A mixed-signal oscillatory neural network for scalable analog computations in phase domain

Corentin Delacour, Stefania Carapezzi, Gabriele Boschetto, Madeleine Abernot, Thierry Gil, Nadine Azemard, Aida Todri-Sanial

A mixed-signal oscillatory neural network for scalable analog computations in phase domain

NEUROMORPHIC
Computing and Engineering

PAPER

A mixed-signal oscillatory neural network for scalable analog computations in phase domain

Corentin Delacour[1,*], **Stefania Carapezzi**[1], **Gabriele Boschetto**[1], **Madeleine Abernot**[1], **Thierry Gil**[1], **Nadine Azemard**[1] and **Aida Todri-Sanial**[1,2]

[1] Microelectronics Dept., LIRMM, University of Montpellier, CNRS, Montpellier, France
[2] Electrical Engineering Dept., Eindhoven Technical University, Eindhoven, The Netherlands
[*] Author to whom any correspondence should be addressed.

E-mail: corentin.delacour@lirmm.fr

Keywords: oscillatory neural network, analog computing, mixed-signal design, NP-hard problems

Supplementary material for this article is available online

OPEN ACCESS

RECEIVED
13 March 2023

REVISED
6 July 2023

ACCEPTED FOR PUBLICATION
24 July 2023

PUBLISHED
4 August 2023

Abstract

Digital electronics based on von Neumann's architecture is reaching its limits to solve large-scale problems essentially due to the memory fetching. Instead, recent efforts to bring the memory near the computation have enabled highly parallel computations at low energy costs. Oscillatory neural network (ONN) is one example of in-memory analog computing paradigm consisting of coupled oscillating neurons. When implemented in hardware, ONNs *naturally* perform gradient descent of an energy landscape which makes them particularly suited for solving optimization problems. Although the ONN computational capability and its link with the Ising model are known for decades, implementing a large-scale ONN remains difficult. Beyond the oscillators' variations, there are still design challenges such as having compact, programmable synapses and a modular architecture for solving large problem instances. In this paper, we propose a mixed-signal architecture named *Saturated Kuramoto ONN* (SKONN) that leverages both analog and digital domains for efficient ONN hardware implementation. SKONN computes in the analog phase domain while propagating the information digitally to facilitate scaling up the ONN size. SKONN's separation between computation and propagation enhances the robustness and enables a feed-forward phase propagation that is showcased for the first time. Moreover, the SKONN architecture leads to unique binarizing dynamics that are particularly suitable for solving NP-hard combinatorial optimization problems such as finding the weighted Max-cut of a graph. We find that SKONN's accuracy is as good as the Goemans–Williamson 0.878-approximation algorithm for Max-cut; whereas SKONN's computation time only grows logarithmically. We report on Weighted Max-cut experiments using a 9-neuron SKONN proof-of-concept on a printed circuit board (PCB). Finally, we present a low-power 16-neuron SKONN integrated circuit and illustrate SKONN's feed-forward ability while computing the XOR function.

1. Introduction

1.1. Oscillatory neural networks (ONNs)

The synchronization of oscillators is a fascinating phenomenon studied for a long time. In the XVII century, Huygens noticed that two identical clocks attached to the same beam synchronize to an anti-phase state where the two pendulums have the same frequency and move in opposite directions [1]. It is in the 1950s that scientists imagined computing with coupled oscillators. The main motivation at that time was to replace the bulky and slow vacuum tubes used for digital computations. In 1954, von Neumann proposed in a patent to use LC resonant circuits driven by a harmonic signal to compute digital functions [2]. At the same time in Japan, Goto developed a similar paradigm called the *parametron*, which consists of an LC resonant circuit

oscillating at one-half the driving frequency. Using transformers as coupling elements, Goto was able to build multiple large-scale digital computers with up to 9600 parametrons in 1958 [3]. However, parametrons became obsolete in the 1960s due to the advent of transistors that were faster and more scalable [4].

With the emergence of artificial intelligence and neural networks, researchers brought back the idea of phase computing to solve complex tasks like pattern recognition [5–8] and NP-hard combinatorial optimization problems (COPs) [9–12] that are challenging for conventional digital electronics. Inspired by Hopfield neural networks [13] and by the Kuramoto model [14], Aoyagi [15], Hoppensteadt and Izhikevich [16] have proposed a computing paradigm called ONN that models the phase dynamics of coupled oscillators. ONNs are particularly interesting for solving COPs as they are dynamical systems that converge *naturally* to fixed points corresponding to the minima of some energy landscape. In other words, there is no external control that makes the neuron state evolve. Instead, all the phases evolve in parallel and in continuous time, enabling fast and energy-efficient inferences in the analog domain [17].

1.2. State-of-the-art ONNs

Recently, a new interest in ONN has risen thanks to the emergence of novel oscillating devices that enable the fabrication of efficient ONNs [22]. Such as, spin-torque and spin Hall devices [23, 24], micro-electromechanical systems [25, 26], and transition metal oxide devices are all candidates for implementing ONNs using their oscillatory behavior and synchronization properties [6, 10, 27–29]. Beyond-CMOS devices are promising as they generally allow a compact oscillator design using a single device that could be scaled down to the nanoscale. Nevertheless, CMOS-based ONNs benefit from the mature CMOS technology which enables rapid ONN development and facilitates its co-integration with conventional digital circuits [5, 12, 20, 21]. In this work, we focus on ONNs that compute in phase domain, i.e. with neurons that oscillate at the same frequency. However, note that it is also possible to compute with various frequencies [30, 31]. Regardless of the technology, we identify three important criteria for designing a competitive ONN that computes in phase domain. It should have:

(1) Homogeneous oscillating frequencies
(2) Compact and linearly programmable signed synapses
(3) A scalable architecture

Even with the mature CMOS technology, achieving perfect matching between hundreds of oscillators is unfeasible for small-scale oscillators due to device-to-device variations. Hopefully, some techniques can overcome frequency mismatches such as calibration or sub harmonic injection locking (SHIL). SHIL consists in driving the oscillators with a harmonic signal that can lock to a Fourier harmonic of the oscillating signal [32]. In case of large frequency mismatches, the injection of a strong SHIL signal ensures phase locking among the oscillators [33, 34]. The second criterion promotes synapses that are compact, programmable with signed weights, and have a value proportional to their conceptual weight. Some architectures can lead to a non-linear mapping between the conceptual weights and their hardware implementation [35], or even be unknown due to the high complexity of the dynamics. Finally, the ONN architecture must be scalable to compete with conventional computing and solve large-scale problems involving thousands or millions of synapses. For this reason, we believe that the ONN architecture should be modular, i.e. to support the interconnection of smaller sub-ONNs to build a larger system and avoid the implementation of a fully-connected network.

Table 1 presents the state-of-the-art ONN architectures and their features. We only consider ONN computing in the phase domain and based on electrical oscillators. For solving COPs, a general approach is to map the input graph to the ONN where vertices are oscillators, and edges are synapses. Some architectures such as [10, 20] are dedicated to finding the maximum cut of a graph with weights of the same sign, as the synapses only implement negative weights. The main drawback is that both coupling capacitors and resistors are required to program negative and positive weights, respectively. Other architectures using differential LC oscillators enable signed weights using resistors only [18, 19] but are not scalable on chip due to the bulky LC tanks and resistors. Digital ONNs are promising as they are scalable and modular, as demonstrated by Moy *et al* with their 1968 ring oscillators chip [12]. A recent promising fully-analog architecture for solving COPs has also been proposed by Graber *et al* [21] that consists of 400 oscillators coupled with nearest neighbors.

1.3. A mixed-signal approach

In this work, we introduce a new mixed-signal ONN architecture, named *Saturated Kuramoto ONN* (SKONN) that leverages both analog and digital domains to satisfy the three ONN design criteria. SKONN takes inspiration from the state-of-the-art analog ONN architectures for which the dynamics evolve *naturally* in continuous time and can easily be described by phase models like Kuramoto's or Izhikevich's [14], thus

Table 1. State-of-the-art ONN architectures.

	Goto [3]	Jackson *et al* [5]	Wang *et al* [18]	Chou *et al* [19]	Bashar *et al* [20]	Dutta *et al* [10]	Moy *et al* [12]	Graber *et al* [21]	This work
Size	9600	100	240	4	30	8	1968	400	16
Oscillator	Analog LC	Digital	Analog LC	Analog LC	Analog relaxation	Analog relaxation (PTNO)	Ring Oscillator	Analog differential	Analog relaxation
SHIL or Calibration	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Coupling	Transformers	Resistors	Resistors	Resistors	Capacitors	Capacitors Resistors	Transmission gates	Current sources with DACs	Capacitors
Signed weights	No	Yes	Yes	Yes	No	No	Yes	Yes	Yes
Weight precision	1 bit	5 bits	8 bits	5 bits	1 bit	—	5 levels	6 bits	5 bits
Modular	Yes	—	Yes	—	—	—	Yes	Yes	Yes
Feedforward	Yes	Yes	—	Yes	No	No	No	Yes	Yes
Initial phase control	Yes	Yes	—	—	—	—	—	No	Yes
Application	Digital logic	Pattern recognition	COP	COP	COP	COP	COP	COP	COP Image processing

facilitating the exploration of potential applications. SKONN's main novelty consists in setting the computation and propagation in the analog and digital domains, respectively. Figure 1 illustrates SKONN with 4 fully-coupled neurons. Digital propagation has several advantages such as greater noise immunity, a higher fan-out, and smoother interfacing with other digital circuits. Moreover, the separation between the computation and propagation induces a natural implementation of feed-forward synapses that have never been implemented in literature, to the best of our knowledge.

The paper is organized as follows. First, SKONN architecture is described and its dynamics are derived, highlighting the link between the Kuramoto model and SKONN, and its ability to solve NP-hard weighted Max-cut problems. Then, we introduce a SKONN-printed circuit board (PCB) proof-of-concept that solves weighted Max-cut problems with nine nodes. A 16-neuron 65 nm-ASIC chip recently taped out is also presented, demonstrating SKONN feed-forward ability with a XOR example. Finally, we report on SKONN's performance scaling in solving the weighted Max-cut problem, and benchmark with state-of-the-art solvers on G-SET instances.

2. Methods

2.1. SKONN architecture overview

2.1.1. A mixed-signal oscillating neuron

A SKONN neuron consists of a relaxation oscillator producing analog and digital oscillations with period T at its input and output nodes, respectively. Figure 2(a) shows the block diagram of the oscillating neuron. It consists of a hysteresis circuit that commands a shaper block to charge and discharge a capacitor C_L with constant current I_{bias}. The voltage across the capacitor V_i^{in} is fed back to the hysteresis comparator that switches between V_{DD} and 0 when V_i^{in} reaches the thresholds V_H and V_L, thus producing oscillations. V_i^{out} holds the phase state in the digital domain, whereas V_i^{in} is the analog evolution of the oscillation. Note that the input impedance of the oscillator is purely capacitive in the ideal case so that any charge sent to the input node causes an instantaneous phase shift. The analog waveform V_i^{in} supports the computation and is separated from the digital propagation V_i^{out}, enabling a feed-forward propagation of the phase information.

Figure 2(c) shows an example of feed-forward propagation between two oscillators. The computation occurs in the analog domain at the input node of neuron i that gathers the output signals from neuron j. The oscillator output signal is a square digital-like signal that carries the oscillator state and evolves until the phase dynamics settle to a fixed point. Choosing a triangular waveform at the analog input leads to simple yet rich phase dynamics that are similar to the Kuramoto model, which is known to have interesting computational properties [14]. Moreover, it skips the use of bulky LC tanks needed for producing sinusoidal oscillations. The neuron voltage dynamics are expressed in appendix A for completeness, although this work rather focuses on phase dynamics that are more suitable to study phase-based ONNs. The comparison between SKONN circuit dynamics and phase dynamics is presented in appendix D.

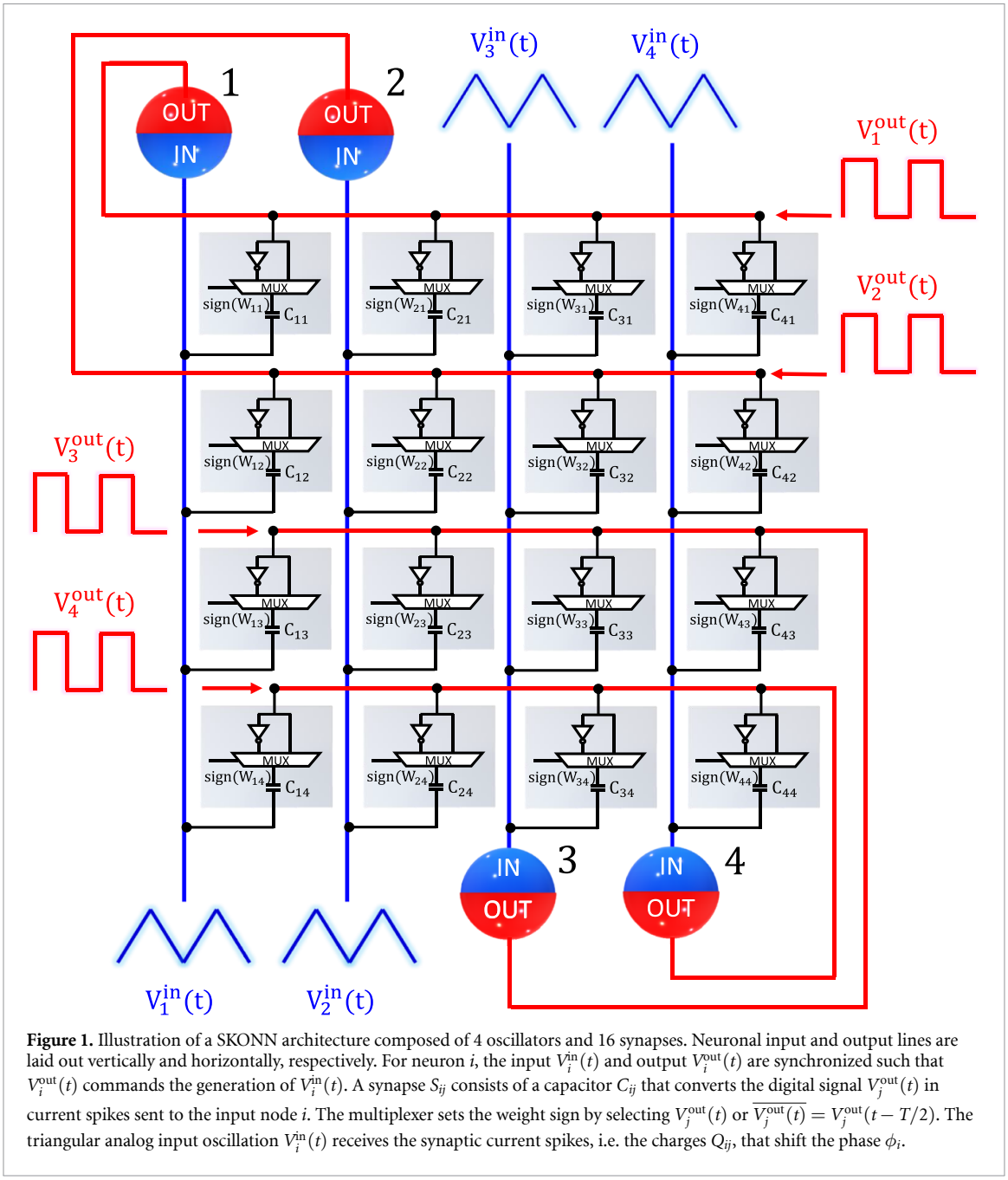

Figure 1. Illustration of a SKONN architecture composed of 4 oscillators and 16 synapses. Neuronal input and output lines are laid out vertically and horizontally, respectively. For neuron i, the input $V_i^{in}(t)$ and output $V_i^{out}(t)$ are synchronized such that $V_i^{out}(t)$ commands the generation of $V_i^{in}(t)$. A synapse S_{ij} consists of a capacitor C_{ij} that converts the digital signal $V_j^{out}(t)$ in current spikes sent to the input node i. The multiplexer sets the weight sign by selecting $V_j^{out}(t)$ or $\overline{V_j^{out}(t)} = V_j^{out}(t - T/2)$. The triangular analog input oscillation $V_i^{in}(t)$ receives the synaptic current spikes, i.e. the charges Q_{ij}, that shift the phase ϕ_i.

2.1.2. Synaptic design and weight sign

A SKONN synapse S_{ij} consists of a capacitor C_{ij} that transmits current pulses, i.e. charges Q_{ij}, from the output of oscillator j to the input of oscillator i. C_{ij} can easily be programmed using a capacitor bank and a register, as shown in figure 2(b). Instead of propagating the sensitive analog signal, SKONN only transmits the oscillator phase information in a robust manner. The digital output voltage V_j^{out} is applied to C_{ij} that creates current spikes holding the phase information ϕ_j. The synaptic spike train can be expressed as follows:

$$I_{ij} = C_{ij}\left(\frac{dV_j^{out}}{dt} - \frac{dV_i^{in}}{dt}\right) \quad (1)$$

The synaptic capacitor can be thought of as a digital-to-analog phase converter. The synaptic weight consists of the capacitance value C_{ij} that linearly modulates the charge sent to the oscillating input node i as $Q_{ij} = C_{ij}V_{DD}$, thus inducing phase shifts in the oscillation i as shown in figure 2(c). To implement a negative weight, the complementary of V_j^{out} defined as $\overline{V_j^{out}(t)} = V_j^{out}(t - T/2)$ is selected using a multiplexer and applied to C_{ij}. Compared to resistors, synaptic capacitors have several advantages for upscaling the ONN:

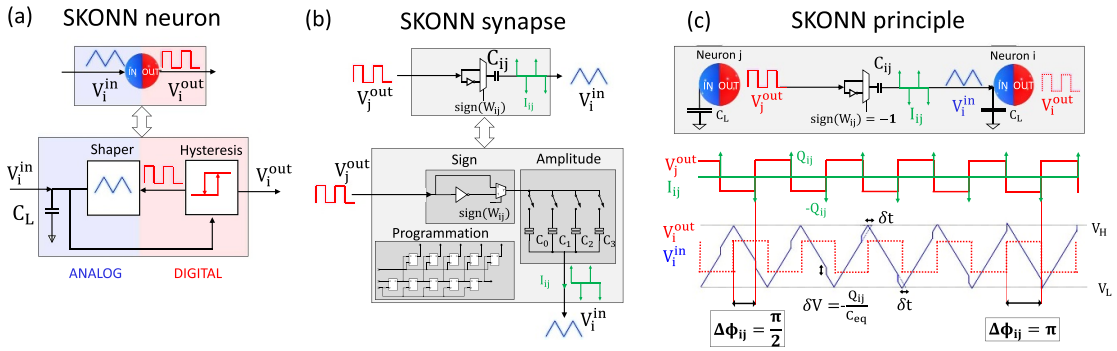

Figure 2. (a) A SKONN neuron is a relaxation oscillator composed of two blocks: the hysteresis circuit that holds the neuron state and drives the shaper stage. The latter produces an analog triangular waveform at the input whereas the hysteresis block outputs a digital waveform. (b) A SKONN synapse consists of a capacitor bank setting the weight amplitude. A multiplexer selects V_j^{out} or $\overline{V_j^{\text{out}}}$ to set the weight sign. (c) SKONN principle of computation illustrated with a negative weight. The multiplexer selects $\overline{V_j^{\text{out}}}$ that is fed into C_{ij}, creating current spikes I_{ij} aligned with the rising and falling edges of $\overline{V_j^{\text{out}}}$. The injected charges $\pm Q_{ij}$ to V_i^{in} induce voltage jumps $\pm\delta V$ that cause time shifts δt. After a few cycles, the two oscillators lock to $\Delta\phi_{ij} = \pi$.

(1) ONN computation models are generally based on the weak coupling assumption [9, 14] and necessitate weak synaptic signals. This means the ONN needs either large coupling resistors or small capacitors, the latter being much more scalable in a chip.
(2) For a limited neuron output strength, the only way of increasing the synaptic fan-out is to reduce the synaptic current, which again would lead to bulky resistors or smaller capacitors in the case of SKONN.

2.2. SKONN phase dynamics

2.2.1. 2 coupled oscillators

SKONN computing mechanism is illustrated in figure 2(c) with the case of a neuron j feeding its phase to another neuron i in a feed-forward manner with a negative weight. This means that $\overline{V_j^{\text{out}}}$ is selected by the multiplexer and applied to C_{ij}, thus creating current spikes $+Q_{ij}\delta(t)$ and $-Q_{ij}\delta(t - T/2)$ that are aligned with the rising and falling edges of $\overline{V_j^{\text{out}}}$, respectively. Each injected charge $\pm Q_{ij}$ induces a voltage jump $\delta V = \pm Q_{ij}/C_{\text{eq}}$ at the input node, with $C_{\text{eq}} = C_L + C_{ij}$. As V_i^{in} is a triangular waveform, δV provokes a time shift $\delta t = \pm C_{\text{eq}}\delta V/I_{\text{bias}}$, where $\pm$ indicates here the sign of V_i^{in}'s slope. Knowing the period of the triangular oscillation $T = 2C_{\text{eq}}\Delta V/I_{\text{bias}}$ where ΔV is V_i^{in}'s peak-to-peak amplitude, we can then express the phase shift related to a single current spike:

$$\begin{aligned}\delta\phi &= 2\pi\frac{\delta t}{T}\\ &= \pi\frac{\pm Q_{ij}}{C_{\text{eq}}\Delta V}\\ &= \pi\frac{\pm C_{ij}}{C_{\text{eq}}}\frac{V_{DD}}{\Delta V}\\ &\approx \pi\frac{\pm C_{ij}}{C_L}\frac{V_{DD}}{\Delta V}\ \text{ if } C_L >> C_{ij}.\end{aligned} \tag{2}$$

SKONN's unique feature consists of this simple relation (2) between coupling capacitors and phase shift, thus enabling well-controlled phase dynamics and a precise weight mapping to the coupling capacitor C_{ij}. As we will see later, the quantity $\beta_0 = |\delta\phi/Q_{ij}|$ provides the neuron phase sensitivity with respect to the charge perturbation. It is linked to the phase perturbation vector (PPV) of the oscillator which is key for deriving SKONN's phase dynamics [33]. SKONN's PPV is defined and derived in appendix B.

Two coupled oscillators converge either in- or out-of-phase, depending on the synaptic sign. To show this property, we use SKONN's phase dynamics that are derived in appendix C using the PPV formalism [33]. Under the weak coupling assumption ($C_{ij} << C_L$), the phase dynamics of oscillator i can be expressed as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\phi_i = 2\beta_0\frac{Q_{ij}}{T}\,\text{square}(\phi_i - \phi_j). \tag{3}$$

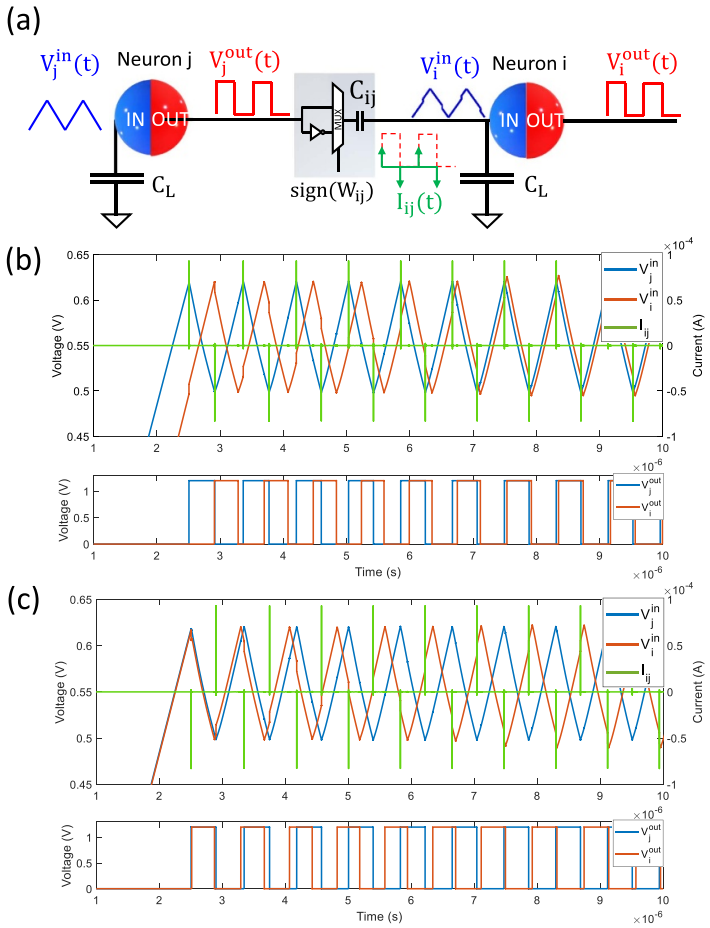

Figure 3. (a) Illustration of a neuron that drives a second neuron in a feed-forward manner. The multiplexer sets the weight sign and the capacitor C_{ij} converts the digital signal $V_j^{\text{out}}(t)$ into current spikes $I_{ij}(t)$ that induce phase shifts of the input $V_i^{\text{in}}(t)$. (b) Transistor-level transient simulation with a positive weight $W_{ij} > 0$. The second phase catches up the first one after few cycles such that $\Delta\phi_{ij} \approx 0°$. (c) Simulated dynamics with a negative weight $W_{ij} < 0$. The second phase is pushed such that $\Delta\phi_{ij} \approx 180°$ after few cycles.

With the 2π-periodic function

$$\text{square}(\theta) = \begin{cases} -1, & \text{if } 0 < \theta < \pi \\ +1, & \text{if } \pi < \theta < 2\pi \end{cases}. \tag{4}$$

The phase fixed points can be derived from (3) and are expressed in the next proposition.

Proposition 1. *If the injected charge $Q_{ij} \neq 0$ then the two SKONN oscillators admit a unique stable fixed-point $\Delta\phi^* = (\phi_i - \phi_j)^*$ such that*

$$\Delta\phi^* = \begin{cases} 0, & \text{if } Q_{ij} > 0 \\ \pi, & \text{if } Q_{ij} < 0 \end{cases}. \tag{5}$$

The proof is shown in appendix C. In other words, propagating a spike train defined as (1) induces an in-phase or out-of-phase locking, depending on the polarity of Q_{ij}. Each current spike produces a local phase shift to the analog input oscillation, resulting in an average phase shift $\Delta\phi = \pm 2\beta_0 Q_{ij}$ after each cycle (3). Figure 3(b) shows a transistor-level simulation of the positive weight case. I_{ij} perturbs V_i^{in} until the oscillators converge in phase. Similarly, figure 3(c) shows the same configuration with a negative weight and the oscillators are out-of-phase.

Note that the phases measured from the rising edges of V_i^{out} and V_j^{out} are slightly shifted from the theoretical fixed points (5). This is mainly due to the limited bandwidth of the hysteresis block which does not switch instantaneously when reaching its thresholds. This non-ideality can be compensated and is further discussed in appendix F. Interestingly, this phase shift disappears with symmetric synapses as both oscillators are equally delayed (see figure 5(c)).

2.2.2. N coupled oscillators

The phase dynamics of N sinusoidal coupled oscillators are often expressed using the Kuramoto model [9, 14, 16]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\phi_i = -\omega_0 \sum_{j=1}^{N} K_{ij} \sin\left(\phi_i - \phi_j\right) \tag{6}$$

where ω_0 is the frequency in rad/s^{-1} and K_{ij} the coupling coefficients. Similarly, we derive SKONN's phase dynamics for N oscillators as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\phi_i = \omega_0 \frac{V_{DD}}{\Delta V} \sum_{j=1}^{N} \frac{C_{ij}}{C_L} \mathrm{square}\left(\phi_i - \phi_j\right) \tag{7}$$

where we replaced β_0 and Q_{ij} from (3) by their expressions $\beta_0 = \pi/(\Delta V C_L)$ and $Q_{ij} = C_{ij} V_{DD}$. The derivation is detailed in appendix C. V_{DD} is the digital voltage swing, ΔV is the peak-to-peak triangular voltage amplitude at the input, C_{ij} is the synaptic capacitance value, and C_L is the neuron input capacitance.

SKONN's phase dynamics are very similar to the Kuramoto model (6) except for its sinusoidal function replaced by a *saturated* square function in this work. It induces a binarization behavior that is useful for solving some optimization problems as shown next. Note that similar dynamics have already been explored in simulation by Wang *et al* in their work about oscillatory Ising machines (OIMs) [9]. The authors studied the case where the sinusoidal term $\sin(\Delta\phi)$ from Kuramoto (6) is replaced by $\tanh(\alpha \sin \Delta\phi)$ with $\alpha = 10$. As SKONN's square interaction can be thought as $\mathrm{square}(\Delta\phi) \approx -\tanh(\alpha \sin \Delta\phi)$ for $\alpha >> 1$, we expect SKONN to have good performances when solving NP-hard COPs.

2.3. SKONN fixed points

2.3.1. Energy landscape

SKONN stability can be proved by applying the convergence theorem for ONNs derived by Hoppensteadt and Izhikevich [36]. With an odd coupling function (4) and symmetric coupling $Q_{ij} = Q_{ji}$, the theorem ensures that the phase differences converge to a stable fixed point. The proof consists of finding a Lyapunov function for the dynamics (7) that is bounded below and minimized through time. A candidate for the SKONN Lyapunov function is:

$$E = \frac{\beta_0}{T} \sum_{i,j}^{N} Q_{ij}\, \mathrm{triangle}\left(\phi_i - \phi_j\right). \tag{8}$$

With:

$$\mathrm{triangle}(\theta) = \begin{cases} \theta - \pi/2, & \text{if } 0 \leqslant \theta \leqslant \pi \\ 3\pi/2 - \theta, & \text{if } \pi \leqslant \theta \leqslant 2\pi \end{cases}. \tag{9}$$

Under the assumption that $Q_{ij} = Q_{ji}$, one can check that:

$$\begin{aligned} \frac{\partial E}{\partial \phi_k} &= \frac{\beta_0}{T} \left(-\sum_{j=1}^{N} Q_{kj}\, \mathrm{square}\left(\phi_k - \phi_j\right) + \sum_{i=1}^{N} Q_{ik}\, \mathrm{square}\left(\phi_i - \phi_k\right) \right) \\ &= -\frac{\mathrm{d}\phi_k}{\mathrm{d}t} \end{aligned} \tag{10}$$

Thus, SKONN minimizes E over time:

$$\begin{aligned} \frac{\mathrm{d}E}{\mathrm{d}t} &= \sum_{k=1}^{N} \frac{\partial E}{\partial \phi_k} \frac{\mathrm{d}\phi_k}{\mathrm{d}t} \\ &= -\sum_{k=1}^{N} \left(\frac{\mathrm{d}\phi_k}{\mathrm{d}t} \right)^2 \leqslant 0. \end{aligned} \tag{11}$$

SKONN's and Kuramoto's energy landscapes are represented for the three fully-coupled oscillators case in figure 4(a). In this simulation, we randomly initialized the phases, numerically solved (6), (7), and measured the final phases with respect to the reference ϕ_1. It can be seen from the distribution of final phases for 1000 trials that SKONN can settle to arbitrary analog phases, whereas the Kuramoto-ONN always converges to a

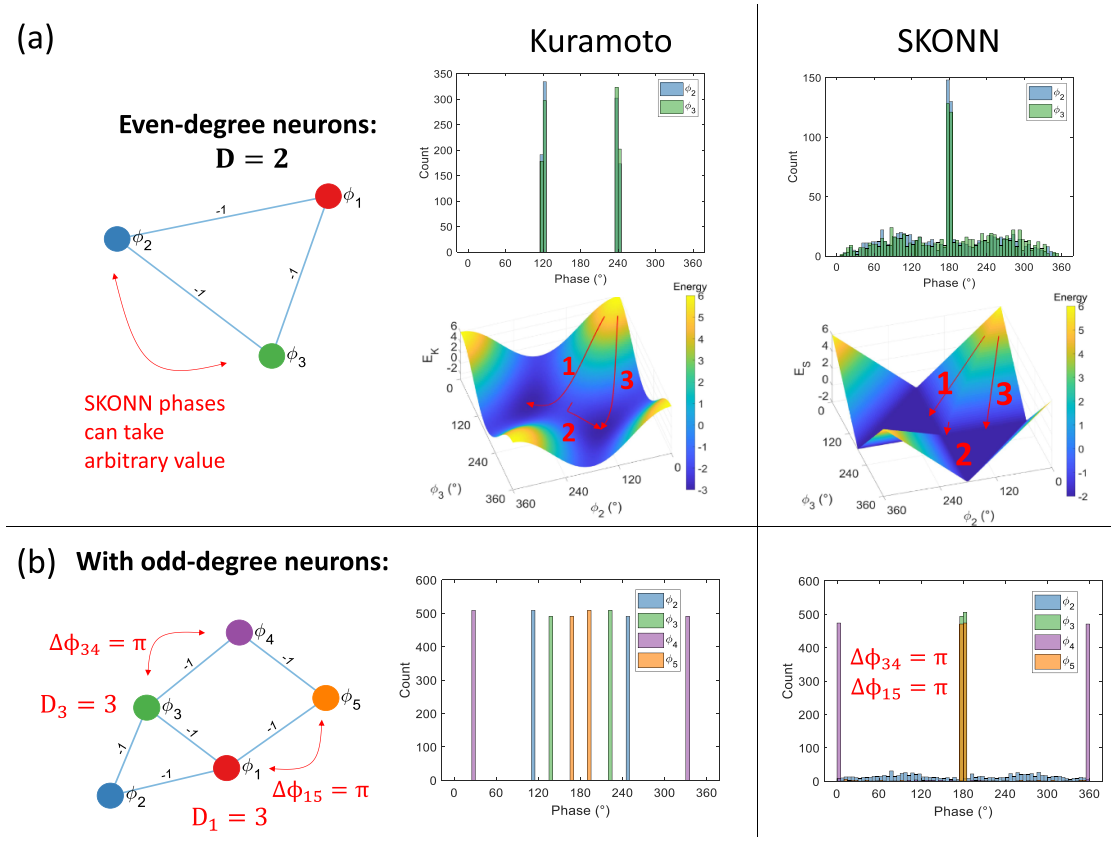

Figure 4. (a) Three oscillators coupled by negative weights. The right hand side shows the Kuramoto and SKONN distributions of the final phases for 1000 random initializations (uniform distribution) and the corresponding Kuramoto and SKONN energy landscapes. The arrows represent three examples of trajectories with various initializations, highlighting Kuramoto's minima and saddle point. In this example, SKONN's energy minima consist of plateaus that are linked to the even-degree property (proposition 2). (b) Five coupled oscillators where D_1, D_3 and D_2, D_4, D_5 are odd and even, respectively. SKONN tends to binarize phases except for ϕ_2 which can rather converge to any phase value (proposition 2).

single phase fixed point $\phi^* = (0°, 120°, 240°)$. Moreover, we observe that SKONN's energy landscape has two plateaus where the phases can settle and remain stable, which is consistent with the associated phase distribution extracted from simulations. Whereas the Kuramoto energy landscape consists of two minima and a saddle point, as highlighted by the three simulated trajectories of figure 4(a). Unfortunately, we cannot visualize the energy landscape for larger networks but a local analysis around a phase fixed point can reveal the landscape around it (discussed in the supplementary material). There can be sharp hills in some directions and plateaus in others. For instance in the five-node case of figure 4(b), most of the fixed points are surrounded by hills in all directions except for the direction ϕ_2 which consists of a trench with constant E.

2.3.2. Link between graph degree and SKONN fixed points

The three-node case of figure 4(a) revealed a different distribution of SKONN's phase fixed points compared to Kuramoto. A simple analysis of the dynamics for N oscillators gives a relationship between the connectivity of the network and the phase fixed points. Specifically, the next proposition links the degree of a SKONN neuron, i.e. the number of synaptic inputs, with the value of its phase fixed point.

Proposition 2. *Consider a neuron i of degree D, i.e. driven by D neurons j with weighted charges $Q_{ij} \in \{-q, +q\}$ $q \neq 0$.*

(1) If D is odd and $\mathrm{d}\phi_i/\mathrm{d}t = 0$, then there is at least one input neuron j such that $(\phi_i - \phi_j)$ is a multiple of π.
(2) If D is even, then there is at least one ϕ_i and one set of input phase ϕ_j such that $\mathrm{d}\phi_i/\mathrm{d}t = 0$ and $\forall j\,(\phi_i - \phi_j)$ is not a multiple of π.

The proof is shown in appendix C. Interestingly, odd-degree neurons will phase-lock in- or out-of-phase with at least one input phase. The odd-degree property will be advantageous for solving some optimization problems (COPs) on graphs as it can prevent the use of sub-harmonic injection to binarize phases. SKONN

for COPs is further discussed in section 3.4. On another hand, an even number of inputs instead leads to a relaxed scenario where the neuron can settle into an infinite number of phases as seen in the 3-node case of figure 4(a). When SKONN has both odd and even numbers of input synapses, we heuristically find that most of phases tend to binarize as illustrated with the five-node graph in figure 4(b), although some phases (such as ϕ_2) can still converge to fixed points with arbitrary phase values. This aspect will be further discussed when solving larger graphs in section 3.4.

So far, we have seen that SKONN has simple yet rich phase dynamics with unique phase binarization properties resumed in proposition 2. Moreover, SKONN's phase evolution can be interpreted as the minimization of a N-dimensional energy landscape E (8) and thus naturally performs the gradient descent of E through time. One can harness this concept for solving optimization problems that consist in finding minima of a cost function. In the next sections, we focus on the NP-hard Max-cut problem and present two approaches for solving Max-cut with SKONN.

2.4. SKONN for solving the NP-hard Max-cut problem

Given a graph with a set V of N vertices connected by weighted edges $W_{ij} = W_{ji}$, the Max-cut problem consists in *cutting* the graph in two complementary subsets of vertices V_1 and V_2 such that the sum of weights between V_1 and V_2 is maximum. The Max-cut problem can be formulated as follows [37]:

$$\begin{aligned} &\text{Max}\frac{1}{2}\sum_{i,j} W_{ij}(1 - S_iS_j) \\ &\text{subject to:} \\ &S_i \in \{-1, +1\}\ \forall i \in V. \end{aligned} \tag{12}$$

Solving the Max-cut problem is NP-hard and the best-known approximation algorithm is the semidefinite programming (SDP) algorithm found by Goemans and Williamson [37] and denoted GW throughout the paper. By relaxing the binary spins S_i to unit vectors v_i in $\mathbb{R}^N$, GW relaxes the NP-hard Max-cut problem to an SDP convex problem for which optimality can be found in polynomial time:

$$\begin{aligned} &\text{Max}\frac{1}{2}\sum_{i,j} W_{ij}(1 - v_i.v_j) \\ &\text{subject to:} \\ &v_i \in \mathbb{R}^N \\ &|v_i| = 1\ \forall i \in V. \end{aligned} \tag{13}$$

To compute the cut, the vectors are finally assigned to binary spins by splitting in two the N-dimensional sphere with a random hyper plan. Repeating this final rounding step provides a cut whose expectation is:

$$E[\text{cut}] = \frac{1}{\pi}\sum_{i,j} W_{ij}\arccos(v_i.v_j) > 0.878\,\text{Max-cut}. \tag{14}$$

However, due to the high dimension of the problem relaxation ($\mathbb{R}^N$), GW is costly for large instances [38, 39] and alternative approaches using physical systems such as quantum annealers [40], coherent Ising machines [39], memristors [41] or coupled oscillators are being investigated [9, 10, 12, 20].

2.4.1. The Ising approach

One of the most studied formalisms applied to ONN is from Ising which was initially derived to study magnetism in materials [42]. Given interaction coefficients $J_{ij} \in \mathbb{R}$ between particles that can have two spins $S_i \in \{-1; +1\}$, the particles relax to a state that minimizes the Ising Hamiltonian (we skip the external fields for simplicity):

$$H = -\frac{1}{2}\sum_{i,j}^{N} J_{ij}S_iS_j. \tag{15}$$

Thanks to Lucas' seminal work [43], all Karp's 21 NP-complete problems can be mapped to the Ising formalism and the solutions can be approximated by any physical machine that minimizes the Ising

Hamiltonian (15). If SKONN phases take binary values $\phi_i = (1 - S_i)\pi/2 \in \{0, \pi\}$, its Lyapunov function (8) becomes:

$$\begin{aligned} E &= \frac{\beta_0}{T} \sum_{i,j}^{N} Q_{ij}\, \text{triangle}\left(\frac{\pi}{2}(S_j - S_i)\right) \\ &= -\frac{\pi\beta_0}{2T} \sum_{i,j}^{N} Q_{ij} S_i S_j \\ &\propto H. \end{aligned} \tag{16}$$

Each synaptic current spike can be thought of as a downward step (due to equation (11)) in the energy landscape (8) which corresponds to the Ising Hamiltonian (16) if the final phases are binary. However, having binary phases is not guaranteed in general. To force phase binarization, it is common practice to inject into the oscillators a SHIL periodic signal at twice the oscillating frequency [9] and described in appendix E for SKONN.

The Max-cut problem can easily be mapped to an OIM with spins corresponding to the graph vertices by setting $J_{ij} = -W_{ij}$ with W_{ij} the graph weights [43]. Then, the OIM performs the following minimization which is equivalent to the Max-cut (12):

$$\begin{aligned} &\text{Min}\, H = \text{Max}\left(-\frac{1}{2}\sum_{i,j} W_{ij} S_i S_j\right) \\ &\text{subject to:} \\ &S_i \in \{-1, +1\}\ \forall i \in V. \end{aligned} \tag{17}$$

The general strategy is to (1) map the graph to the OIM, (2) start the OIM while ramping up a 2-SHIL signal to binarize the phases, and (3) read the stable phase state [9]. Forcing phase binarization is common practice as it maps the OIM Lyapunov function to the Ising Hamiltonian (16). However, how to binarize is not straightforward. If the injected signal is too strong, it may 'freeze' the phases to sub-optimal local minima [9]. Whereas if the signal is too weak, it might increase the OIM computation time. As described next, we rather harness the free SKONN dynamics without SHIL to compute the Max-cut in this work.

2.4.2. A Rank-2 relaxation approach

Erementchouk *et al* [44] have recently shown that the free OIM relaxation can be harnessed to solve the Max-cut problem. Their recent results demonstrate that letting a Kuramoto-ONN settle to analog phase values is equivalent to solving a rank-2 relaxation problem for the NP-hard Max-cut problem. Such phase dynamics are used in the CirCut solver [38]. Similarly to GW, the CirCut algorithm relaxes spins S_i to 2D unit vectors $x_i \in \mathbb{R}^2$ such that $x_i = (\cos(\phi_i)\ \sin(\phi_i))$ that can take arbitrary values on the unit circle. The objective of the rank-2 relaxation is:

$$\begin{aligned} &\text{Max}\frac{1}{2}\sum_{i,j} W_{ij}(1 - x_i.x_j) = \text{Max}\frac{1}{2}\sum_{i,j} W_{ij}(1 - \cos(\phi_i - \phi_j)) \\ &\text{subject to:} \\ &x_i \in \mathbb{R}^2 \\ &|x_i| = 1\ \forall i \in V. \end{aligned} \tag{18}$$

Then, a rounding procedure produces spins to compute the graph cut. Unfortunately, this rank-2 algorithm cannot guarantee a lower bound on the cut as it remains a non-convex optimization problem. Nevertheless, its accuracy is comparable to the GW algorithm in practical use [38].

2.4.3. SKONN's approach and link with GW

In this paper, we only explore the relaxation approach where we let SKONN settle without forcing binarization. We will see that SKONN's phase binarization property (proposition 2.1) is particularly useful in this case. For the Ising approach, we invite the reader to consult the excellent work from Wang and Roychowdhury [9] as the reported dynamics are equivalent to SKONN's.

Moreover, it is worth mentioning that SKONN's energy landscape can be linked to the GW algorithm restricted to vectors of dimension 2 (GW2), as recently noticed in [45]. Indeed, SKONN's triangular interaction can be written as:

$$\text{triangle}(\phi_i - \phi_j) + \pi/2 = \arccos(x_i.x_j). \tag{19}$$

By setting SKONN's synapses as the negative of the graph weights $J_{ij} = -W_{ij}$, the energy becomes:

$$E \propto -\sum_{i,j} W_{ij} \arccos(x_i.x_j) + C. \tag{20}$$

With C a real constant. Hence, SKONN's energy minimization is equivalent to GW2's maximization task (14) as:

$$\begin{aligned} \text{Min}E = \text{Max} \sum_{ij} W_{ij} \arccos(x_i.x_j) \\ \text{subject to:} \\ x_i \in \mathbb{R}^2 \\ |x_i| = 1 \; \forall i \in V. \end{aligned} \tag{21}$$

3. Results

3.1. 3×3 SKONN PCB

We designed a 3×3 SKONN on PCB with fully-connected capability and 81 synapses (figure 5(a)) as a proof of concept for the SKONN architecture. Due to area constraints, we only implemented negative weights that we program by placing discrete capacitors C_{ij} manually. Figure 5(b) shows the oscillating neuron based on a Schmitt trigger (U1) with feedback resistor R_3 that charges/discharges a load capacitor C_L, producing a triangular-like waveform with 720 mVpp amplitude. The neuron output is a square-like waveform oscillating between $V_{DD} = +0.9$V and $V_{SS} = -0.9$V. Using an FPGA, we set the initial phase state by delaying the oscillator's starting time (Q1). The FPGA measures the neurons' output voltages and allows phase post-processing with a maximum precision of $\epsilon = 360° f_0/f_{\text{FPGA}}$. In our experiments we set $f_0 = 4$kHz, $f_{\text{FPGA}} = 50$ MHz and $\epsilon \approx 0.03°$.

Figure 5(c) shows an experiment of two oscillators weakly coupled by $C_{12} = C_{21} = 1\%C_L$ whereas figure 5(d) is a strong coupling with $C_{12} = C_{21} = 10\%C_L$. In both experiments, the oscillators are out-of-phase but the strong coupling case leads to a frequency reduction of −34% as the voltage jumps $\Delta V = (V_{DD} - V_{SS})C_{12}/C_L$ produced by each current spike are too large with respect to the analog amplitude. This phenomenon can induce frequency mismatches between groups of strongly coupled oscillators and groups of weakly coupled oscillators. Frequency mismatches still need to be investigated and here we empirically choose $C_{ij} < 5\%C_L$ to guarantee phase locking among oscillators.

3.2. Weighted Max-cut on PCB

The test case consists of the Max-cut problem with 2-bit positive weights. We generate random instances of Erdos–Rényi graphs $G(N,p)$ [46] with $N = 9$ nodes and p is the probability to have an edge between a pair of vertices such that the total number of edges $m = pN(N-1)/2$. For each graph edge, the weight is randomly selected from the list [0 10 22 47]/47 that corresponds to discrete capacitors used experimentally.

Figure 6(a) shows an example of a dense random graph instance with $p = 0.75$. We map the graph edges to the synaptic matrix and run 100 trials with random phase initializations. For each trial, the nine phases are sampled every oscillation period during 1000 oscillation cycles. Figure 6(e) shows the final phases $\phi_i(t = 1000\,T)$ measured for each trial, the latter indicated as the amplitude in the polar plot. The right-hand side of the polar plot corresponds to positive spins, whereas the left-hand side corresponds to negative spins. It appears that some phases such as ϕ_2, ϕ_4 and ϕ_6 are always assigned to the same spin polarity whereas most of the phases can end up in both half-circles, depending on the phase initialization. Hence, SKONN final states depend on the initialization and several trials ensure obtaining a good solution. Figure 6(d) shows the histogram of solution and the settling time. SKONN finds the graph Max-cut with 75% probability in less than 100 oscillation cycles on average. Figure 7 presents another Max-cut problem with $G(9, 0.5)$. This instance is easier to solve as there are two optimal spin states as seen in figure 7(a) and SKONN reaches 97% accuracy. Whereas Kuramoto simulations led to a lower accuracy due to the large phase distribution causing errors when rounding phases to spin values. Surprisingly, SKONN favors a single optimal state and never finds the second one. We observed a similar behavior even for graph instances with three optimal states. This suggests there are optimal critical points that are unstable (such as saddle points) as recently highlighted by Bashar *et al* [47].

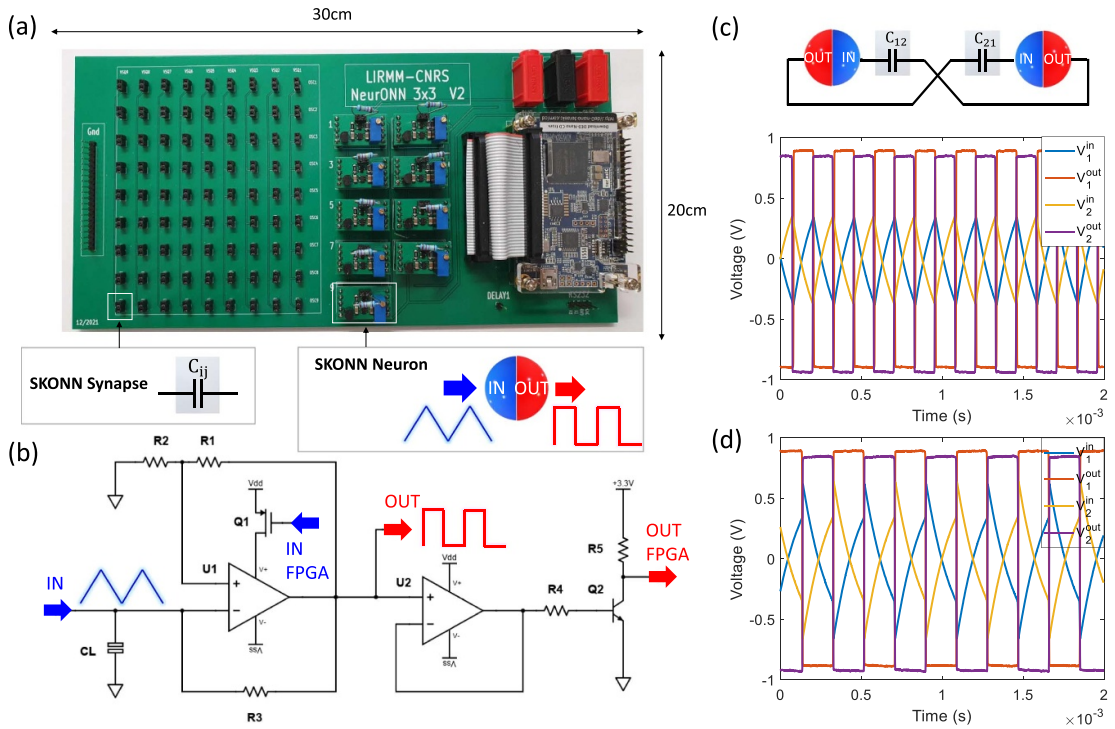

Figure 5. (a) SKONN on PCB with 9 oscillators and 81 synapses. We set the weight amplitude with the synaptic capacitance value C_{ij}. The FPGA initializes the phases by delaying the oscillators' starting time by switching Q1 and measures the digital oscillations buffered by the output stage (U2 and Q2). (b) The oscillator consists of a OP-amp Schmidt trigger feedback by the resistor R_3 to produce self-oscillations. $R_1//R_2$ sets the analog oscillation amplitude. (c) Two coupled oscillators with $C_{12}/C_L = C_{21}/C_L = 1\%$. (d) Two coupled oscillators with $C_{12}/C_L = C_{21}/C_L = 10\%$.

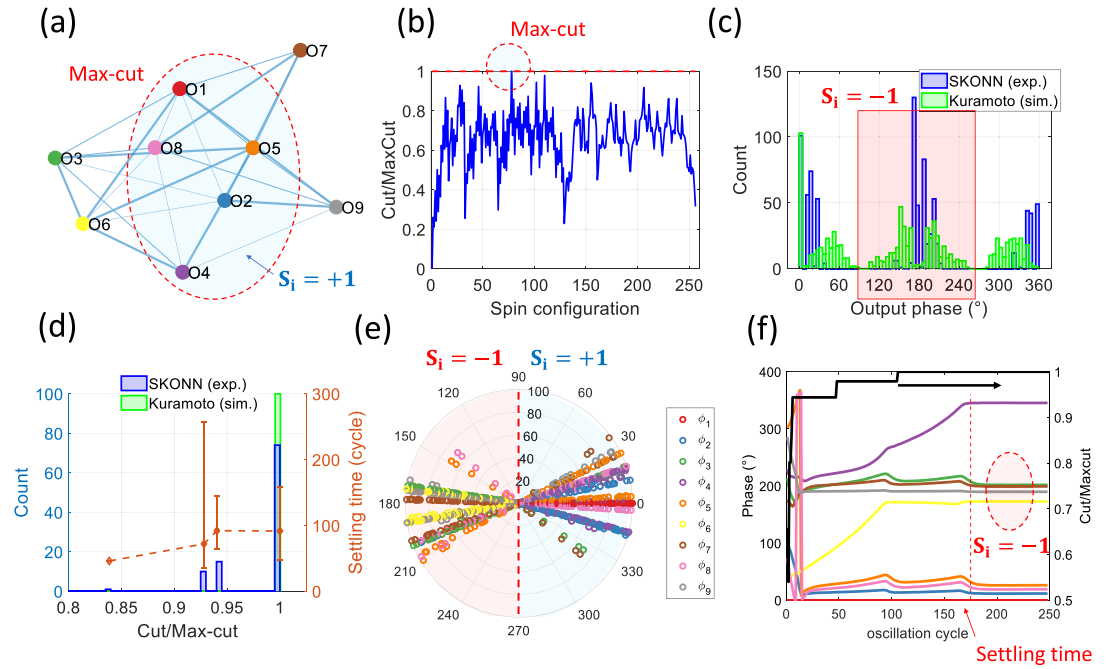

Figure 6. (a) Random instance of $G(9, 0.75)$ with 2-bit weighted edges. (b) Cut value vs spin configuration. (c) SKONN phases measured after 1000 oscillation cycles and compared with Kuramoto simulations. (d) Histogram of Max-cut solutions for 100 trials and measured settling time (e) SKONN experimental phase distribution. The polar amplitude represents the trial number (100 trials). We assign $-90° < \phi_i < 90° \rightarrow S_i = +1$ and $S_i = -1$ otherwise. (f) Example of phase dynamics and cut evolution.

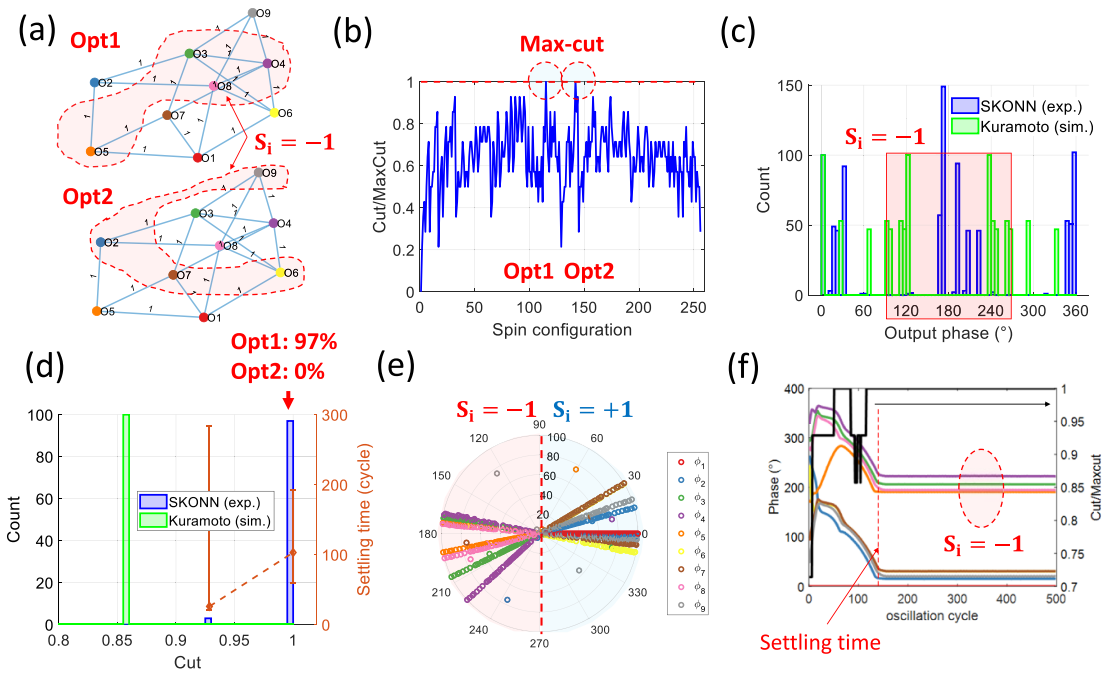

Figure 7. (a) Random instance of $G(9, 0.5)$ with 1-bit weighted edges and the two optimal states. (b) Cut value vs spin configuration. (c) SKONN phases measured after 1000 oscillation cycles and compared with Kuramoto simulations. (d) Histogram of Max-cut solutions for 100 trials and measured settling time (e) SKONN experimental phase distribution. The polar amplitude represents the trial number (100 trials). We assign $-90° < \phi_i < 90° \rightarrow S_i = +1$ and $S_i = -1$ otherwise. (f) Example of phase dynamics and cut evolution.

3.3. 4x4 SKONN CMOS design

3.3.1. SKONN integrated circuit

To further assess SKONN performances, we taped out an ASIC chip using a 65 nm technology. Figure 8(a) shows the chip layout view and table 2 contains the chip specifications. For this first ASIC version, we have focused on transistor matching to minimize the variations among oscillators and enhance the robustness. The second main objective was a low power consumption (160 μW without IOs). Although there are 16× more synapses than neurons, the synapses represent only 20% of the power consumption and are promising for solving dense networks. The constraint on the oscillator matching led to a large oscillator area (5000 μm^2) but could be overcome with some calibration technique. Although the total chip area is 3.9 mm^2, there is room for improvement as the core area is 1.1 mm^2, and 71% of the total area consists of routing lines from the core to the high number of IOs and test pads (90).

Similarly to the SKONN design on PCB, the oscillating neuron consists of a hysteresis regenerative comparator whose digital output drives the charge and discharge of the capacitor $C_L = 500$ fF in the shaper block (figure 8(d)). The comparator switches whenever the input voltage V_i^{in} reaches one of the two voltage thresholds V_H and V_L that define the analog voltage amplitude as $\Delta V = V_H - V_L = 120$ mV and $(V_H + V_L)/2 = V_{REF} = 600$ mV. The synaptic block consists of a capacitor bank ranging from 0 to 37.5 fF that linearly maps the weight amplitude from 0 to 15 ($C_0 = C_1/2 = C_2/4 = C_3/8 = 2.5$ fF). The weight sign is selected by the multiplexer commanded by the sign bit B_S. The synaptic matrix is programmed by sending serially the $5 \times 256 = 1280$ bits through the registers.

Table 3 presents the specifications of the state-of-the-art ONNs designed using CMOS technology, transition metal oxide, and spintronic devices. The connectivity scheme can be all-to-all for small-sized ONNs ($N \leqslant 100$) but is obviously reduced to nearest neighbor connections for larger ONNs such as in [12] and [21] that both chose 8 neighbors (King's graph). SKONN's input/output separation allows any type of modular connectivity in a robust manner as the synaptic signal from 2-coupled oscillators cannot leak to any other oscillator. This contrasts with fully-analog architectures such as [10, 20] that merge input/output nodes, resulting in undesired current paths between non-adjacent oscillators. It appears that digital ONNs such as [5] and [12] are very energy-efficient as they produce a single oscillation with only 300 fJ and 21 fJ. In contrast, analog oscillators found in [21] and this work consume 2.3 pJ and 10 pJ per oscillation, respectively. However, we believe that SKONN's energy could be reduced by relaxing the constraints on the analog oscillator matching and using a calibration scheme instead, as proposed by Graber *et al* [21]. Finally, novel

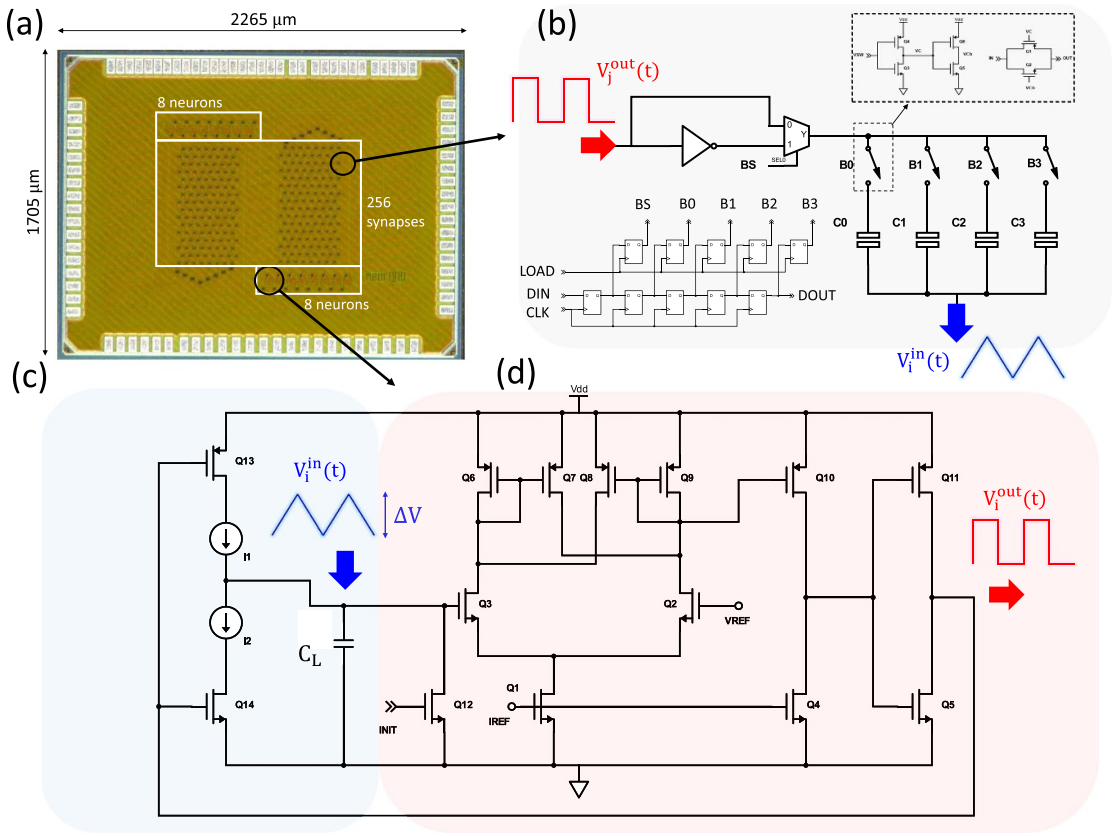

Figure 8. (a) Photography of SKONN ASIC. There are 16 neurons and 256 synapses laid out as in figure 1. (b) SKONN synapse with 5 bits precision. The four parallel capacitors set the weight amplitude from 0 to 15. The bit B_S selects the weight sign from the multiplexer. (c) Shaper circuit producing the input triangular oscillation. (d) Regenerative comparator that holds the neuron state and commands the shaper stage. An hysteresis behavior is obtained by setting the feedback transistors Q_7 and Q_8 such that $W_{Q7}/W_{Q6} = W_{Q8}/W_{Q9} > 1$ for the same transistor length.

Table 2. SKONN ASIC specifications.

Technology	Neurons	Synapses	Synaptic precision	f_{osc}	SHIL inputs	Area	Power (core@1.2V)
65 nm	16	256	5 bits	1–4 MHz	16	3.9 mm^2 Neurons: 2% Synapses: 24% Biasing: 3% Routing, IOs: 71%	160 μW@1MHz Neurons: 75% Synapses: 20% Biasing: 5%

devices such as vanadium dioxide (VO_2 [10]) and spin-torque oscillators [48] are promising for future ONN implementation thanks to their potential dense integration and high frequency.

3.3.2. Feed-forward network with SKONN

Propagating the information in a feed-forward manner is useful in some applications that require driving neurons. For instance, when training an ONN with the equilibrium propagation method [49, 50], one must nudge the output oscillators toward the desired value which is challenging to obtain with recurrent synapses. Instead, teaching oscillators could drive the output oscillators using feed-forward connections without being impacted during the learning phase.

To demonstrate SKONN's feed-forward ability, we program the ASIC to solve a simple 2-input XOR operation. Inspired by the Parametron built by Goto in the 1950s [3], we use a 3-input SKONN neuron as a majority gate

$$\phi_M = (\phi_X.\phi_Y) + (\phi_X.\phi_Z) + (\phi_Y.\phi_Z) \tag{22}$$

where $\phi_X, \phi_Y, \phi_Z \in \{0°; 180°\}$ are the input binary phases thought as Boolean variables; i.e. ϕ_M is true when $\phi_M = 180°$. Interestingly, SKONN's odd-degree property (proposition 2) ensures that ϕ_M is binary when its inputs are also binary.

Table 3. Comparison between state-of-the-art ONNs.

	Jackson *et al* [5]	Moy *et al* [12]	Bashar *et al* [20]	Graber *et al* [21]	Dutta *et al* [10]	Romera *et al* [48]	This work
Technology	28 nm	65 nm	65 nm	28 nm	Simulations: 28 nm CMOS + PTNO (VO_2)	Spin-torque	65 nm
Neurons	100	1968	30	400	100	4	16
Connectivity	all-to-all	King's graph	all-to-all	King's graph	all-to-all	all-to-all	all-to-all
Power	303 mW	42 mW	1.76 mW	182 mW	2.56 mW	4 mW	160 μW (core)
Frequency	1 GHz	1 GHz	45 kHz	200 MHz	87 MHz	300 MHz	1 MHz
Energy/osc	300 fJ	21 fJ	1.3 nJ	2.3 pJ	300 fJ	3.3 pJ	10 pJ
Chip area	3.24 mm^2	2.1 mm^2	—	2.2 mm^2	—	—	3.9 mm^2

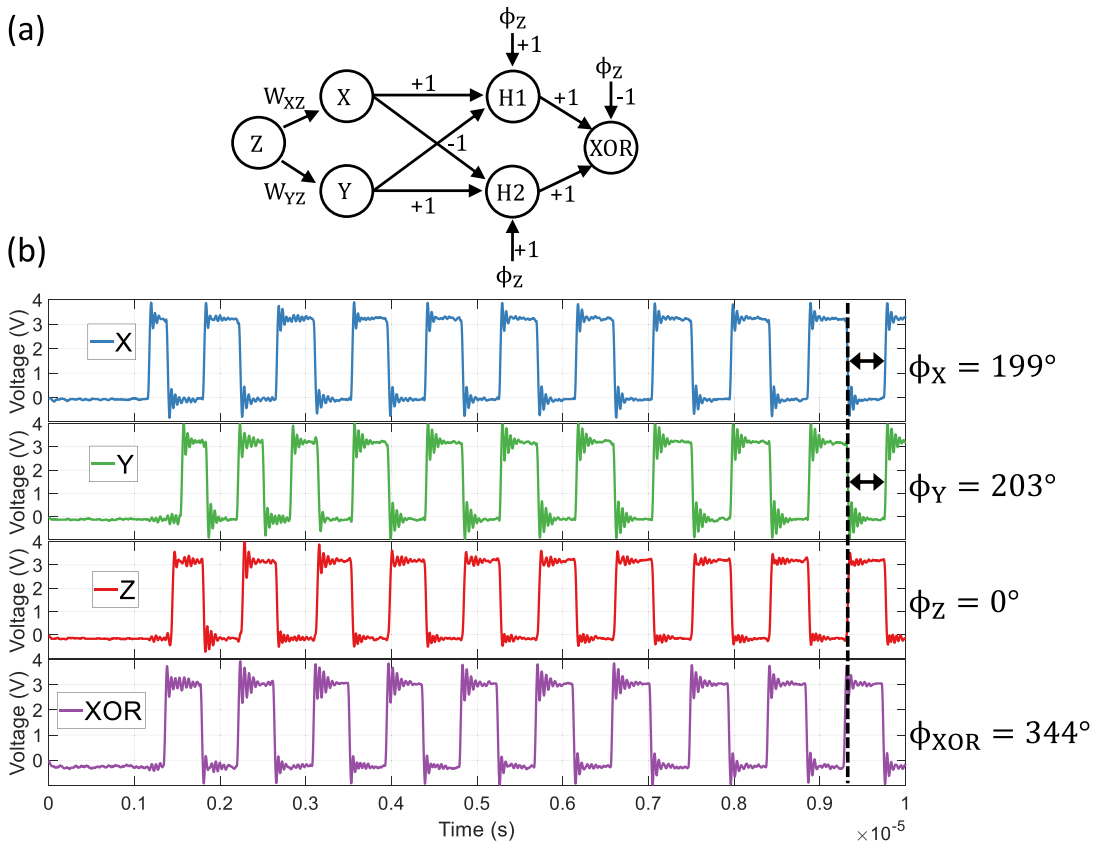

Figure 9. (a) XOR(X), (Y) circuit using SKONN and feed-forward synapses. Neuron Z is the reference oscillator and can be thought of as the neurons' bias, similar to perceptrons. The weights W_{XZ} and W_{YZ} are the inputs and set the initial phases ϕ_X and ϕ_Y. (b) ASIC experimental results when $W_{XZ} = W_{XZ} = -1$ which set $\phi_X \approx \phi_Y \approx 180°$ and corresponds to the Boolean case where $X = Y = 1$.

The XOR(X,Y) circuit is implemented by writing the XOR Boolean expression $\phi_{XOR} = (\phi_X.\overline{\phi_Y}) + (\phi_Y.\overline{\phi_X})$. Considering ϕ_Z as the reference oscillator, AND and OR gates are obtained when feeding ϕ_Z or $\overline{\phi_Z}$ to the majority gates, respectively. Figure 9(a) shows the obtained network with two hidden neurons (H1,H2), one neuron acting as a bias unit (Z), and one output neuron. The reference oscillator Z sets the two input phases ϕ_X and ϕ_Y by feeding its output signal in a feed-forward manner. When the weight W_{XZ} is -1 or +1, it sets $\phi_X \approx 0°$ or $\phi_X \approx 180°$, respectively, and the same rule applies for ϕ_Y and W_{YZ}. The oscillators are randomly initialized before the input phases ϕ_X and ϕ_Y settle to the desired inputs. Then, the network further relaxes to a stable phase state after a few oscillations and we read the output phase ϕ_{XOR}. Figure 9(b) shows the ASIC measurement in the case $\phi_{XOR}(180°, 180°) = 0°$. The three other cases are shown in the supplementary material. Table 4 summarizes the results for the 4 possible inputs W_{XZ} and W_{YZ}. By assigning

Table 4. Solving XOR(X,Y) with SKONN ASIC.

W_{XZ}	W_{YZ}	ϕ_X	ϕ_Y	ϕ_{XOR}	XOR(X,Y)
+1	+1	359°	56°	342°	0
+1	−1	358°	195°	161°	1
−1	+1	203°	0°	187°	1
−1	−1	199°	203°	344°	0

the bit 0 when $-90° \leqslant \phi_i \leqslant 90°$ and 1 otherwise, it can be seen that the proposed network computes XOR(X,Y) in a feed-forward manner.

3.4. SKONN scaling and benchmarking

3.4.1. Weighted Max-cut of random graphs

To assess how SKONN's computational performances scale, we run large-scale simulations of random Weighted Max-cut problems for $N = 8$, 16, 32, 64, 128, 256, 512, and 1024 nodes. For each graph density $d = 0.25$, 0.5, and 0.75, 10 random graphs $G(N, d)$ are generated such that the total number of edges $m = d\,N(N-1)/2$. The graph edges are randomly weighted with positive values from 0 to 15 that correspond to the ASIC synaptic range. We use the ASIC parameters and solve SKONN's dynamics (7) with MATLAB using the built-in ODE solver ode15s (see supplementary material). For each graph instance, we run 10 trials with random phase initialization, for a total of 100 trials per graph size and density. As a ground truth, we consider the best solution $\mathrm{Cut_{GW}}$ provided by the Goemans–Williamson algorithm, out of 100 random projections defining the cut [37] and computed with the CVX solver on MATLAB [51]. The distance between the SDP cut $\mathrm{Cut_{SDP}}$ and $\mathrm{Cut_{GW}}$ is represented in figure 10(a). As $\mathrm{Cut_{SDP}} \geqslant$ Max-cut, the ratio $\mathrm{Cut_{GW}}/\mathrm{Cut_{SDP}}$ gives a lower bound on the chosen GW cut. For all the trials, the results are compared with Kuramoto dynamics with the same parameters and initializations.

Figure 10(b) shows SKONN's and Kuramoto's phase distributions for each ONN size. Here again, it appears that SKONN phases tend to be clustered near 0° and 180° whereas Kuramoto phases seem more uniformly distributed. Figure 10(c) present the obtained cuts when considering the first oscillator as the reference, and normalized by $\mathrm{Cut_{GW}}$. We first notice that the results are quite homogeneous with respect to the graph densities. Secondly, it appears that SKONN produces high-quality cuts as $\mathrm{Cut}/\mathrm{Cut_{GW}} \approx 1$ for all ONN sizes. In contrast, Kuramoto-ONNs have a lower accuracy for sizes between $N = 16$ and $N = 256$. Interestingly, the settling time (time to reach a steady phase state) seems to grow according to a logarithmic law with the ONN size. This result refines some previous scaling observations mentioning a quasi-constant settling time [9, 17]. It also confirms the high ONN parallelism and ability to compute in a few tens of cycles, even for large graphs.

Similarly to the CirCut algorithm [38] (rank-2 relaxation approach), we also investigate the Kuramoto accuracy when changing the reference oscillator and name it the 'Kuramoto-CirCut' scheme. Figure 10(d) presents the case where the best Kuramoto cut is chosen out of N possible cuts, whereas the SKONN reference oscillator remains the first one. It can be seen that SKONN provides the same quality cut as GW and Kuramoto-CirCut. However, compared to GW and Kuramoto-CirCut, SKONN's cut is solely obtained by reading out the phases with respect to a single oscillator and does not need N different cut evaluations that linearly increase the time to solution.

3.4.2. G-set benchmark

The previous study concerned random graphs. Here, we benchmark SKONN for solving Max-cut using the G-set benchmark that includes various graph topologies [52]. Table 5 shows the cuts obtained for a single trial with SKONN and Kuramoto using the same random phase initialization, and considering the first oscillator as the phase reference. The cuts are compared against the state-of-the-art GW [37] and the Kuramoto-CirCut scheme [38]. GW's cut is the best cut obtained out of 100 random projections and computed with the CVX solver on MATLAB [51]. For graphs with $N > 3000$, GW values are taken from another state-of-the-art SDP solver [53] due to memory constraints. The Kuramoto-CirCut values correspond to the best cut extracted from the Kuramoto dynamics, out of N possible reference oscillators.

With a single run, Kuramoto-CirCut and SKONN solvers produce, on average, better results than the GW algorithm which picks up the best spin configuration out of 100 random projections. The average SKONN cut value is 94.6% of the best-known cuts [52] and the highest among the four methods. The accuracy obtained by simulating SKONN motivates its real hardware implementation as the time-to-solution could be drastically reduced compared to a CPU. For instance, solving the smallest graph G11 requires 12 s of GW runtime on a laptop (i7 Intel core @1.6 GHz and 32 GB of RAM). In contrast, SKONN's settling time does not vary much with the ONN size and could enable a large-scale cut computation in less than 431 cycles

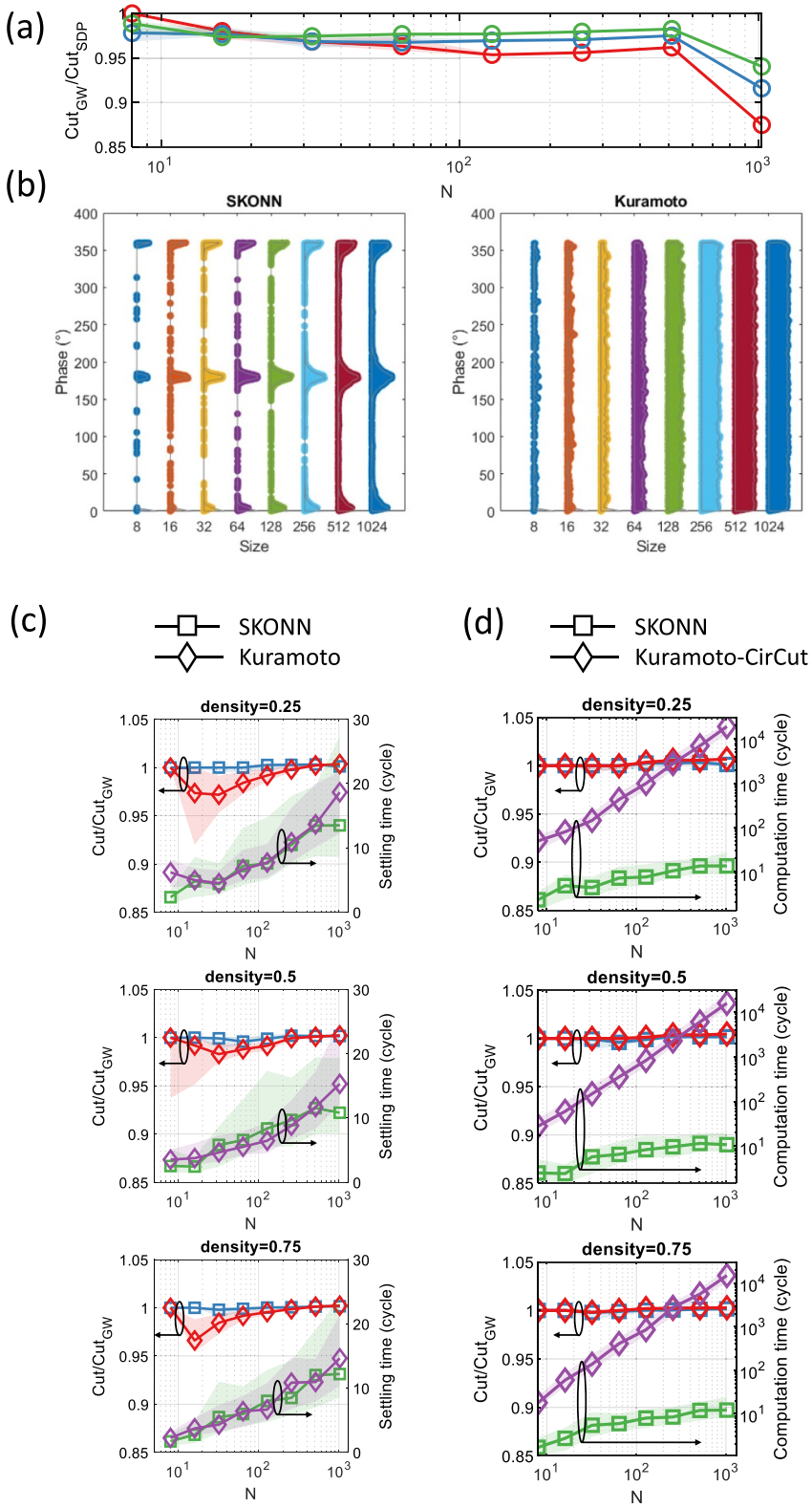

Figure 10. (a) Ratio between Cut_{GW} and Cut_{SDP} that gives a lower bound on Cut_{GW} as $Cut_{SDP} \geqslant$Max-cut. (b) Phase distribution at steady state for SKONN (left) and Kuramoto (right) for various ONN sizes. (c) Cuts obtained by SKONN and Kuramoto when the first oscillator is the phase reference and for various graph densities. (d) Cut obtained when SKONN's reference is the first oscillator, compared to the case where the Kuramoto reference is changed N times, similarly to the CirCut algorithm [38].

on average. With oscillators running at 1 MHz, the runtime per trial would only be 431 μs which is 2.8 $10^4\times$ faster than GW's execution. However, reaching excellent cuts such as 99.9% of the best-known cut requires more trials and annealing the ONN to escape local minima. For more results using SHIL and various annealing schemes, please refer to [9].

Table 5. Comparison between CirCut, Kuramoto, and SKONN solvers for G-SET Max-cut instances. The bold values indicate the best cut obtained among the four approaches. The number in parenthesis is the cut normalized by the best-known cut found in [52].

Graph				Value					Settling time (cycles)	
Name	$\|(V\|, \|E\|)$	Weights	Type	GW	Kuramoto-CirCut	Kuramoto	SKONN	Best-known [52]	Kuramoto	SKONN
G11	(800, 1600)	−1,+1	Toroidal	530 (0.940)	514 (0.911)	502 (0.890)	**540 (0.957)**	564	480	368
G12	(800, 1600)	−1,+1	Toroidal	**532 (0.957)**	510 (0.917)	496 (0.892)	528 (0.950)	556	216	272
G13	(800, 1600)	−1,+1	Toroidal	**554 (0.952)**	538 (0.924)	524 (0.900)	544 (0.935)	582	368	224
G14	(800, 4694)	+1	Planar	2978 (0.972)	3005 (0.981)	2974 (0.970)	**3020 (0.986)**	3064	416	496
G15	(800, 4661)	+1	Planar	2963 (0.971)	**2992 (0.981)**	2970 (0.974)	2979 (0.977)	3050	704	240
G20	(800, 4672)	−1,+1	Planar	849 (0.902)	**871 (0.926)**	818 (0.869)	858 (0.912)	941	504	208
G21	(800, 4667)	−1,+1	Planar	849 (0.911)	**868 (0.932)**	839 (0.901)	852 (0.915)	931	184	104
G22	(2000, 19 990)	+1	Random	12 936 (0.968)	**13 095 (0.980)**	13 026 (0.975)	12 988 (0.972)	13 359	280	104
G23	(2000, 19 990)	+1	Random	12 946 (0.970)	**13 106 (0.982)**	13 078 (0.980)	12 950 (0.970)	13 344	864	216
G24	(2000, 19 990)	+1	Random	12 966 (0.972)	**13 135 (0.985)**	13 050 (0.978)	12 997 (0.974)	13 337	928	160
G30	(2000, 19 990)	−1,+1	Random	3014 (0.883)	**3200 (0.938)**	3175 (0.930)	3095 (0.907)	3413	376	192
G31	(2000, 19 990)	−1,+1	Random	2885 (0.872)	**3089 (0.933)**	3063 (0.925)	3015 (0.911)	3310	1424	456
G32	(2000, 4000)	−1,+1	Toroidal	1290 (0.915)	1284 (0.911)	1280 (0.908)	**1332 (0.944)**	1410	616	520
G33	(2000, 4000)	−1,+1	Toroidal	1266 (0.916)	1254 (0.907)	1244 (0.900)	**1294 (0.936)**	1382	424	248
G34	(2000, 4000)	−1,+1	Toroidal	1274 (0.920)	1258 (0.909)	1224 (0.884)	**1306 (0.943)**	1384	704	456
G50	(3000, 6000)	−1,+1	Toroidal	**5880 (1.00)**	5784 (0.983)	5764 (0.980)	5818 (0.989)	5880	544	1968
G56	(5000, 12 498)	−1,+1	Random	3634 (0.904)	3700 (0.921)	3622 (0.901)	**3733 (0.929)**	4017	712	296
G57	(5000, 10 000)	−1,+1	Toroidal	**3320 (0.950)**	3138 (0.898)	3054 (0.874)	3270 (0.936)	3494	792	576
G60	(7000, 17 148)	+1	Random	13 610 (0.959)	**13 730 (0.968)**	13 669 (0.963)	13 717 (0.967)	14 188	824	384
G61	(7000, 17 148)	−1,+1	Random	5252 (0.906)	5322 (0.918)	5257 (0.907)	**5327 (0.919)**	5796	1648	312
G62	(7000, 14 000)	−1,+1	Toroidal	4612 (0.947)	4394 (0.902)	4358 (0.895)	**4642 (0.953)**	4870	520	1328
G64	(7000, 41 459)	−1,+1	Planar	7624 (0.871)	8046 (0.919)	7946 (0.908)	**8135 (0.930)**	8751	800	352
Average				0.934	0.938	0.923	**0.946**	1	623	431

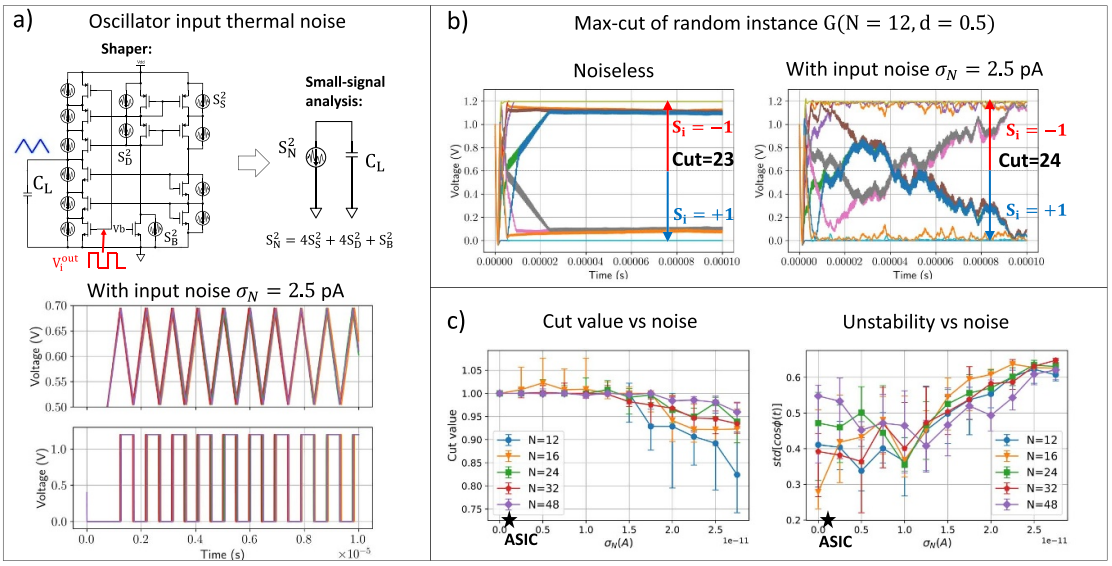

Figure 11. (a) Schematic of the shaper stage with current noise sources. Bottom figure shows 10 stochastic transient simulations with $\sigma_N = 2.5$ pA. (b) Example of phase dynamics with and without noise for a 12-node graph with 50% density. The Y-axis show the output voltages V_f of each analog filter measuring the phases (see figure 14). (c) Left: cut-value vs standard deviation of noise σ_N. Right: measure of unstability vs σ_N.

3.5. Impact of noise

Here, we study the impact of noise at the hardware level when solving the Max-cut task for various graph sizes. The objective is twofold: (1) assess whether SKONN is robust to noise and (2) quantify the maximum tolerated noise for practical implementation. Due to resource constraints during transient simulations, this study does not include every single noise source from each transistor. As an approximation, we perform stochastic simulations at the level of circuit equations by including the main thermal noise contributions at the oscillating input node (equation (32) derived in the appendix A). To compensate for this approximation, the level of input noise is swept to 70× larger values than the calculated ASIC input noise. Note that the input noise also contributes to jitter noise at the output due to the stochasticity on the oscillator transition times, as shown by the 10 stochastic oscillator trajectories in figure 11(a).

Figure 11(a) presents the main thermal noise sources of the shaper block that perturb the analog input waveform. Assuming the noise amplitude is small (small-signal calculation [54]), we find that the main source of thermal noise originates from the current source transistors in our design that also carry noise from biasing stages. During the load capacitor charge, four transistors acting as current sources and four diode-connected transistors contribute to current noise at the input. Further assuming transistor noises are uncorrelated, the current noise power spectral density (PSD) at the input is expressed as:

$$S_N^2 = 4S_S^2 + 4S_D^2 + S_B^2 = \sigma_N^2 \tag{23}$$

where S_S^2, S_D^2, and S_B^2 are the PSDs of the current source, diode-connected, and bias transistors, respectively. Note that the noise contribution from the switches is negligible as their PSDs are divided by the high output impedance of the current sources. The transistor parameters used to compute the noise variance are listed in table 6. The small-signal parameters are evaluated using the model from Enz, Krummenacher and Vittoz (EKV) [55] where the inversion factor is defined as $I_F = I_{\text{bias}}/I_S$ with $I_S = 2\mu C_{\text{ox}}(W/L)U_T^2$ the specific current of the transistor, and $U_T = k_B T/q$ is the thermal voltage. The transconductance g_m is calculated as:

$$\frac{g_m}{I_{\text{bias}}} = \frac{1}{U_T}\frac{2}{1+\sqrt{4I_F+1}}. \tag{24}$$

As the $1/f$ noise PSD is inversely proportional to the transistor dimensions, we assume that thermal noise is dominant and express the current PSD of each transistor i as:

$$S_i^2 \approx 4k_B T g_m \gamma. \tag{25}$$

With $\gamma = 1/2$ or $\gamma = 2/3$ when the transistor is in saturation and weak or strong inversion [55].

Table 6. ASIC transistor parameters used in the stochastic simulations.

Transistor	Length	Width	I_F	g_m or R	Noise standard deviation
Current source	6 μm	12 μm	0.5	$g_{m_S} = 5.6\,\mu$A/V	$\sigma_S = 0.25$ pA
Diode-connected	6 μm	12 μm	0.5	$g_{m_D} = 5.6\,\mu$A/V	$\sigma_D = 0.25$ pA
Chip current mirror	16 μm	2 μm	8	$g_{m_C} = 2.3\,\mu$A/V	$\sigma_C = 0.16$ pA
External resistor	NA	NA	NA	$R = 1\,M\Omega$	$\sigma_R = 0.13$ pA
Shaper bias	20 μm	10 μm	2	$g_{m_B} = 3.8\,\mu$A/V	$\sigma_B^2 = \sigma_R^2 + 4\sigma_C^2 + 0.18^2\sigma_B$ =0.4 pA
Total input current noise	$\sigma_N^2 = 4\sigma_S^2 + 4\sigma_D^2 + \sigma_B^2$ $\sigma_N = 0.8$ pA				

To model noise in time domain, we consider Gaussian white noise (GWN) which also has a similar PSD, i.e. a constant spectrum, with the additional assumption that the current noise samples follow a Gaussian distribution [56, 57]. Hence, we express the current noise in time domain as:

$$I_N(t) = \sigma_N \xi(t) \tag{26}$$

where $\xi(t)$ is a GWN with $E[\xi(t)] = 0$, and correlation $E[\xi(t+\tau)\xi(t)] = \delta(\tau)$ with $\delta(t)$ the Dirac delta function. It follows that $E[I_N(t+\tau)I_N(t)] = \sigma_N^2\delta(\tau)$ and the PSD is:

$$S_N^2 = \sigma_N^2. \tag{27}$$

This relation enables the numerical evaluation of the σ_N-values using the ASIC transistor parameters of table 6 while running stochastic transient simulations. From the circuit of figure 11(a), we see that the current noise is integrated by the capacitor C_L during the dynamics. Overall, the addition of GWN turns the ODE system into a stochastic differential equation (SDE) system that is integrated by a dedicated Julia solver [58]. Especially, the ODE for the input voltage V_i^{in} of neuron i becomes the following SDE:

$$C_L \mathrm{d}V_i^{\mathrm{in}} = I_{\mathrm{bias}}\left(1 - 2\frac{V_i^{\mathrm{out}}}{V_{dd}}\right)\mathrm{d}t + \sum_j C_{ij} dV_j^{\mathrm{out}} + \sigma_N \mathrm{d}W \tag{28}$$

where $\mathrm{d}W$ is a Wiener process defined as $\xi(t) = \mathrm{d}W/\mathrm{d}t$ [57].

Figure 11(c) shows the cut values obtained for 10 random graphs $G(N,d)$ of sizes $N \in \{12, 16, 24, 32, 48\}$, densities $d \in \{0.1, 0.2 \ldots 0.9, 1\}$ and various noise variances. For all graph instances, the cut value is evaluated after 100 oscillation cycles and normalized by the cut value obtained in the noiseless case. Note that from table 6 we calculate the input current noise in the ASIC as $\sigma_N \approx 0.8$ pA, which does not include the output noise from the hysteresis block. However, the simulations suggest robustness to larger input noise up to $\sigma_N = 15$ pA which also contributes to output noise. For lower noise levels, noise can sometimes increase the accuracy as shown in the example of figure 11(b) where a noisy SKONN with $\sigma_N = 2.5$ pA finds a larger cut compared to the noiseless case. Furthermore, the stability of the system is estimated by analyzing the time evolution of the spin signals expressed as $S_i(t) = \cos\phi_i(t)$. Computing the standard deviation std$[S_i(t)]$ over time from t = 0 to t = $100/f_{\mathrm{osc}}$ indicates whether the spins settle or not. In the ideal case, std$[S_i(t)] \approx 0$ whereas in the worst case, the spin varies between -1 and +1 with std$[S_i(t)] \approx 1$. Note that the transient dynamics are also included in the metric std$[S_i(t)]$ which hence never reaches 0 in practice. Figure 11(c) shows a correlation between the decrease of cut value and the decrease of stability for $\sigma_N \geqslant 15$ pA for all SKONN sizes.

From this study, we conclude that only large levels of noise have detrimental effects on SKONN performances. Moreover, moderate levels of noise can slightly increase the accuracy which we believe is linked to the escape from local minima, as already observed by Wang *et al* [9] at the level of phase modeling.

4. Discussion

SKONN mixed-signal architecture facilitates scaling up the ONN size thanks to the separation between analog computation and digital propagation. While the neurons exchange information in the digital domain, SKONN architecture ensures that the sensitive analog computation remains within modules and can be shielded from the outside. However, it remains some design challenges that are currently being investigated:

(1) Any delay added to the digital propagation is equivalent to a synaptic phase shift. This must be considered for a high-frequency operation when buffers' delays become non-negligible and can be an issue for advanced technologies.

(2) Large synaptic values can induce large voltage jumps at the analog input that can decrease the frequency for recurrent synapses, as highlighted in figure 5(d). Without compensation, this strong coupling scenario could lead to multiple oscillation frequencies and prevent global phase-locking.

5. Conclusion

This article introduced SKONN, a novel mixed-signal ONN architecture that enables large-scale analog computations in phase domain. We presented how the association of simple circuitry can enable robust analog phase computation; while propagating the information in the digital domain to facilitate the implementation of large networks. We first reported experiments on a 9-neuron SKONN on PCB that finds the maximum cut of weighted graphs with high accuracy. Then, we presented the design of a CMOS 16-neuron integrated circuit (IC), highlighting SKONN programmability, modularity, and benchmarking with state-of-the-art ONN ICs. Furthermore, our design choice led to interesting phase dynamics that are a *saturated* version of the Kuramoto model and have unique properties. Such as, we found that such dynamics tend to binarize phases and are very efficient to solve NP-hard problems based on binary variables like Max-Cut. It appeared that SKONN's accuracy on random graphs is as good as the state-of-the-art Goemans-Williamson (GW) and CirCut algorithms, and even higher when benchmarked with graphs with up to 7000 nodes from the G-set. Our study revealed that SKONN's computation time grows logarithmically with the network size which is promising for solving large-scale problems. For example, SKONN neurons oscillating at 1 MHz would provide four orders of magnitude runtime improvement compared to the GW algorithm run on a CPU.

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments

We thank Laurent Deknyff from LIRMM, CNRS, for his circuit design expertise and for fabricating the PCB proof-of-concept. We also thank Jeremie Salles and Fathi Ben Ali from LIRMM, CNRS, for their support during the IC design.

Declarations

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.

Appendix A. SKONN voltage and current dynamics

A.1. Neuron voltage dynamics

By denoting I_{ij} the input synaptic currents, the voltage dynamics of neuron i can be modeled as follows:

$$\begin{cases} C_L \frac{\mathrm{d}V_i^{\mathrm{in}}}{\mathrm{d}t} = I_{\mathrm{bias}}\left(1 - 2V_i^{\mathrm{out}}/V_{DD}\right) + \sum_j I_{ij} \\ \tau_H \frac{\mathrm{d}V_i^{\mathrm{out}}}{\mathrm{d}t} = V_{DD} f_H\left(V_i^{\mathrm{in}}, V_i^{\mathrm{out}}, V_L, V_H\right) - V_i^{\mathrm{out}} \end{cases} \tag{29}$$

where C_L is the input capacitance, I_{bias} the current that charges and discharges C_L, V_{DD} is the amplitude of V_i^{out}, V_L and V_H are the lower and upper thresholds of the hysteresis block, and τ_H is the time constant linked to the output load of the hysteresis block. The term f_H expresses the output switching with hysteresis behavior. As in [59], one can model the hysteresis behavior using a tanh function with slope γ:

$$f_H = 0.5\left(1 + \tanh\left(\gamma\left(V_i^{\mathrm{in}} - V_H - \frac{V_L - V_H}{V_{DD}} V_i^{\mathrm{out}}\right)\right)\right). \tag{30}$$

When $f_H = 0$, $V_i^{\mathrm{out}} = 0$ and C_L charges. When $f_H = 1$, $V_i^{\mathrm{out}} = V_{DD}$ and C_L discharges. The two switching occur when $V_i^{\mathrm{in}} = V_L$ and $V_i^{\mathrm{in}} = V_H$, respectively. Figure 12 shows an example of numerical solution for the equation (29).

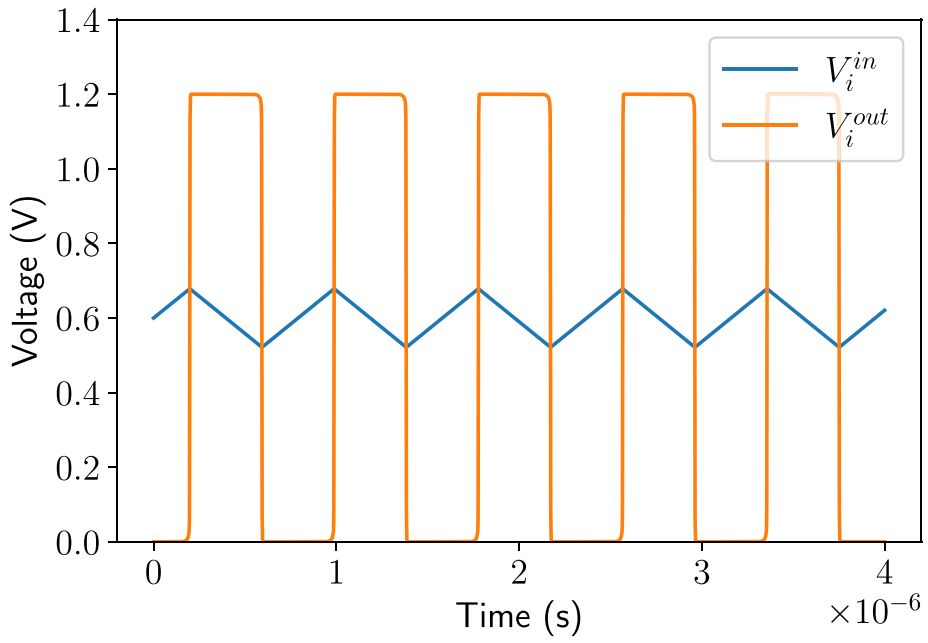

Figure 12. Voltage dynamics of a single neuron obtained by solving numerically (29). In this example, $C_L = 500$ fF, $I_{\text{bias}} = 200$ nA, $V_L = 0.5$ V, $V_H = 0.7$ V, $V_{DD} = 1.2$ V, $\gamma = 100$ and $\tau_H = 1$ ns.

A.2. Synaptic currents

In SKONN, the digital output voltage V_j^{out} goes through the synaptic capacitance C_{ij} that creates current spikes holding the phase information ϕ_j. The synaptic spike train can be expressed as follows:

$$I_{ij} = C_{ij}\left(\frac{\mathrm{d}V_j^{\text{out}}}{\mathrm{d}t} - \frac{\mathrm{d}V_i^{\text{in}}}{\mathrm{d}t}\right). \tag{31}$$

A.2.1. Voltage dynamics of coupled neurons

By injecting the synaptic current expression in equation (29), we obtain:

$$\begin{cases} C_{\text{eq}}\frac{\mathrm{d}V_i^{\text{in}}}{\mathrm{d}t} = I_{\text{bias}}\left(1 - 2V_i^{\text{out}}/V_{DD}\right) + \sum_j C_{ij}\frac{\mathrm{d}V_j^{\text{out}}}{\mathrm{d}t} \\ \tau_H\frac{\mathrm{d}V_i^{\text{out}}}{\mathrm{d}t} = V_{DD}f_H\left(V_i^{\text{in}}, V_i^{\text{out}}, V_L, V_H\right) - V_i^{\text{out}} \end{cases}. \tag{32}$$

With C_{eq} the equivalent capacitance:

$$C_{\text{eq}} = C_L + \sum_j C_{ij} \tag{33}$$

(33) indicates that the synaptic capacitances are added to the oscillator load and slow down the charge and discharge of the input node. Large synaptic capacitances could potentially induce heterogeneous frequencies within SKONN and still needs to be explored.

Appendix B. SKONN phase perturbation vector

The PPV is a T-periodic function $\vec{v}(t)$ that quantifies the phase shift of an oscillator subject to a perturbation occurring at time t [33]. One way of computing $\vec{v}(t)$ is to inject a pulsed perturbation to the oscillator at time t, measure the induced phase shift and normalize by the perturbation's strength [10]. In SKONN, the synaptic current I_{ij} perturbs the triangular oscillation V_i^{in} and the scalar PPV $v(t)$ can be derived by computing the phase shift $\mathrm{d}\phi$ when injecting current pulses $I(t') = \mathrm{d}Q\delta(t' - t)$ with $t \in [0; T]$. From figure 13, we distinguish three cases:

(1) $0 < t < T/2$: the perturbed oscillation is shifted toward the left by the same amount of time $-\mathrm{d}t$.
(2) $T/2 < t < T$: the perturbed oscillation is shifted toward the right by the same amount of time $+\mathrm{d}t$.
(3) $t \in \{0; T/2\}$: the time shift is undefined as V_i^{in}'s slope is undefined (not derivable).

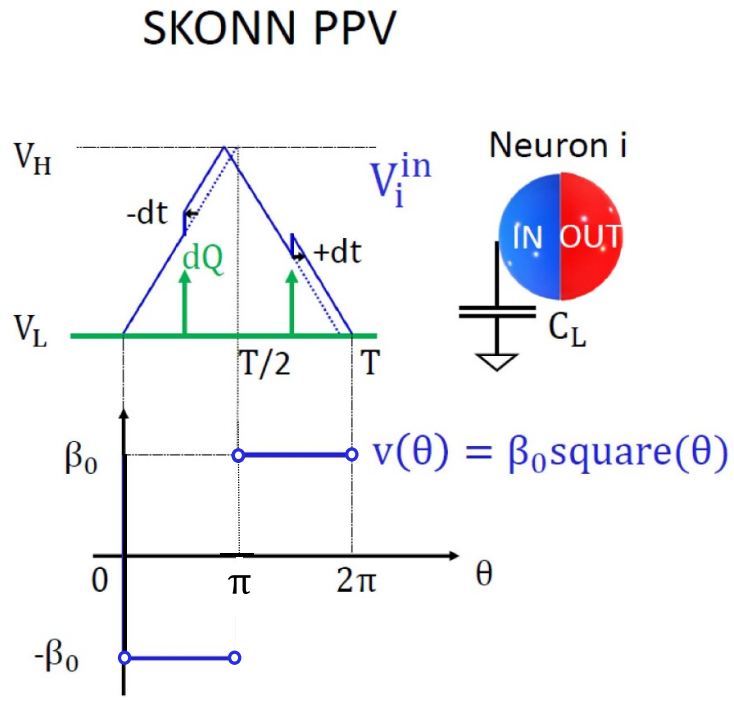

Figure 13. SKONN phase perturbation vector (PPV). The injection of a charge dQ induces a time shift $\pm$dt, which in turn creates a phase shift $\mathrm{d}\phi = 2\pi\,\mathrm{d}t/T$. The phase shift sign changes when there is a change in V_i^{in}'s slope.

Injecting dQ to C_L induces a voltage jump $\mathrm{d}V = \mathrm{d}Q/C_L$. This results in a time delay $-\mathrm{d}t$ and $+\mathrm{d}t$ when C_L charges and discharges, respectively, with $|\mathrm{d}t| = C_L\mathrm{d}V/I_{\text{bias}}$. The amount of phase shift can then be expressed as $\mathrm{d}\phi = 2\pi\,\mathrm{d}t/T$. The oscillation period T is expressed by $T = 2C_L\Delta V/I_{\text{bias}}$, with $\Delta V = V_H - V_L$. Finally, merging the equations leads to

$$\frac{\mathrm{d}\phi}{\mathrm{dQ}} = \pm\frac{\pi}{C_L\Delta V} = \pm\beta_0. \tag{34}$$

Which is the phase shift caused by the injection of 1 Coulomb. The phase shift sign depends on whether the charge is injected during the charge or discharge of the triangular waveform. Considering the three previous cases and changing the time variable t to phase θ, we express SKONN's PPV as follows:

$$v(\theta) = \beta_0\,\mathrm{square}(\theta) \tag{35}$$

where:

$$\mathrm{square}(\theta) = \begin{cases} -1, & \text{if } 0 < \theta < \pi \\ +1, & \text{if } \pi < \theta < 2\pi \end{cases}. \tag{36}$$

Appendix C. SKONN phase dynamics

C.1. Two coupled oscillators

The phase dynamics of a single oscillator of frequency $\omega_0 = 2\pi/T$ are:

$$\frac{\mathrm{d}}{\mathrm{d}t}\phi(t) = \omega_0. \tag{37}$$

When the oscillator receives a pre-synaptic signal $\vec{b}(t)$, it undergoes a time shift $\alpha(t)$ associated with the perturbation $\vec{b}(t)$. If the variation of the oscillating amplitude remains small [33], $\alpha(t)$ dynamics can be expressed as follows:

$$\frac{\mathrm{d}}{\mathrm{d}t}\alpha(t) = \vec{v}(t+\alpha(t)).\vec{b}(t) \tag{38}$$

where $\vec{v}(t)$ is the T-periodic phase perturbation vector (PPV) associated with the oscillator; and describes the phase sensitivity of the oscillator under injections at different nodes. In our case, we consider scalars $b(t)$ and $v(t)$ as the pre-synaptic signal $b(t)$ is injected to a unique input node. As $b(t)$ also oscillates at frequency ω_0 and with phase $\phi_b(t) = \omega_0 t$, we introduce $\Delta\phi(t) = \phi(t) - \phi_b(t) = \omega_0\alpha(t)$ that expresses the phase difference between post and presynaptic signals. The latter can be considered as the reference as it is driving the oscillator. To simplify equations, we define the 2π-periodic PPV and perturbation as $v^{2\pi}(\omega_0 t) = v(t)$ and $b^{2\pi}(\omega_0 t) = b(t)$, respectively. The phase dynamics become:

$$\frac{\mathrm{d}}{\mathrm{d}t}\Delta\phi = \omega_0\, v^{2\pi}\left(\phi_b + \Delta\phi\right) b^{2\pi}\left(\phi_b\right). \tag{39}$$

We assume that under weak coupling, the phase difference $\Delta\phi$ evolves slowly compared to the presynaptic phase ϕ_b and it is common practice to average out $\Delta\phi$ over one period:

$$\begin{aligned}\frac{1}{2\pi}\int_{-\pi}^{\pi}\frac{\mathrm{d}}{\mathrm{d}t}\Delta\phi\, d\phi_b &\approx \frac{\mathrm{d}}{\mathrm{d}t}\Delta\phi \\ &= \frac{1}{T}\int_{-\pi}^{\pi} v^{2\pi}\left(\phi_b + \Delta\phi\right) b^{2\pi}\left(\phi_b\right)\mathrm{d}\phi_b.\end{aligned} \tag{40}$$

We saw previously that a pre-synaptic signal consists of current pulses that are aligned with the rising and falling edges of the digital pre-synaptic voltage. Then, we consider the case where $b^{2\pi}(\theta)$ consists of a train of Dirac pulses:

$$b^{2\pi}(\theta) = \sum_{n=0}^{\infty} p(\theta - n2\pi). \tag{41}$$

With:

$$p(\theta) = Q\left(\delta(\theta) - \delta(\theta - \pi)\right). \tag{42}$$

Under this assumption, (40) becomes:

$$\frac{\mathrm{d}}{\mathrm{d}t}\Delta\phi = \frac{Q}{T}\left(v^{2\pi}(\Delta\phi) - v^{2\pi}(\Delta\phi + \pi)\right). \tag{43}$$

In SKONN, the analog input oscillation is a symmetric triangular waveform that has a simple 2π-periodic PPV expressed as follows:

$$v^{2\pi}(\theta) = \beta_0\,\mathrm{square}(\theta) \tag{44}$$

where:

$$\mathrm{square}(\theta) = \begin{cases} -1, & \text{if } 0 < \theta < \pi \\ +1, & \text{if } \pi < \theta < 2\pi \end{cases} \tag{45}$$

and β_0 is the phase shift induced by the injection of 1 Coulomb to the oscillating node. Finally, we express the phase dynamics of the driven oscillator:

$$\frac{\mathrm{d}}{\mathrm{d}t}\Delta\phi = 2\beta_0\,\frac{Q}{T}\,\mathrm{square}(\Delta\phi). \tag{46}$$

We notice that the average of the phase dynamics are very similar to the Kuramoto model except that its sinusoidal interaction term is replaced by a *saturated* square function in our case.

Proposition 1. *If the injected charge $Q \neq 0$ then the two SKONN oscillators admit a unique stable fixed-point $\Delta\phi^* = (\phi_i - \phi_j)^*$ such that:*

$$\Delta\phi^* = \begin{cases} 0, & \text{if } Q > 0 \\ \pi, & \text{if } Q < 0 \end{cases}. \tag{47}$$

Proof. The proof consists in finding a Lyapunov function for the dynamics (46). Consider the bounded continuous Lyapunov function:

$$E = 2\beta_0 \frac{Q}{T} \text{triangle}(\Delta\phi). \tag{48}$$

With:

$$\text{triangle}(\theta) = \begin{cases} \theta - \pi/2, & \text{if } 0 \leqslant \theta \leqslant \pi \\ 3\pi/2 - \theta, & \text{if } \pi \leqslant \theta \leqslant 2\pi \end{cases}. \tag{49}$$

We have :

$$\frac{\partial E}{\partial \Delta\phi} = -2\beta_0 \frac{Q}{T} \text{square}(\Delta\phi) = -\frac{\mathrm{d}\Delta\phi}{\mathrm{d}t}. \tag{50}$$

E is minimized through time as follows:

$$\frac{\mathrm{d}E}{\mathrm{d}t} = \frac{\partial E}{\partial \Delta\phi}\frac{\mathrm{d}\Delta\phi}{\mathrm{d}t} = -\left(\frac{\mathrm{d}\Delta\phi}{\mathrm{d}t}\right)^2 \leqslant 0. \tag{51}$$

(1) If $Q > 0$, the minima of E are $\Delta\phi^* = 0\,[2\pi]$ and correspond to the phase fixed points of the dynamics (46).
(2) If $Q < 0$, the minima of E are $\Delta\phi^* = \pi\,[2\pi]$ and correspond to the phase fixed points of the dynamics (46).

□

In other words, propagating a spike train that consists of positive and negative current spikes spaced in time by $T/2$ induce an in-phase or out-of-phase locking, depending on the polarity of Q. The latter can be set by choosing one of the two complementary digital post synaptic voltages as shown in figure 3.

C.2. N coupled oscillators

When an oscillator i is perturbed by N other oscillators with same pulsation ω_0, (38) can be generalized in the scalar case:

$$\frac{\mathrm{d}}{\mathrm{d}t}\alpha_i = \sum_{j=1}^{N} v_{ij}^{2\pi}\left(\phi_i\right) b_j^{2\pi}\left(\phi_j\right). \tag{52}$$

Similarly to the two-oscillators case, averaging out the previous equation along the fast variable ϕ_j leads to:

$$\frac{\mathrm{d}}{\mathrm{d}t}\alpha_i = \sum_{j=1}^{N} \frac{1}{2\pi}\int_{-\pi}^{\pi} v_{ij}^{2\pi}\left(\Delta\phi_{ij} + \phi_j\right) b_j^{2\pi}\left(\phi_j\right) \mathrm{d}\phi_j. \tag{53}$$

We use the spike train expression (41) to obtain:

$$\frac{\mathrm{d}}{\mathrm{d}t}\alpha_i = \frac{1}{2\pi}\sum_{j=1}^{N}\left(v_{ij}^{2\pi}\left(\Delta\phi_{ij}\right) - v_{ij}^{2\pi}\left(\Delta\phi_{ij} + \pi\right)\right). \tag{54}$$

As we inject pre-synaptic signals to the same node, we have $v_{ij}^{2\pi} = v^{2\pi}$ and we use the SKONN oscillator PPV $v^{2\pi}$ (44) to finally get:

$$\frac{\mathrm{d}}{\mathrm{d}t}\phi_i = \frac{2\beta_0}{T}\sum_{j=1}^{N} Q_{ij}\,\text{square}\left(\phi_i - \phi_j\right). \tag{55}$$

Note that we omitted the term ω_0 in the right-hand side of (55) as in practice we refer to the relative phase relationship between oscillators instead of the absolute values that linearly increase with $\omega_0 t$.

Considering SKONN's hardware implementation, we saw that β_0 is equal to:

$$\beta_0 = \frac{\pi}{\Delta V C_L} \tag{56}$$

and:

$$Q_{ij} = V_{DD} C_{ij} \tag{57}$$

where ΔV is the peak-to-peak triangular amplitude at the input, C_L is the neuron input capacitance, V_{DD} is the digital voltage swing, and C_{ij} is the synaptic capacitance value. SKONN's phase dynamics become:

$$\frac{\mathrm{d}}{\mathrm{d}t}\phi_i = \omega_0 \frac{V_{DD}}{\Delta V} \sum_{j=1}^{N} \frac{C_{ij}}{C_L} \mathrm{square}\left(\phi_i - \phi_j\right). \tag{58}$$

SKONN has an interesting phase binarization property resumed in the following proposition:

Proposition 2. *Consider a neuron i of degree D, i.e. driven by D neurons j with weighted charges* $Q_{ij} \in \{-q, +q\}$ $q \neq 0$.

(1) If D is odd and $\mathrm{d}\phi_i/\mathrm{d}t = 0$*, then there is at least one input neuron j such that* $(\phi_i - \phi_j)$ *is a multiple of* π.
(2) If D is even, then there is at least one ϕ_i *and one set of input phase* ϕ_j *such that* $\mathrm{d}\phi_i/\mathrm{d}t = 0$ *and* $\forall j (\phi_i - \phi_j)$ *is not a multiple of* π.

Proof.

(1) By assuming the opposite, i.e. that $\forall j (\phi_i - \phi_j) \neq 0\ [\pi]$, it leads to $\forall j \mathrm{square}(\phi_i - \phi_j) = \pm 1$, using (4). Noticing that $D = m + l$ with $m \neq l$, and writing SKONN's phase dynamics (7) leads to:

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = 0 \implies \sum_{j=1}^{D} \pm 1 = \sum_{j=1}^{m} 1 - \sum_{j=1}^{l} 1 = 0. \tag{59}$$

Which is not possible as $m \neq l$ and proves the proposition.

(2) Consider the integers m and l such that there are m weights $Q_{ij} = +q$ and l weights $Q_{ij} = -q$, with $D = m + l = 2k$. We can choose:
(a) $\phi_j = 0$ for the l (resp. m) input neurons.
(b) $\phi_j = \pi$ for $k - l$ (resp. $k - m$) other input neurons.
(c) $\phi_j = 0$ for the remaining k input neurons.
From SKONN's phase dynamics (7) we obtain:

$$\frac{T}{2\beta_0}\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = -q \sum_{j \leqslant l} \mathrm{square}(\phi_i) + q \sum_{l < j \leqslant k} \mathrm{square}(\phi_i - \pi) + q \sum_{k < j \leqslant 2k} \mathrm{square}(\phi_i).$$

Assuming that $\phi_i \neq 0\ [\pi]$, it follows from (4):

$$\begin{aligned} \frac{T}{2\beta_0}\frac{\mathrm{d}\phi_i}{\mathrm{d}t} &= -ql(\pm 1) - q(k-l)(\pm 1) + kq(\pm 1) \\ &= 0. \end{aligned}$$

□

Appendix D. Comparison between SKONN phase model and circuit dynamics

The phase dynamics defined in (58) only provide the phases ϕ_i^m and do not include voltage and current equations. Here, we compare ϕ_i^m with ϕ_i^c, the phases extracted from the circuit dynamics expressed in (32). For every oscillator i, the pair of output voltages V_i^{out} and V_1^{out} are fed to a 2-XOR gate which produces a signal V_i^{XOR} whose duty cycle is proportional to the phase between the first (reference) and the ith oscillator (figure 14(a)). Then, the voltages V_i^{XOR} are filtered out with a second-order low-pass filter to obtain the phase dynamics ϕ_i^c. Figures 14(b) and (c) show an example of circuit simulation for 4 fully-coupled oscillators with $W_{ij} = -1$ and a random initialization. Figure 14(d) shows the corresponding phases ϕ_i^c and ϕ_i^m (dashed lines) obtained from the circuit and the phase model, respectively. Note that the XOR-based phase measurement is insensitive to the phase sign and so ϕ_3^c departs from ϕ_3^m when $\phi_3^m > \phi_3^c = 180°$. However, computing $\cos(\phi_i^c)$ and $\cos(\phi_i^m)$ provide the spin dynamics for the two approaches and are shown at the bottom of figure 14(d). Overall, in the weak-coupling regime, we observe a good agreement between the phase and circuit models. This comparison motivates the use of the phase model (58) as it only contains N ODEs for emulating an N-node SKONN, compared to at least 2 N ODEs for the circuit approach (32).

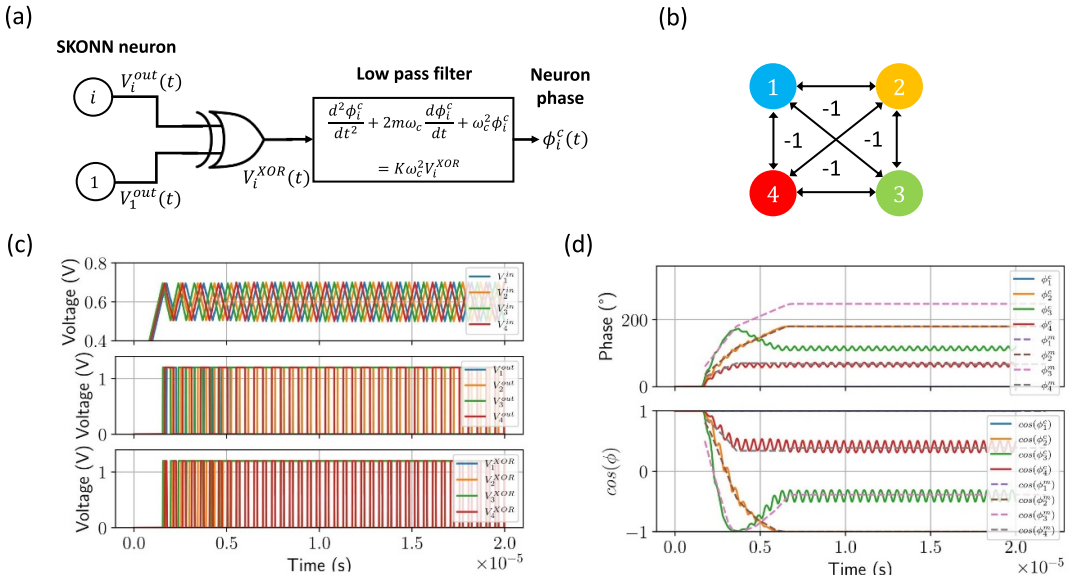

Figure 14. (a) Proposed phase read-out to analyze the circuit phase dynamics. We set the cut-off frequency $\omega_c = \omega_0/2 = 500$ MHz and damping factor $m = \sqrt{2}/2$. The gain K converts voltage to phase as $K = \pi/V_{DD}$. Circuit parameters are $C_L = 500$ fF, $I_{\text{bias}} = 200$ nA, $V_L = 0.5$ V, $V_H = 0.7$ V, $V_{DD} = 1.2$ V, $\gamma = 100$ and $\tau_H = 1$ ns. (b) Example of SKONN network. $|W_{ij}| = 1$ corresponds to a coupling capacitance $C_{ij} = 2.5$ fF. (c) Circuit dynamics obtained by solving numerically (32). (d) Comparison between SKONN's phase model ϕ_i^m (58) and the circuit phase ϕ_i^c.

Appendix E. Sub harmonic injection locking in SKONN

To binarize the phases, one can inject a signal at twice the oscillating frequency $V_{\text{SHIL}}(t) = A\sin(4\pi\omega_0 t)$ to an oscillator's node for which its scalar PPV contains a second-order harmonic $P_2 \neq 0$ in its Fourier decomposition [32]. In SKONN, we cannot inject the 2-SHIL signal to the input oscillating node $V_i^{\text{in}}(t)$ as the associated scalar PPV only contains odd harmonics (square (35)). In practice, we inject the 2-SHIL signal to a biasing node that allows SKONN binary phase locking. In this case, the SKONN Lyapunov function becomes

$$E = \frac{\beta_0}{T}\sum_{i,j}^{N} Q_{ij}\,\text{triangle}\left(\phi_i - \phi_j\right) + \sum_{i}^{N} A_i P_2 \cos(2\phi_i). \tag{60}$$

When SHIL amplitudes A_i are large enough, phases are binarized $\phi_i = (1 - S_i)\pi/2 \in \{0, \pi\}$ and the SKONN Lyapunov function corresponds to the Ising Hamiltonian H with an additional offset:

$$\begin{aligned} E &= -\frac{\pi\beta_0}{2T}\sum_{i,j}^{N} Q_{ij}S_iS_j + \sum_{i}^{N} A_i P_2 \\ &= H + \text{constant} \end{aligned}. \tag{61}$$

Appendix F. Impact of SKONN's limited bandwidth

When SKONN is implemented in hardware, we observe some phase deviation with respect to the theoretical phase fixed points, as shown in figure 15. The main reason is the hysteresis block that does not switch instantaneously when the synaptic current spikes induce voltage jumps above or below the hysteresis thresholds V_H and V_L. To better understand this phenomenon, we ran two transistor-level simulations of two coupled neurons in feed-forward mode with a strong weight $W_{21} = +15$ and different frequencies (figure 15). When the oscillation frequency is low (300 kHz), the hysteresis switching delay is negligible and there is only a small phase deviation $\delta\phi = 4°$. However, when the oscillation frequency increases to 1.2 MHz, the limited bandwidth of the hysteresis circuit causes a switching delay and a larger phase deviation $\delta\phi = 13°$.

For a given phase precision required by the application, this error could be mitigated by increasing the bandwidth of the hysteresis circuit or slowing down the oscillators, and constitute a trade-off with the energy consumption [17]. Interestingly, we have observed in experiments that having recurrent synapses $W_{ij} = W_{ji}$

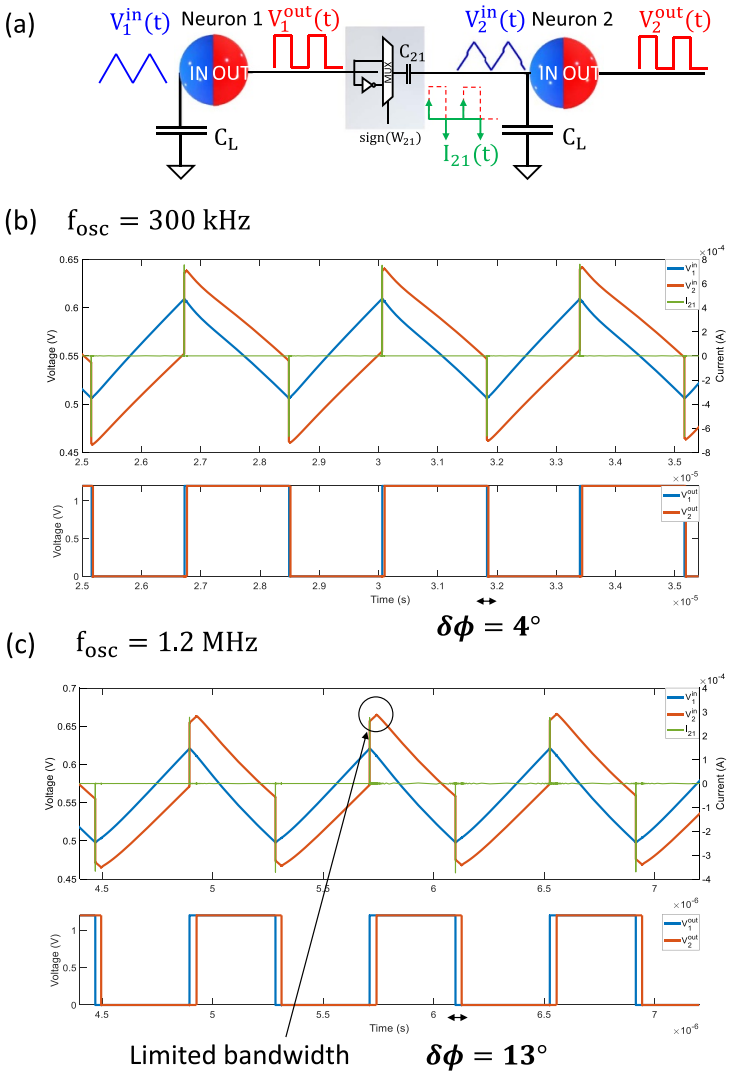

Figure 15. (a) Two coupled oscillator in feed-forward mode with $+C_{21} = 7.5\%C_L$ which implements $W_{21} = +15$ in the ASIC. (b) Transistor-level simulation with $C_L = 2$ pF to decrease the frequency to $f_{osc} = 300$ kHz. (c) Transistor level simulation in nominal case where $C_L = 500$ fF and $f_{osc} = 1.2$ MHz.

compensate the hysteresis delay induced in both neurons and the theoretical phase fixed point is reached in that case (see figures 5(c) and (d)).

ORCID iDs

Corentin Delacour https://orcid.org/0000-0001-6386-9588
Stefania Carapezzi https://orcid.org/0000-0002-9271-1189
Gabriele Boschetto https://orcid.org/0000-0001-8830-3572
Madeleine Abernot https://orcid.org/0000-0002-8267-2972
Aida Todri-Sanial https://orcid.org/0000-0001-8573-2910

References

[1] Huygens C 1888 *Oeuvres complètes de Christiaan Huygens* Société Hollandaise des Sciences vol 1, ed M Nijhoff (Martinus Nijhoff)
[2] von Neumann J 1954 *Non-linear capacitance or inductance switching, amplifying and memory devices* 426149
[3] Goto E 1959 The parametron, a digital computing element which utilizes parametric oscillation *Proc. IRE* **47** 1304–16
[4] Takahashi S 1996 A brief history of the Japanese computer industry before 1985 *IEEE Ann. Hist. Comput.* **18** 76
[5] Jackson T, Pagliarini S and Pileggi L 2018 An oscillatory neural network with programmable resistive synapses in 28 nm cmos *2018 IEEE Int. Conf. on Rebooting Computing (ICRC)* pp 1–7
[6] Corti E, Antonio Cornejo Jimenez J, Niang K M, Robertson J, Moselund K E, Gotsmann B, Ionescu A M and Karg S 2021 Coupled VO_2 oscillators circuit as analog first layer filter in convolutional neural networks *Front. Neurosci.* **15** 628254

[7] Delacour C, Carapezzi S, Abernot M, Boschetto G, Azemard N, Salles J, Gil T and Todri-Sanial A 2021 Oscillatory neural networks for edge AI computing *2021 IEEE Computer Society Annual Symp. on VLSI (ISVLSI)* pp 326–31

[8] Abernot M and Todri-Sanial A 2023 Simulation and implementation of two-layer oscillatory neural networks for image edge detection: bidirectional and feedforward architectures *Neuromorph. Comput. Eng.* **3** 014006

[9] Wang T, Leon W, Nobel P and Roychowdhury J 2021 Solving combinatorial optimisation problems using oscillator based Ising machines *Nat. Comput.* **20** 287–306

[10] Dutta S, Khanna A, Assoa A S, Paik H, Schlom D G, Toroczkai Z, Raychowdhury A and Datta S 2021 An Ising Hamiltonian solver based on coupled stochastic phase-transition nano-oscillators *Nat. Electron.* **4** 502–12

[11] Mallick A, Bashar M K, Truesdell D S, Calhoun B H, Joshi S and Shukla N 2020 Using synchronized oscillators to compute the maximum independent set *Nat. Commun.* **11** 4689

[12] Moy W, Ahmed I, Chiu P-wei, Moy J, Sapatnekar S S and Kim C H 2022 A 1,968-node coupled ring oscillator circuit for combinatorial optimization problem solving *Nat. Electron.* **5** 310–7

[13] Hopfield J J 1982 Neural networks and physical systems with emergent collective computational abilities *Proc. Natl Acad. Sci.* **79** 2554–8

[14] Izhikevich E and Kuramoto Y 2006 Weakly coupled oscillators *Encyclop. Math. Phys.* **1** 448–53

[15] Aoyagi T 1995 Network of neural oscillators for retrieving phase information *Phys. Rev. Lett.* **74** 4075–8

[16] Hoppensteadt F C and Izhikevich E M 2000 Pattern recognition via synchronization in phase-locked loop neural networks *IEEE Trans. Neural Netw.* **11** 734–8

[17] Delacour C, Carapezzi S, Abernot M and Todri-Sanial A 2023 Energy-performance assessment of oscillatory neural networks based on VO_2 devices for future edge ai computing *IEEE Trans. Neural Netw. Learn. Syst.* 1–14

[18] Wang T, Leon W and Roychowdhury J 2019 New computational results and hardware prototypes for oscillator-based Ising machines *Proc. 56th Annual Design Automation Conf. 2019, DAC '19 (New York, USA)* (Association for Computing Machinery) (https://doi.org/10.1145/3316781.3322473)

[19] Chou J, Bramhavar S, Ghosh S and Herzog W 2019 Analog coupled oscillator based weighted Ising machine *Sci. Rep.* **9** 14786

[20] Khairul Bashar M, Mallick A and Shukla N 2021 Experimental investigation of the dynamics of coupled oscillators as Ising machines *IEEE Access* **9** 148184–90

[21] Graber M and Hofmann K 2022 A versatile & adjustable 400 node CMOS oscillator based Ising machine to investigate and optimize the internal computing principle *2022 IEEE 35th Int. System-on-Chip Conf. (SOCC)* pp 1–6

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

[23] Torrejon J *et al* 2017 Neuromorphic computing with nanoscale spintronic oscillators *Nature* **547** 428–31

[24] Grollier J, Querlioz D, Camsari K Y, Everschor-Sitte K, Fukami S and Stiles M D 2020 Neuromorphic spintronics *Nat. Electron.* **3** 360–70

[25] Maffezzoni P, Bahr B, Zhang Z and Daniel L 2016 Analysis and design of Boolean associative memories made of resonant oscillator arrays *IEEE Trans. Circuits Syst.* I **63** 1964–73

[26] Hoppensteadt F C and Izhikevich E M 2001 Synchronization of MEMS resonators and mechanical neurocomputing *IEEE Trans. Circuits Syst.* I **48** 133–8

[27] Shukla N *et al* 2014 Synchronized charge oscillations in correlated electron systems *Sci. Rep.* **4** 4964

[28] Carapezzi S, Boschetto G, Delacour C, Corti E, Plews A, Nejim A, Karg S and Todri-Sanial A 2021 Advanced design methods from materials and devices to circuits for brain-inspired oscillatory neural networks for edge computing *IEEE J. Emerg. Sel. Top. Circuits Syst.* **11** 586–96

[29] Jackson T C, Sharma A A, Bain J A, Weldon J A and Pileggi L 2015 Oscillatory neural networks based on TMO nano-oscillators and multi-level RRAM cells *IEEE J. Emerg. Sel. Top. Circuits Syst.* **5** 230–41

[30] Dutta S, Khanna A, Chakraborty W, Gomez J, Joshi S and Datta S 2019 Spoken vowel classification using synchronization of phase transition nano-oscillators *2019 Symp. on VLSI Circuits* pp T128–9

[31] Nikonov D, Kurahashi P, Ayers J, Hai Li, Kamgaing T, Dogiamis G, Lee H-J, Fan Y and Young I 2020 Convolution inference via synchronization of a coupled CMOS oscillator array *IEEE J. Exploratory Solid-State Comput. Devices Circuits* **6** 1

[32] Neogy A and Roychowdhury J 2012 Analysis and design of sub-harmonically injection locked oscillators *2012 Design, Automation & Test in Europe Conf. & Exhibition (DATE)* pp 1209–14

[33] Bhansali P and Roychowdhury J 2009 Gen-Adler: The generalized Adler's equation for injection locking analysis in oscillators *2009 Asia and South Pacific Design Automation Conf.* pp 522–7

[34] Todri-Sanial A *et al* 2022 How frequency injection locking can train oscillatory neural networks to compute in phase *IEEE Trans. Neural Netw. Learn. Syst.* **33** 1996–2009

[35] Delacour C and Todri-Sanial A 2021 Mapping Hebbian learning rules to coupling resistances for oscillatory neural networks *Front. Neurosci.* **15** 694549

[36] Hoppensteadt F C and Izhikevich E M 1997 Weakly Connected Oscillators *Weakly Connected Neural Networks* Applied Mathematical Sciences (Springer) pp 247–93

[37] Goemans M X and Williamson D P 1995 Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming *J. ACM* **42** 1115–45

[38] Burer S, Monteiro R and Zhang Y 2001 Rank-two relaxation heuristics for Max-cut and other binary quadratic programs *SIAM J. Optim.* **12** 503–21

[39] Honjo T *et al* 2021 100,000-spin coherent Ising machine *Sci. Adv.* **7** eabh0952

[40] Hamerly R *et al* 2019 Experimental investigation of performance differences between coherent Ising machines and a quantum annealer *Sci. Adv.* **5** eaau0823

[41] Cai F *et al* 2020 Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks *Nat. Electron.* **3** 409–18

[42] Ising E 1925 Beitrag zur theorie des ferromagnetismus *Z. Phys.* **31** 253–8

[43] Lucas A 2014 Ising formulations of many NP problems *Front. Phys.* **2** 5

[44] Erementchouk M, Shukla A and Mazumder P 2022 On computational capabilities of Ising machines based on nonlinear oscillators *Physica* D **437** 133334

[45] Erementchouk M, Shukla A and Mazumder P 2023 Self-contained relaxation-based dynamical Ising machines (arXiv:2305.06414 [cs, math])

[46] Erdös P *et al* 1960 On the evolution of random graphs *Publ. Math. Inst. Hung. Acad. Sci* **5** 17–60

[47] Khairul Bashar M, Lin Z and Shukla N 2023 Stability of oscillator Ising machines: not all solutions are created equal (arXiv:2301.07601)
[48] Romera M *et al* 2018 Vowel recognition with four coupled spin-torque nano-oscillators *Nature* **563** 230–4
[49] Scellier B and Bengio Y 2017 Equilibrium propagation: bridging the gap between energy-based models and backpropagation *Front. Comput. Neurosci.* **11** 24
[50] Zoppo G, Marrone F, Bonnin M and Corinto F 2022 Equilibrium propagation and (memristor-based) oscillatory neural networks *2022 IEEE Int. Symp. on Circuits and Systems (ISCAS)* pp 639–43
[51] Grant M and Stephen B 2014 CVX: Matlab software for disciplined convex programming, version 2.1 (available at: http://cvxr.com/cvx)
[52] Martí D 2009 G-set data for Max-cut (https://grafo.etsii.urjc.es/optsicom/maxcut.html)
[53] Choi C and Yinyu Y 2000 Solving sparse semidefinite programs using the dual scaling algorithm with an iterative solver *Manuscript* (Department of Management Sciences, University of Iowa) p 52242
[54] Sansen W 2006 *Noise Performance of Elementary Transistor Stages* (Springer) pp 117–47
[55] Enz C C, Krummenacher F and Vittoz E A 1995 An analytical MOS transistor model valid in all regions of operation and dedicated to low-voltage and low-current applications *Analog Integr. Circuits Signal Process.* **8** 83–114
[56] Penski C 2000 A new numerical method for SDEs and its application in circuit simulation *J. Comput. Appl. Math.* **115** 461–70
[57] Sickenberger T and Winkler R 2006 Efficient transient noise analysis in circuit simulation *Proc. Appl. Math. Mech.* **6** 55–58
[58] Rackauckas C and Nie Q 2017 DifferentialEquations.jl–a performant and feature-rich ecosystem for solving differential equations in Julia *J. Open Res. Softw.* **5**
[59] Maffezzoni P, Daniel L, Shukla N, Datta S and Raychowdhury A 2015 Modeling and simulation of vanadium dioxide relaxation oscillators *IEEE Trans. Circuits Syst.* I **62** 2207–15