EP-Net: Learning Cardiac Electrophysiology Models for Physiology-based Constraints in Data-Driven Predictions

Ibrahim Ayed, Nicolas Cedilnik, Patrick Gallinari, Maxime Sermesant

▶ **To cite this version:**

HAL Id: hal-02106618
https://inria.hal.science/hal-02106618

Submitted on 23 Apr 2019

EP-Net: Learning Cardiac Electrophysiology Models for Physiology-based Constraints in Data-Driven Predictions

Ibrahim Ayed[1], Nicolas Cedilnik[2,3], Patrick Gallinari[1], and Maxime Sermesant[2]

[1] Sorbonne University, LIP6, Paris
[2] Université Côte d'Azur, Epione Research Project, Inria, Sophia Antipolis
[3] Liryc Institute, Bordeaux

Abstract. Cardiac electrophysiology (EP) models achieved good progress in simulating cardiac electrical activity. However numerical issues and computational times hamper clinical applicability of such models. Moreover, personalisation can still be challenging and model errors can be difficult to overcome. On the other hand, deep learning methods achieved impressive results but suffer from robustness issues in healthcare due to their lack of physiological knowledge. We propose a novel approach which is based on deep learning in order to replace numerical integration of partial differential equations. This has the advantage to directly learn spatio-temporal correlations, which increases stability. Moreover, once trained, solutions are very fast to compute. We present first results in state estimation based on few measurements and evaluate the forecasting power of the trained network. The proposed method performed very well on this preliminary evaluation. It opens up possibilities towards data-driven personalisation, to overcome model error by learning from the data.

Keywords: Electrophysiology · Deep learning · Simulation

1 Introduction

Mathematical modelling of the cardiac cell has been an active research area for the last decades. Cardiac electrophysiology models can accurately reproduce cardiac cells electrical behaviour. This provides a mathematical framework to simulate cardiac activity, however it remains challenging to achieve in 3D due to numerical issues and computational time. These difficulties become all the more apparent when trying to personalise such model by matching patient data. This matching can also be hampered by the approximations of the model.

The last decade has seen huge progress in data-driven approaches, with deep learning achieving impressive results in numerous tasks. It has been particularly successful in image and signal processing, where spatio-temporal correlations can be learned. However it suffers from robustness issues in healthcare due to its lack of physiological knowledge. More recently, physics-based learning proposed to

use machine learning in order to solve physics equations [9,7,5]. This has the potential to alleviate some numerical and computational time issues, and also to provide a framework to overcome model error. Additionally, it is a way to introduce physics-based prior knowledge in learning methods.

In this manuscript, we present a method to learn the dynamics from a database of cardiac electrophysiology simulations and use the trained network to forecast the behaviour of unseen simulations. A state estimation method is coupled with a forecasting network to produce predictions for a few iterations.

These preliminary results are promising in terms of learning the dynamical behaviour of cardiac electrophysiology models.
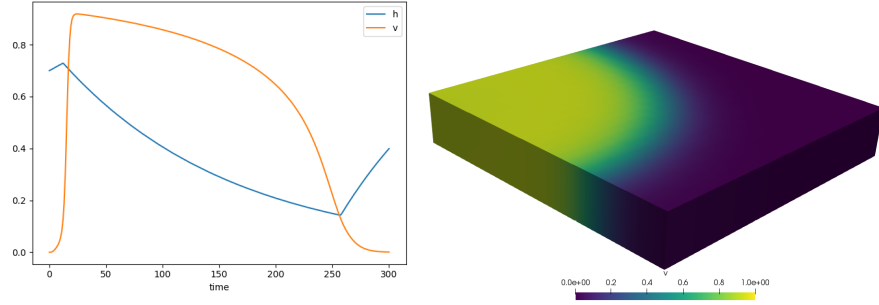
2   Electrophysiological Modelling

Fig. 1. Cardiac electrophysiology model simulation. (Left) v and h values along time for a given point of the domain, (Right) spatial propagation of v at a given time point.

Cardiac electrophysiology modelling is a very active research area, with a large variety in terms of model complexity. In this manuscript, we used the Mitchell-Schaeffer model [8] which is a two variables model that has been successfully used in patient-specific modelling [11]. The variable v represents the transmembrane potential while the "gating" variable h controls the repolarisation:

$$\partial_t v = \Delta v + \frac{h v^2 (1 - v)}{\tau_{in}} - \frac{v}{\tau_{out}} + J_{stim} \tag{1}$$

$$\partial_t h = \begin{cases} \frac{1-v}{\tau_{open}} & \text{if } v < v_{gate} \\ \frac{v}{\tau_{close}} & \text{if } v > v_{gate} \end{cases} \tag{2}$$

In order to use convolutional neural networks, it is much more efficient to work on a Cartesian grid. To this end, we used a Lattice Boltzmann method to solve the EP model [10].

The domain is a slab of cardiac tissue of size $25 \times 25 \times 5 \ mm^3$, which we will denote Ω, discretised in voxels of $1 \ mm^3$. To initiate the propagation (J_{stim}), a current of 1 normalised transmembrane potential unit was applied on a single voxel for 10 ms. The Mitchell-Schaeffer model parameters are taken from the original paper, except h_0 which was set to 0.7. The simulation was conducted for $300 \ ms$, and stored every ms with a discrete time step of $0.1 \ ms$. The simulation database was created by stimulating the domain from every voxel. We therefore have 3 125 simulations in the database. Let V and H be the complete solutions in space and time of these equations.

3 Learning Method

In this section, we reformulate the problem we want to solve with learning and outline the features of the learning model we used.

3.1 Formulation

Combining equations (1) and (2), the studied dynamical system can be written in the form:

$$\frac{dX_t}{dt} = F(X_t) \tag{3}$$

where X is a spatio-temporal three-dimensional vector field over the domain $\Omega \subset \mathbb{R}^3$. In this particular case, we have:

$$X = \begin{pmatrix} V \\ H \end{pmatrix}$$

In other words, for any given time t and point of the domain $x \in \Omega$, we have $X_t(x)$, a two-dimensional vector.

In practice, while the value of V can be measured, H is a hidden variable which is difficult to estimate. We model this fact by adding to our system an observation operator $\mathcal{H}$ defined in our case by:

$$\mathcal{H}(X) = V \tag{4}$$

which makes our model a partially observed one, as only some parts of the full state can be directly measured.

It is important to note that $\mathcal{H}$ can be used to model more general observation operators which can be any transformation of the state. Obviously, the more information is lost through the observation process, the harder it will be to reconstruct the dynamics of the system. Moreover, for practical purposes, we also require $\mathcal{H}$ to be differentiable. This might seem as a strong hypothesis but, in practice, the operators which are used can generally be smoothed without a significant departure from reality.

The general system that we model can thus be written:

$$\begin{cases} X_0 \\ \dfrac{dX_t}{dt} = F(X_t) \\ \quad Y_t = \mathcal{H}(X_t) \end{cases} \tag{5}$$

3.2   Approach

In the system (5), starting from observations Y, we need to estimate both the initial state and the dynamics it should follow. Obviously, from one observation alone, it is often impossible to estimate the full state but hopefully this becomes possible for most operators $\mathcal{H}$ when a long enough sequence of observations is used.

Our idea is to use an operator g_θ to estimate X_0 from a sequence of past observations $Y_{-k} = (Y_{-k+1}, ..., Y_0)$ of length k and a second operator F_θ to model the ODE governing the dynamics of X. Thus, θ is a variable summarizing the parameterization of both the dynamics and the initial state.

This means that the system can be written as:

$$\begin{cases} X_0 = g_\theta(Y_{-k}) \\ \dfrac{dX_t}{dt} = F_\theta(X_t) \\ \quad Y_t = \mathcal{H}(X_t) \end{cases} \tag{6}$$

We assume that F and g are parameterized so that the above system has a unique solution which will be denoted X^θ.

Defining a cost functional:

$$\mathcal{J}(Y, \widetilde{Y}) = \int_0^T \|Y_t - \widetilde{Y}_t\|^2 \mathrm{dt} \tag{7}$$

we can then frame the statistical learning problem as:

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & \mathbb{E}_{Y \in \text{Dataset}} \mathcal{J}(Y, \mathcal{H}(X)) \\ \text{subject to} \quad & \frac{dX_t}{dt} = F_\theta(X_t), \\ & X_0 = g_\theta(Y_{-k}) \end{aligned} \tag{8}$$

This optimization problem is generally a difficult one and cannot be solved exactly for most choices of parameterization families for F and g. This makes necessary the use of gradient descent algorithms for which it is necessary to estimate the derivative $w.r.t.$ θ of $\mathbb{E}_{Y \in \text{Dataset}} \left[\mathcal{J}(Y, \mathcal{X}^\theta) \right]$ which justifies the use of differentiable parameterizations.

In our case, starting from a random initialization of θ, we follow the steps:

1. Solve the forward state equation (6) to find X^θ with an explicit differentiable solver;
2. Get the gradient of $\theta \to \mathbb{E}_{Y \in \text{Dataset}} \left[\mathcal{J}(Y, \mathcal{X}^\theta) \right]$ with automatic differentiation tools;
3. Update θ in the steepest descent direction.

In our case, we parameterize g and F as neural networks so that the corresponding automatic differentiation tools can be used. The simplest example for explicit solvers is a Euler scheme but more complicated numerical methods can also be used. As shown in [1], this framework can also be extended to the modelling of stiff equations by using implicit solvers through the more general adjoint method.

4   Experiments

In this section we present the experiments we undertook. Even though we are still at a preliminary stage regarding the choice of architectures and hyperparameters, the results are already encouraging.
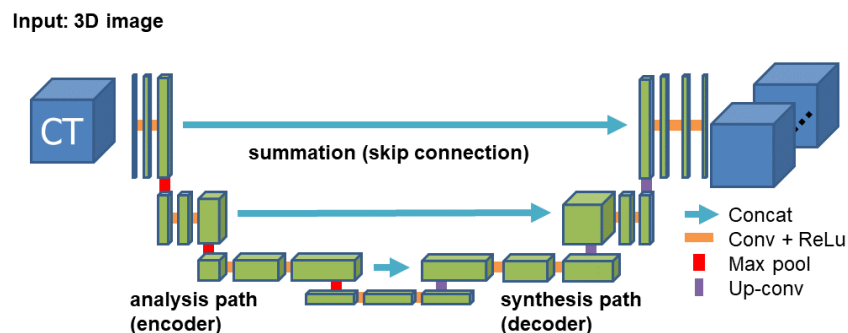
4.1   Implementation details

Fig. 2. Figure illustrating the general form of the U-Net architecture. Illustration taken with a slight modification from [12].

Operator g is implemented as a three-dimensional U-Net inspired from [3] with 25 filters at the initial stage, figure 2 shows the general structure of this architecture. F is implemented as a Residual Network [4] with three-dimensional convolutions, two downsampling initial layers, three intermediary blocks (figure 3 shows the structure of one) and an inner dimension of 24. In order to solve the forward equation, we use an explicit Euler scheme.

To construct a training dataset, we have randomly selected 500 simulations, temporally downsampled three times. A validation set of 300 simulations and a
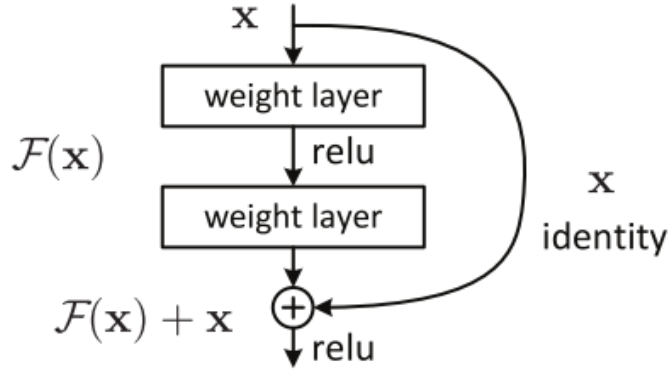
Fig. 3. Figure showing a typical residual block for the ResNet. Illustration taken from [4].

Table 1. Relative mean-squared error of normalised transmembrane potential per time-step for different forecasting horizons.

Time horizon K	2 (6 ms)	5 (15 ms)	8 (24 ms)	11 (33 ms)	14 (42 ms)	17 (51 ms)
Mean Squared Error	0.005	0.012	0.048	0.060	0.14	0.58

test set of 200 were also randomly selected from the remaining simulations and downsampled likewise. In particular, this means that test and training simulations have different initial conditions so that the model is tested with data it has never seen. Only the V variable was used and we have taken $k = 4$ and a training horizon of 8 time-steps.

The optimization over θ uses the ADAM optimizer [6] with a learning rate of 10^{-3}. We also use exponential scheduled sampling [2] with parameter 0.9999 during training and start with a reweighted orthogonal initialization for the parameters of F.

4.2   Results

We present here results on the forecast over 8 time frames after assimilating the first 4 frames (see Fig. 4). We can observe very good agreement with the ground truth on this forecast, which represents an important part of cardiac dynamics within this virtual slab of tissue, from early depolarisation to full depolarisation.

Table 1 shows MSE results for our algorithm for different forecasting horizons: for a horizon K, this means that we feed the model with three initial measurements then let it predict K steps forward without any additional information. We can see that up to a reasonable number of steps, here corresponding to 51 ms, our model does reasonably well which is very encouraging given that we are still at an
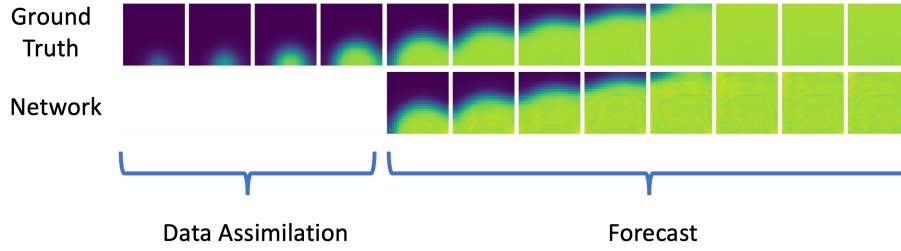
Fig. 4. Transmembrane potential ground-truth (top) and forecasted (bottom) for one slice of the tissue slab.

early stage of experimenting with it. Figure 4 visually confirms those numerical results.

It has to be noted that for such cardiac model, predicting the propagation between time 10 and 60 ms on such a slab of tissue represents the whole depolarisation wave, which is the most important part of cardiac electrophysiology dynamics.

Moreover, it is important to notice that, while the model takes a few hours to train, once this is done the inference is very quick and does not require any recalibration.

5   Discussion

We have presented a learning model able to learn the considered dynamics and thus showing promise regarding the automated learning of cardiac electrophysiology models. Moreover, while we only presented results which are completely unsupervised regarding the full state of the studied system, which makes us unable to reconstruct H for example, it is possible with our approach to add in stronger physical priors by pre-training the model over fully observed states, adding some information about the equation into F. The model we present is very versatile and can thus be used with different types of dynamics, degrees of supervision... which could open the way for many other applications where quick simulation is needed.

However, this is still a first step and we need to experiment with our model in more challenging settings *i.e.*, with a more realistic $\mathcal{H}$ which is not a simple and dense linear projection in real-world applications. Another shortcoming of our model is the fact that uncertainty is not taken into account, at input as well as at output level, while it is an important feature of many real-world applications especially in healthcare. This should be an essential direction of research for our future work. Finally, it is also important to note that, while Mean Squared Error was chosen for training as well as for testing, it is not necessarily the best metric for all applications: depending on the use case, it might be useful to design a different one, more adapted to the problem.

6   Conclusion

We proposed in this manuscript an approach to learn the dynamics of a cardiac electrophysiology model along with a data assimilation procedure, so that limited measurements can be used to estimate an initial state and predict the future evolution of the electrophysiology model.

In clinical practice, it is challenging to completely map cardiac activation, especially in case of arrhythmia. This method could be used to complete in space and time limited clinical measurements.

Such coupling between simulation and learning opens up many possibilities, especially in order to make such approach more clinically applicable and patient-specific. It has the potential to tackle both computational as well as robustness issues that appear when using computer models or machine learning separately.

References

1. Ayed, I., de Bezenac, E., Pajot, A., Brajard, J., Gallinari, P.: Learning dynamical systems from partial observations (2019), `http://arxiv.org/abs/1902.11136`
2. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled sampling for sequence prediction with recurrent neural networks. Advances in Neural Information Processing Systems 28 (NIPS 2015) (2015)
3. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: Learning dense volumetric segmentation from sparse annotation. Proceedings of MICCAI 2016
4. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2015)
5. Herzog, S., Wörgötter, F., Parlitz, U.: Data-Driven Modeling and Prediction of Complex Spatio-Temporal Dynamics in Excitable Media. Frontiers in Applied Mathematics and Statistics **4** (2018). https://doi.org/10.3389/fams.2018.00060
6. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. Proceedings of International Conference for Learning Representations, San Diego, 2015
7. Long, Z., Lu, Y., Ma, X., Dong, B.: PDE-Net: Learning PDEs from Data. In: International Conference on Machine Learning. pp. 3208–3216 (Jul 2018), `http://proceedings.mlr.press/v80/long18a.html`
8. Mitchell, C.C., Schaeffer, D.G.: A two-current model for the dynamics of cardiac membrane. Bulletin of mathematical biology **65**(5), 767–793 (Sep 2003)
9. Raissi, M., Karniadakis, G.E.: Hidden physics models: Machine learning of nonlinear partial differential equations. Journal of Computational Physics **357**, 125–141 (Mar 2018). https://doi.org/10.1016/j.jcp.2017.11.039, `http://www.sciencedirect.com/science/article/pii/S0021999117309014`
10. Rapaka, S., Mansi, T., Georgescu, B., Pop, M., Wright, G., Kamen, A., Comaniciu, D.: LBM-EP: Lattice-Boltzmann method for fast cardiac electrophysiology simulation from 3d images. Medical Image Computing and Computer-Assisted Intervention–MICCAI 2012 pp. 33–40 (2012)
11. Relan, J., Chinchapatnam, P., Sermesant, M., Rhode, K., Ginks, M., Delingette, H., Rinaldi, C.A., Razavi, R., Ayache, N.: Coupled personalization of cardiac electrophysiology models for prediction of ischaemic ventricular tachycardia. Interface Focus **1**(3), 396–407 (2011)

12. Roth, H.R., Shen, C., Oda, H., Oda, M., Hayashi, Y., Misawa, K., Mori, K.: Deep learning and its application to medical image segmentation. Medical Imaging Technology, Volume 36 (2018), Issue 2, p. 63-71 (2018)