



## Accelerated simulation of a neuronal population via mathematical model order reduction

### Citation

Lehtimäki, M., Seppälä, I., Paunonen, L., & Linne, M-L. (2020). Accelerated simulation of a neuronal population via mathematical model order reduction. In *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)* (pp. 118-122). IEEE. <https://doi.org/10.1109/AICAS48895.2020.9073844>

### Year

2020

### Version

Peer reviewed version (post-print)

### Link to publication

[TUTCRIS Portal \(http://www.tut.fi/tutcris\)](http://www.tut.fi/tutcris)

### Published in

2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)

### DOI

[10.1109/AICAS48895.2020.9073844](https://doi.org/10.1109/AICAS48895.2020.9073844)

### Copyright

This publication is copyrighted. You may download, display and print it for Your own personal use. Commercial use is prohibited.

### Take down policy

If you believe that this document breaches copyright, please contact [cris.tau@tuni.fi](mailto:cris.tau@tuni.fi), and we will remove access to the work immediately and investigate your claim.

# Accelerated Simulation of a Neuronal Population via Mathematical Model Order Reduction

Mikko Lehtimäki\*, Ippa Seppälä\*, Lassi Paunonen†, Marja-Leena Linne\*

\*Faculty of Medicine and Health Technology

†Faculty of Information Technology and Communication Sciences  
Tampere University, Tampere, Finland.

Email: mikko.lehtimaki@tuni.fi, ippa.seppala@tuni.fi, lassi.paunonen@tuni.fi, marja-leena.linne@tuni.fi

**Abstract**—Mathematical modeling of biological neuronal networks is important in order to increase understanding of the brain and develop systems capable of brain-like learning. While mathematical analysis of these comprehensive, stochastic, and complex models is intractable, and their numerical simulation is very resource intensive, mean-field modeling is an effective tool in enabling the analysis of these models. The mean-field approach allows the study of populations of biophysically detailed neurons with some assumptions of the mean behaviour of the population, but ultimately requires numerical solving of high-dimensional differential equation systems. Mathematical model order reduction methods can be employed to accelerate the analysis of high-dimensional nonlinear models with a purely software-based approach. Here we compare state-of-the-art methods for improving the simulation time of a neuronal mean-field model and show that a nonlinear Fokker-Planck-McKean-Vlasov model can be accurately approximated in low-dimensional subspaces with these methods. Using Proper Orthogonal Decomposition and different variations of the Discrete Empirical Interpolation Method, we improved the simulation time by over three orders of magnitude while achieving low approximation error.

## I. INTRODUCTION

The human brain possesses unmatched information processing and generalization capabilities. Hence, machine learning has drawn inspiration from neuroscience and understanding the mechanisms of learning in biological neuronal networks of the brain continues to be of utmost importance for developing efficient and effective machine learning algorithms [1]. Some interesting properties of these networks are their natural capability to process data with a temporal dimension (see Reservoir Computing) [2], use of an energy efficient continuous computation paradigm which can be imitated with neuromorphic hardware [3], [4], and diverse synaptic plasticity rules [5]. Utilizing these features together with brain-like computation units could result in improved performance in classification and regression tasks [6]. Indeed, by understanding how the brain implements biological intelligence, progress can also be made in deep learning [7], [8]. To that end, insight into biological neuronal networks can be gained with mean-field methods [9]. Here, we show how the simulation of a computationally expensive mean-field model can be accelerated with mathematical model order reduction (MOR) methods [10].

Biological systems are naturally noisy and rarely behave identically between repeated measurements [11]. This stochasticity, be it intrinsic or external, is believed to serve a

purpose such as enhancing detection of weak signals [12]. The effects of noise in neuronal signaling have been studied with stochastic models of neuronal systems in the molecular [13], single cell [14] and network [15] levels. However, the added biological relevance achieved with stochastic modeling comes with the burden of mathematical intractability and increased computation time, as specialized numerical methods are needed for simulating stochastic systems and analysis of system dynamics relies on Monte Carlo methods. Understanding network plasticity in the presence of this randomness is one step towards more brain-like machine learning algorithms.

The dynamics of stochastic neuronal network circuits can be studied with mean-field models that use deterministic descriptions of the underlying network [9]. The mean-field framework encompasses multiple methods that result in models of different levels of complexity, ranging from simple firing rate equations to probability density functions to models with multiple spatial dimensions. In this work, we will study a Fokker-Planck-McKean-Vlasov-type mean-field model that describes the time evolution of the probability density function of the state of a large neuronal population of stochastic FitzHugh-Nagumo (FN) neurons [16]–[18]. The model is mathematically intractable, and studying it requires numerical solving methods.

We show that numerical simulation of a mean-field model can be made significantly faster by employing *reduced order models*, created with mathematical MOR methods. MOR methods require no simplifications of the modeled system and allow every variable therein to be reconstructed at any time. When choosing MOR methods appropriately, no linearization is required, which allows more dynamics to be retained in the reduced model [19]. Hence, these methods can be applied directly to nonlinear systems, which in the field of computational neuroscience is highly advantageous. Moreover, the chosen MOR methods allow approximating linear and nonlinear components independently, in order to approximate the original system accurately.

In Section II, the model we studied is introduced together with mean-field theory, and the MOR methods employed here are described. In Section III, numerical simulation results of the original and reduced models are shown. Finally, in Section IV the significance of our results is discussed together with directions for future research.

## II. METHODS

Mean-field approximation was originally used to describe the spin of electrons in theoretical physics. In the field of computational neuroscience the method can be used to model the behaviour of populations of neurons [9]. The single neuron model, which will be taken to the mean-field limit, is in itself stochastic. These neurons can prove to be challenging to incorporate into models, and are often preprocessed by introducing tractable randomness in the form of Markov chains. In such a Markov chain the transition probabilities of neuronal states obey the *master equation*, from which the *Fokker-Planck equation* can be derived. In general, the mean-field approximation consists of dividing neurons into statistically similar populations, in which the population behaviour is uncorrelated. This is true when the population size is theoretically infinite.

In this work we examine a mean-field model derived in [16], representing a population of FN neurons. This second-order nonlinear partial differential equation (PDE) has three independent variables and describes the time evolution of the probability density function  $p(t, V, W, Y)$  of the state of the neuron population. It gives a deterministic description of the underlying stochastic system. The model is

$$\begin{aligned}
& \partial_t p(t, V, W, Y) \\
&= \frac{1}{2} \sigma_J^2 \bar{y}^2(t) \frac{\partial^2}{\partial V^2} \left[ (V - V_{rev})^2 p(t, V, W, Y) \right] \\
&+ \frac{1}{2} \frac{\partial^2}{\partial Y^2} \left[ \sigma_Y^2 (V, Y) p(t, V, W, Y) \right] \\
&+ \frac{1}{2} \sigma_{ext}^2 \frac{\partial^2}{\partial V^2} \left[ p(t, V, W, Y) \right] \\
&- \frac{\partial}{\partial V} \left[ \left( V - \frac{V^3}{3} - W + I_{ext} + \bar{J}(V - V_{rev}) \bar{y}(t) \right) \right. \\
&\quad \left. \times p(t, V, W, Y) \right] \\
&- \frac{\partial}{\partial W} \left[ a(V + b - cW) p(t, V, W, Y) \right] \\
&- \frac{\partial}{\partial Y} \left[ \left( \alpha_r S(V)(1 - Y) - \alpha_d Y \right) p(t, V, W, Y) \right], \tag{1}
\end{aligned}$$

where  $\bar{y}(t) = \iiint yp(t, v, w, y) dv dw dy$ ,  $V$  is the neuronal membrane voltage,  $W$  is the recovery variable of the FN model,  $Y$  is the synaptic conductance of the neurons in this population, and  $I_{ext}$  is external current stimulus. For additional details of the model, see [16].

Equation (1) must be discretized in space prior to numerical simulation. The discretization results in a system of ordinary differential equations of dimension (number of equations to solve)  $\nu^3$  with  $\nu$  being the number of discretization points in one variable of the PDE, assuming an equal amount of discretization points in every variable. A fine discretization grid is required so that fast dynamics of the model are captured, making  $\nu$  a large integer.

Spatial discretization of Equation (1) is carried out with a fourth-order central difference scheme and the triple integral

$\bar{y}(t)$  is evaluated with the Newton-Cotes method of order six. After discretization, we write the system in state-space format

$$x'(t) = Ax(t) + f(x(t)), \tag{2}$$

where  $x \in \mathbb{R}^n$  is the current state of the system,  $A \in \mathbb{R}^{n \times n}$  is the state matrix with linear coefficients,  $f(x(t)) \in \mathbb{R}^n$  is a vector of nonlinear functions,  $n = \nu^3$ , and  $\nu = 50$ . For numerical simulations we use parameters from [16].

We construct reduced order models (ROMs) with the Discrete Empirical Interpolation Method (DEIM) [20] and two of its advanced variants, namely LDEIM and QDEIM. DEIM is a MOR method that is used in conjunction with the Proper Orthogonal Decomposition (POD) [21] and is based on the method from [19]. These methods are applicable to general nonlinear systems such as the model used here.

POD is a projection based MOR method that approximates the original system of dimension  $n$  in a reduced linear subspace. A reduced basis with orthonormal column vectors  $V_k \in \mathbb{R}^{n \times k}$  where  $k < n$  is computed using singular value decomposition (SVD). This POD basis is constructed from snapshots  $Y = [y_1, y_2, \dots, y_s]$  that are a set of solutions to the original system [22], collected for example with numerical simulation. Then, a reduced state vector  $V_k^T x(t) = \tilde{x}(t) \in \mathbb{R}^k$  is obtained by a linear transformation. Projecting the system described in Equation (2) onto  $V_k$  by Galerkin projection results in a reduced system

$$\tilde{x}'(t) = \underbrace{V_k^T A V_k}_{\tilde{A}} \tilde{x}(t) + V_k^T f(V_k \tilde{x}(t)) \tag{3}$$

where  $\tilde{A}$  can be precomputed before the online (simulation) phase. At any point, an approximation of the original, full-dimensional state vector can be computed with  $x(t) \approx V_k \tilde{x}(t)$ .

However, while POD itself can be applied to nonlinear systems, there is no guarantee of computational savings as the nonlinear part  $f(V_k \tilde{x}(t))$  of the reduced system must be evaluated in the original space. Efficient evaluation of the nonlinear term can be achieved with DEIM [19], [20]. DEIM extends the subspace projection approach with an interpolation step for nonlinear functions. To construct an approximation of the nonlinear term, the algorithm gives

$$\tilde{f}(x, t) \approx U_m (P_m^T U_m)^{-1} P_m^T f(x, t), \tag{4}$$

where the DEIM basis  $U_m = [u_1, u_2, \dots, u_m]$ ,  $m < n$  is computed via SVD of the snapshots of the nonlinear function outputs,  $P_m^T f(x, t) := f_m(x, t)$  is a nonlinear function with  $m$  components chosen from  $f$  according to DEIM determined interpolation points  $p_1, \dots, p_m$  and  $P_m = [e_{p_1}, e_{p_2}, \dots, e_{p_m}]$  with  $e_{p_i}$  being the standard basis vector  $i$  of  $\mathbb{R}^n$ . Together POD and DEIM form a ROM

$$\tilde{x}'(t) = \tilde{A} \tilde{x}(t) + \underbrace{V_k^T U_m (P_m^T U_m)^{-1}}_N f_m(V_k \tilde{x}(t)), \tag{5}$$

where  $N \in \mathbb{R}^{n \times m}$  can be precomputed in the offline phase. Thus in the online phase only  $m$  nonlinear functions are evaluated using  $N f_m(V_k \tilde{x}(t))$ . Note that the dimension  $k$  of

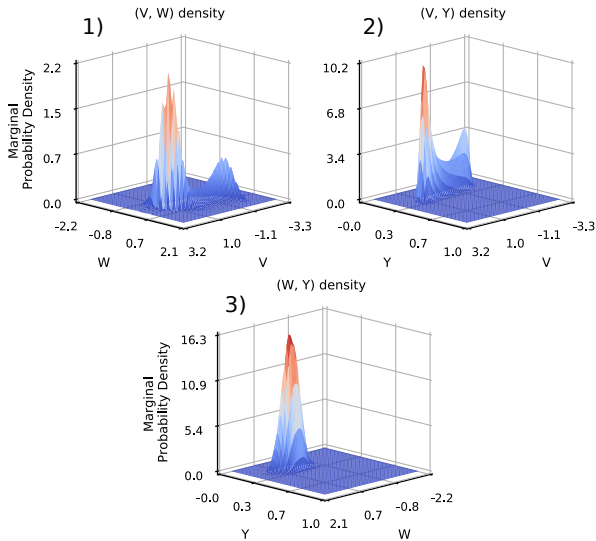


Fig. 1. Original model simulated for  $t = 2.2$  s. The probability density function is integrated over each of the three independent variables, Y, W and V (plots 1) to 3) respectively) for visualization purposes.

the linear component of the reduced system does not need to equal the dimension  $m$  of the nonlinear component.

Another algorithm we use to reduce the mean-field model is QDEIM [23]. In DEIM, the computed basis functions  $U_m$  and interpolation points  $P_m$  depend on the sequence in which snapshots were collected. The idea of QDEIM is to find  $P_m$  independent of this specific sequence, and to compute a more numerically stable basis  $M_m$  that replaces  $U_m$ .  $P_m$  and  $M_m$  are obtained with QR factorization of column pivoted  $U_m^*$ . Efficient implementations of these steps are readily available in high performance computing software packages.

Additionally, we reduce the model with Localized Discrete Empirical Interpolation Method (LDEIM) [24]. In LDEIM, a clustering algorithm is employed in the offline phase to group solution snapshots before DEIM basis generation. Several basis and interpolation point pairs  $(U_{m_1}, P_{m_1}), \dots, (U_{m_p}, P_{m_p})$  are computed to obtain a set of local bases, one from each cluster of snapshots, in contrast to the global basis used in DEIM. In the online phase a local precomputed DEIM basis is chosen with nearest neighbor classification. LDEIM requires the number of clusters and features as user defined parameters, and the size of the feature vector is a decision between classification power and computational efficiency. The premise of LDEIM is to use multiple smaller yet accurate reduced subspaces to compensate for the extra online computation time that is needed for basis selection.

In general, hardware requirements of our approach are decided by the model that is reduced. The reduction algorithms need additional memory proportionally to  $n$ ,  $m$  and  $k$ . To simulate the models we use four Intel Xeon E5-2680 v3 cores and 350GB RAM. The original model requires a considerable amount of memory in the state space format as there are  $n^2$ ,  $n = \nu^3$  floats in  $A$ , here  $\sim 125$ Gb for  $A$ . However, we did not exploit the sparsity of  $A$  to reduce memory load, thus

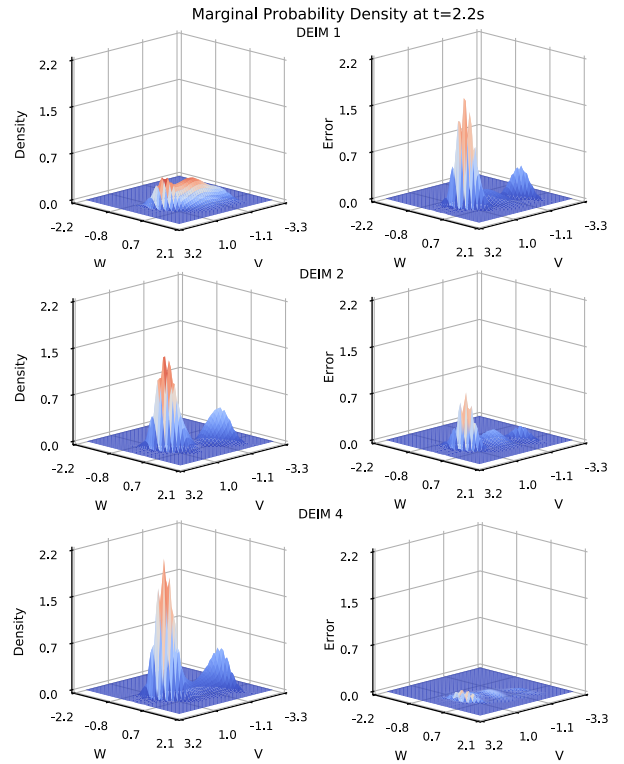


Fig. 2. Reduced models constructed with DEIM. Left column shows the approximation and right column the absolute difference between the state seen in Figure 1 in (V, W) space. Lower rows indicate approximation of higher dimension.

a more economical implementation is achievable. The models take advantage of multiple cores through differential equation solvers. We use the 4<sup>th</sup> order Runge-Kutta method with fixed step of  $dt = 0.01$  s.

### III. RESULTS

The original model from Equation (1) was simulated for  $t = 2.2$  s with the Gaussian distribution as the initial state. On average, this takes 200 minutes. The state of the model after simulation is seen in Figure 1. As the modeled probability density function has a 3-dimensional domain, for visualization purposes integration over one variable is required. In plot 1) of Figure 1, integration is done over the Y-variable, in plot 2) over W-variable and finally over V-variable in plot 3). The reduced models should reach the same state with minimal error. From this point forward, only the (V, W) space will be visualized.

Figure 2 shows reduction results with the DEIM method, after an approximation of the original system is reconstructed using the low-dimensional model. Left column shows the state of the reduced model integrated over the Y-variable, and right column shows the absolute difference between the original and reduced model at every point in the (V, W) space. Dimension of the reduced model grows in each row, with dimensions 1, 2 and 4 illustrated. It can be seen how the reduced model rapidly converges to the same solution as the original model.

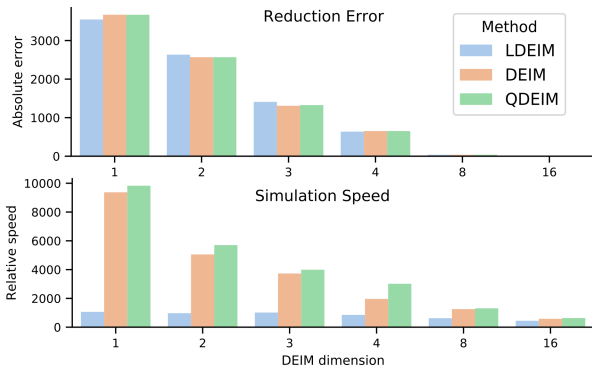


Fig. 3. Top row: approximation error from reduced models in  $(V, W)$  space. Bottom row: factor of acceleration gained using the reduced models. Reduced model dimension grows to the right and color indicates reduction method.

In Figure 3, upper row shows approximation error as the sum over point-wise absolute differences between the original and each reduced model in  $(V, W)$  space. Lower row indicates achieved speedup as median simulation time of each reduced model divided by the median time of the original model, with data from 20 simulations per model. Hue indicates MOR method. Error decreases with increasing dimension using any method, while the gained acceleration also declines. DEIM and QDEIM are seen to be equally accurate and fast, while LDEIM is slower due to the online adaptivity. The cost of adaptivity diminishes as the dimension of the reduced model increases. With the present model, LDEIM does not achieve an increase in accuracy compared to DEIM and QDEIM of similar dimensions.

#### IV. DISCUSSION AND CONCLUSIONS

Mathematical model order reduction (MOR) of nonlinear systems in neuroscience has been studied before in limited settings. In [25]–[27] a model of a branching neuron was reduced. A chemical reaction based model of synaptic plasticity was reduced in [28]. MOR studies of nonlinear neuronal populations have been conducted in a hippocampal network [29], in cardiac and muscular systems [30], [31] and in mono- and bidomain electromyography models [32], [33]. Mean-field models have been reduced in fluid dynamics in the context of e.g. water flow and flame behaviour [34], [35]. The authors are not aware of previous applications of mathematical MOR to neuronal Fokker-Planck mean-field models.

We have shown how the numerical simulation of a high-dimensional neuronal mean-field model can be accelerated significantly by the use of mathematical MOR methods. We achieved an improvement of over three orders of magnitude in simulation time, with low approximation error. Performing this type of numerical approximation does not render the model any less biologically relevant, as happens with simplification approaches that remove variables and make assumptions about the dynamics of the system. We note that the magnitude of acceleration gained using reduced models depends on the number of discretization points required to numerically solve

the partial differential equation. To accurately reach a steady-state solution a fine grid is required, and the potential speedup is greater. The main bottleneck is the rapidly growing memory consumption of the original model.

Improving the simulation time of computational models of neural populations is important as the number of neural cells, and hence variables in the model, must be large in order to reach satisfactory levels of biological realism. While Fokker-Planck-type mean-field models make the study of stochastic networks easier, solving them requires spatial discretization over every variable of the neuron model, resulting in a rapidly growing number of dimensions and hence long computation times. In [36] the present model was simulated efficiently with graphical processing units (GPUs). The alternative approach described in this study does not rely on increasing hardware resources and instead improves simulation time with mathematical methods based on low-dimensional subspace approximation.

The present, purely software-based implementation of mathematical MOR methods is especially interesting in terms of integration into neuronal simulators such as NEURON [37], NEST [38] and The Virtual Brain [39]. In these simulators, approximated models could be made readily available as components for network or compartmental cell simulations. Alternatively, the simulator software could compute reduced models during long simulations and finish the simulation efficiently using the low-dimensional model. Integration is feasible because MOR methods do not require any special hardware, and software only needs to support matrix arithmetic and differential equation solvers.

Neuromorphic hardware is state-of-the-art in low energy computation and is used in neuroscience, robotics and artificial intelligence research [3], [4]. Neuromorphic chips have parallels with MOR methods in striving for accelerated simulation of (neuronal) models. Additionally, implementing reduced models on neuromorphic hardware, such as the SpiNNaker system [3], could enable the study of neuronal networks in even larger scales than before. However, not all neuromorphic chips can be combined with MOR methods.

Based on our results, we suggest MOR methods to be applied to other mean-field models to see whether we can reproduce such results with different populations of neurons. An excellent candidate would be the Hodgkin-Huxley neuronal mean-field model discussed in [16], for which applying the present method is straightforward. Follow-up studies should also address the situation when simulation parameters differ from parameters used in the snapshot collection phase. Additionally, the benefit gained from the methods introduced in [40] should be addressed in the context of computational neuroscience.

#### ACKNOWLEDGMENTS

Funding: 298182, 297893, 310489 (Academy of Finland; M.-L.L., L.P), 785907 (EU FET Flagship Human Brain Project; M.-L.L.), TUNI Graduate School (M.L.).

## REFERENCES

- [1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
- [2] D. Buonomano and W. Maass, "State-dependent computations: spatiotemporal processing in cortical networks," *Nature Reviews Neuroscience*, vol. 10, no. 2, p. 113, 2009.
- [3] S. Furber, F. Galluppi, S. Temple, and L. Plana, "The SpiNNaker project," *Proceedings of the IEEE*, vol. 102, no. 5, pp. 652–665, 2014.
- [4] S. van Albada, A. Rowley, J. Senk, M. Hopkins, M. Schmidt, A. Stokes, D. Lester, M. Diesmann, and S. Furber, "Performance comparison of the digital neuromorphic hardware SpiNNaker and the neural network simulation software NEST for a full-scale cortical microcircuit model," *Frontiers in Neuroscience*, vol. 12, p. 291, 2018.
- [5] F. Zenke, E. Agnes, and W. Gerstner, "Diverse synaptic plasticity mechanisms orchestrated to form and retrieve memories in spiking neural networks," *Nature Communications*, vol. 6, p. 6922, 2015.
- [6] J. Guerguiev, T. P. Lillicrap, and B. Richards, "Towards deep learning with segregated dendrites," *ELife*, vol. 6, p. e22901, 2017.
- [7] A. Marblestone, G. Wayne, and K. Kording, "Toward an integration of deep learning and neuroscience," *Frontiers in Computational Neuroscience*, vol. 10, p. 94, 2016.
- [8] W. Maass, "Searching for principles of brain computation," *Current Opinion in Behavioral Sciences*, vol. 11, pp. 81–92, 2016.
- [9] G. Deco, V. Jirsa, P. Robinson, M. Breakspear, and K. Friston, "The dynamic brain: from spiking neurons to neural masses and cortical fields," *PLoS Computational Biology*, vol. 4, no. 8, p. e1000092, 2008.
- [10] P. Benner, S. Gugercin, and K. Willcox, "A survey of projection-based model reduction methods for parametric dynamical systems," *Society for Industrial and Applied Mathematics Review*, vol. 57, no. 4, pp. 483–531, 2015.
- [11] M. Churchland and K. Shenoy, "Temporal complexity and heterogeneity of single-neuron activity in premotor and motor cortex," *Journal of Neurophysiology*, vol. 97, no. 6, pp. 4235–4257, 2007.
- [12] K. Wiesenfeld and F. Moss, "Stochastic resonance and the benefits of noise: from ice ages to crayfish and SQUIDS," *Nature*, vol. 373, no. 6509, p. 33, 1995.
- [13] T. Manninen, M.-L. Linne, and K. Ruohonen, "Developing Itô stochastic differential equation models for neuronal signal transduction pathways," *Computational Biology and Chemistry*, vol. 30, no. 4, pp. 280–291, 2006.
- [14] A. Saarinen, M.-L. Linne, and O. Yli-Harja, "Stochastic differential equation model for cerebellar granule cell excitability," *PLoS computational biology*, vol. 4, no. 2, p. e1000004, 2008.
- [15] A. Destexhe and D. Contreras, "Neuronal computations with stochastic network states," *Science*, vol. 314, no. 5796, pp. 85–90, 2006.
- [16] J. Baladron, D. Fasoli, O. Faugeras, and J. Touboul, "Mean-field description and propagation of chaos in networks of Hodgkin-Huxley and FitzHugh-Nagumo neurons," *The Journal of Mathematical Neuroscience*, vol. 2, no. 1, p. 10, 2012.
- [17] R. FitzHugh, "Impulses and physiological states in theoretical models of nerve membrane," *Biophysical Journal*, vol. 1, no. 6, pp. 445–466, 1961.
- [18] J. Nagumo, S. Arimoto, and S. Yoshizawa, "An active pulse transmission line simulating nerve axon," *Proceedings of the IRE*, vol. 50, pp. 2061–2070, 1962.
- [19] M. Barrault, Y. Maday, N. C. Nguyen, and A. T. Patera, "An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations," *Comptes Rendus Mathématique*, vol. 339, no. 9, pp. 667–672, 2004.
- [20] S. Chaturantabut and D. Sorensen, "Nonlinear model reduction via discrete empirical interpolation," *SIAM Journal on Scientific Computing*, vol. 32, no. 5, pp. 2737–2764, 2010.
- [21] J. Lumley, G. Berkooz, and P. Holmes, "The Proper Orthogonal Decomposition in the analysis of turbulent flows," *Annual Review of Fluid Mechanics*, vol. 25, pp. 539–575, 1993.
- [22] L. Sirovich, "Turbulence and the dynamics of coherent structures. I-III," *Quarterly of Applied Mathematics*, vol. 45, no. 3, pp. 561–590, 1987.
- [23] Z. Drmac and S. Gugercin, "A new selection operator for the Discrete Empirical Interpolation Method—improved a priori error bound and extensions," *SIAM Journal on Scientific Computing*, vol. 38, no. 2, pp. A631–A648, 2016.
- [24] B. Peherstorfer, D. Butnaru, K. Willcox, and H. Bungartz, "Localized discrete empirical interpolation method," *SIAM Journal on Scientific Computing*, vol. 36, no. 1, 2014.
- [25] A. Kellems, S. Chaturantabut, D. Sorensen, and S. Cox, "Morphologically accurate reduced order modeling of spiking neurons," *Journal of Computational Neuroscience*, vol. 28, no. 3, 2010.
- [26] B. Du, D. Sorensen, and S. Cox, "Model reduction of strong-weak neurons," *Frontiers in Computational Neuroscience*, vol. 8, no. 164, 2014.
- [27] D. Amsallem and J. Nordström, "Energy stable model reduction of neurons by nonnegative discrete empirical interpolation," *SIAM Journal on Scientific Computing*, vol. 38, no. 2, pp. B297–B326, 2016.
- [28] M. Lehtimäki, L. Paunonen, S. Pohjolainen, and M.-L. Linne, "Order reduction for a signaling pathway model of neuronal synaptic plasticity," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7687–7692, 2017.
- [29] M. Lehtimäki, L. Paunonen, and M.-L. Linne, "Projection-based order reduction of a nonlinear biophysical neuronal network model," in *2019 Proceedings of the IEEE Conference on Decision and Control (CDC)*. IEEE, accepted.
- [30] M. Boulakia, E. Schenone, and J.-F. Gerbeau, "Reduced-order modeling for cardiac electrophysiology. Application to parameter identification," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 28, no. 6-7, pp. 727–744, 2012.
- [31] H. Yang and A. Veneziani, "Efficient estimation of cardiac conductivities via pod-deim model order reduction," *Applied Numerical Mathematics*, vol. 115, pp. 180–199, 2017.
- [32] M. Mordhorst, T. Strecker, D. Wirtz, T. Heidlauf, and O. Röhrle, "Pod-deim reduction of computational emg models," *Journal of Computational Science*, vol. 19, pp. 86–96, 2017.
- [33] N. Emamy, P. Litty, T. Klotz, M. Mehl, and O. Röhrle, "POD-DEIM model order reduction for the monodomain reaction-diffusion sub-model of the neuro-muscular system," in *IUTAM Symposium on Model Order Reduction of Coupled Systems, Stuttgart, Germany, May 22–25, 2018*, J. Fehr and B. Haasdonk, Eds. Cham: Springer International Publishing, 2020, pp. 177–190.
- [34] T. Lassila, A. Manzoni, A. Quarteroni, and G. Rozza, "Model order reduction in fluid dynamics: challenges and perspectives," in *Reduced Order Methods for modeling and computational reduction*. Springer, 2014, pp. 235–273.
- [35] S. Mowlavi and T. P. Sapsis, "Model order reduction for stochastic dynamical systems with continuous symmetries," *SIAM Journal on Scientific Computing*, vol. 40, no. 3, pp. A1669–A1695, 2018.
- [36] J. Pezoa, J. Baladron, D. Fasoli, and O. Faugeras, "Three applications of GPU computing in neuroscience," *Computing in Science & Engineering*, vol. 14, no. 3, pp. 40–47, 2011.
- [37] M. Hines and N. Carnevale, "The neuron simulation environment," *Neural Computation*, vol. 9, no. 6, pp. 1179–1209, 1997.
- [38] M.-O. G. and M. D., "Nest (neural simulation tool)," *Scholarpedia*, vol. 2, no. 4, p. 1430, 2007.
- [39] P. Sanz Leon, S. Knock, M. Woodman, L. Domide, J. Mersmann, A. McIntosh, and V. Jirsa, "The Virtual Brain: a simulator of primate brain network dynamics," *Frontiers in Neuroinformatics*, vol. 7, p. 10, 2013.
- [40] B. Peherstorfer, Z. Drmač, and S. Gugercin, "Stabilizing discrete empirical interpolation via randomized and deterministic oversampling," *arXiv preprint arXiv:1808.10473*, 2018.