How Frequency Injection Locking Can Train Oscillatory Neural Networks to Compute in Phase

Aida Todri-Sanial, Stefania Carapezzi, Corentin Delacour, Madeleine Abernot, Thierry Gil, Elisabetta Corti, Siegfried F. Karg, Juan Nunez, Manuel Jimenez, Maria J. Avedillo, Bernabe Linares-Barranco

***Abstract*—Brain-inspired computing employs devices and architectures that emulate biological functions for more adaptive and energy efficient systems. Oscillatory neural networks are an alternative approach in emulating biological functions of the human brain and suitable for solving large and complex associative problems. In this work, we investigate the dynamics of coupled oscillators to implement such oscillatory neural networks. By harnessing the complex dynamics of coupled oscillatory systems, we forge a novel computation model – information is encoded in the phase of oscillations. Coupled interconnected oscillators can exhibit various behaviors due to the strength of the coupling. Here, we present a novel method based on subharmonic injection locking (SHIL) for controlling the oscillatory states of coupled oscillators that allow them to lock in frequency with distinct phase differences. Circuit-level simulation results indicate SHIL effectiveness and its applicability to large-scale oscillatory networks for pattern recognition.**

***Index Terms*—oscillatory neural networks, subharmonic injection locking, oscillator dynamics, pattern recognition**

I. Introduction

INNOVATIONS in CMOS technology and the continuous scaling roadmap of transistors outlined by Moore's prediction have enabled today's powerful computers and handheld devices. Mere miniaturization of devices was initially sufficient to reduce transistors' area and power requirements, yet scaling for sub-100nm technology nodes was not enough [1], [2], [3], [4]. Two main paths were taken to change the device materials to reduce its parasitics, and change the device geometry for better channel control.

Nevertheless, despite advancements in the transistor device and fabrication technologies, CMOS faces physical barriers – as scaling is approaching a fundamental physical limit with the transistor channel length becoming comparable to the size of a handful of atoms. Such channel lengths lead to significant leakage currents and suffer from lower yield due to high process variations. Consequently, this would translate to more power consumption and more expensive chips, which would be an overkill to what Moore's law has been promising so far.

Despite the ongoing research on novel device geometries and channel materials, there is a tremendous effort to explore innovative non-Von Neumann computing architectures to meet the requirements of data-centric applications [5], [6], [7]. In the classical von Neuman architecture, data moves from memory to the processor, which for processing large datasets becomes infeasible as a large amount of power is consumed in data movement, hence, the memory-wall problem arises [5]. This problem is exacerbated for more data-centric applications, such as image segmentation and pattern recognition that require online training and learning, such as in the autonomous car technology or edge computing in IoTs [6].

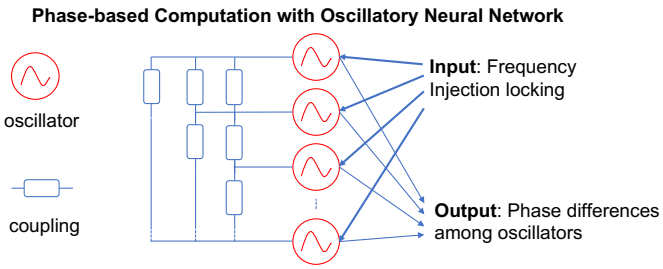


Fig. 1. Illustration of the proposed phase-based computation via coupled oscillators for implementing an oscillatory neural network (ONN).

Non-Von Neumann architectures like brain-inspired architectures based on neural networks have drawn a lot of interest as we gained more understanding of how the brain and neurons work. Neural networks aim to mimic the parallelism of the brain and their implementation in resource-intensive hardware such as CPUs, GPUs and TPUs have revolutionized AI applications [7]. For example, current CMOS implementations of neural networks such as Google's Tensor Processing Unit can offer up 86X more computations per watt [8]. Even though these systems are more power-efficient than a CPU due to their architecture, the CMOS implementations of neural networks will eventually face the problems described earlier (power consumption and memory-wall problems). Thus, ideally, one needs devices, materials, and computing architecture that offer the advantages of the biological system [11].

Novel brain-inspired neuromorphic architectures such as oscillatory neural networks (ONNs) [10] and coupled oscillator networks [22] have emerged as an alternative and energy-efficient architecture (Fig. 1), especially for associative memory applications [11]. They are inspired by neuroscience and understanding of brain activity – neurons in the human brain fire periodically [23] and synchronization of neurons might correspond to recognize faces or to learn of words and music [24], [25]. Highly inspired from Hopfield neural

This work was supported by the European Union's Horizon 2020 research and innovation program, EU H2020 NEURONN (www.neuronn.eu) project under Grant 871501.

A. Todri-Sanial, S. Carapezzi, C. Delacour, M. Abernot, and Th. Gil are with the Microelectronics Department, LIRMM, University of Montpellier, CNRS, Montpellier, France, e-mail: todri@lirmm.fr.

E. Corti and S. F. Karg are with the IBM Research Zurich, Switzerland.

J. Nunez, M. Jimenez, M. J. Avedillo and B. Linares-Barranco are with the Institute of Microelectronics of Sevilla (IMSE), CSIC, Spain.

Manuscript received Month, Day, Year; revised Month, Day, Year.

networks [9], ONNs compute based on the coupling weights connected among oscillators which also serves as its memory.

Mathematical concepts to computing with oscillators have been established since the late 90s [10], [12], [13]. The main attribute of ONN is its suitability for associative memory problems such as pattern recognition. However, ONNs have drawn a lot of interest in recent years due to their low-power computing capability, and it is being explored for other types of applications. Others have investigated ONNs for building Ising Machines [20], [21] or solving NP-hard combinatorial optimization problems, such as traveling salesman problem [19]. Moreover, ONNs have been experimentally validated for other applications such as image saliency detection [14], graph coloring [15], as digital filters for speeding up computations in convolutional neural networks [16], implementing Hopfield neural networks for pattern recognition applications [17], [18], and deep associative neural network implementation [51].

In this work, we investigate the dynamics of coupled oscillators as remarkably, such networks can implement a wide range of mathematical functions relating input states to output states. Furthermore, we introduce a subharmonic injection method which, when associated with learning rules for unsupervised learning such as Hebbian learning rules, ONN can learn efficiently by allowing oscillators to lock and synchronize. Our contributions can be summarized as 1) development of an analytical platform for deriving coupled oscillator dynamics, 2) development and implementation of subharmonic injection method for inducing oscillators to lock in the same frequency with distinctive phase differences, 3) reporting on ONN learning capability on various benchmarks for pattern recognition applications. These results will incite further investigations to study ONN scalability, capacity, and their specialized hardware implementations.

II. Introduction of Oscillatory Neural Networks

A. ONN Computing Principle

ONNs are inspired by the synchronization behavior of oscillators found in nature, such as in the human brain [11], [25]. Regularly firing (or spiking) neurons can be described as oscillators. At the level of large neuron ensembles, the synchronized activity of a large number of neurons gives rise to macroscopic oscillations, which can be observed as brain waves on electroencephalogram (EEG) [12], [13], [27]. By emulating neurons as oscillators, we investigate the frequency domain dynamics between neurons or the phase relations between oscillators.

ONNs differ from artificial neural networks (ANNs) or spiking neural networks (SNNs). In ANNs, information is encoded on the voltage amplitude of the neuron activation by calculating the weighted sum of the states of pre synapses [43]. In contrast, in SNNs, the time of individual spikes constitutes the basis of information encoding [44], [45], [46], [47], [48], [49]. An in-depth review on ANN and SNNs can be found in [50]. Alternatively, in an ONN, the information is encoded on the phase difference between oscillating neurons. Computing in phase allows for ultra-low power computing, as the signal voltage amplitude can be low [35], [36], [17]. To illustrate, phase difference 0^o between any two oscillators means they are in-phase, or their logic value is 0. When an oscillator has a phase difference of 180^o with respect to a reference oscillator, then the oscillator is out-of-phase, or it has a logic value 1.

Stable phase relations are obtained from the synchronization of the coupled oscillator dynamics. Stable phase patterns correspond to the memorized patterns in the network [11]. Oscillators synchronize at the same switching frequency and converge to a phase-locked pattern [11], [12], [27]. But to exploit such a computing principle, ideally, one needs uniform oscillators switching at the same frequency. Although at a small-scale ONN, oscillator uniformity can be somewhat attained depending on the oscillator design; however, to implement large-scale ONN and exploit its functionality, a robust method is needed to allow oscillators to lock and enable ONN learning. In the following subsections, we describe the building blocks of ONN, such as the implementation of the oscillator, coupling elements, and parameters used in this work.

B. Neuron – Oscillator Implementation

In literature, there are various approaches for implementing artificial neurons based on oscillating principle [11], [12], [27]. Spin-torque oscillators [28], [29], [30], [31], in which the magnetization of thin ferromagnetic layer is induced into sustained oscillations through the application of bias current or external magnetic field, have been shown to be capable of frequency locking, enabling the path toward neuromorphic computing. Other approaches are based on implementing micro electro-mechanical-systems (MEMS) resonators that are configured to self-oscillate as the oscillating element in the neural networks [32]. Digital architectures based on CMOS-based ring oscillators were developed by [35], [36] for pattern recognition. Alternatively, analog ONN architectures based on phase-locked loops (PLLs) as neurons were developed by [37].

Alternatively, novel devices and materials are explored to emulate biological oscillatory behavior. Oscillators based on phase change materials such as vanadium oxide (VO_2) have been recently developed. VO_2 relaxation oscillators, which rely on a precise switching between metallic and insulating states, have been successfully synchronized using resistive and capacitive coupling to implement associative memory operations and image recognition [17], [18], [39]. VO_2 device acts as a hysteresis resistor with two states, insulating and metallic, which changes phase around a critical temperature of 68^oC. In this work, we model such oscillators fabricated with phase change material VO_2 as in [18]. Devices have a 50-nm thick layer of VO_2 grown on top of 1 μm thermal SiO_2 on a silicon substrate. A two-terminal device is realized by etching rectangles into the VO_2 and depositing two Ni/Au contacts at both ends. The size of the VO_2 devices has 0.2 μm channel length and 1 μm channel width. The resistivity of the VO_2 in the insulator state was of ρ_{ins} = 10 Ω cm, while in the metallic state was of ρ_{met} = 0.8 Ω cm.

C. Synapse – Coupling Element Implementation

Precise modulation of the coupling conductance between neurons has a significant role in efficiently representing the

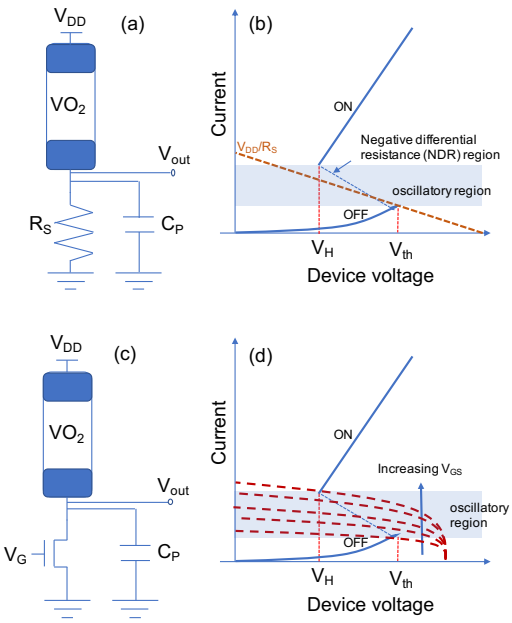


Fig. 2. Illustration of (a) VO_2 device with a resistor in series, R_S, (b) I-V characteristic of device, (c) VO_2 device with NMOS transistor in series, and (d) I-V characteristics with transistor's loadline. Gate voltage of transistor can be used to bias the VO_2 device at various current levels at the negative differential resistance (NDR) region allowing more oscillatory states.

TABLE I
LIST OF PARAMETERS USED FOR SIMULATIONS IN THIS WORK.

Parameter	Value
V_{DD}	2.5 V
V_G	2.5 V peak-to-peak (1.25 V DC)
F_{osc}	700 kHz (natural frequency)
T_{osc}	$1/F_{osc}$
R_{ins}	100.2 kΩ
R_{met}	0.99 kΩ
V_H	1 V
V_{th}	1.99 V
C_P	100 pF
C_C	0.05 pF
R_C	vary from 1 kΩ to 1MΩ
NMOS (W,L)	HSPICE default values (1e-04 m)

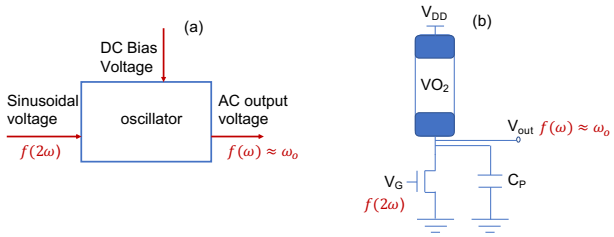


Fig. 3. Illustration of (a) sub-harmonic injection locking (SHIL) technique, where w_o is the natural frequency of oscillator, and (b) SHIL applied to VO_2 oscillator where the output signal oscillates at the half frequency of V_G sinusoidal signal when $w = w_o$.

synaptic weights in oscillatory neural networks. Several approaches have been explored, from metal-oxide resistive devices (RRAM), phase change memristors to novel memristor devices based on 1D/2D materials. In [33], memristive synaptic architectures have been proposed that showed their feasibility and effectiveness for both spiking and non-spiking neural networks. Phase change memory devices have been widely explored to implement the synaptic weights. In [34], RRAM-based oscillators and synapses are used to implement the analog functionality of ONNs. An ONN with programmable resistive synapses was implemented in a 28nm CMOS technology node for pattern recognition [38]. Alternatively, the coupling between oscillators can be realized by either, capacitive C coupling only, resistive R coupling only, or both resistive and capacitive, RC coupling [11], [17], [18], [39].

In this work, we focus on coupled oscillators to study their dynamics and oscillatory stable states in ONN for pattern recognition application. In biological neurons, the coupling has been measured by electrophysiological observation where the postsynaptic potential is less than 1 mV (i.e., representing a weak coupling) and the action potential is approximately 100 mV (i.e., representing a strong coupling) [11], [12]. In this work, we investigate RC coupling to understand the dynamics of coupled oscillators and their phase differences. Table 1 gives the list of parameters and their respective values that are used in this work.

III. Controlling the Dynamics of Coupled Oscillators

A. Analytical formulation

Here, we present our analytical method for deriving the phase dynamics of coupled oscillators. First, we illustrate the necessary mathematical formulations for a single oscillator, and then we turn to networks of coupled oscillators as the central focus. We describe how the concept of encoding in phase can be applied to oscillatory networks. We describe i) initialization or oscillator initial switching time and ii) frequency locking via load transistor gate voltage V_G for investigating ONN synchronization and phase dynamics.

Coupled oscillators can exhibit attractive dynamics that represent synchronized states. In Fig. 2a, the concept of driving the oscillator dynamics via supply voltage V_{DD} initialization with a resistive load [17], [18], [39] is illustrated. Note that oscillations happen when the voltage across the VO_2 device increases above a threshold voltage V_{th} and the device changes from insulating to metallic state with resistance R_{met}. When the voltage decreases below a lower threshold voltage, V_H, it changes back to the insulating state with resistance R_{ins}. The external capacitance C_P to the device ensures gradual transition (i.e., charging and discharging) of the voltage across the device. The series resistance with the VO_2 device provides the load line V_{DD}/R_S as shown in Fig.2b. The output voltage oscillates when the load line resides in the negative differential resistance (NDR) region. But, in contrast to previous works that control the supply voltage V_{DD} to initiate oscillations, we use an NMOS transistor where its gate voltage dynamically controls the load resistance, as illustrated in Fig. 2c. Controlling the load line via NMOS transistor's gate voltage enables us to explore more oscillatory states, hence, more states to encode information in an ONN (as shown in Fig. 2d).

An important feature of an oscillator that we exploit is how its phase responds to external signals – in this study, the NMOS transistor gate voltage V_G. External signals can be

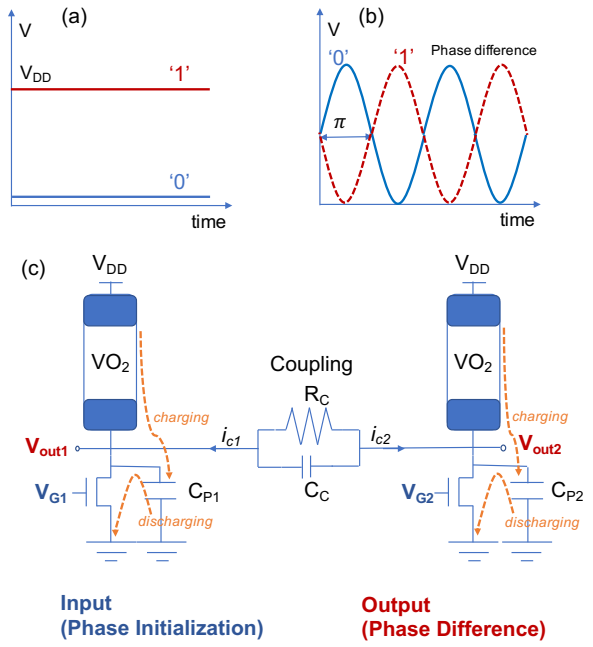


Fig. 4. Illustration of computing in phase, (a) logical 1 and 0 values based on voltage levels, (b) encoding of '1' and '0' on phase difference. A phase difference of 180^o is a logic '1' value and a phase difference of 0^o is a logic '0' value. (c) Two coupled oscillators inputs (transistors gate voltage) and outputs (VO_2 device output voltage).

unwanted, such as noise, jitter, or wanted, as in our case, where we inject a sinusoidal signal at the gate voltage for locking oscillators at the same frequency regardless of their individual non-uniform switching frequencies. Thus, we harness the V_G signal as the controlling knob for oscillatory states that enables oscillators to lock in the same frequency with distinct phases. In other words, we perform sub-harmonic injection locking (SHIL) via V_G to encode information on ONN and obtain the output as the phase differences among oscillators. Fig. 3 illustrates the sub-harmonic injection locking technique to drive oscillatory states through the signal applied at the transistor's gate voltage. The oscillation dynamics of a single oscillator (as in Fig. 2c) can be explained by the transition between two conductive states of the VO_2 device – insulating and metallic state. Therefore, the resistance R^{vo2} of the VO_2 device alternates between two values:

$$R^{vo2} = \begin{cases} R^{vo2}_{met} & \text{(charging)} \\ R^{vo2}_{ins} & \text{(discharging)} \end{cases} \tag{1}$$

By applying Ohm's law, we can analyze the dynamics of the oscillator output voltage in the two different states:

$$C_p \frac{\mathrm{d}V_{out}}{\mathrm{d}t} = \begin{cases} \frac{V_{DD}-V_{out}}{R^{vo2}_{met}} - I_{nmos} & \text{(charging)} \\ \frac{V_{DD}-V_{out}}{R^{vo2}_{ins}} - I_{nmos} & \text{(discharching)} \end{cases} \tag{2}$$

In the case of two coupled oscillators with resistance R_C and capacitance C_C (as in Fig. 4c), oscillations can be explained by the complementary transitions between the oscillators – one oscillator is in the metallic state while the other is in the insulating state, causing continuous charging and discharging of the external capacitors. This can be described as:

$$C_{P1} \frac{\mathrm{d}V_{out1}}{\mathrm{d}t} = \begin{cases} \frac{V_{DD1}-V_{out1}}{R^{vo2}_{met1}} - I_{nmos1} + i_{c1} & \text{(charging)} \\ \frac{V_{DD1}-V_{out1}}{R^{vo2}_{ins1}} - I_{nmos1} + i_{c1} & \text{(discharching)} \end{cases} \tag{3}$$

$$C_{P2} \frac{\mathrm{d}V_{out2}}{\mathrm{d}t} = \begin{cases} \frac{V_{DD2}-V_{out2}}{R^{vo2}_{met2}} - I_{nmos2} + i_{c2} & \text{(charging)} \\ \frac{V_{DD2}-V_{out2}}{R^{vo2}_{ins2}} - I_{nmos2} + i_{c2} & \text{(discharching)} \end{cases} \tag{4}$$

where i_{c1} and i_{c2} are the currents flowing through the RC coupling element and described as:

$$i_{c1} = -i_{c2} = \frac{V_{out2} - V_{out1}}{R_C} + (\frac{\mathrm{d}V_{out2}}{\mathrm{d}t} - \frac{\mathrm{d}V_{out1}}{\mathrm{d}t})C_C \tag{5}$$

We are using the NMOS transistors in strong inversion ($V_{GS} \geq V_T$) and either in saturation or linear regime, depending on the DC gate bias as shown in (6) and (7). By neglecting the short channel effects, the current flowing in the NMOS transistor $i \in \{1, 2\}$ can be expressed in saturation regime as:

$$V_{out_i} \geq V_{GS_i} - V_T \implies I_{nmos_i} = \frac{1}{2}\mu_n C_{ox} \frac{W}{L}(V_{GS_i} - V_T)^2 \tag{6}$$

or in linear regime as:

$$V_{out_i} < V_{GS_i} - V_T \implies I_{nmos_i} = \mu_n C_{ox} \frac{W}{L}(V_{GS_i} - V_T)V_{out_i} \tag{7}$$

where W and L are the width and length geometries of the NMOS transistor. C_{ox} is the oxide capacitance and V_T is the transistor threshold voltage. The voltage V_{GS} can be described by the injected wave signal with SHIL as:

$$V_{GS1} = V_{G1} = V_{G0} + A_1 sin(2\pi w_o^1 t) \tag{8}$$

$$V_{GS2} = V_{G2} = V_{G0} + A_2 sin(2\pi w_o^2 t) \tag{9}$$

where V_{G0} is the DC gate bias voltage, A_1, A_2, w_o^1 and w_o^2 are the amplitudes and frequencies of the sinusoidal signals injected at the gate voltages of each NMOS transistor. In this work, we apply SHIL to all NMOS transistors by injecting the same sinusoidal signal. Hence, V_{GS_i} signals have the same amplitude and frequency while their phases can differ.

As expressed in (1), VO_2 device resistance alternates between two values, metallic and insulating state. We choose to represent VO_2 resistance in time dependency as $R_1^{vo2}(t)$ and $R_2^{vo2}(t)$, respectively. Then, equations (3) and (4) can be formulated in a matrix form as:

$$C\frac{\mathrm{d}V}{\mathrm{d}t} = G_A(t)V + G_B(t)V_{DD} - I \tag{10}$$

where

$$G_A(t) = \begin{bmatrix} -\frac{1}{R_1^{vo2}(t)} - \frac{1}{R_C} & \frac{1}{R_C} \\ \frac{1}{R_C} & -\frac{1}{R_2^{vo2}(t)} - \frac{1}{R_C} \end{bmatrix} \tag{11}$$

$$G_B(t) = \begin{bmatrix} \frac{1}{R_1^{vo2}(t)} & 0 \\ 0 & \frac{1}{R_2^{vo2}(t)} \end{bmatrix} \tag{12}$$

$$C = \begin{bmatrix} C_{P1} + C_C & -C_C \\ -C_C & C_{P2} + C_C \end{bmatrix} \tag{13}$$

$$V_{DD} = \begin{bmatrix} V_{DD1} \\ V_{DD2} \end{bmatrix}; I = \begin{bmatrix} I_{nmos1} \\ I_{nmos2} \end{bmatrix}; V = \begin{bmatrix} V_{out1} \\ V_{out2} \end{bmatrix} \tag{14}$$

For a system of n coupled oscillators, $G_A(t)$ and $G_B(t)$ are n-by-n conductance matrices, C is the n-by-n capacitance matrix, V_{DD} and I are n-by-1 vectors representing the voltage and current biases of the oscillators. V is the n-by-1 vector that contains the output node voltages of the oscillatory network. Note that if the NMOS transistors are biased in linear regime (7), I is proportional to V and can be rewritten as:

$$I = \begin{bmatrix} \mu_n C_{ox} \frac{W}{L}(V_{GS1} - V_T) & 0 \\ 0 & \mu_n C_{ox} \frac{W}{L}(V_{GS2} - V_T) \end{bmatrix} \begin{bmatrix} V_{out1} \\ V_{out2} \end{bmatrix} \tag{15}$$

To obtain the output voltages, we solve the nonlinear system in (10). Solving the set of equations when subjected to a sinusoidal input voltage, oscillators lock in frequency which is close to the oscillator's natural (or free-running) frequency, $w_o = 1/T_o$. The output voltages will be in the form [40] of:

$$V_{out_i}(t) = A_i cos(2\pi w_o(t + \theta_i(t))) \tag{16}$$

where A_i is the oscillation amplitude and $\theta_i(t)$ is the instantaneous phase. To compute in phase, we determine the phase differences between oscillators. For example, once frequency locking occurs, we compute the phase difference between oscillator V_{out1} and V_{out2} as $\theta_2(t) - \theta_1(t)$ to represent the encoded output as either a logic 0 or 1. For example, as illustrated in Fig. 4a and Fig. 4b, a logic 0 is obtained when the phase difference between oscillators is close to 0, whereas a logic 1 is obtained when the phase difference between oscillators is 180^o. But oscillator dynamics and phase difference depend on several parameters, such as i) uniformity or variability of oscillators and their respective R_{met}, R_{ins} V_H, V_{th}, ii) applied supply voltages, V_{DD}, iii) coupling elements R_C and C_C to induce either weak or strong coupling, iv) applied gate voltage V_G frequencies, and v) applied learning rules and algorithm to train ONN. In section IV, we answer these questions by studying ONN dynamics for pattern recognition application.

B. ONN Numerical Solver

The system of equations (10) is difficult to solve analytically because of the VO_2 nonlinear behavior. However, it can be solved numerically (such as in Matlab) by approximating the derivative at time $t = k\,dt$ with integration time step of 1 ns:

$$\frac{\mathrm{d}V(t)}{\mathrm{d}t} \approx \frac{V(k\,dt) - V((k-1)\,dt)}{dt} \tag{17}$$

The insulating-metal transition (IMT) thresholds of the VO_2 device can be expressed as:

$$\begin{cases} V_{th+} = V_{DD} - V_H \\ V_{th-} = V_{DD} - V_{th} \end{cases} \tag{18}$$

To emulate the IMT behavior of VO_2 devices while solving the problem numerically, we keep track of the state of each device $i \in \{1, 2\}$ and update its resistance value. More precisely, we test the following sets of conditions:

$$\begin{cases} V_{out_i}[k] - V_{out_i}[k-1] \geq 0 \\ V_{out_i}[k] \geq V_{th+} \end{cases} \implies R_i^{vo2}[k+1] = R_{ins}^{vo2} \tag{19}$$

$$\begin{cases} V_{out_i}[k] - V_{out_i}[k-1] \leq 0 \\ V_{out_i}[k] \leq V_{th-} \end{cases} \implies R_i^{vo2}[k+1] = R_{met}^{vo2} \tag{20}$$

If none of these conditions is fulfilled, then the VO_2 device stays in the same resistive state as $R_i^{vo2}[k+1] = R_i^{vo2}[k]$. It is important to note that this approach ignores many features of the VO_2 device, such as its intrinsic time constant to switch from one state to another. Therefore, the frequency of oscillators differs from circuit simulations (i.e., HSpice). The frequency deviation of the numerical solver vs. circuit simulation is around 4.5% difference. Nevertheless, our model accurately captures the transient oscillatory dynamics and the exact phase difference between oscillators (Fig. 5), which are also the most important characteristics for computing with ONNs.

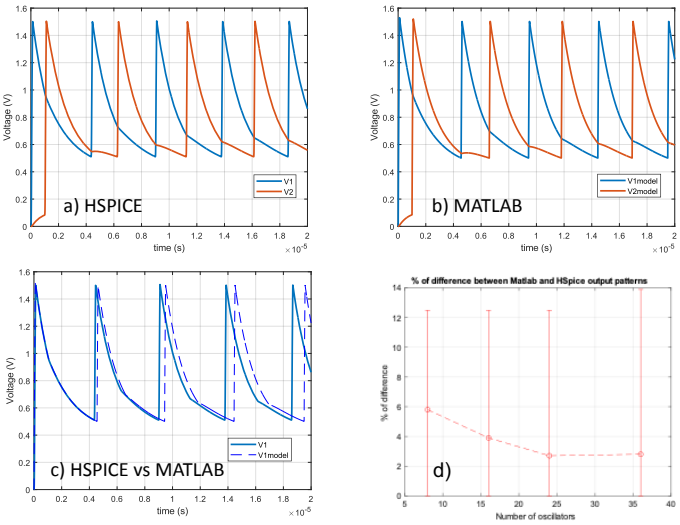


Fig. 5. Two coupled oscillators are simulated with a resistive load of 20 kΩ (as in Fig. 2a) and with R_C = 100 kΩ. The second oscillator is turned on 1 μs after the first one. a) HSpice simulations. b) Matlab simulations with integration time step of 1 ns. c) Comparison between the HSpice and Matlab waveforms. The transient dynamics and the phase relations at steady-state match whereas the frequencies do not. Matlab simulation shows a frequency deviation of 4.5 % with respect to the steady-state frequency f = 206.2 kHz computed with HSpice. d) Discrepancy between HSpice vs. Matlab for different size ONN.

IV. Controlling the Dynamics of Oscillatory Neural Networks

The thrust of this work is to use a network of driven coupled oscillators to recognize stored patterns based on their collective behavior and phase dynamics. First, we present a network of two coupled oscillators as a case study to illustrate how to compute with phase and highlight the importance of coupling parasitics, the initial state of the network (input delays) and learning rules. Second, we describe how the SHIL method can be injected via V_G to lock oscillators for pattern recognition.

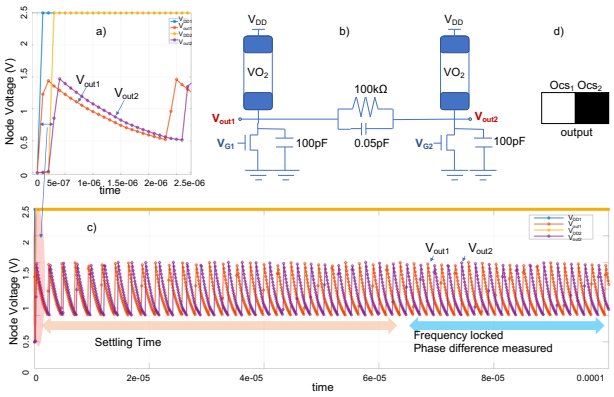


Fig. 6. Two coupled oscillator simulation. a) Initialization of oscillators. The second oscillator is initialized with an initial switching delay after the first oscillator. b) Schematic of two coupled oscillators with R_C=100 kΩ and C_C=0.05 pF. c) Oscillation cycles depicting the time for oscillators to settle and lock. Once the frequency is locked, then the phase difference between oscillators is obtained. d) Image representation of the phase difference. Oscillator with a phase difference of 180^o, the first oscillator is logic 0 (or white pixel), whereas the second oscillator is logic 1 (or black pixel).

A. Initialization of Oscillators

Initialization is the switching start time of oscillators. At first glance, one can think to start all the oscillators simultaneously, but this leads to incorrect or even chaotic ONN dynamics. Therefore, the switching start time of oscillators plays an important role in the ONN dynamics as it represents the input test pattern. Thus, the start of oscillators switching time represents the phases of the encoded test pattern. For example, in Fig. 6a, for two-coupled oscillators, the first oscillator turns on at T_1=0. The second oscillator turns on at T_2=230 ns representing 10% of oscillator period T_{osc}=2.3 μs and corresponding to an input test pattern of a phase of 0^o for the first oscillator and 36^o for the second oscillator. In other words, a white pixel for the first oscillator and a light-gray pixel for the second oscillator. Hence, the switching start time of oscillators must be set as an input switching delay, SW as a fraction of the oscillator period (0 to 50% T_{osc}) corresponding to an input phase $\theta = \frac{SW}{T_{osc}} 360^o$. Once oscillator dynamics settle in a few cycles, the output phase difference can be measured. The first oscillator has an output phase of 0^o (white pixel) and the second oscillator has a phase of 180^o (black pixel).

Initialization impacts the time needed for the oscillators to reach steady-state (i.e., settling time), but also it affects the final phase difference between oscillators, hence the success of identifying the correct stored pattern. Both initialization and coupling weights (resistive and capacitance coupling) can alter ONN settling time and final output phase. Fig. 7a represents the impact of initialization of two oscillators when an input switching delay of 10% T_{osc} is introduced. We observe cycle-by-cycle changes in the phase difference between the two oscillators until they reach a steady-state. The final oscillator waveforms are in 180^o phase difference, indicating the correct identification of the stored pattern. Fig. 7b shows the evolution of instantaneous phase differences between the two oscillators when varying the coupling resistance R_C, which shows the interdependence between initialization and coupling.

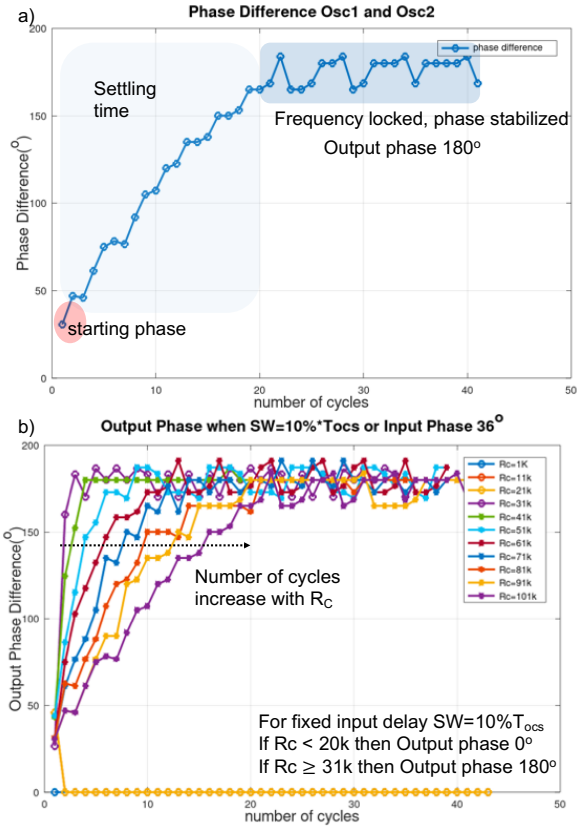


Fig. 7. a) The instantaneous phase difference between the two oscillators for each cycle. The second oscillator starts with an initial delay corresponding to 10% of oscillator period T_{osc} or an initial phase of 36^o. b) The instantaneous phase difference between the two oscillators for each cycle with varying coupling resistance R_C from 1k Ω to 101 kΩ.

B. Impact of Coupling

Coupling between oscillators can vary in topology (such as all-connected or subset of connected oscillators) and type of coupling (such as resistive-only, capacitive-only, or both resistive and capacitive). In this work, we investigate RC coupling between oscillators, as illustrated in Fig.6. The capacitive coupling is set to 0.05pF for all coupling, while the resistance coupling is varied to represent weak or strong coupling. Typically, the weak coupling is a large resistance value, whereas a strong coupling has a low resistance value. Fig. 8a shows the output phase difference between the two oscillators when resistive coupling varies from 1 kΩ to 100 kΩ – such range represents a strong coupling between oscillators. We notice for small resistance values (< 20 kΩ), oscillators are in-phase (phase difference 0^o) but with coupling resistances > 20 kΩ oscillators turn out-of-phase (phase difference 180^o). Simultaneously, we also vary the input switching delay of the second oscillator (i.e., switching delay (SW) from 10% to 40% of T_{osc}) while varying resistance coupling, as shown in Fig. 8b. We observe that with the increase of input switching delay, oscillators require a lower coupling resistance to change from in-phase to out-of-phase.

We also investigate large coupling resistance values from 100 kΩ to 1 MΩ – such range represents weak coupling between oscillators. Fig. 9a shows the output phase differences

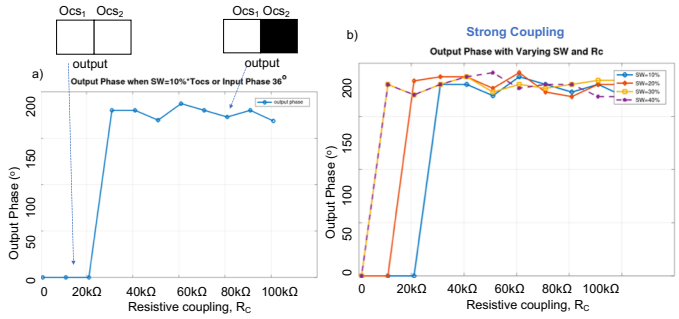


Fig. 8. Strong coupling simulations. a) The output phase difference between the two oscillators when varying resistive coupling R_C between 1 kΩ to 100 $k\Omega$ with 10% of T_{osc} input switching delay for the second oscillator. b) Output phase difference for strong coupling while varying the input switching delay on the second oscillator.

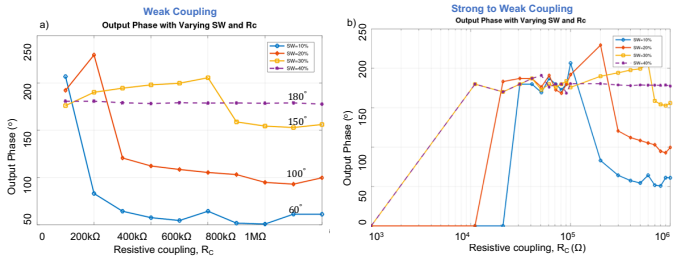


Fig. 9. a) Output phase difference with weak coupling of resistance values 100 $k\Omega$ to 1 $M\Omega$ for various switching delays. Phase dynamics differ with weak coupling providing more possibilities to encode different phase configurations between 0^o to 180^o that can enable various shades of grey for images. b) Overall phase difference evolution with resistive coupling (from strong to weak coupling) while varying input switching delay of the second oscillator.

between the two oscillators with resistance coupling values from 100 kΩ to 1 MΩ and various input switching delays. Interestingly, we observe that various output phases can be obtained for different input switching delays. Such weak coupling enables to encode more distinct and subtler phase differences such as 150^o, 100^o, or 60^o to allow more possibility to encode various phase configurations such as shades of grey in an image. The overall impact of resistance coupling is represented in Fig. 9b.

C. Sub-harmonic Injection Frequency Locking

Injection locking (IL) is an attractive phenomenon in non-linear coupled oscillators as it enables frequency locking among oscillators [29]. When an external signal is applied to an oscillator, the oscillator locks on to the external signal frequency whose frequency is close to the oscillator's natural frequency, also termed as fundamental harmonic IL. It is also possible for an oscillator to lock at a frequency that is an exact sub-multiple frequency of the externally applied signal – or sub-harmonic injection locking. In this work, we apply the sub-harmonic injection locking as an effective method to facilitate frequency locking among oscillators to reach steady-state so phase differences can be computed. We apply the external frequency signal to the gate voltage of the NMOS transistor. For a single oscillator case, we use the same configuration as in Fig. 2c with V_{DD}=2.5 V and C_p=100 pF. Fig. 10 shows

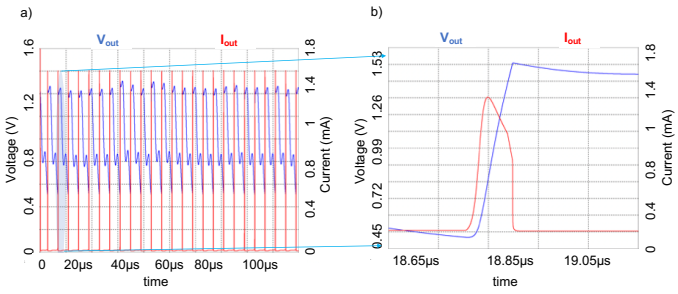


Fig. 10. Output voltage waveform for a single oscillator case. The output frequency follows the natural frequency of the oscillator of 250 kHz, whereas applied V_G is 500 kHz.

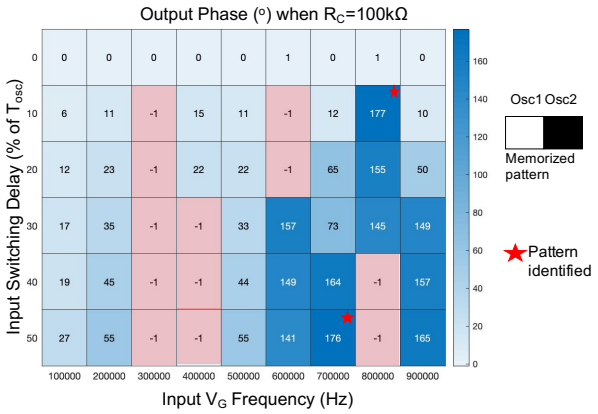


Fig. 11. Map of oscillatory states and output phase difference (^{o}) between two oscillators with V_G frequency injection locking from $100kHz$ to $900kHz$ as a function of input switching delay (0% to 50%) on the second oscillator with coupling resistance R_C=100 kΩ. The boxes with a value of -1 represent the chaotic oscillatory states where phase dynamics are unstable. V_G frequency tuning allows to identify the oscillatory states that retrieve the memorized pattern.

the output voltage waveform for a single oscillator where the applied V_G is of 2.5 V peak-to-peak (1.25 V DC level) at 500 kHz frequency. The output voltage oscillates with 250 kHz frequency, half of the V_G input frequency, as expected based on the SHIL method.

In the case of two coupled oscillators (Fig. 6b), we apply the same sinusoidal external signal at the gate terminals of both NMOS transistors. To understand the SHIL method, we vary the applied sinusoidal signal frequency from 100 kHz to 900 kHz while introducing an input switching delay to the second oscillator from 0% to 50% of T_{osc}. Both oscillators lock in frequency, which is a sub-multiple of the input V_G frequency. Measured phase differences are shown in Fig. 11. We observe at low frequencies, even though the oscillators lock in frequency, the correct pattern is not found as the phase differences are less than 180^o. We also notice oscillatory states with unstable phase differences from cycle to cycle, and they are marked with '-1'. We notice several cases where the correct pattern is found at higher frequencies, such as 700 kHz or 800 kHz. It is also important to note that SHIL can also be used as a frequency tuning method to find the correct pattern, such as when the second oscillator has an input delay of 10% T_{osc}. Overall, the SHIL method allows oscillators to

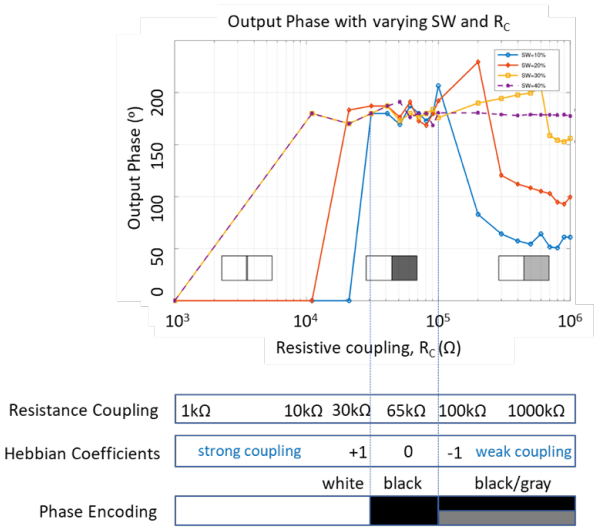


Fig. 12. Relationship between coupling resistance, phase encoding and Hebbian learning coefficients.

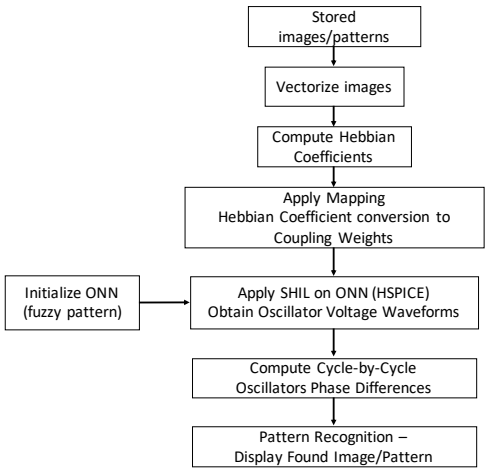


Fig. 13. Description of the pattern recognition flow for ONN based on phase dynamics. The pattern recognition flow is based on the sub-harmonic injection locking method with Hebbian learning rules.

lock and gives a wide range of oscillatory states (with incorrect and correct patterns). Next, we explore how to use the SHIL method together with learning rules to lead to oscillatory states with stored patterns only.

D. Learning Rules

We apply Hebbian learning to ONN, which is one of the most commonly used learning rules based on the dynamics of biological systems [41], [42]. The learning rules can memorize black-and-white images represented in a vectorized form as $\xi^k=(\xi_1^k, \xi_2^k,..., \xi_n^k)$ where k=1,...,m, and m is the number of stored patterns. ξ_i^k represents the oscillatory state of ith neuron. If two oscillators should oscillate in-phase such as $\xi_i^k=\xi_j^k$, then their phase difference is zero or $\theta_i=\theta_j$, and if $\xi_i^k=-\xi_j^k$ means oscillators are out-of-phase or $\theta_i=\theta_j+180^o$. The connection matrix element c_{ij} [10], [11], [12] using the Hebbian learning rule can be derived as:

$$c_{ij} = \frac{1}{n}\sum_{k=1}^{m} \xi_i^k \overline{\xi_j^k} \tag{21}$$

where n is the number of neurons or oscillators. By intertwining the SHIL method with Hebbian learning rules, we develop an ONN system that allows oscillators to collectively lock in frequency (sub-multiple of input V_G frequency) with distinct phase relations. Determining the coupling weights via Hebbian learning rules together with injection locking adds plasticity to the ONN system; in a sense, the system can change its parameters to learn the frequencies of the periodic input signal. The interest in combining injection locking with Hebbian learning is that the whole learning process is dynamic and does not require any external signal processing. It means that ONN can adapt its frequency to any periodic input. As shown in the next section, ONN can learn, achieve frequency locking with distinct phases and retrieve memorized patterns. Such learning shares similarities with the learning in biological neural networks [44].

To apply the learning rule to ONN, we must translate or map the Hebbian coefficients c_{ij} to resistive coupling values. To do so, we rely on our simulation insights from two-coupled oscillators to deduce a one-to-one mapping between Hebbian coefficients and resistance coupling. Thorough simulations with varying resistive coupling values (as in Fig. 9b) prompt a direct mapping approach between Hebbian coefficients and resistive coupling values. Fig. 12 depicts the relationship between the coupling resistance (i.e., strong and weak coupling), phase encoding (white for 0^o phase difference, black for 180^o phase difference) and learning Hebbian coefficients. As we are interested in encoding only black or white patterns, we apply non-complex Hebbian coefficients. Coefficients for out-of-phase encoding (or black) are represented in the range of 30 kΩ to 100 kΩ, and coefficients for in-phase (or white) encoding are represented in the range of 1 kΩ to 30 kΩ.

V. Pattern Recognition with ONN

We have developed a pattern recognition flow for ONN based on the sub-harmonic injection locking method applied with Hebbian learning rules. The flow includes several steps as described in Fig. 13. The first step is to construct a vectorized image of the stored pattern in the ONN. The number of stored patterns in ONN can vary with ONN size. Once the patterns are vectorized, the Hebbian learning rule (15) is applied to obtain the connection matrix with coefficients, c_{ij}. Based on the obtained Hebbian coefficients, a direct mapping is applied to translate the Hebbian coefficients to resistive coupling values. The initialization of the ONN is applied based on the pattern to be recognized, such as a fuzzy pattern or image. We apply a sub-harmonic injection method to the ONN circuit and obtain each oscillator waveform. We post-process each oscillator waveform in Matlab to get the phase difference among oscillators. The first oscillator serves as a reference, and the phase differences are measured with respect to the first oscillator. The recognized pattern or image is reconstructed from the computed phase differences. In the following subsections, we describe ONN pattern recognition

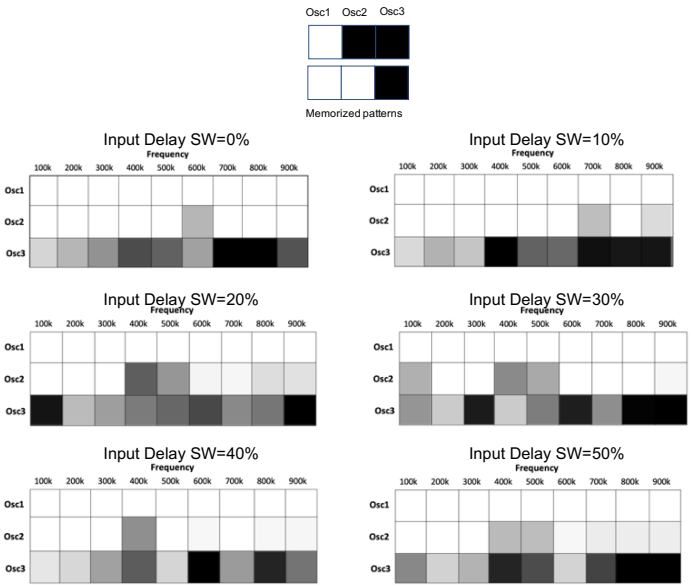


Fig. 14. Simulations of three coupled oscillator network. First oscillator switches at T_1=0, third oscillator switches at T_3=50% T_{osc} (T_{osc}=1.42 μs), whereas the initialization of the second oscillator varies from 0% to 50% of T_{osc}. The restored pattern for each initialization case while V_G input frequency is varying 100 kHz to 900 kHz. Both memorized patterns are restored, but for some states, phase differences are not completely 180^o, hence the grey states.

on ONN circuits with three coupled oscillators and 10x6 coupled oscillators.

A. Three Coupled Oscillators Study

To illustrate the phase dynamics in an ONN for pattern recognition, we study three-coupled oscillators, where all oscillators are identical. The list of parameters is in Table I. There are two stored patterns (as shown on the top of Fig. 14) and the Hebbian coefficients are derived as c_{12}=0, c_{13}=-1 and c_{23}=0, where white (in-phase) is encoded as '1' and black (out-of-phase) as '-1'. We apply the same sinusoidal input signal to V_G for each oscillator. Initialization is applied to all three oscillators, such as the first oscillator switches at T_1=0, third oscillator switches at T_3=50% T_{osc}, whereas the initialization of the second oscillator varies from 0% to 50% of T_{osc}. Fig. 14 shows the restored patterns while frequency tuning is applied using V_G with frequency from 100 kHz to 900 kHz. As both V_G frequency and switching delay are varied, the stored patterns are successfully retrieved for most cases. All the restored patterns are valid as either one of the memorized patterns is recognized, even though the phase differences are not entirely 180^o (grey pixels). We suspect that the applied direct mapping method for converting Hebbian coefficients to resistive coupling values could be the primary reason for such variance. We illustrate one of these restored patterns. The input conditions are shown in Fig. 15a, and the output voltage waveforms for each oscillator are plotted in Fig. 15b. Frequency locking (Fig. 15c) at 150 kHz, final phase differences and restored pattern (Fig. 15d) and frequency harmonics via Fourier transform (Fig. 15e) are shown. We observe that the initial cycles are unstable, hence the settling

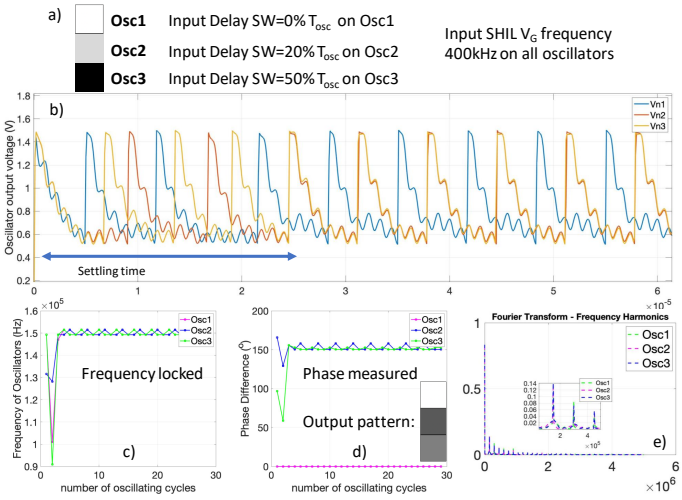


Fig. 15. a) Initialization of three coupled oscillators. b) Voltage waveform of oscillators showing their settling time. c) Frequency harmonics of oscillators showing they are locked at 150 kHz. d) Phase difference of oscillator per each cycle of simulation and output pattern, e) Fourier transform showing locked frequency harmonics of three oscillators.

time needed for oscillators to lock in the same frequency and reach a steady state. Once all three oscillators lock in frequency, they achieve distinct phases of 155^o for the second oscillator and 150^o for the third oscillator. The stored patterns are retrieved with distinct phase differences up to 155^o but not fully 180^o (or black pixel), hence, they are represented in dark grey pixel.

B. 10x6 Coupled Oscillators Study

Here we implement a 10x6 oscillatory neural network. Five patterns are stored with the digits 0, 1, 2, 3, and 4 (Fig. 16) and parameters used are in Table I. By applying the Hebbian learning rule, we obtain coefficients (with values -1 to 1) that we map to resistance coupling values (Fig. 12). The switching time of oscillators (or initialization) is applied to represent the test pattern (or a fuzzy digit). For example, as shown in Fig. 17, switching input delays on each oscillator are applied to show the fuzzy digit 0 as a start image. The applied V_G sinusoidal input signal at each oscillator has a frequency of 1.2 MHz. Such frequency is chosen based on our preliminary assessment with V_G frequency tuning from 200 kHz to 2 MHz. The applied frequency 1.2 MHz provides stable oscillations (not in a chaotic regime); however, other suitable frequencies such as 900 kHz or 1 MHz can also be used. The phase difference is computed relative to the first oscillator (top left-hand corner). This also explains that some restored images have an inverse color (white digit on black background) due to the phase difference with respect to the first oscillator. The cycle-by-cycle evolution of the restored digit 0 is represented in Fig. 17. We observe that the digit 0 is almost restored, although some oscillators (on top and bottom) did not settle at 180^o phase difference, hence, they are grey pixels.

Similarly, we also investigate the other stored digits and their restored patterns, as shown in Fig. 18. Overall, we notice that the memorized digits were restored, but some oscillators still

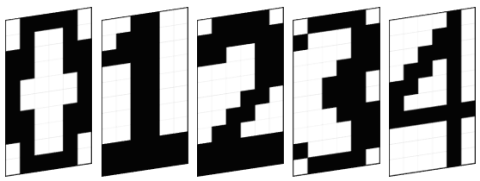


Fig. 16. Stored images of the digits 0, 1, 2, 3, and 4 on the 10x6 ONN.

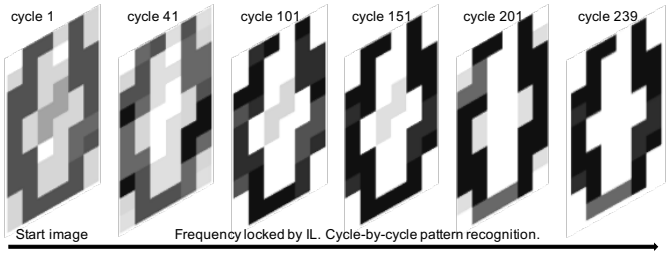


Fig. 17. Cycle-by-cycle representation of the pattern restoration process for digit 0. Sub-harmonic injection locking allows oscillators to lock in frequency, while phase varies from cycle to cycle, oscillator settle and pattern is restored.

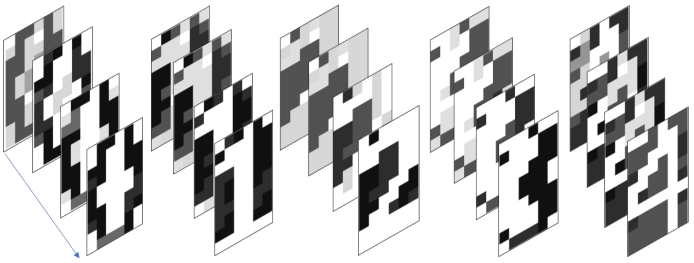


Fig. 18. 10x6 coupled oscillators network and phase difference with V_G frequency injection locking. Both memorized patterns are found and depending on the V_G frequency and chosen resistive coupling values.

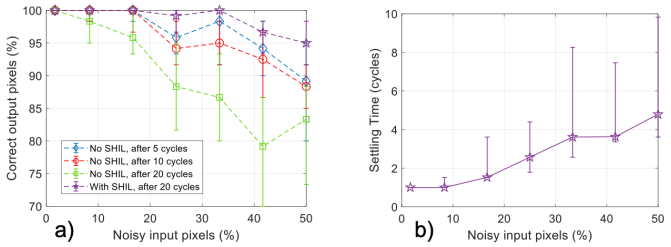


Fig. 19. a) The number of correct ouput pixels measured after 5, 10, 20 cycles without SHIL. Phase drifts induces a decrease in accuracy with time. (b) ONN settling time vs. noise pixels.

do not achieve a complete either in-/out-of-phase state. From our assessment, we find that the mapping process of determining the resistive coupling values plays a crucial role in the coupling between oscillators and requires further optimization. Nevertheless, the proposed method of frequency locking with Hebbian learning rules shows promising results in training 60 oscillators and restoring memorized patterns. Additionally, we measure the accuracy of the 10x6 ONN by varying the number of noisy input pixels from 0 to 50%. It is important to note that 50% is the maximum noise as more noise would revert the image (such as black pixels into white and vice versa). For example, a 60% noise would mean a reverted image with 10% noise. We randomly choose noisy pixels with values from -1 to +1 according to a uniform random distribution. We measure the number of correct output pixels after 5, 10, 20 cycles without SHIL, where data points are obtained after 30 trials each. We observe that accuracy decreases with time as oscillators' phases drift without SHIL. Phase drift means that the phase difference between oscillators changes from cycle to cycle. If phase measurements are done after 5, 10, or 20 cycles, phase drifts among oscillators increase. In contrast, when we apply SHIL to ONN, accuracy is improved, and no phase drifting is observed. This is because SHIL allows keeping oscillators locked once they are settled. Fig. 19 shows the accuracy and settling time vs. noise input pixels.

C. SHIL on ONN Hardware Experimental Results

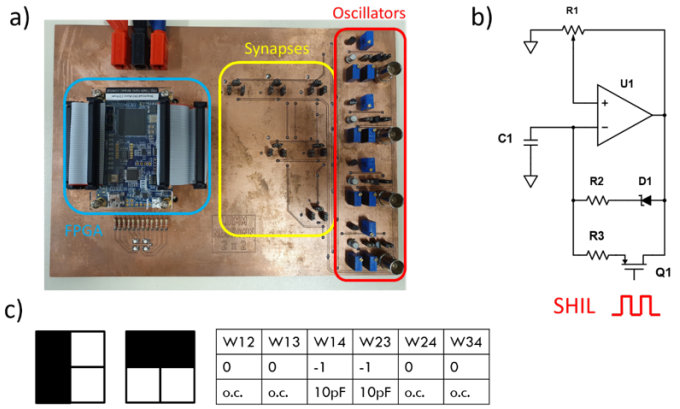
To validate our proposed SHIL on ONN approach, we developed a small scale ONN hardware on a printed circuit board (PCB) with off-the-shelf components. We test the SHIL method with four single-ended coupled oscillators (Fig. 20). ONN PCB is composed of four oscillators, six fully connected synapses (coupling elements) and an FPGA to interface the PCB with Matlab to issue commands and read the phases.

1) Two coupled oscillators experiment results: We couple two oscillators with a 100 k Ω resistor to assess if oscillators can lock to both 0^o and 180^o phase states. We notice that the 180^o phase state is less stable than the 0^o phase state. In Fig. 21a, we observe that oscillators do not lock to 180^o phase after initialization. We believe the type of waveform, noise, and variability impact the 180^o phase state stability. To mitigate this problem, we apply SHIL by injecting a high-frequency signal like in Fig. 21b, and the two oscillators lock to the desired 180^o phase state.

2) Four coupled oscillators experiment results: We perform a simple pattern recognition experiment using four coupled oscillators, as in Fig. 22. As in the two coupled oscillators experiment, we observe that the four oscillators do not lock to 180^o phases without SHIL due to oscillator non-uniformities and variabilities. We apply SHIL and ONN locks to either 0^o and 180^o phase states and retrieves the correct pattern. This shows that the SHIL method will be important to use in ONN hardware implementation where oscillator non-uniformities and variabilities are present.

VI. Discussion and Future Work

The research presented here started with a mathematical puzzle: How can we collectively control the dynamics of oscillators in an oscillatory neural network to learn and recognize in phase? In the course of addressing this issue with oscillatory neural networks, we found that the sub-harmonic injection locking mechanism can boost the ONN's ability to lock in frequency and by applying Hebbian learning rules, ONNs can learn patterns (by distinct phase differences between oscillators). We also observed that the initialization of the oscillator switching time has a strong impact on the ONN synchronization and learning. As such, this work can further incite research to the new found puzzle of how



W12	W13	W14	W23	W24	W34
0	0	-1	-1	0	0
o.c.	o.c.	10pF	10pF	o.c.	o.c.

Fig. 20. a) Custom PCB design to test SHIL on 4 coupled oscillators. FPGA controls oscillatorsâĂŹ initialization and reads output phases. ONN is fully connected with 6 synapses. b) Op-amp relaxation oscillator is based on an inverting Schmitt trigger. We apply the SHIL signal to the gate of Q1. c) Example of two training patterns and corresponding synaptic values obtained using the Hebbian rule. (o.c. stands for open circuit).

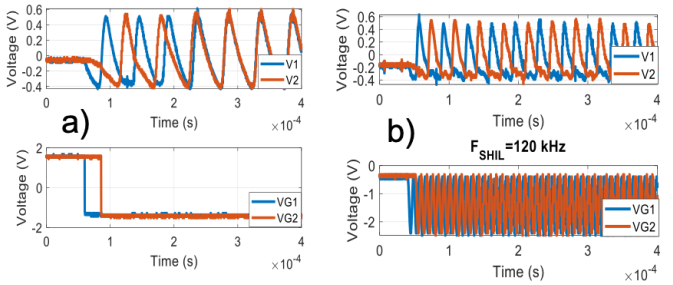


Fig. 21. a) Two oscillators coupled by R=100kΩ. a) Without SHIL, the two oscillators do not converge to a 180^o phase state. b) we apply SHIL by injecting a high-frequency signal @ 120 kHz enables 180^o phase locking between the two oscillators.

sub-harmonic controlled oscillations in ONN contribute to learning. The challenge is now to follow up the preliminary results presented here with a more detailed assessment of how the basic principle of the ONN learning algorithm (sub-harmonic injection locking with Hebbian or other learning rules) and more importantly, the mapping of Hebbian learning coefficients translate to resistance coupling weights can shed light on mathematical [52], oscillatory neural network learning and efficient ONN hardware implementation [53].

VII. Conclusion

In this work, we have shown that the phase dynamics of oscillators can be exploited to achieve pattern recognition with oscillatory neural networks. We have presented a new learning algorithm that leverages sub-harmonic injection locking and Hebbian learning rules to train oscillatory neural networks. Injecting a periodic input signal to the oscillators can be used to lock them in the same periodic or sub-periodic frequency, which can be used to compute phase differences among oscillators. We apply Hebbian learning rules to update coupling weights and translate the obtained Hebbian coefficients into resistance coupling values. The value of resistive coupling varies as a function of the sign of Hebbian coefficients. A

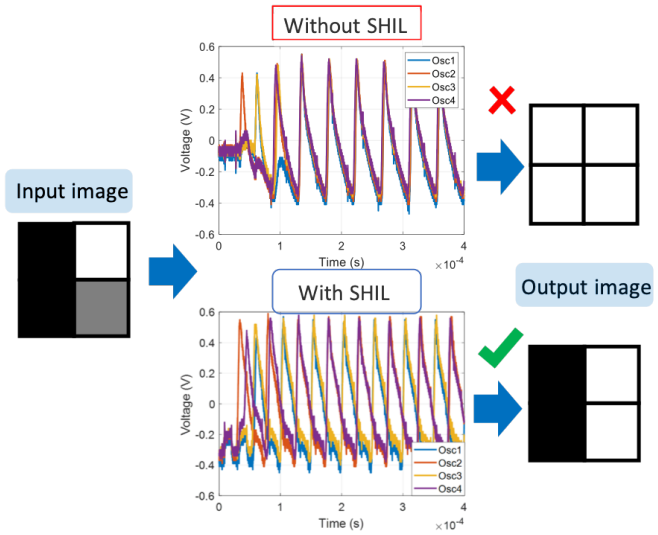


Fig. 22. ONN inference with a noisy input image. Without SHIL, ONN converges towards a wrong pattern. Oscillators do not converge towards 180^o phase due to oscillators non-uniformity. With SHIL applied at $2xf_{osc}$, 180^o phase can be achieved and ONN retrieves the correct pattern.

challenge remains on mapping more accurately the Hebbian coefficients to resistive coupling values. As a consequence, the mapping problem appears as a crucial parameter during the ONN design and training. Our study has shown that the new learning algorithm can recognize a large number of correlated input patterns based on the phase difference among oscillators, and it displays a good recognition capability on various size ONNs.

Acknowledgment

This work is supported by the European Union's Horizon 2020 research and innovation program, EU H2020 NEURONN (www.neuronn.eu) project under Grant No. 871501.

References

[1] K. Mistry, C. Allen, C. Auth, B. Beattie, D. Bergstrom, M. Bost, M. Brazier, M. Buehler, A. Cappellani, R. Chau, et al., A 45nm logic technology with high-k+ metal gate transistors, strained silicon, 9 cu interconnect layers, 193nm dry patterning, and 100% pb-free packaging, *IEEE International Electron Devices Meeting*, pp. 247-250, 2007.

[2] N. Loubet, T. Hook, P. Montanini, C.-W. Yeung, S. Kanakasabapathy, M. Guillom, T. Yamashita, J. Zhang, X. Miao, J. Wang, et al., Stacked nanosheet gate-all-around transistor to enable scaling beyond finfet, *IEEE Symposium on VLSI Technology*, pp. T230-T231, 2017.

[3] Heterogenous Integration Roadmap (HIR), https://eps.ieee.org/technology/heterogeneous-integration-roadmap.html, Online Edition 2020.

[4] M. M. S. Aly, M. Gao, G. Hills, C.-S. Lee, G. Pitner, M. M. Shulaker, T. F. Wu, M. Asheghi, J. Bokor, F. Franchetti, et al., Energy-efficient abundant-data computing: The n3xt 1,000 x, *Computer*, vol. 48, no. 12, pp. 24-33, 2015.

[5] S.-C. Lin, Y. Zhang, C.-H. Hsu, M. Skach, M. E. Haque, L. Tang, J. Mars, The architectural implications of autonomous driving: Constraints and acceleration, *ACM SIGPLAN Notices*, vol. 53, pp. 751-766, ACM, 2018.

[6] M. Merenda, C. Porcaro, and D. Iero, Edge machine learning for AI-enabled IoT devices: a review, *Sensors*, vol. 20, no. 9, 2533, 2020.

[7] A. James, The Why, What and How of Artificial General Intelligence Chip Development, *IEEE Transactions on Cognitive and Developmental Systems* doi: 10.1109/TCDS.2021.3069871

[8] N. Jouppi, C. Young, N. Patil, D. Patterson, Motivation for and evaluation of the first tensor processing unit, *IEEE Micro*, vol. 38, no. 3, pp. 10-19, 2018.

[9] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, *Proceedings of the National Academy of Sciences* 79, 2554-2558, 1982. doi:10.1073/pnas.79.8.2554

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

[11] G. Csaba, and W. Porod, Coupled oscillators for computing: a review and perspective, *Appl. Phys. Rev.*, vol. 7, 011302, 2020.

[12] E. M. Izhikevich, Computing with Oscillators, *Neural Networks*, 2000.

[13] F. C. Hoppensteadt and E. M. Izhikevich. Weakly Connected Neural Networks. *Applied Mathematical Sciences*, 126, Springer, 1997.

[14] N. Shukla, A. Parihar, E. Freeman, H. Paik, G. Stone, V. Narayanan, H.Wen, Z. Cai, V. Gopalan, R. Engel-Herbert, et al., Synchronized charge oscillations in correlated electron systems, *Scientific reports*, vol. 4, p. 4964, 2014.

[15] A. Parihar, N. Shukla, S. Datta, A. Raychowdhury, Synchronization of Pairwise-Coupled, Identical Relaxation Oscillators based on Metal-Insulator Phase Transition Devices: A Model Study, *Journal of Applied Physics*, 117:054902, 2015.

[16] E. Corti, J. Cornejo Jimenez, K.M. Niang, J. Robertson, K. Moselund, B. Gotsmann, et al., Coupled VO_2 oscillators circuit as analog first layer filter in convolutional neural networks. *Frontiers in Neuroscience* 15, 19, 2021. doi:10.3389/fnins.2021.628254

[17] E. Corti, B. Gotsmann, K. Moselund, I. Stolichnov, A. Ionescu, S. Karg, Resistive coupled VO_2 oscillators for image recognition, *IEEE International Conference on Rebooting Computing*, pp. 1-7, 2018. doi:10.1109/ICRC.2018.8638626

[18] E. Corti, A. Khanna, K. Niang, J. Robertson, K. Moselund, B. Gotsmann, S. Datta, S. Karg, Time-delay encoded image recognition in a network of resistively coupled vo on si oscillators, *IEEE Electron Device Letters*, 41, 629-632, 2020. doi:10.1109/LED.2020.2972006

[19] T. Endo, and K. Takeyama, Neural network using oscillators, *Electronics and Communications in Japan – Part III: Fundamental Electronic Science*, 75, 51-59, 1992. doi:https://doi.org/10.1002/ecjc. 352 4430750505

[20] J. Chou, S. Bramhavar, S. Ghosh, W. Herzog, Analog coupled oscillator based weighted Ising machine, *Scientific Reports*, 9:14786, 2019. https://doi.org/10.1038/s41598-019-49699-5

[21] T. Wang and J. Roychowdhury, Oscillator-based Ising machine, 2017. https://arxiv.org/abs/1709.08102

[22] D. Wang, D. Terman, Locally excitatory globally inhibitory oscillator networks, *IEEE transactions on neural networks*, vol. 6, no. 1, pp. 283-286, 1995.

[23] D. Vodenicarevic, Rhythms and oscillations: a vision for nanoelectronics, *PhD thesis*, Paris-Sud University, 2017.

[24] M. Romera, P. Talatchian, S. Tsunegi, F. A. Araujo, V. Cros, P. Bortolotti, J. Trastoy, K. Yakushiji, A. Fukushima, H. Kubota, et al., Vowel recognition with four coupled spin-torque nanooscillators, *Nature*, vol. 563, no. 7730, p. 230, 2018.

[25] M. B. Sachs, H. F. Voigt, and E. D. Young, Auditory nerve representation of vowels in background noise, *Journal of neurophysiology*, vol. 50, no. 1, pp. 27-45, 1983.

[26] T. C. Jackson, A. A. Sharma, J. A. Bain, J. A. Weldon, L. Pileggi, Oscillatory neural networks based on nano-oscillators and multi-level rram cells, *IEEE journal on Emerging and Selected Topics in Circuits and Systems*, vol. 5, no. 2, pp. 230-241, 2015.

[27] F. C. Hoppensteadt and E. M. Izhikevich, Synchronization of MEMS resonators and mechanical neurocomputing. *IEEE Transactions of Circuits and Systems -l: Fundamental Theory and Applications*, 48, 133, 2001.

[28] M. Sharad, D. Fan, K. Roy, Energy-efficient and robust associative computing with injection-locked dual pillar spin-torque oscillators. *IEEE Transactions on Magnetics*, 51, 1 2015.

[29] D. Fan, S. Maji, K. Yogendra, M. Sharad, K. Roy, Injection-locked spin hall-induced coupled-oscillators for energy efficient associative computing. *IEEE Transactions on Nanotechnology*, 14,1083, 2015.

[30] G. Csaba, and W. Porod, Computational study of spin-torque oscillator interactions for non-boolean computing applications. *IEEE Transactions on Magnetics* 49, 4447, 2013.

[31] K. Yogendra, D. Fan, Y. Shim, M. Koo, K. Roy. Computing with coupled spin torque nano oscillators. *21st Asia and South Pacific Design Automation Conference (ASP-DAC)* 312, 2016.

[32] A. Kumar and P. Mohanty, Autoassociative Memory and Pattern Recognition in Micromechanical Oscillator Networks, *Scientific Reports*, 7:41, 2017.

[33] I. Boybat, M. L. Gallo, S.R. Nandakumar, T. Moraitis, Th. Parnell, T. Tuma, B. Rajendran, Y. Leblebici, A. Sebastian, E. Eleftheriou, Neuromorphic Computing with Multi-Memresistive Synapses, *Nature Communications*, 9, 2514, 2018.

[34] T. C. Jackson, A. A. Sharma, J. A. Bain, J. A. Weldon, L. Pileggi, An RRAM-based Oscillatory Neural Network, *IEEE Latin American Symposium on Circuits and Systems*, 2015.

[35] G. Csaba, T. Ytterdal, W. Porod, Oscillatory neural network from ring oscillators. *International Workshop on Cellular Nanoscale Networks and their Applications* pp. 1-2, 2006.

[36] M. J. Cotter, Y. Fang, S. P. Levitan, D. M. Chiarulli, V. Narayanan, Computational architectures based on coupled oscillators. *IEEE Computer Society Annual Symposium*, pp. 130-135, 2014.

[37] R. Shi, T. C. Jackson, B. Swenson, S. Kar, L. Pileggi, On the design of phase locked loop oscillatory neural networks: Mitigation of transmission delay effects. *Proceedings of the Neural Networks (IJCNN)*, pp. 2039-2046, 2016.

[38] T. Jackson, S. Pagliarini, L. Pileggi, An Oscillatory Neural Network with Programmable Resistive Synapses in 28nm CMOS, |*IEEE ICRC*, pp.118-124, 2018.

[39] E. Corti, B. Gotsmann, K. Moselund, A. Ionescu, J. Robertson, S. Karg, Scaled Resistively Coupled VO_2 Oscillators for Neuromorphic Computing, *Solid State Electronics*, 2020.

[40] P. Vanassche, G. Gielen, W. Sansen, On the Difference between Two Widely Publicized Methods for Analyzing Oscillator Phase Behavior, *IEEE ICCAD*, 2002.

[41] N. Caporale and Y. Dan, Spike Timing-Dependent Plasticity: A Hebbian Learning Rule, *Annual Review of Neuroscience*, vol. 31, no. 1, pp. 25-46, 2008.

[42] L. Abbott, S. Nelson, Synaptic plasticity: taming the beast. *Nat Neurosci* 3, 1178-1183 (2000). https://doi.org/10.1038/81453

[43] J. D. Kendall and S. Kumar, The building blocks of a brain-inspired computer, *Applied Physics Reviews*, 7, 011305, 2020.

[44] R. Gutig, To spike, or when to spike?, *Current Opinion in Neurobiology*, vol. 25, pp. 143-139, 2014.

[45] W. Maass, Noise as a Resource for Computation and Learning in Networks of Spiking Neurons, *Proceedings of the IEEE*, vol. 102, no. 5, pp. 860-880, 2014, doi: 10.1109/JPROC.2014.2310593.

[46] A. Steimer, W. Maass, and R. Douglas, Belief Propagation in Networks of Spiking Neurons, *Neural Computation*, vol. 21, no. 9, pp. 2502-2523, Sep. 2009, doi: 10.1162/neco.2009.08-08-837.

[47] S. Shapero, M. Zhu, J. Hasler, and C. Rozell, Optimal Sparse Approximation With Integrate and Fire Neurons, *International Journal of Neural Systems*, vol. 24, no. 05, p. 1440001, 2014, doi: 10.1142/S0129065714400012.

[48] L. Rossello, V. Canals, A. Oliver, and A. Morro, Studying the Role of Synchronized and Chaotic Spiking Neural Ensembles in Neural Information Processing, *International Journal of Neural Systems*, vol. 24, no. 05, p. 1430003, 2014, doi: 10.1142/S0129065714300034.

[49] T. Serrano-Gotarredona, T. Masquelier, T. Prodromakis, G. Indiveri, and B. Linares-Barranco, STDP and STDP variations with memristors for spiking neuromorphic learning systems, *Frontiers in Neuroscience*, vol. 7, 2013, doi: 10.3389/fnins.2013.00002.

[50] H. Paugam-Moisy and S. Bohte, Computing with Spiking Neuron Networks, *Handbook of Natural Computing*, Springer-Verlag, Chapter 10, pp. 335-376, 2012. doi: 10.1007/978-3-540-92910-9-10.

[51] J. Liu, M. Gong, H. He, Deep associative neural network for associative memory based on unsupervised representation learning, *Neural Networks*, 2019.

[52] C. Delacour and A. Todri-Sanial, Mapping Hebbian Learning Rules to Coupling Resistances for Oscillatory Neural Networks, working paper or preprint, https://hal-lirmm.ccsd.cnrs.fr/lirmm-03197299, 2021.

[53] M. Abernot, T. Gil, M. Jimenez, J. Nunez, M. J. Avedillo, B. Linares-Barranco, T. Hardelin, T. Gonos, A. Todri-Sanial, Digital Implementation of Oscillatory Neural Network for Image Recognition Application, working paper or preprint, https://hal-lirmm.ccsd.cnrs.fr/lirmm-03185020, 2021.