A Spatial-Temporally Adaptive PINN Framework for 3D Bi-Ventricular Electrophysiological Simulations and Parameter Inference

Yubo Ye[1], Huafeng Liu[1,3,4(✉)], Xiajun Jiang[2], Maryam Toloubidokhti[2], and Linwei Wang[2]

[1] State Key Laboratory of Modern Optical Instrumentation, Department of Optical Engineering, Zhejiang University, Hangzhou 310027, China
liuhf@zju.edu.cn
[2] Rochester Institute of Technology, Rochester, NY 14623, USA
[3] Jiaxing Key Laboratory of Photonic Sensing and Intelligent Imaging, Jiaxing 314000, China
[4] Intelligent Optics and Photonics Research Center, Jiaxing Research Institute, Zhejiang University, Jiaxing 314000, China

Abstract. Physics-informed neural networks (PINNs) is a new paradigm for solving the forward and inverse problems of partial differential equations (PDEs). Its penetration into 3D bi-ventricular electrophysiology (EP) however has been slow, owing to its fundamental limitations to solve PDEs over large or complex solution domains with sharp transitions. In this paper, we propose a new PINN framework to overcome these challenges via three key innovations: 1) a weak-form PDE residual to bypass the challenges of high-order spatial derivatives over irregular spatial domains, 2) a spatial-temporally adaptive training strategy to mitigate the failure of PINN to propagate correct solutions and accelerate convergence, and 3) a sequential learning strategy to enable solutions over longer time domains. We experimentally demonstrated the effectiveness of the presented PINN framework to obtain the complete forward and inverse EP solutions over the 3D bi-ventricular geometry, which is otherwise not possible with vanilla PINN frameworks.

Keywords: Physics-informed neural network · Parameter Inference · Cardiac electrophysiology · Spatial-temporally adaptive training

1 Introduction

Virtual models of cardiac electrophysiology (EP) have demonstrated significant potential in various clinical tasks, such as stratifying the risk for lethal arrhythmias [2] and predicting responses to cardiac resynchronization therapy [18]. Numerical solutions to the governing partial differential equations (PDEs) for cardiac EP (forward problem), however, are difficult to obtain. The inverse

H. Greenspan et al. (Eds.): MICCAI 2023, LNCS 14226, pp. 163–172, 2023.
https://doi.org/10.1007/978-3-031-43990-2_16

estimation of the parameters of these PDEs given observation data (inverse problem) is even more difficult, due to challenges such as the complex relation between the PDE parameters and the observations, and the need to embed a numerical PDE solver within the inverse optimization process.

Physics-informed neural networks (PINNs) is a new paradigm for solving both the forward and inverse problems of PDEs [16]. The general idea of PINNs is to train neural networks to satisfy physical laws as described by the PDEs, optimized via the so-called PDE residuals. When partial observations of the PDE solutions are available, this PDE residual can be combined with data-residuals to solve the forward and inverse problems at the same time [4,11,13,15,23]. Despite substantial attention and successes, however, the use of PINN in cardiac EP has been rather limited. To date, there have only been two attempts of PINN-based EP simulation that are limited to 1D/2D [9] or atrial surfaces [17].

What fundamental challenges does 3D bi-ventricular EP present for PINNs? First, 3D bi-ventricular EP simulation requires the PDE solutions to be obtained over a complex geometry domain in space and a long duration in time. Second, the PDE solution to bi-ventricular EP – in the form of the spatiotemporal propagation of transmembrane potential (TMP) activation – exhibits sharp spatial and temporal gradients. These characteristics of bi-ventricluar EP simulations present fundamental challenges to the state-of-the-art PINN framework, which has been shown to face potential failure modes when the PDEs are solved over large or complex solution domains with sharp transitions [8,10,14,21,22].

In this paper, we present a novel PINN framework to overcome these challenges and enable the forward and inverse solutions to 3D bi-ventricular EP simulations. This is achieved with three key innovations. First, to avoid dealing with higher-order spatial derivatives over the complex 3D geometry, we formulated the PINN over the meshfree representation of the 3D bi-ventricular geometry with a modified PDE residual incorporating the weak form of the original PDE. Second, to enable PINN solutions over the long temporal domain, we present a temporally adaptive and sequential training strategy to guide the PINN to respect the causality of the underlying physics of wave propagation. Finally, we introduce a spatially adaptive training strategy to guide the PINN to exploit the spatiotemporal sparsity of the sharp gradients exhibited in the PDE solution.

We experimentally demonstrated that the presented PINN framework was able to enable complete simulation of the bi-ventricular EP activation process that is otherwise not possible with vanilla PINN frameworks. We further conducted detailed ablation studies of the benefits of the presented spatial-temporally adaptive and sequential learning strategies, and demonstrated preliminary feasibility of the presented PINN framework for supporting the inverse parameter estimation of 3D bi-ventricular EP models. These represent an innovative first attempt to enable PINN solutions for 3D bi-ventricular EP applications.

2 Background: Bi-Ventricular EP Simulations

Existing ventricular EP models range from macroscopic-level two-variable PDEs to ionic models with tens of variables [5]. In this proof-of-concept study, we

consider the two-variable diffusion-reaction Aliev-Panfilov (AP) PDE [1] for its ability to reproduce key excitation features without formidable computation:

$$\frac{\partial u}{\partial t} = \nabla(\mathbf{D}\nabla u) + f_1(u,v), \quad f_1(u,v) = cu(u-\gamma)(u-1) - uv \tag{1}$$

$$\frac{\partial v}{\partial t} = f_2(u,v), \quad f_2(u,v) = (e_0 + (\mu_1 v)/(u+\mu_2))(-v - cu(u-\gamma-1)) \tag{2}$$

$$\frac{\partial u}{\partial n} = 0\text{: natural boundary condition} \tag{3}$$

where $u \in [0,1]$ is the unit-less TMP and v is the recovery current. The diffusion tensor $\mathbf{D}$ describes local conductivity anisotropy determined by fiber structures. Parameters $\{\gamma, c, e_0, \mu_1, \mu_2\}$ control the temporal dynamics of u and v.

3 Methodology

We present a novel PINN framework to support forward and inverse 3D bi-ventricular EP simulations. As outlined in Fig. 1, it includes three key components: 1) a weak-form PDE residual to bypass the challenges of high-order spatial derivatives over irregular geometry; 2) a novel spatial-temporally adaptive strategy to mitigate PDE propagation failure over large solution domains and accelerate convergence; and 3) a sequential training strategy to enable solutions over the complete bi-ventricular EP activation process.

3.1 Weak-Form PDE Residual over Meshfree Representations

In a vanilla PINN framework, we will use a neural network $u_\theta(x,t)$ to approximate PDE solutions $u(x,t)$ to the AP model. The network will be optimized by the initial, boundary, and PDE residuals ($Loss_I$, $Loss_B$ and $Loss_R$) over a set of points $\{t_k\}_{k=1}^{N_T}$, $\{x_i\}_{i=1}^{N_\Omega}$, $\{x_j\}_{j=1}^{N_{\partial\Omega}}$ sampled in time, space, and boundary domains:

$$Res_{1,ik} = \frac{\partial u_\theta(x_i,t_k)}{\partial t} - \nabla(\mathbf{D}\nabla u_\theta(x_i,t_k) - f_1(u_\theta(x_i,t_k), v_\theta(x_i,t_k)) \tag{4}$$

$$Res_{2,ik} = \frac{\partial v_\theta(x_i,t_k)}{\partial t} - f_2(u_\theta(x_i,t_k), v_\theta(x_i,t_k)) \tag{5}$$

$$Loss_R = \frac{1}{N_\Omega N_T}\sum_{i=1}^{N_\Omega}\sum_{k=1}^{N_T} |Res_{1,ik}|^2 + |Res_{2,ik}|^2 \tag{6}$$

$$Loss_I = \frac{1}{N_\Omega}\sum_{i=1}^{N_\Omega} |u(x_i,0) - u_\theta(x_i,0)|^2 \tag{7}$$

$$Loss_B = \frac{1}{N_{\partial\Omega} N_T}\sum_{j=1}^{N_{\partial\Omega}}\sum_{k=1}^{N_T} \left|\frac{\partial u_\theta(x_j,t_k)}{\partial n}\right|^2 \tag{8}$$

$$Loss = \lambda_I Loss_I + \lambda_B Loss_B + \lambda_R Loss_R \tag{9}$$

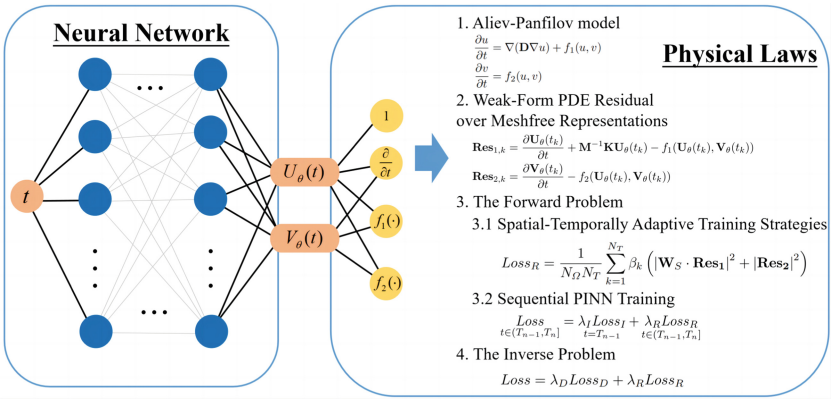

Fig. 1. The schematics of the presented PINN framework

where λ_I, λ_B and λ_R are the hyperparameters that balance these residuals during training. Because $u_\theta(x,t)$ lives on the 3D geometry of the ventricles, it is non-trivial to calculate the second-order spatial derivatives.

To address this, we utilize the weak form of PDE and spatially discretize it on a bi-ventricular mesh through the Mesh-free method [19] as:

$$\frac{\partial \mathbf{U}(t)}{\partial t} = -\mathbf{M}^{-1}\mathbf{K}\mathbf{U}(t) + f_1(\mathbf{U}(t), \mathbf{V}(t)), \quad \frac{\partial \mathbf{V}(t)}{\partial t} = f_2(\mathbf{U}(t), \mathbf{V}(t)) \tag{10}$$

where vectors $\mathbf{U} = [u_1, u_2, \ldots, u_{N_\Omega}]^T$ and $\mathbf{V} = [v_1, v_2, \ldots, v_{N_\Omega}]^T$ consist of u and v from all N_Ω mesh-free points inside the myocardium. Matrices $\mathbf{M}$ and $\mathbf{K}$ represent numerical approximations of the second-order spatial derivatives in Eq. (1), which also automatically incorporate the natural boundary condition [19,24]. We then let the PINN neural network describe the PDE solutions over the discrete ventricular mesh as $\mathbf{U}_\theta(t)$ and $\mathbf{V}_\theta(t)$ as a function of time t, and obtain a modified PDE residual as:

$$\mathbf{Res}_{1,k} = \frac{\partial \mathbf{U}_\theta(t_k)}{\partial t} + \mathbf{M}^{-1}\mathbf{K}\mathbf{U}_\theta(t_k) - f_1(\mathbf{U}_\theta(t_k), \mathbf{V}_\theta(t_k)) \tag{11}$$

$$\mathbf{Res}_{2,k} = \frac{\partial \mathbf{V}_\theta(t_k)}{\partial t} - f_2(\mathbf{U}_\theta(t_k), \mathbf{V}_\theta(t_k)) \tag{12}$$

$$Loss_R = \frac{1}{N_\Omega N_T}\sum_{k=1}^{N_T} |\mathbf{Res}_{1,k}|^2 + |\mathbf{Res}_{2,k}|^2 \tag{13}$$

With this modification, PINN no longer needs to deal directly with the second-order spatial derivatives over the 3D ventricular geometry.

3.2 Spatial-Temporally Adaptive Training Strategies

A common challenge in PINN training is the failure to propagate PDE solutions over a large solution domain [6,14,20]. Fundamentally different from conven-

tional training, only labels of the initial PDE solutions are known in PINN training, from which the correct solutions need to be propagated to the entire solution domain. Before the correct solution arrives, the intermediate (and mostly trivial) solutions also propagate. If dominating the propagation, this will result in a chain reaction that prevents the propagation of correct solutions, leading to propagation failure. This problem is escalated if the propagation gradient is sharp.

Temporally Adaptive Training: We argue that the propagation failure is caused by the inability of vanilla PINNs to respect the causality underlying the physical laws of wave propagation. Therefore, we propose temporal weights $\{\beta_1, \beta_2, \ldots, \beta_{N_T}\}$ to guide PINNs to respect temporal causality during training. The intuition is that, when the correct solution is propagated to time t_c, the previous times $\{t_1, t_2, \ldots, t_{c-1}\}$ should all have lower PDE residuals. We thus propose the temporal weights to be:

$$\beta_k = \begin{cases} 1 & \text{if } \sum_{m=1}^{k-1} Loss_{R_m} < \epsilon_t \\ 0 & \text{else} \end{cases} \tag{14}$$

where ϵ_t is a threshold to be tuned. This will guide the propagation of correct solutions while avoiding obtaining and propagating trivial intermediate solutions.

Spatially Adaptive Training: While sharp gradients in PDE solutions are challenging for PINNs [12,14,22], they are sparse both in space and time in ventricular EP activation. We thus propose to exploit these sharp-gradients to focus PINN training on these sparse regions, thus accelerating PINN convergence. This is achieved by spatial weights $\mathbf{W}_S = [w_1, w_2, \ldots, w_{N_\Omega}]$ defined as:

$$w_i = \begin{cases} w_h & \text{if} \frac{\partial u_i}{\partial t}|_{t_c} > \epsilon_s \\ 1 & \text{else} \end{cases} \tag{15}$$

where $w_h > 1$ and ϵ_s is another threshold to be tuned.

With the above strategy, PDE residual is further modified to:

$$Loss_R = \frac{1}{N_\Omega N_T} \sum_{k=1}^{N_T} \beta_k \left(|\mathbf{W}_S \cdot \mathbf{Res}_{1,k}|^2 + |\mathbf{Res}_{2,k}|^2 \right) \tag{16}$$

Sequential PINN Training: Even when propagation causality and sparse regions of sharp gradients are respected, PINN cannot solve for arbitrarily long time domains because the loss landscape becomes increasingly complex as N_T increases. We thus further utilize a sequential learning method where we first uniformly discretize the time domain $[0, T]$ into n segments, and then train the PINN across these segments sequentially as:

$$\underset{t\in(T_{n-1},T_n]}{Loss} = \lambda_I \underset{t=T_{n-1}}{Loss_I} + \lambda_R \underset{t\in(T_{n-1},T_n]}{Loss_R} \tag{17}$$

where PDE solutions obtained from previous time segments become the initial residual for training the PINN for the next time segment. The complete PDE solution for the entire time domain is obtained at the end of the training.

3.3 Solving the Forward and Inverse Problems

In the forward problem, PINNs approximate the solution of the AP model by optimizing the initial residual $Loss_I$ and the PDE residual $Loss_R$:

$$Loss_I = \frac{1}{N_\Omega} |\mathbf{U}_\theta(0) - \mathbf{U}(0)|^2 + |\mathbf{V}_\theta(0) - \mathbf{V}(0)|^2 \tag{18}$$

$$Loss = \lambda_I Loss_I + \lambda_R Loss_R \tag{19}$$

In the inverse problem, PINNs utilize partially known solutions $\{\mathbf{U}(t_k)\}_{k=1}^{N_T}$ to simultaneously optimize the PINN parameters θ and unknown PDE parameters ϕ with an additional data loss $Loss_D$:

$$Loss_R = \frac{1}{N_\Omega N_T} \sum_{k=1}^{N_T} |\mathbf{Res}_{1,k}|^2_{\phi=\phi_\theta} + |\mathbf{Res}_{2,k}|^2_{\phi=\phi_\theta} \tag{20}$$

$$Loss_D = \frac{1}{N_\Omega N_T} \sum_{k=1}^{N_T} |\mathbf{U}_\theta(t_k) - \mathbf{U}(t_k)|^2 \tag{21}$$

$$Loss = \lambda_D Loss_D + \lambda_R Loss_R \tag{22}$$

4 Experiments and Results

In all experiments, the neural network we used has 5 quadratic residual layers [3] as hidden layers with 512 neurons in each layer, where the input is time t and the output is $N_\Omega(=1862)$ dimensional vectors $\mathbf{U}_\theta$ and $\mathbf{V}_\theta$. The quadratic residual layers have a stronger nonlinearity than the fully connected layers [3]. We used Adam optimizer and the learning rate is set at 1×10^{-3}. The parameters of AP model are fixed to standard values as documented in the literature [1]:$c = 8$, $e_0 = 0.002$, $\mu_1 = 0.2$ and $\mu_2 = 0.3$. The hyperparameters of our proposed PINN framework are set as follows: $\epsilon_t = 0.001$, $\epsilon_s = 0.01$, w_h=4 and $T_n - T_{n-1} = 1$. In the forward problem $\lambda_I : \lambda_R = 10 : 1$, and in the inverse problem $\lambda_D : \lambda_R = 1 : 1000$. All experiments were run on NVIDIA Titan RTX and numerical solution generated through the method in the literature [19].

The Effectiveness of Spatial-Temporally Adaptive PINNs: To verify the effectiveness of the spatial-temporally adaptive strategies for PINN training, we solve the forward problem of the AP model on time domain $t \in [0, 1]$, $t \in [0, 5]$ and $t \in [0, 10]$ considering three ablation models: vanilla PINN, PINN + temporal weights, PINN + temporal and spatial weights. Figure 2 shows MSE error between PINN output and numerical solution during the training process.

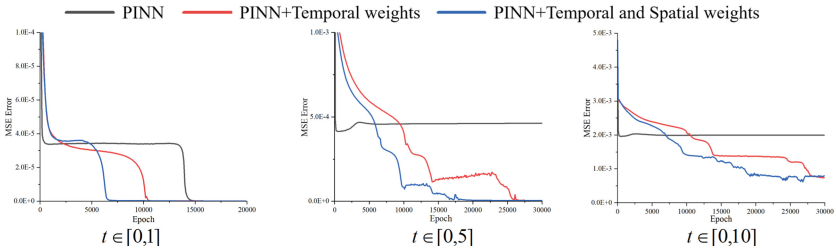

Fig. 2. Training MSE between PDE and PINN solutions for three ablation models. (Color figure online)

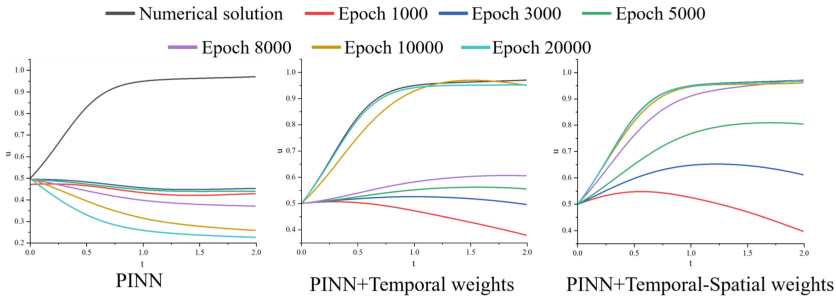

Fig. 3. Examples of PDE solutions obtained by PINN during training.

As shown, the vanilla PINN (black) can only solve for a short time domain ($t \in [0, 1]$). Temporal weights (red) helped mitigate propagation failure over longer time domains, while spatial weights (blue) further accelerated convergence.

To understand why spatial-temporal adaptive training works, we further observed the PINN outputs during the training process. Figure 3 provides an example of the waveform of a PDE solution output by PINN (training on $t \in [0, 5]$) during training. As shown, the propagation of the trivial solution trumped the propagation of the correct solution in the vanilla PINN. Temporal weights were able to guide the correct solution to propagate. The addition of spatial weights was able to further speed up this process.

Figure 4 provides examples of how the correct solution, PINN solutions, and spatial weights were propagated over space. As shown, the spatial weights were able to track the "wavefront" of the TMP activation – a sparse set of locations where high gradients exist both in space and time. This succeeded in guiding the optimizer to focus more on these regions, thus accelerating the convergence.

The Effectiveness of Sequential Training: We then compared the computation cost and the accuracy of the PDE solution achieved by the presented PINNs with and without sequence learning, across different lengths of time domains as summarized in Table 1. As shown, the sequential training was able to achieve a 1.3–1.8× speedup with increased accuracy. Also note that without sequence

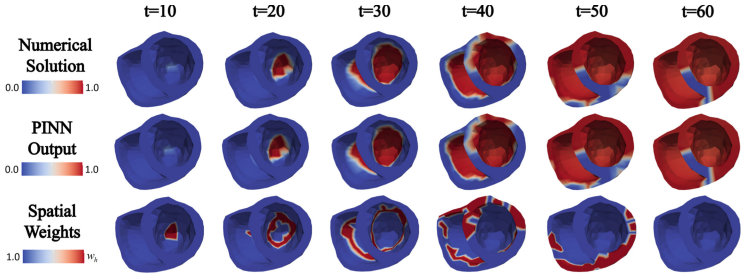

Fig. 4. Propagation of the correct solutions, PINN output, and spatial weights

Table 1. Comparison of time cost and accuracy with and without sequence learning

	Without Sequential Learning		With Sequential Learning	
Time Domain	Time Consuming	MSE Error	Time Consuming	MSE Error
$t \in [0, 2]$	5 min	1.80E−08	4 min	5.23E−09
$t \in [0, 5]$	14 min	6.77E−06	9 min	3.08E−08
$t \in [0, 7]$	24 min	7.17E−05	13 min	2.21E−07
$t \in [0, 10]$	26 min	6.18E−04	20 min	9.59E−07
$t \in [0, 20]$	59 min	1.39E−02	41 min	9.51E−06

learning, the MSE error increases rapidly as the time domain increases, making it difficult to scale to longer time domains. In comparison, sequence learning enables the PDE solution (training enough time) to preserve a low MSE error($= 1.02 \times 10^{-5}$) throughout the complete ventricular activation ($t \in [0, 60]$), enabling simulation of the complete ventricular activation process as shown in Fig. 4 that is otherwise not possible with the other ablation models.

PINN for Supporting PDE Parameter Inference: Finally, we tested the feasibility of the presented PINNs to support parameter estimation of the AP model. We assumed measurements of TMP solutions to be available and considered unknown parameter γ in the AP model (Eq. 1) due to its relatively large influence on the PDE solution [7]. We considered spatially varying values of γ representing infarct tissues, and tested joint optimization of the PINN and the unknown γ for six different cases. The estimated parameter, as shown in Fig. 5, exhibited small MSEs in comparison to the ground truth.

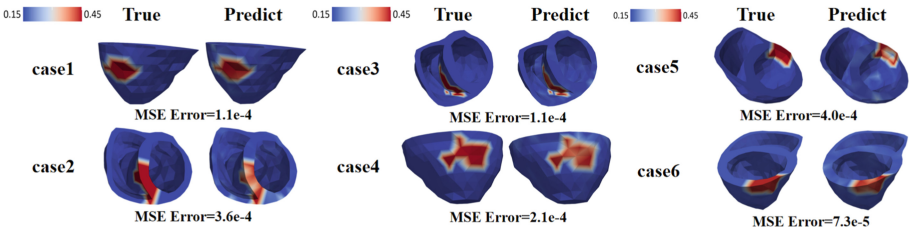

Fig. 5. True parameter γ and PINN predicted parameter $\hat{\gamma}$

5 Conclusion

We presented a spatial-temporally adaptive PINN framework to solve the forward and inverse problems of cardiac EP over 3D bi-ventricular models, overcoming the current limitations of PINNs in solving PDEs with sharp gradients/transitions over tricky spatial domains and long time domains. However, our PINN framework still has a much larger computational cost than traditional numerical methods, which is a drawback of PINN itself. Future works will pursue the use of this PINN framework for more complex PDEs of bi-ventricular EP, improve its computational efficiency as well as enable inverse EP using surface measurements such as electrocardiograms (ECGs) in real-data settings.

Acknowledgements. This work is supported in part by the National Key Research and Development Program of China(No: 2020AAA0109502); the National Natural Science Foundation of China (No: U1809204); the Talent Program of Zhejiang Province (No: 2021R51004); NIH/NHLBI under Award Numbers R01HL145590; and NSF grants OAC-2212548.

References

1. Aliev, R.R., Panfilov, A.V.: A simple two-variable model of cardiac excitation. Chaos Solitons Fract. **7**(3), 293–301 (1996)
2. Arevalo, H.J., et al.: Arrhythmia risk stratification of patients after myocardial infarction using personalized heart models. Nat. Commun. **7**(1), 11437 (2016)
3. Bu, J., Karpatne, A.: Quadratic residual networks: a new class of neural networks for solving forward and inverse problems in physics involving PDEs. In: Proceedings of the 2021 SIAM International Conference on Data Mining (SDM), pp. 675–683. SIAM (2021)
4. Chen, Y., Lu, L., Karniadakis, G.E., Dal Negro, L.: Physics-informed neural networks for inverse problems in nano-optics and metamaterials. Opt. Express **28**(8), 11618–11633 (2020)
5. Clayton, R., et al.: Models of cardiac tissue electrophysiology: progress, challenges and open questions. Prog. Biophys. Mol. Biol. **104**(1–3), 22–48 (2011)
6. Daw, A., Bu, J., Wang, S., Perdikaris, P., Karpatne, A.: Rethinking the importance of sampling in physics-informed neural networks. arXiv preprint arXiv:2207.02338 (2022)

7. Dhamala, J., et al.: Embedding high-dimensional Bayesian optimization via generative modeling: parameter personalization of cardiac electrophysiological models. Med. Image Anal. **62**, 101670 (2020)
8. Hao, Z., et al.: Physics-informed machine learning: a survey on problems, methods and applications. arXiv preprint arXiv:2211.08064 (2022)
9. Herrero Martin, C., et al.: Ep-pinns: cardiac electrophysiology characterisation using physics-informed neural networks. Front. Cardiovasc. Med. **8**, 2179 (2022)
10. Jagtap, A.D., Shin, Y., Kawaguchi, K., Karniadakis, G.E.: Deep Kronecker neural networks: a general framework for neural networks with adaptive activation functions. Neurocomputing **468**, 165–180 (2022)
11. Jin, X., Cai, S., Li, H., Karniadakis, G.E.: NSFnets (Navier-Stokes flow nets): physics-informed neural networks for the incompressible Navier-Stokes equations. J. Comput. Phys. **426**, 109951 (2021)
12. Karniadakis, G.E., Kevrekidis, I.G., Lu, L., Perdikaris, P., Wang, S., Yang, L.: Physics-informed machine learning. Nat. Rev. Phys. **3**(6), 422–440 (2021)
13. Kissas, G., Yang, Y., Hwuang, E., Witschey, W.R., Detre, J.A., 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. Methods Appl. Mech. Eng. **358**, 112623 (2020)
14. Krishnapriyan, A., Gholami, A., Zhe, S., Kirby, R., Mahoney, M.W.: Characterizing possible failure modes in physics-informed neural networks. In: Advances in Neural Information Processing Systems, vol. 34, pp. 26548–26560 (2021)
15. Mathews, A., Francisquez, M., Hughes, J.W., Hatch, D.R., Zhu, B., Rogers, B.N.: Uncovering turbulent plasma dynamics via deep learning from partial observations. Phys. Rev. E **104**(2), 025205 (2021)
16. Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: a deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. J. Comput. Phys. **378**, 686–707 (2019)
17. Sahli Costabal, F., Yang, Y., Perdikaris, P., Hurtado, D.E., Kuhl, E.: Physics-informed neural networks for cardiac activation mapping. Front. Phys. **8**, 42 (2020)
18. Sermesant, M., et al.: Patient-specific electromechanical models of the heart for the prediction of pacing acute effects in CRT: a preliminary clinical validation. Med. Image Anal. **16**(1), 201–215 (2012)
19. Wang, L., Zhang, H., Wong, K.C., Liu, H., Shi, P.: Physiological-model-constrained noninvasive reconstruction of volumetric myocardial transmembrane potentials. IEEE Trans. Biomed. Eng. **57**(2), 296–315 (2009)
20. Wang, S., Sankaran, S., Perdikaris, P.: Respecting causality is all you need for training physics-informed neural networks. arXiv abs/2203.07404 (2022)
21. Wang, S., Teng, Y., Perdikaris, P.: Understanding and mitigating gradient flow pathologies in physics-informed neural networks. SIAM J. Sci. Comput. **43**(5), A3055–A3081 (2021)
22. Wang, S., Yu, X., Perdikaris, P.: When and why PINNs fail to train: a neural tangent kernel perspective. J. Comput. Phys. **449**, 110768 (2022)
23. Xu, K., Darve, E.: Physics constrained learning for data-driven inverse modeling from sparse observations. J. Comput. Phys. **453**, 110938 (2022)
24. Zhang, H., Shi, P.: A meshfree method for solving cardiac electrical propagation. In: 2005 IEEE Engineering in Medicine and Biology 27th Annual Conference, pp. 349–352. IEEE (2006)