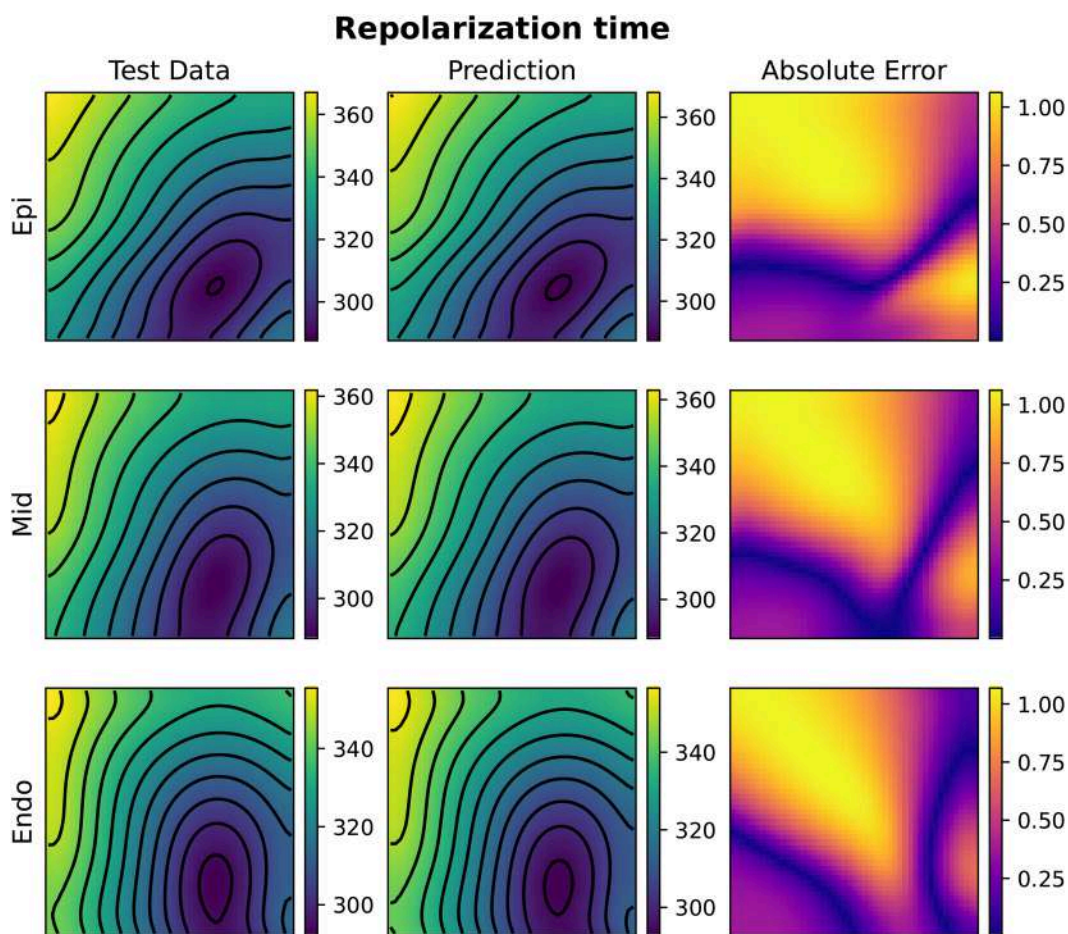


Fig 13. Example of KOL repolarization time prediction for the 3D case (repo with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g013>

Also in this case, KOL (endowed with iq4 kernel) significantly outperforms FNO in terms of test error, achieving a test error as low as 2.53×10^{-3} on the repolarization dataset for $N = 2000$, compared to FNO's 2.42×10^{-2} . The difference is even more pronounced for the activation dataset, where KOL attains a test error of 1.36×10^{-2} , while FNO's best result remains at 5.86×10^{-2} . Additionally, KOL exhibits lower Pearson dissimilarity values, indicating a better linear correlation between the ground truth test data and the corresponding predictions. Solutions for a specific test case, for both activation and repolarization, are shown in Figs 18 and 19. The absolute error plots highlight the lower error achieved by KOL, with a maximum absolute error of 4 ms, compared to larger regions reaching approximately 8.6 ms in the FNO case. However, both architectures successfully captured the qualitative distribution of activation and repolarization times.

Computationally, FNO has an advantage in terms of inference time. A single prediction with FNO on the larger dataset, consisting of 2000 samples with 1600 for training and 400 for testing, takes approximately 8.5 ms, whereas KOL requires 56 ms for the same task. Both methods outperform the solution of a single Monodomain model implemented using the

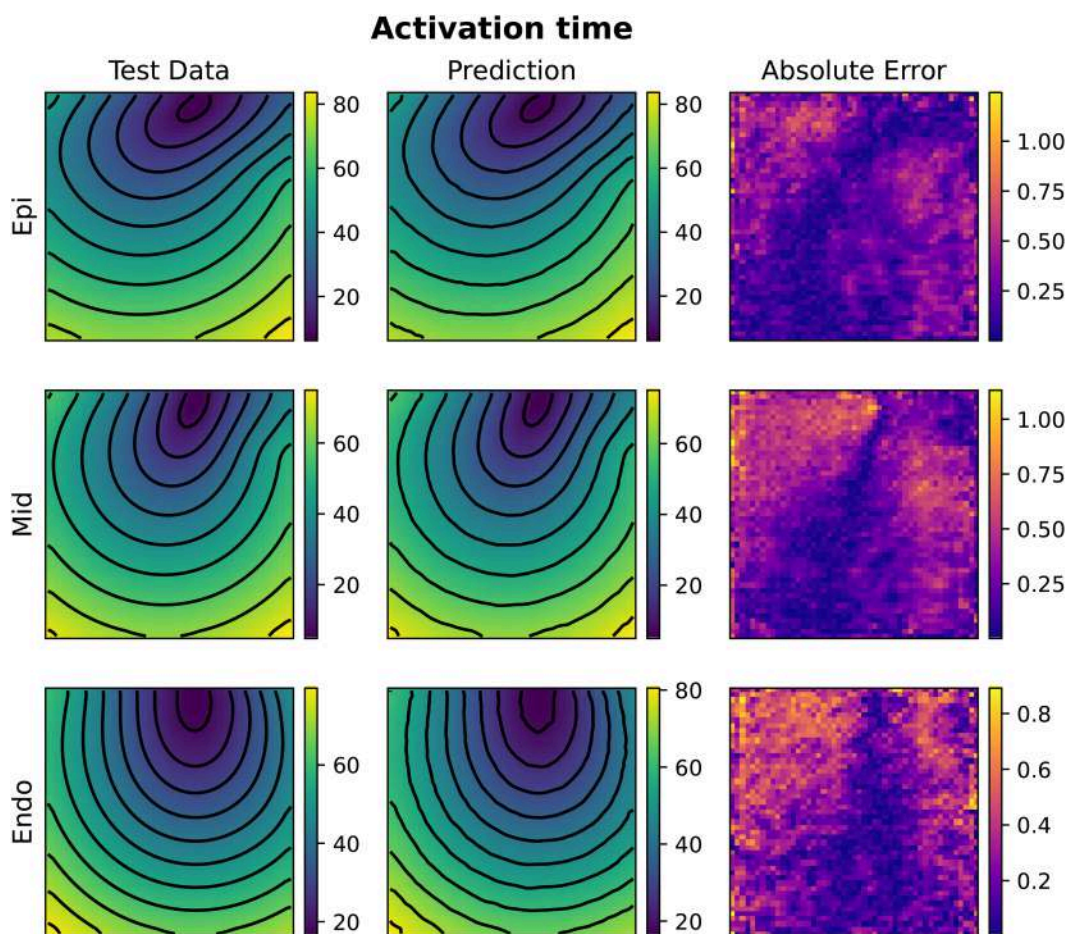


Fig 14. Example of FNO activation time prediction for the 3D heterogeneous case (acti with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g014>

PETSc library [4] on a node equipped with an Intel(R) Xeon(R) Silver 4316 CPU @ 2.30GHz and two Nvidia H100 GPUs, which requires about 4 minutes on 2 cores with GPU acceleration. In order to have a fair comparison from a user's point of view, we have performed the single prediction timing tests on a laptop equipped with an Apple M1 Pro chip, while the trainings have been performed on a machine equipped with an NVIDIA Quadro RTX 5000 GPU for FNO, while on an Intel i7 machine for KOL.

Furthermore, in this case training KOL is significantly faster than FNO, requiring only a few minutes compared to FNO's training time, which ranges from 58 to 84 minutes depending on the configuration. KOL is also much more memory efficient, using just over 1GB of CPU memory, compared to FNO's GPU memory consumption of around 6.1GB. These results suggest that KOL is lighter, faster and more accurate than FNO, even though we expect the latter to be more time-efficient when predicting a large number of occurrences is required.

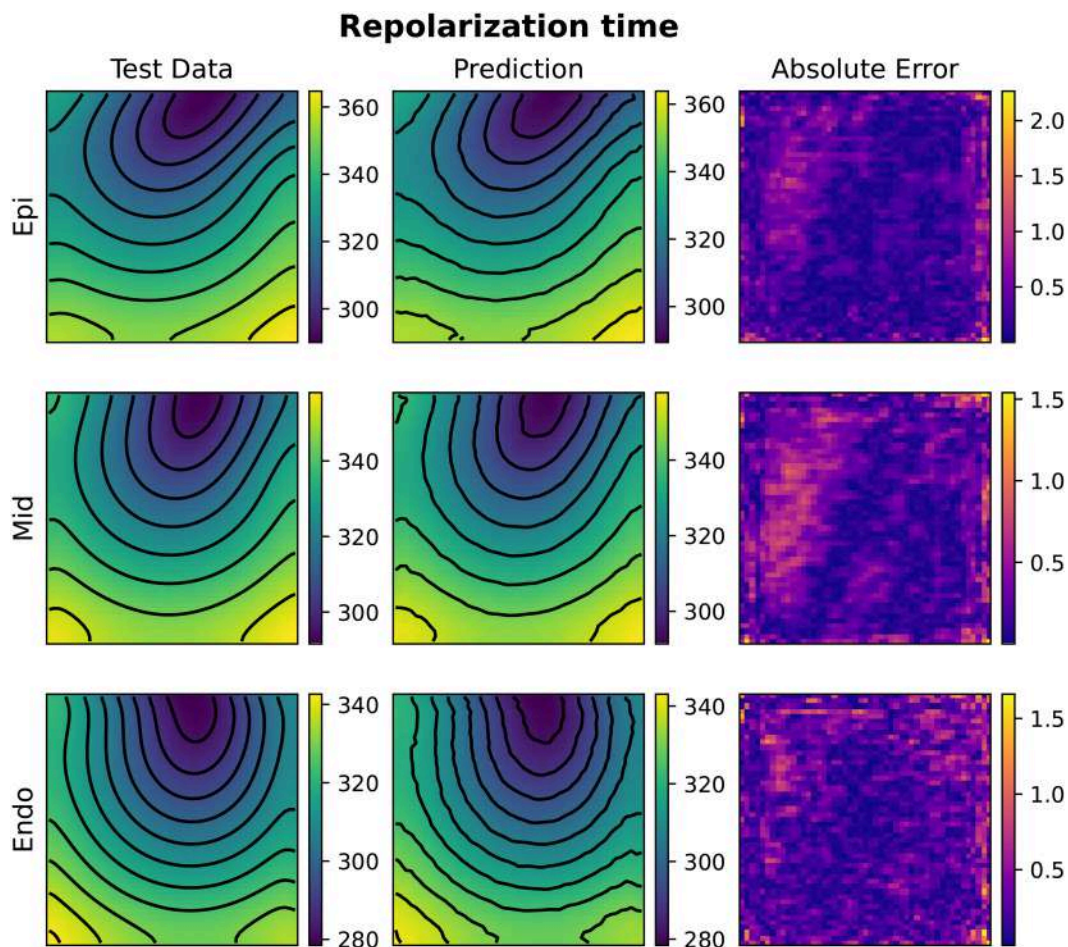


Fig 15. Example of FNO repolarization time prediction for the 3D heterogeneous case (repo with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g015>

Conclusions

In this study, we developed and formalized operator learning approaches to reconstruct activation and repolarization times in cardiac tissue, given an input activation region corresponding to electrically stimulated cells. This problem is particularly significant for clinicians, as utilizing computational architectures for patient-specific simulations can improve clinical decision-making. Although activation and repolarization times can be derived from PDE-based models, simulating these processes entails solving large-scale systems, resulting in high computational costs. The time savings offered by surrogate models, compared to high-fidelity PDE-based approaches, become particularly evident in multi-query applications which require many evaluations of the same model with slight modifications of some quantities of interest, such as inverse problems, parameter estimation, and uncertainty quantification.

For this purpose, we adapted and evaluated two operator learning strategies: Fourier Neural Operators (FNO), based on the convolution theorem for the Fourier transform, and Kernel Operator Learning (KOL), based on kernel regression. These trained operator learning techniques yield accurate and computationally efficient approximations of the target maps when evaluated on new samples. Notably, in the repolarization case, we successfully approximated an operator map

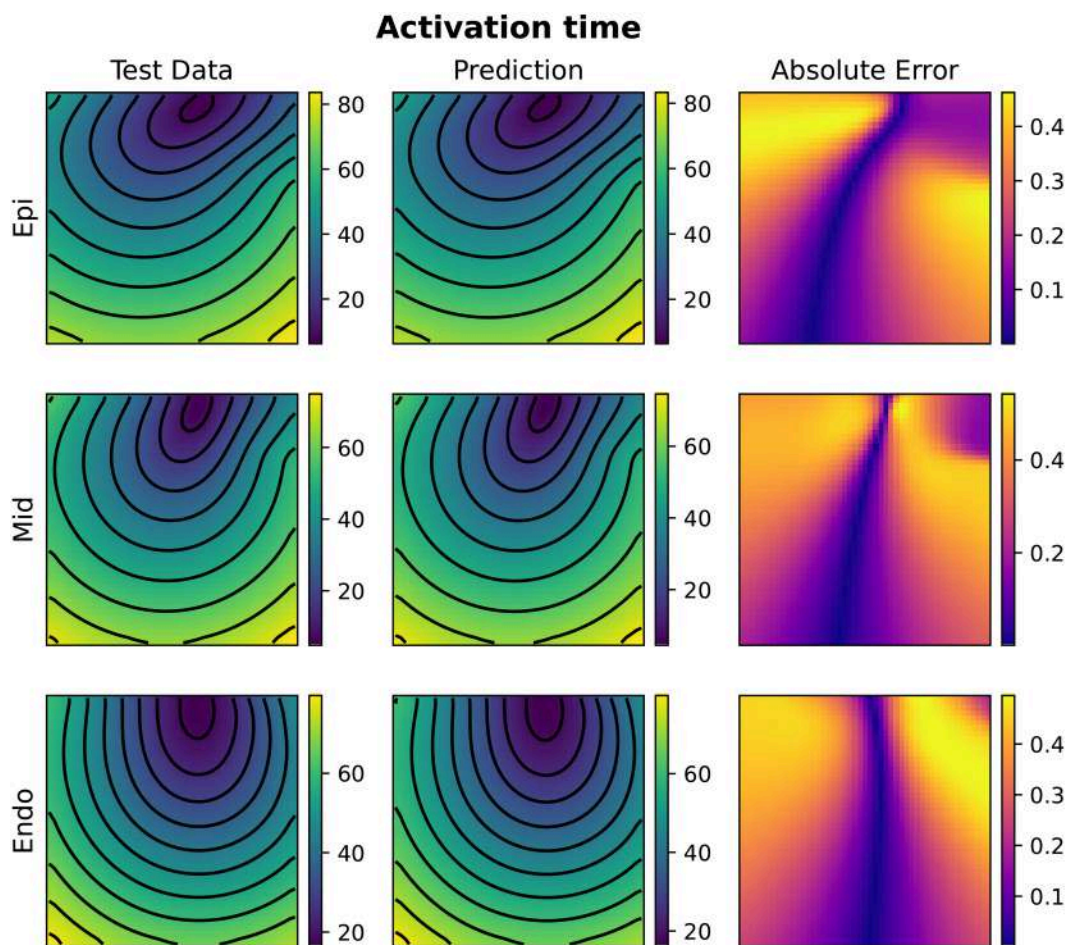


Fig 16. Example of KOL activation time prediction for the 3D heterogeneous case (acti with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g016>

for which no corresponding PDE model is currently available. Training data were generated by solving the Monodomain model with spatial randomly distributed pulses with different ionic models using finite element method on 2D and 3D structured meshes, as well as on a physiologically realistic left ventricle. Both methods proved robust and accurate performance (with errors generally below 1%) while significantly reducing computational costs compared to classical FEM-based simulations in a high number of evaluations. Additionally, a systematic sensitivity analysis was conducted for the 2D case to assess hyperparameter dependence of both architectures.

Our numerical experiments demonstrate that KOL outperforms FNO in terms of accuracy and training time, even in the challenging case of 3D unstructured meshes. However, the computational gains of KOL are partially offset by the significant cost associated with kernel selection, which may pose a limitation. This latter problem may be mitigated by choosing the kernel adaptively, e.g. relying on the so called parametric Kernel Flows approaches [36]. Additionally, our results validate the feasibility of applying FNO to unstructured cardiac meshes, provided that suitable architectural modifications are implemented to accommodate non-uniform data structures. Finally, while KOL offers superior accuracy, it comes at the expense of an increased inference time compared to the FNO counterpart. Hence, FNO can be the eligible choice for very high number of validation data.

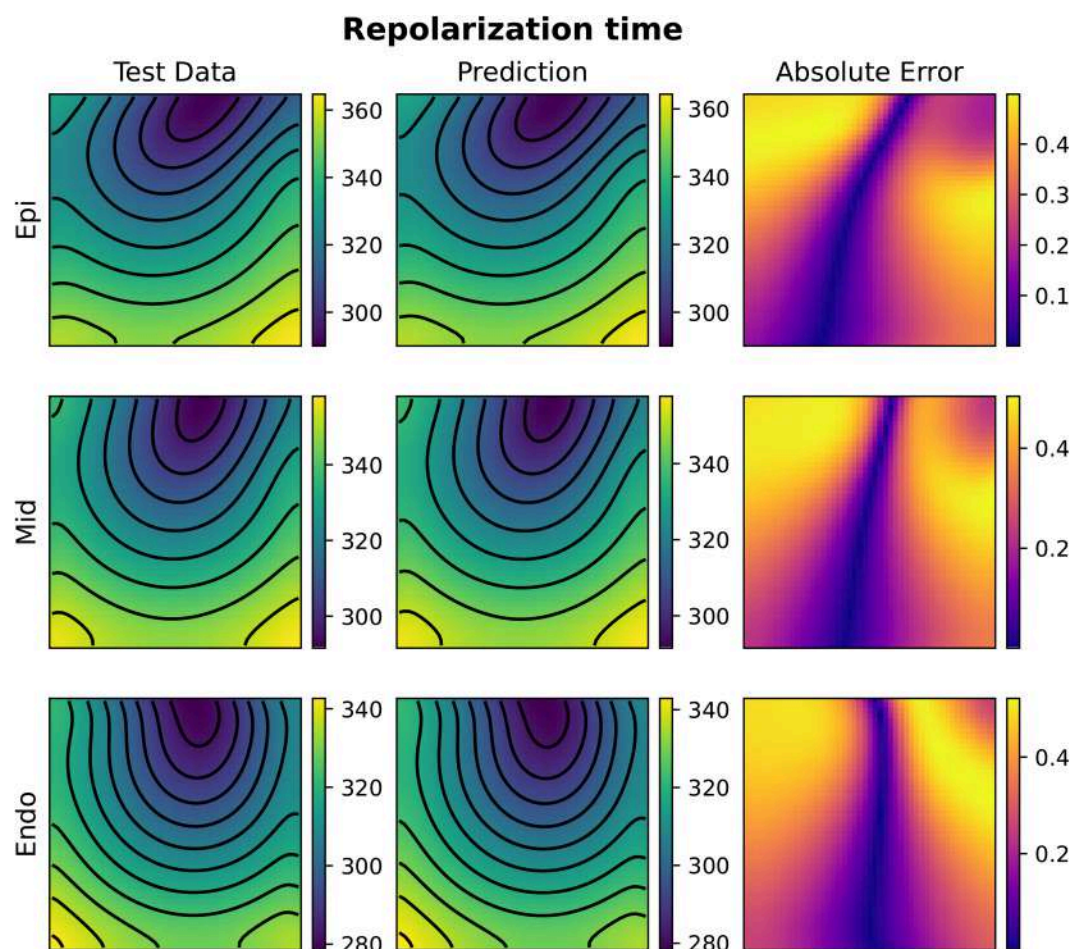


Fig 17. Example of KOL repolarization time prediction for the 3D heterogeneous case (repo with 2000 samples). The picture represents three slices of tissue: epicardium (top), middle (center) and endocardium (bottom). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g017>

Table 4. Performance comparison of FNO and KOL on 3D unstructured datasets (cfr. Table D in S1 File for the performance comparison of FNO architectures with different layers and widths).

	Dataset (size)	Test Error	GPU Memory	Training Time	Test Time	Pearson (test)
FNO with lr policy <code>reduceOnPlateau</code>	acti (2000)	5.86E-02 ± 2.98E-03	6.13GB	80 min	8.4E-03 sec	5.6E-03
	repo (2000)	2.42E-02 ± 1.00E-02	6.15GB	84 min	9.6E-03 sec	2.6E-03
	Dataset (size)	Test Error	CPU Memory	Training Time	Test Time	Pearson (test)
KOL with <code>iq4</code> kernel	acti (2000)	1.36E-02	1.37GB	4 min	4.7E-02 sec	6.6E-05
	repo (2000)	2.53E-03	1.34GB	4 min	4.8E-02 sec	6.1E-05

<https://doi.org/10.1371/journal.pcbi.1013920.t004>

Future work will leverage on the proposed models, which have proven to be fast and accurate surrogate for forward simulations, for inverse problem applications which are extremely relevant in computational cardiology. In particular, a long-term goal is to enable the reconstruction of the likely site of excitation origin based on observed activation maps obtained from electroanatomical mapping or ECG imaging, such as in cases of focal arrhythmias.

Despite the promising potential of KOL and FNO for forward EP problems, a major challenge remains their extension to diverse patient-specific geometries, which will require substantially larger datasets to account for intra-sample variability during training and also test. Model generalization across geometries will be the subject of future work. One possible

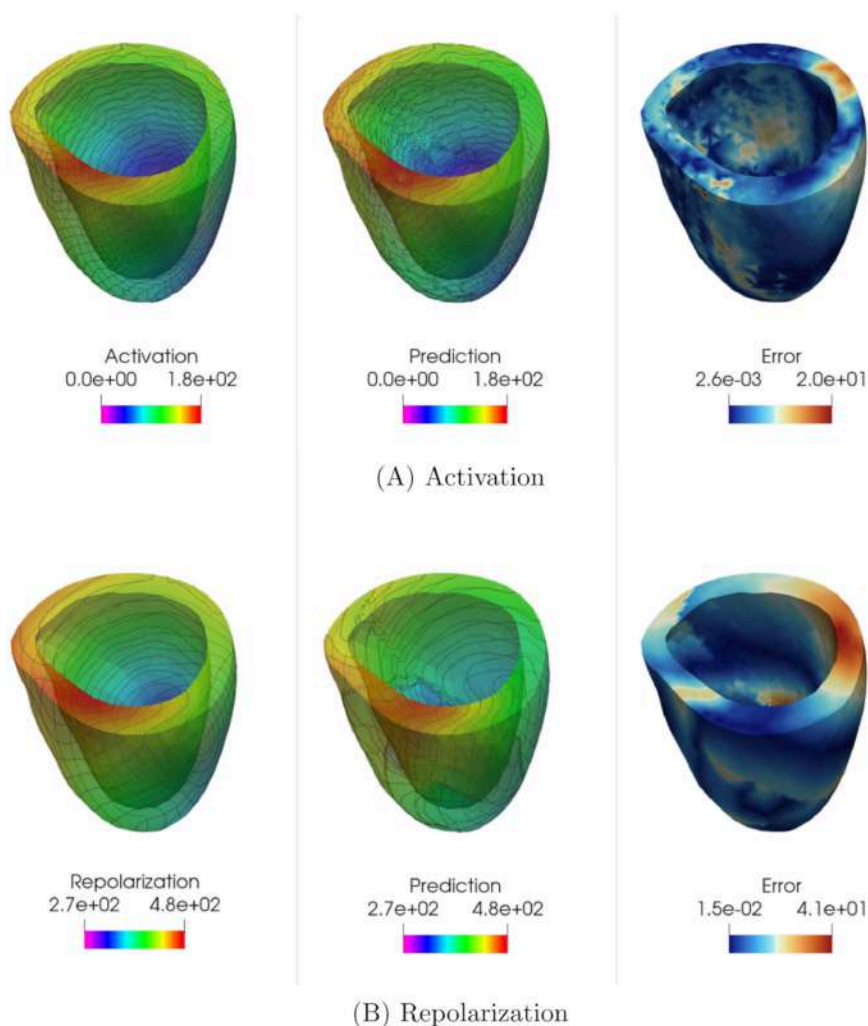


Fig 18. Example of FNO predictions for the 3D unstructured case: (A) Activation times (acti_{2000}), (B) Repolarization times (acti_{2000}). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g018>

direction consists in exploring Universal Solution Manifold Networks (USM-Nets) [41], which are designed to learn mappings over families of PDE solutions defined on different domains. Another viable route is the integration of data assimilation techniques [40] to personalize or adapt pretrained operator models using sparse or partial observations from individual patients.

Building on the flexibility of the proposed operator learning framework, future work will explore the incorporation of additional input parameters to better represent both structural and electrophysiological heterogeneity of the cardiac tissue. In particular, we plan to encode spatially localized features, such as ischemic regions or zones of reduced conductivity, together with parameters describing inter-individual variability in ionic kinetics and electrophysiological remodeling. This extension will enable the surrogate models to generalize across a broader spectrum of physiological and pathological conditions, including drug-induced alterations, abnormal restitution dynamics, and disease-related propagation impairments. However, realizing this extension will require the generation or collection of sufficiently large and diverse training

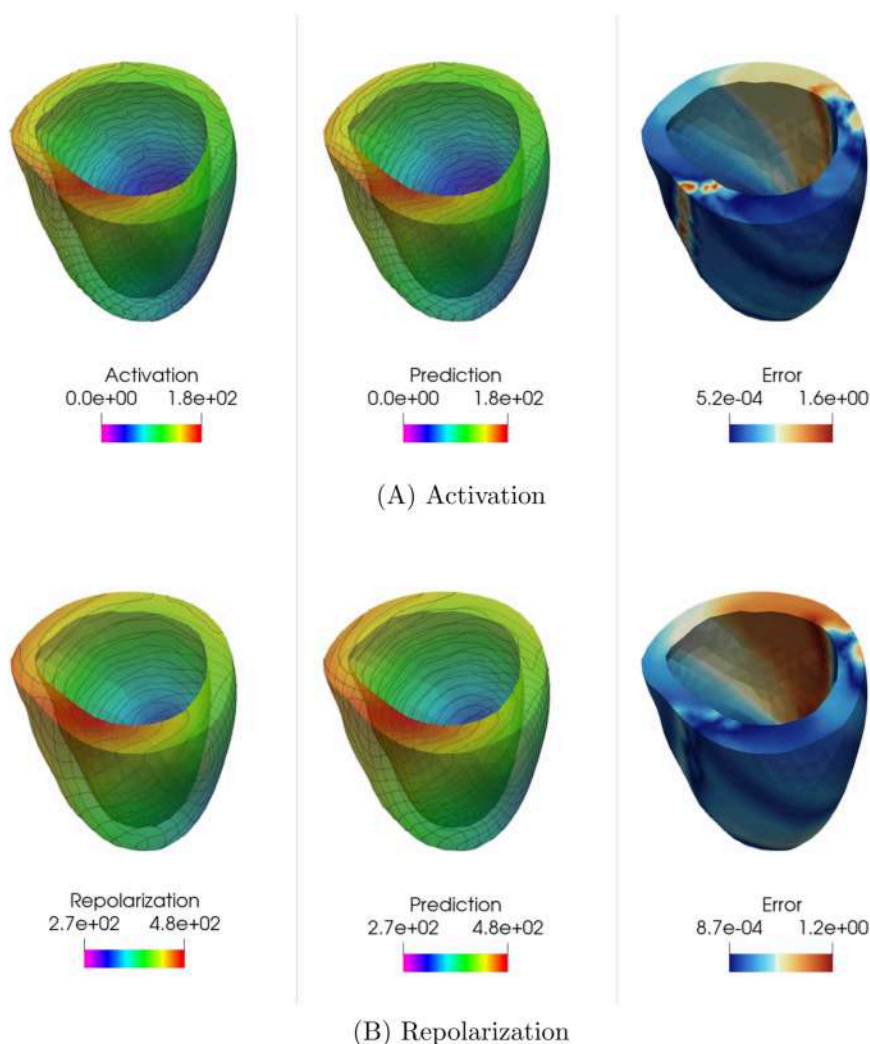


Fig 19. Example of KOL predictions for the 3D unstructured case: (A) Activation times (acti_2000), (B) Repolarization times (acti_2000). Colorbars indicate time in milliseconds (ms).

<https://doi.org/10.1371/journal.pcbi.1013920.g019>

datasets that include both physiological and pathological samples, in order to ensure proper generalization and to prevent overfitting on specific configurations.

Data collection and software

Data employed for the numerical tests presented in this work are available at <https://zenodo.org/records/16913206>, whilst the software is available at <https://github.com/edoardo100/timeactML.git>.

Supporting information

S1 File. Supplementary material. File containing all the supplementary mathematical methods, tables and figures. (PDF)

Author contributions

Conceptualization: Giovanni Ziarelli, Edoardo Centofanti, Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

Data curation: Giovanni Ziarelli, Edoardo Centofanti.

Formal analysis: Giovanni Ziarelli, Edoardo Centofanti, Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

Funding acquisition: Giovanni Ziarelli, Edoardo Centofanti, Simone Scacchi, Luca F. Pavarino.

Investigation: Giovanni Ziarelli, Edoardo Centofanti, Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

Methodology: Giovanni Ziarelli, Edoardo Centofanti, Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

Project administration: Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

Resources: Simone Scacchi.

Software: Giovanni Ziarelli, Edoardo Centofanti, Simone Scacchi.

Supervision: Giovanni Ziarelli, Edoardo Centofanti, Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

Validation: Giovanni Ziarelli, Edoardo Centofanti, Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

Visualization: Giovanni Ziarelli, Edoardo Centofanti.

Writing – original draft: Giovanni Ziarelli, Edoardo Centofanti.

Writing – review & editing: Giovanni Ziarelli, Edoardo Centofanti, Nicola Parolini, Simone Scacchi, Marco Verani, Luca F. Pavarino.

References

1. Africa PC. life: A flexible, high performance library for the numerical solution of complex finite element problems. *SoftwareX*. 2022;20:101252. <https://doi.org/10.1016/j.softx.2022.101252>
2. Amuzescu B, Airini R, Epureanu FB, Mann SA, Knott T, Radu BM. Evolution of mathematical models of cardiomyocyte electrophysiology. *Math Biosci*. 2021;334:108567. <https://doi.org/10.1016/j.mbs.2021.108567> PMID: 33607174
3. Anderson ME, Al-Khatib SM, Roden DM, Califf RM, Duke Clinical Research Institute/American Heart Journal Expert Meeting on Repolarization Changes. Cardiac repolarization: Current knowledge, critical gaps, and new approaches to drug development and patient management. *Am Heart J*. 2002;144(5):769–81. <https://doi.org/10.1067/mhj.2002.125804> PMID: 12422144
4. Balay S, Abhyankar S, Adams M, Brown J, Brune P, Buschelman K, et al. PETSc users manual. 2019.
5. Batlle P, Darcy M, Hosseini B, Owahdi H. Kernel methods are competitive for operator learning. *J Comput Phys*. 2024;496:112549.
6. Bucelli M. The lifex library version 2.0. arXiv preprint. 2024. <https://doi.org/arXiv:2411.19624>
7. Capuano E, Regazzoni F, Maines M, Fornara S, Locatelli V, Catanzariti D, et al. Personalized computational electro-mechanics simulations to optimize cardiac resynchronization therapy. *Biomech Model Mechanobiol*. 2024;23(6):1977–2004. <https://doi.org/10.1007/s10237-024-01878-8> PMID: 39192164
8. Centofanti E, Scacchi S. A comparison of algebraic multigrid bidomain solvers on hybrid CPU–GPU architectures. *Comput Meth Appl Mech Eng*. 2024;423:116875. <https://doi.org/10.1016/j.cma.2024.116875>
9. Centofanti E, Ghiotto M, Pavarino LF. Learning the Hodgkin–Huxley model with operator learning techniques. *Comput Meth Appl Mech Eng*. 2024;432:117381. <https://doi.org/10.1016/j.cma.2024.117381>
10. Clayton RH, Bernus O, Cherry EM, Dierckx H, Fenton FH, Mirabella L, et al. Models of cardiac tissue electrophysiology: Progress, challenges and open questions. *Prog Biophys Mol Biol*. 2011;104(1–3):22–48. <https://doi.org/10.1016/j.pbiomolbio.2010.05.008> PMID: 20553746
11. Colli-Franzone P, Pavarino LF, Scacchi S. Exploring anodal and cathodal make and break cardiac excitation mechanisms in a 3D anisotropic bidomain model. *Math Biosci*. 2011;230(2):96–114. <https://doi.org/10.1016/j.mbs.2011.02.002> PMID: 21329705
12. Colli Franzone P, Pavarino LF, Scacchi S. *Mathematical cardiac electrophysiology*. Springer; 2014.

13. Colli Franzone P, Guerri L, Rovida S. Wavefront propagation in an activation model of the anisotropic cardiac tissue: Asymptotic analysis and numerical simulations. *J Math Biol.* 1990;28(2):121–76. <https://doi.org/10.1007/BF00163143> PMID: 2319210
14. Coronel R, Wilms-Schopman FJG, Opthof T, Janse MJ. Dispersion of repolarization and arrhythmogenesis. *Heart Rhythm.* 2009;6(4):537–43. <https://doi.org/10.1016/j.hrthm.2009.01.013> PMID: 19324316
15. Mendonca Costa C, Gemmell P, Elliott MK, Whitaker J, Campos FO, Strocchi M, et al. Determining anatomical and electrophysiological detail requirements for computational ventricular models of porcine myocardial infarction. *Comput Biol Med.* 2022;141:105061. <https://doi.org/10.1016/j.combiomed.2021.105061> PMID: 34915331
16. Coveney S, Cantwell C, Roney C. Atrial conduction velocity mapping: Clinical tools, algorithms and approaches for understanding the arrhythmogenic substrate. *Med Biol Eng Comput.* 2022;60(9):2463–78. <https://doi.org/10.1007/s11517-022-02621-0> PMID: 35867323
17. Dell'Era G, Gravellone M, Scacchi S, Franzone PC, Pavarino LF, Boggio E, et al. A clinical-in silico study on the effectiveness of multipoint bicathodic and cathodic-anodal pacing in cardiac resynchronization therapy. *Comput Biol Med.* 2021;136:104661. <https://doi.org/10.1016/j.combiomed.2021.104661> PMID: 34332350
18. He K, Zhang X, Ren S, Sun J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: *Proc IEEE Int Conf Comput Vis*; 2015. p. 1026–34.
19. Fitzhugh R. Impulses and physiological states in theoretical models of nerve membrane. *Biophys J.* 1961;1(6):445–66. [https://doi.org/10.1016/s0006-3495\(61\)86902-6](https://doi.org/10.1016/s0006-3495(61)86902-6) PMID: 19431309
20. Fresca S, Manzoni A, Dedè L, Quarteroni A. Deep learning-based reduced order models in cardiac electrophysiology. *PLoS One.* 2020;15(10):e0239416. <https://doi.org/10.1371/journal.pone.0239416> PMID: 33002014
21. Goswami S, Yin M, Yu Y, Karniadakis GE. A physics-informed variational DeepONet for predicting crack path in quasi-brittle materials. *Comput Meth Appl Mech Eng.* 2022;391:114587. <https://doi.org/10.1016/j.cma.2022.114587>
22. Izhikevich E, FitzHugh R. FitzHugh-Nagumo model. *Scholarpedia.* 2006;1(9):1349. <https://doi.org/10.4249/scholarpedia.1349>
23. Keener JP, Sneyd J. *Mathematical physiology.* 2nd ed. Springer; 2009.
24. Keener JP. An eikonal-curvature equation for action potential propagation in myocardium. *J Math Biol.* 1991;29(7):629–51. <https://doi.org/10.1007/BF00163916> PMID: 1940663
25. Kissas G, Yang Y, Hwuang E, Witschey WR, Detre JA, Perdikaris P. Machine learning in cardiovascular flows modeling: Predicting arterial blood pressure from non-invasive 4D flow MRI data using physics-informed neural networks. *Comput Meth Appl Mech Eng.* 2020;358:112623. <https://doi.org/10.1016/j.cma.2019.112623>
26. Kovachki N, Li Z, Liu B, Azizzadenesheli K, Bhattacharya K, Stuart A, et al. Neural operator: Learning maps between function spaces with applications to PDEs. *J Mach Learn Res.* 2023;24(89):1–97.
27. Krishnamoorthi S, Perotti LE, Borgstrom NP, Ajjola OA, Frid A, Ponnaluri AV, et al. Simulation methods and validation criteria for modeling cardiac ventricular electrophysiology. *PLoS One.* 2014;9(12):e114494. <https://doi.org/10.1371/journal.pone.0114494> PMID: 25493967
28. Li Z, Kovachki N, Azizzadenesheli K, Liu B, Bhattacharya K, Stuart A, et al. Fourier neural operator for parametric partial differential equations. *arXiv preprint.* 2020;arXiv:2010.08895..
29. Lingsch L, Michelis MY, de Bezenac E, Perera SM, Katzschmann RK, Mishra S. Beyond regular grids: Fourier-based neural operators on arbitrary domains. *arXiv preprint.* 2023;arXiv:2305.19663.
30. Lu L, Jin P, Pang G, Zhang Z, Karniadakis GE. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nat Mach Intell.* 2021;3(3):218–29.
31. Lu L, Meng X, Cai S, Mao Z, Goswami S, Zhang Z, et al. A comprehensive and fair comparison of two neural operators (with practical extensions) based on FAIR data. *Comput Meth Appl Mech Eng.* 2022;393:114778. <https://doi.org/10.1016/j.cma.2022.114778>
32. Martinez E, Moscoloni B, Salvador M, Kong F, Peirlinck M, Marsden AL. Full-field surrogate modeling of cardiac function encoding geometric variability. 2025. <https://doi.org/10.48550/arXiv.2504.20479>
33. Meanti G, Carratino L, Rosasco L, Rudi A. Kernel methods through the roof: Handling billions of points efficiently. *Adv Neural Inf Process Syst.* 2020;33:14410–22.
34. Niederer SA, Lumens J, Trayanova NA. Computational models in cardiology. *Nat Rev Cardiol.* 2019;16(2):100–11. <https://doi.org/10.1038/s41569-018-0104-y> PMID: 30361497
35. Noble D, Rudy Y. Models of cardiac ventricular action potentials: Iterative interaction between experiment and simulation. *Philos Trans R Soc Lond Ser A: Math Phys Eng Sci.* 2001;359(1783):1127–42. <https://doi.org/10.1098/rsta.2001.0820>
36. Owhadi H, Yoo GR. Kernel flows: From learning kernels from data into the abyss. *J Comput Phys.* 2019;389:22–47.
37. Piersanti R. Mathematical and numerical modeling of cardiac fiber generation and electromechanical function: Towards a realistic simulation of the whole heart. *Politecnico di Milano*; 2022.
38. Punske BB, Ni Q, Lux RL, MacLeod RS, Ershler PR, Dustman TJ, et al. Spatial methods of epicardial activation time determination in normal hearts. *Ann Biomed Eng.* 2003;31(7):781–92. <https://doi.org/10.1114/1.1581877> PMID: 12971611
39. Regazzoni F, Dedè L, Quarteroni A. Machine learning of multiscale active force generation models for the efficient simulation of cardiac electromechanics. *Comput Meth Appl Mech Eng.* 2020;370:113268. <https://doi.org/10.1016/j.cma.2020.113268>

40. Regazzoni F, Chapelle D, Moireau P. Combining data assimilation and machine learning to build data-driven models for unknown long time dynamics—Applications in cardiovascular modeling. *Int J Numer Method Biomed Eng*. 2021;37(7):e3471. <https://doi.org/10.1002/cnm.3471> PMID: 33913623
41. Regazzoni F, Pagani S, Quarteroni A. Universal solution manifold networks (USM-Nets): Non-intrusive mesh-free surrogate models for problems in variable domains. *J Biomech Eng*. 2022;144(12):121004. <https://doi.org/10.1115/1.4055285> PMID: 35993790
42. Regazzoni F, Pagani S, Salvador M, Dede' L, Quarteroni A. Learning the intrinsic dynamics of spatio-temporal processes through Latent Dynamics Networks. *Nat Commun*. 2024;15(1):1834. <https://doi.org/10.1038/s41467-024-45323-x> PMID: 38418469
43. Rogers JM, McCulloch AD. A collocation—Galerkin finite element model of cardiac action potential propagation. *IEEE Trans Biomed Eng*. 1994;41(8):743–57. <https://doi.org/10.1109/10.310090> PMID: 7927397
44. Rudy Y, Silva JR. Computational biology in the study of cardiac ion channels and cell electrophysiology. *Q Rev Biophys*. 2006;39(1):57–116. <https://doi.org/10.1017/S0033583506004227> PMID: 16848931
45. Santiago A, Aguado-Sierra J, Zavala-Aké M, Doste-Beltran R, Gómez S, Arís R, et al. Fully coupled fluid-electro-mechanical model of the human heart for supercomputers. *Int J Numer Method Biomed Eng*. 2018;34(12):e3140. <https://doi.org/10.1002/cnm.3140> PMID: 30117302
46. Shi L, Zhang Z. Iterative kernel regression with preconditioning. *Anal Appl*. 2024;22(6):1095–131.
47. Spach MS, Dolber PC. Relating extracellular potentials and their derivatives to anisotropic propagation at a microscopic level in human cardiac muscle. Evidence for electrical uncoupling of side-to-side fiber connections with increasing age. *Circ Res*. 1986;58(3):356–71. <https://doi.org/10.1161/01.res.58.3.356> PMID: 3719925
48. Steinhaus BM. Estimating cardiac transmembrane activation and recovery times from unipolar and bipolar extracellular electrograms: A simulation study. *Circ Res*. 1989;64(3):449–62. <https://doi.org/10.1161/01.res.64.3.449> PMID: 2917378
49. Sundnes J, Lines GT, Cai X, Nielsen BF, Mardal KA, Tveito A. Computing the electrical activity in the heart. Springer; 2007.
50. ten Tusscher KHWJ, Noble D, Noble PJ, Panfilov AV. A model for human ventricular tissue. *Am J Physiol Heart Circ Physiol*. 2004;286(4):H1573–89. <https://doi.org/10.1152/ajpheart.00794.2003> PMID: 14656705
51. Tenderini R, Pagani S, Quarteroni A, Deparis S. PDE-aware deep learning for inverse problems in cardiac electrophysiology. *SIAM J Sci Comput*. 2022;44(3):B605–39. <https://doi.org/10.1137/21m1438529>
52. Trayanova NA, Popescu DM, Shade JK. Machine learning in arrhythmia and electrophysiology. *Circ Res*. 2021;128(4):544–66. <https://doi.org/10.1161/CIRCRESAHA.120.317872> PMID: 33600229
53. Trayanova NA, Lyon A, Shade J, Heijman J. Computational modeling of cardiac electrophysiology and arrhythmogenesis: Toward clinical translation. *Physiol Rev*. 2024;104(3):1265–333. <https://doi.org/10.1152/physrev.00017.2023> PMID: 38153307
54. Tripura T, Chakraborty S. Wavelet neural operator for solving parametric partial differential equations in computational mechanics problems. *Comput Meth Appl Mech Eng*. 2023;404:115783. <https://doi.org/10.1016/j.cma.2022.115783>
55. Tveito A, Jaeger KH, Kuchta M, Mardal KA, Rognes ME. A cell-based framework for numerical modeling of electrical conduction in cardiac tissue. *Front Phys*. 2017;5:48.
56. Varró A, Baczkó I. Cardiac ventricular repolarization reserve: A principle for understanding drug-related proarrhythmic risk. *Br J Pharmacol*. 2011;164(1):14–36. <https://doi.org/10.1111/j.1476-5381.2011.01367.x> PMID: 21545574
57. Vergara C, Lange M, Palamara S, Lassila T, Frangi AF, Quarteroni A. A coupled 3D–1D numerical monodomain solver for cardiac electrical activation in the myocardium with detailed Purkinje network. *J Comput Phys*. 2016;308:218–38.
58. Vergara C, Stella S, Maines M, Africa PC, Catanzariti D, Demattè C, et al. Computational electrophysiology of the coronary sinus branches based on electro-anatomical mapping for the prediction of the latest activated region. *Med Biol Eng Comput*. 2022;60(8):2307–19. <https://doi.org/10.1007/s11517-022-02610-3> PMID: 35729476
59. Henson VE, Yang UM. BoomerAMG: A parallel algebraic multigrid solver and preconditioner. *Appl Numer Math*. 2002;41(1):155–77. [https://doi.org/10.1016/s0168-9274\(01\)00115-5](https://doi.org/10.1016/s0168-9274(01)00115-5)
60. Ziarelli G, Parolini N, Verani M. Learning epidemic trajectories through kernel operator learning: From modelling to optimal control. *NMTMA*. 2025;18(2):285–324. <https://doi.org/10.4208/nmtma.aa-2024-0097>