POLITECNICO DI TORINO
Repository ISTITUZIONALE

Computing with Memristor-based Nonlinear Oscillators

Original
Computing with Memristor-based Nonlinear Oscillators / Zoppo, G; Marrone, F; Bonnin, M; Corinto, F. - In: PROCEEDINGS OF THE ... IEEE CONFERENCE ON NANOTECHNOLOGY. - ISSN 1944-9399. - ELETTRONICO. - (2022), pp. 401-404. (Intervento presentato al convegno 2022 IEEE 22nd International Conference on Nanotechnology (NANO) tenutosi a Palma de Mallorca, Spain nel 04-08 July 2022) [10.1109/NANO54668.2022.9928754].

Availability:
This version is available at: 11583/2975397 since: 2023-02-06T07:37:48Z

Publisher:
IEEE

Published
DOI:10.1109/NANO54668.2022.9928754

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2022 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

17 October 2023

Computing with Memristor-based Nonlinear Oscillators

Gianluca Zoppo, Francesco Marrone, Michele Bonnin and Fernando Corinto
Department of Electronics and Telecommunications
Politecnico di Torino, Turin, Italy

***Abstract*—Among the recent disruptive technologies, volatile/nonvolatile memory–resistor (memristor) has attracted the researchers' attention as a fundamental computation element. It has been experimentally shown that memristive elements can emulate synaptic dynamics and are even capable of supporting spike timing dependent plasticity (STDP), an important adaptation rule for neuromorphic computing systems. The overall goal of this work is to provide an unconventional computing platform exploiting memristor-based nonlinear oscillators described by means of phase deviation equations. Experimental results show that the approach significantly outperforms conventional architectures used for pattern recognition tasks.**

I. Introduction

Recently, novel information processing solutions have been proposed to perform energy efficient data centric computation that is called to satisfy the need of the present interconnected society. Most of them are similar in that they exploit memristors as an alternative key-enabling technology [1], [2]. Approaches ranges from more conservative ones trying to close the distance between logic and memory [3] to more innovative ones pushing towards bio-inspired architectures, i.e. neural and neuromorphic networks [4]–[8]. While the former are expected to provide a substantial but still incremental improvement to the Big Data issue, the latter, the neuromorphic computation, which potentially constitutes a real breakthrough, is still hindered by the non–idealities of memristor devices [9]. Conversely, in recent works [10], [11], oscillator–based computing has been demonstrated to exploit the inherent complex dynamics of memristors to perform computation with significant energy benefits. In this framework the memristor plays a crucial role in the realization of nonlinear oscillators displaying a tunable range of periodic and chaotic self-oscillations. Their importance in neuromorphic applications, such as pattern recognition and signal processing tasks in real time, have been demonstrated in hardware [12].

The aim of this work is twofold: (a) to provide a comphrensive mathematical framework to describe nonlinear oscillators by means of phase models. This approach is presented for classical nonlinear oscillators but can be extended, *mutatis mutandi*, to memristor oscillators described in the flux–charge domain [13]; (b) to show how an Oscillatory Neural Network (ONN), i.e. an array of nonlinear oscillators (with or without memristor) coupled by means of programmable memristive synapses, can be trained with the Equilibrium Propagation algorithm by exploiting phase models.

Numerical simulations show that the proposed ONN significantly outperforms conventional approaches used for pattern reconstruction tasks.

II. Nonlinear oscillator and phase model

The fundamental building blocks of ONNs are nonlinear oscillators, that we describe by means of the ordinary differential equation (ODE)

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{f}(\boldsymbol{x}) \tag{1}$$

where $\boldsymbol{x} : \mathbb{R} \mapsto \mathbb{R}^n$ is the state of the oscillator, and $\boldsymbol{f} : \mathbb{R}^n \mapsto \mathbb{R}^n$ is a smooth enough vector valued function, that describes the oscillator's internal dynamics. For the sake of simplicity we shall assume that all oscillators are identical. We shall also assume that equation (1) admits an asymptotically stable T-periodic solution $\boldsymbol{x}_s(t) = \boldsymbol{x}_s(t+T)$, corresponding to a limit cycle $\boldsymbol{\gamma}$ in its state space.

The phase of a nonlinear oscillator can be defined introducing an homeomorphism $\phi(\boldsymbol{x}) : \boldsymbol{\gamma} \mapsto [0, 2\pi[$, mapping the periodic solution $\boldsymbol{x}_s(t)$ to the unit circle. Consider a reference initial point on the limit cycle $\boldsymbol{x}_0 \in \boldsymbol{\gamma}$, and assign phase zero to this point $\phi(\boldsymbol{x}_0) = 0$. Let $\boldsymbol{x}_s(0) = \boldsymbol{x}_0$, then obviously $\phi(\boldsymbol{x}_s(T)) = 2\pi$. The phase of the solution $\boldsymbol{x}_s(t)$ at any arbitrary time instant is $\phi(\boldsymbol{x}_s(t)) = t/T = \omega t$, where $\omega = 2\pi/T$ is the oscillator free running frequency.

The phase just defined can be extended to the whole basin of attraction of the limit cycle, introducing the concept of *isochrons*. Isochrons are $(n-1)$–dimensional manifolds transverse to the limit cycle. The isochron transverse to the cycle at $\boldsymbol{x}_0 \in \boldsymbol{\gamma}$ is defined as

$$I_{\boldsymbol{x}_0} = \left\{ \boldsymbol{x}_\alpha \in \mathbb{R}^n \ : \ \lim_{t \to +\infty} ||\boldsymbol{x}_\alpha(t) - \boldsymbol{x}_0(t)|| = 0 \right\} \tag{2}$$

The isochrons are the set of initial conditions $\boldsymbol{x}_\alpha(0)$ such that the trajectories starting from $\boldsymbol{x}_\alpha(0)$ eventually meet on $\boldsymbol{\gamma}$ at $\boldsymbol{x}_0(t)$. Therefore, they represent the stable manifold foliation of $\boldsymbol{x}_0(t)$.

The phases of the points inside the basin of attraction of a stable limit cycle are defined by assigning the same phase to points lying on the same isochron. The phase of the point $\boldsymbol{x}_\alpha(t)$ is $\phi(\boldsymbol{x}_\alpha(t)) = \omega t + \phi(\boldsymbol{x}_\alpha(0))$. In other words, isochrons are the level sets of the scalar field $\phi(\boldsymbol{x})$.

Together with the phase function we introduce a vector valued function $\boldsymbol{R}(\boldsymbol{x}) : \mathbb{R}^n \mapsto \mathbb{R}^{n-1}$, that we call *amplitude deviation* or simply *amplitude*, that represents a deviation

from the limit cycle. The amplitude deviation is most conveniently measured on the linear subspace locally tangent to the isochrons on the limit cycle. This counterintuitive choice guarantees that phase and amplitude variables are linearly independent in a small enough neighborhood of the limit cycle. Therefore the coordinate transformation is invertible and we can write $\boldsymbol{x} = \boldsymbol{x}(\phi, \boldsymbol{R})$. Moreover, such a choice guarantees that amplitude and phase dynamics are uncoupled up to linear terms [14], [15].

By their very definitions, it follows that

$$\frac{d\phi}{dt} = \nabla\phi(\boldsymbol{x}) \cdot \boldsymbol{f}(\boldsymbol{x}) = \omega \tag{3}$$

and, $\boldsymbol{R}(\boldsymbol{x}) \to 0$ for $t \to +\infty$.

III. Oscillatory neural networks

An ONN composed by N coupled nonlinear oscillators can be conveniently described by the ODEs (from now on, subscript indexes will identify the oscillator)

$$\frac{d\boldsymbol{x}_i}{dt} = \boldsymbol{f}(\boldsymbol{x}_i) + \varepsilon \sum_{j=1}^{N} \boldsymbol{g}_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad i = 1, \ldots, N \tag{4}$$

where $\varepsilon \ll 1$ is a parameter that measures the coupling strength, and $\boldsymbol{g}_{ij} : \mathbb{R}^{n\times n} \mapsto \mathbb{R}^n$ are vectors field describing the coupling between the i-th and the j-th oscillators. In this example, only pairwise interactions are assumed, but the theory can be easily generalized to more complex coupling functions.

Introducing the phase and amplitude of each oscillator $\phi_i(\boldsymbol{x}_i)$ and $\boldsymbol{R}_i(\boldsymbol{x}_i)$, it is straightforward deriving the equations

$$\frac{d\phi_i}{dt} = \omega + \varepsilon \sum_{j=1}^{N} \Gamma_{i,j}(\phi_i, \phi_j, \boldsymbol{R}_i, \boldsymbol{R}_j) \tag{5a}$$

$$\frac{d\boldsymbol{R}_i}{dt} = \boldsymbol{F}_i(\phi_i, \boldsymbol{R}_i) + \varepsilon \sum_{j=1}^{N} \boldsymbol{G}_{ij}(\phi_i, \phi_j, \boldsymbol{R}_i, \boldsymbol{R}_j) \tag{5b}$$

where, taking into account that $\boldsymbol{x}_i = \boldsymbol{x}_i(\phi_i, \boldsymbol{R}_i)$,

$$\Gamma_{ij}(\phi_i, \phi_j, \boldsymbol{R}_i, \boldsymbol{R}_j) = \nabla\phi_i(\boldsymbol{x}_i) \cdot \boldsymbol{g}_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{6}$$

$$\boldsymbol{F}_i(\phi_i, \boldsymbol{R}_i) = \frac{\partial \boldsymbol{R}_i}{\partial \boldsymbol{x}_i} \boldsymbol{f}(\boldsymbol{x}_i) \tag{7}$$

$$\boldsymbol{G}_{ij}(\phi_i, \phi_j, \boldsymbol{R}_i, \boldsymbol{R}_j) = \frac{\partial \boldsymbol{R}_i}{\partial \boldsymbol{x}_i} \boldsymbol{g}_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) \tag{8}$$

and $\partial \boldsymbol{R}_i / \partial \boldsymbol{x}_i$ is the Jacobian matrix of partial derivatives.

The asymptotic stability hypothesis of the limit cycles implies that, as long as the coupling strength remains small enough, then the amplitude deviation remains close to zero. Thus it seems appropriate to make the approximation $\boldsymbol{R}_i \approx 0$ for all $i = 1, \ldots, N$. The phase equation (5a) reduces to

$$\frac{d\phi_i}{dt} = \omega + \varepsilon \sum_{j=1}^{N} \Gamma_{ij}(\phi_i, \phi_j) \tag{9}$$

Equation (9) can be simplified further. Introduce the *phase deviation* $\psi_i = \phi_i - \omega t$, that represents the difference between the phase in presence and in absence of coupling, obtaining the *phase deviation equation*

$$\frac{d\psi_i}{dt} = \varepsilon \sum_{j=1}^{N} \Gamma_{ij}(\psi_i + t, \psi_j + t) \tag{10}$$

Eq. (10) shows that, for small values of ε, the phase deviation is a slow (or nearly constant) variable. Thus we can average over one period without introducing a great error, thus obtaining the averaged phase deviation equation

$$\frac{d\psi_i}{d\tau} = \sum_{j=1}^{N} \overline{\Gamma}_{ij}(\psi_j - \psi_i) \tag{11}$$

where $\tau = \varepsilon t$ is the slow time, and

$$\overline{\Gamma}_{ij}(\psi_j - \psi_i) = \frac{1}{T} \int_0^T \Gamma_{ij}(\psi_i + t, \psi_j + t)\, dt \tag{12}$$

IV. Associative Memories using Oscillatory Networks

The usual approach for implementing neural associative memories consists in teaching the system to converge to prescribed equilibria. The requirement of this dynamic learning is the stability of the equilibrium points. As the Hopfield network, oscillatory networks may exhibit associative properties that can be used to recognize binary patterns. If all oscillators have equal frequencies and functions $\overline{\Gamma}_{ij}$ have pairwise odd form, then the phase model defined in (11) is guaranteed to converge to one of the many possible equilibrium points of the system [16]. This means that networks of the form (4) implement neural systems whose phase dynamics (11) can be used to solve pattern retrieval tasks.

The Kuramoto oscillator is one of the most used mathematical models for associative memory that makes use of the phase dynamics defined in Eq. (11). The dynamics of the network is described in terms of the phase equations:

$$\frac{d\psi_i}{d\tau} = \sum_{j=1}^{N} w_{ij} \sin(\psi_j - \psi_i) \quad \forall i = 1, \ldots, N \tag{13}$$

where $\boldsymbol{\psi} = [\psi_1, \ldots, \psi_N]^T$ is the state variable vector and $\mathbf{W} \in \mathbb{R}^{N\times N}$ is the coupling weight matrix. If the connections are assumed to be symmetric, i.e. $w_{ij} = w_{ji}$ and $w_{ii} = 0$, the stability of the system dynamics can be deduced by reformulating it as a gradient system

$$\frac{d\psi_i}{d\tau} = -\frac{\partial E}{\partial \psi_i}(\boldsymbol{\psi}, \mathbf{W}) \tag{14}$$

where

$$E(\boldsymbol{\psi}, \mathbf{W}) = -\frac{1}{2} \sum_{i,j=1}^{N} w_{ij} \cos(\psi_i - \psi_j). \tag{15}$$

Equilibria of (13) are minima of the scalar (or potential) function defined in Eq. (15).

Consider a set of m binary patterns $\mathbf{T}_k \in \mathbb{R}^N$ $\forall k = 1, \ldots, m$ such that $T_k^j \in \{0, \pi\}$ $\forall j = 1, \ldots, N$). The objective of an equilibrium point learning scheme is to find

the model's parameter so that, for a given initial condition, the fixed point of the system (14) corresponds to a desired target value. Donald O. Hebb [17] suggested that connections between neurons should be strengthened when a simultaneous activity under an external input occurs, and reduced otherwise. This can be mathematically reformulated in the following construction of the weight matrix [18]:

$$\mathbf{W} = \frac{1}{m}\sum_{i=1}^{m}[\cos(\mathbf{T}_i)\cos(\mathbf{T}_i)^T - \mathbb{I}_{N\times N}]$$

where $\cos(\mathbf{T}_i) = [\cos(T_i^1), \ldots, \cos(T_i^N)]^T$. Although this is a basic local learning rule, it can lead to powerful self-organization effects in relatively simple neural networks models. This learning rule is prone to local minima, but a small amount of noise can be useful for escape from these spurious state.

From a different perspective, gradient descent offers a more sophisticated technique that provides to the system some guiding directions to reorganize the connections changes throughout the network. This is achieved by minimizing the cost $\mathbf{C}$ for a single pair of points $\mathbf{T}$ and $\boldsymbol{\psi} = [\psi_1, \ldots, \psi_N]^T$:

$$C(\mathbf{T}, \boldsymbol{\psi}) = N - \sum_{i=1}^{N}\cos(T_i - \psi_i) \tag{16}$$

which measures the distance between a desired target $\mathbf{T}$ and the output state of the system $\boldsymbol{\psi}$. Observe that $C(\mathbf{T}, \boldsymbol{\psi}) \geq 0$ and $C(\mathbf{T}, \boldsymbol{\psi}) = 0 \Leftrightarrow T_i = \psi_i \ \forall i$. Recently, Bengio and the authors in [19] proposed a promising local learning technique used for Energy-based models. In particular, the authors focused on recurrent neural networks that admit a gradient formulation [20]. Our work generalizes the algorithm to the case of ONNs whose phase dynamics is defined as in Eq. (11), and admit a gradient formulation. We illustrate the analysis focusing on the Kuramoto model described in Eq. (13).

Let us define the following augmented energy function:

$$F(\boldsymbol{\psi}^\beta, \mathbf{T}, \mathbf{W}, \beta) = E(\boldsymbol{\psi}^\beta, \mathbf{W}) + \beta C(\mathbf{T}, \boldsymbol{\psi}^\beta) \tag{17}$$

where $\beta \geq 0$ is the forcing parameter, and consider the corresponding gradient system:

$$\frac{d\psi_i^\beta}{d\tau} = -\frac{\partial F}{\partial \psi_i^\beta}(\boldsymbol{\psi}^\beta, \mathbf{T}, \mathbf{W}, \beta) \tag{18}$$

for all $i = 1, \ldots, N$. Let $\hat{\boldsymbol{\psi}}$ and $\hat{\boldsymbol{\psi}}^\beta$ be the fixed points of systems (14) and (18), respectively. By recasting Theorem 1 in [19] to the case of the Kuramoto model dynamics, the learning scheme can be defined as:

$$\begin{aligned}\Delta w_{ij} &\propto -\frac{d}{d\beta}\left[\frac{\partial F}{\partial W_{ij}}(\hat{\boldsymbol{\psi}}^\beta, \mathbf{T}, \mathbf{W}, \mathbf{I}, \beta)\right]_{\beta=0} = \\ &= \frac{d}{d\beta}\cos(\hat{\psi}_i^\beta - \hat{\psi}_j^\beta)\Big|_{\beta=0} = \\ &\approx \frac{\cos(\hat{\psi}_i^\beta - \hat{\psi}_j^\beta) - \cos(\hat{\psi}_i - \hat{\psi}_j)}{\beta}\end{aligned} \tag{19}$$

with $\beta \simeq 0$.

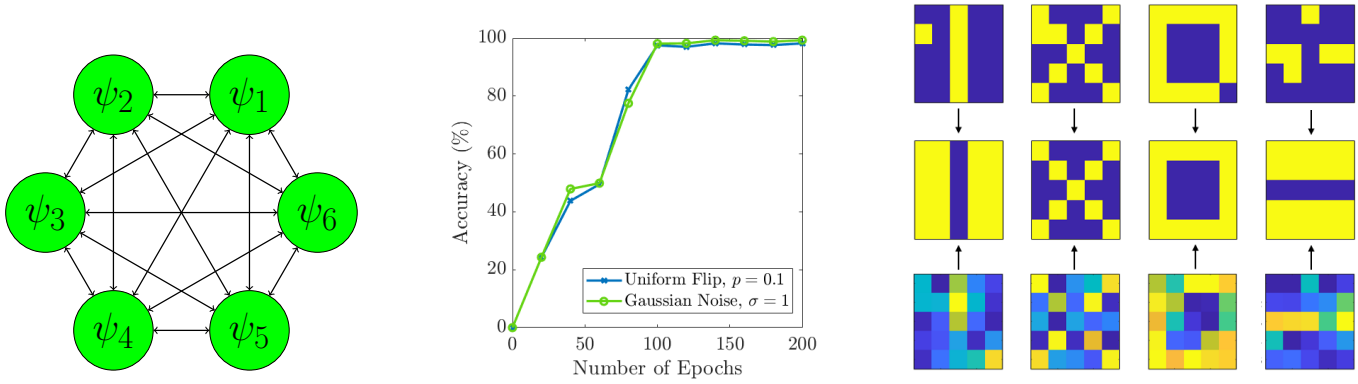
V. Simulations and Results

The network architecture consists of a fully connected ONN with $N = 64$ oscillators and symmetric weights as depicted in Fig. 1(a). Oscillators are individually controllable through the coupling with an additional driving oscillator unit. In the simulations, the driving unit oscillators have two different important roles:

- to set the phase of each oscillator as equal to the perturbed/target pattern ($\beta \gg 0$);
- in case of *in-situ* training, to model the teaching signal of the second phase of Equilibrium Propagation as defined in the augmented energy defined in Eq. (17).

In our example, van der Pol oscillators are used as working units. The system is trained using either the Hebbian learning rule or Equilibrium Propagation, to learn the four different pattern shown in Fig. 1(c). The former as a one shot computation whereas the latter starts with a randomly initialized weight matrix, whose entries are sampled from a uniform distribution. Weights are found with off-line training phase, simulating the Kuramoto model. Once the training process ends, weights are mapped into the adjustable connections of the oscillatory units using memristors as ideal programmable resistors, coupled in a differential pair configuration. The update of the weights is performed by averaging the back propagated errors over the total number of training images. This approach allows us to lower the amount of total updates of the weight matrix. The learning rate is $\eta = 0.0001$ and it decreases during the iterations using a step decay schedule. The forcing parameter is set to $\beta = 0.1$ and the training process ends once a prefixed accuracy is achieved. The learning curve is shown in Fig. 1(b). The information of the patterns are encoded into the network with the following rule: a yellow square corresponds to the phase difference between each oscillator and a reference unit equal to 0, and a blue square corresponds to the phase difference equal to π. Patterns are successively corrupted using either a uniform flipping of the pixels with probability $p = 0.1$ or an additive Gaussian noise with standard deviation $\sigma = 1$. Initial patterns composed by 0 and/or π represent stable equilibria of the free dynamical system defined in Eq. (13). Thus, a small perturbation is added to the phase initial conditions of the oscillators.

Phase trajectories might not converge to multiple values of 0 and π. This is probably due to the cosinusoidal combination in the potential function that allows the system to have many possible equilibria. Therefore, once the cosine of the output phase differences at equilibrium is computed, results are then saturated to the closest values $-1 = \cos(\pi)$ or $1 = \cos(0)$. A pattern is recognized as correctly reconstructed if the Hamming difference between the reconstruction and the target image is zero. As shown in Fig. 1(c), results provide evidence that Oscillatory Networks trained with Equilibrium Propagation are perfectly able to reconstruct the corrupted patterns. Even though the network trained with Equilibrium Propagation needs a larger amount of epochs compared to the Hebbian learning scheme, the former method reaches 98% of



(a) (b) (c)

Figure 1: (a) Illustration of an ONN with symmetric weights and 6 working units. (b) Accuracy of the network trained with Equilibrium Propagation to reconstruct perturbed patterns: in blue, pixels are perturbed by flipping the pixels from yellow to blue and vice versa with probability $p = 0.1$; in green, the same patterns are corrupted with a Gaussian noise with $\sigma = 1$. (c) Examples of reconstruction: the second row shows the four learned patterns that are found by the network when perturbing with either a uniform flip (first row) or a Gaussian noise (third row).

accuracy over 1000 different perturbed patterns whereas the latter has only 48% of correct retrieval on the same test set.

VI. Conclusions

In this paper, the dynamics of oscillatory neural networks with memristive synapses has been analyzed. The mathematical framework to describe the oscillatory array in terms of the corresponding phase model is introduced. Then, the phase model is used to adjust synaptic weights accoding to Equilibrium Propagation algorithm. Finally, numerical simulations show that proposed oscillatory neural network exhibits 98% of accuracy in reconstructing noise corrupted patterns.

Acknowledgments

This work was supported by the Ministero dell'Istruzione, dell'Università e della Ricerca (MIUR) under contract 2017LSCR4K-003.

References

[1] X. Yang, B. Taylor, A. Wu, Y. Chen, and L. O. Chua, "Research progress on memristor: From synapses to computing systems," *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2022.

[2] T. Guo, K. Pan, Y. Jiao, B. Sun, C. Du, J. P. Mills, Z. Chen, X. Zhao, L. Wei, N. Zhou *et al.*, "Versatile memristor for memory and neuromorphic computing," *Nanoscale Horizons*, 2022.

[3] M. Zahedi, M. A. Lebdeh, C. Bengel, D. Wouters, S. Menzel, M. Le Gallo, A. Sebastian, S. Wong, and S. Hamdioui, "Mnemosene: Tile architecture and simulator for memristor-based computation-in-memory," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 18, no. 3, pp. 1–24, 2022.

[4] R. Li, P. Huang, Y. Feng, Z. Zhou, Y. Zhang, X. Ding, L. Liu, and J. Kang, "Hardware demonstration of srdp neuromorphic computing with online unsupervised learning based on memristor synapses," *Micromachines*, vol. 13, no. 3, p. 433, 2022.

[5] N. Ilyas, J. Wang, C. Li, H. Fu, D. Li, X. Jiang, D. Gu, Y. Jiang, and W. Li, "Controllable resistive switching of sto: Ag/sio2-based memristor synapse for neuromorphic computing," *Journal of Materials Science & Technology*, vol. 97, pp. 254–263, 2022.

[6] L. Zhang, Z. Tang, D. Yao, Z. Fan, S. Hu, Q.-J. Sun, X.-G. Tang, Y.-P. Jiang, X. Guo, M. Huang *et al.*, "Synaptic behaviors in flexible au/wox/pt/mica memristor for neuromorphic computing system," *Materials Today Physics*, vol. 23, p. 100650, 2022.

[7] F. Marrone, G. Zoppo, F. Corinto, and M. Gilli, "A dynamic system approach to spiking second order memristor networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, 2021.

[8] K.-S. Min and F. Corinto, "Memristor computing for neuromorphic systems," *Frontiers in Computational Neuroscience*, vol. 15, 2021.

[9] G. Zoppo, A. Korkmaz, F. Marrone, S. Palermo, F. Corinto, and R. S. Williams, "Analog solutions of discrete Markov chains via memristor crossbars," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 12, pp. 4910–4923, 2021.

[10] S. Kumar, J. P. Strachan, and R. S. Williams, "Chaotic dynamics in nanoscale nbo2 mott memristors for analogue computing," *Nature*, vol. 548, no. 7667, pp. 318–321, 2017.

[11] M. Weiher, M. Herzig, R. Tetzlaff, A. Ascoli, T. Mikolajick, and S. Slesazeck, "Improved vertex coloring with nbo_x memristor-based oscillatory networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 68, no. 5, pp. 2082–2095, 2021.

[12] J. Shamsi, M. J. Avedillo, B. Linares-Barranco, and T. Serrano-Gotarredona, "Hardware implementation of differential oscillatory neural networks using vo 2-based oscillators and memristor-bridge circuits," *Frontiers in Neuroscience*, vol. 15, 2021.

[13] F. Corinto, M. Forti, and L. Chua, "Nonlinear circuits and systems with memristors," *Nonlinear Dynamics and Analogue Computing via the Flux-Charge Analysis Method. Springer Verlag. Available online at: https://www. springer. com/gp/book/9783030556501*, 2020.

[14] M. Bonnin, F. Corinto, and M. Gilli, "Phase space decomposition for phase noise and synchronization analysis of planar nonlinear oscillators," vol. 59, no. 10, pp. 638–642, 2012.

[15] M. Bonnin, "Amplitude and phase dynamics of noisy oscillators," *International Journal of Circuit Theory and Applications*, 2016.

[16] E. M. Izhikevich, Y. Kuramoto *et al.*, "Weakly coupled oscillators," *Encyclopedia of mathematical physics*, vol. 5, p. 448, 2006.

[17] D. O. Hebb, *The organization of behavior: a neuropsychological theory*. Science Editions, 1962.

[18] F. C. Hoppensteadt and E. M. Izhikevich, "Pattern recognition via synchronization in phase-locked loop neural networks," *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 734–738, 2000.

[19] B. Scellier and Y. Bengio, "Equilibrium propagation: Bridging the gap between energy-based models and backpropagation," *Frontiers in computational neuroscience*, vol. 11, p. 24, 2017.

[20] G. Zoppo, F. Marrone, and F. Corinto, "Equilibrium propagation for memristor-based recurrent neural networks," *Frontiers in Neuroscience*, vol. 14, 2020.