Computing with oscillators from theoretical underpinnings to applications and demonstrators

Check for updates

Aida Todri-Sanial[1] ✉, Corentin Delacour[2], Madeleine Abernot[2] & Filip Sabo[1]

Networks of coupled oscillators have far-reaching implications across various fields, providing insights into a plethora of dynamics. This review offers an in-depth overview of computing with oscillators covering computational capability, synchronization occurrence and mathematical formalism. We discuss numerous circuit design implementations, technology choices and applications from pattern retrieval, combinatorial optimization problems to machine learning algorithms. We also outline perspectives to broaden the applications and mathematical understanding of coupled oscillator dynamics.

While the von Neumann computing architecture[1] is most likely to persist for many more years, the current demands of artificial intelligence (AI) necessitate a fundamental rethinking of how we compute, sense and communicate information[2–5]. The proliferation of electronic devices, edge devices, sensor-based electronics, and wearables, has led to a significant surge in raw data generation, demanding more power, real-time processing, and decision-making capabilities[5–7]. Although digital computing has revolutionized various industries and bolstered economies[8], the requirements of AI computing call for a different information processing paradigm and architecture[9,10]. In this evolving big data landscape, the conventional von Neumann computing architecture with separated memory and processing is no longer suitable[11]. Computing architectures must evolve to not only enable energy efficient in-memory computations but also facilitate the deployment of robust neural networks for the execution of AI tasks[12,13] that infer and learn interactively with the environment.

An emerging and innovative computing paradigm gaining momentum is *physical computing* or *natural computing*, which relies on principles rooted in physics[14]. In physical computing, physical variables are encoded into a dynamical system that evolves in accordance with the laws of classical or quantum mechanics[15,16]. This marks a departure from instruction-based information processing, where information is encoded in logical bits of either 0 or 1. In physical computing, information is represented as a continuous variable encoded either in amplitude, frequency or phase differences between signals[17]. In this paper, we focus mainly on physical computing based on coupled oscillators with information encoded in the phase differences between oscillators. Figure 1 illustrates the concept of physical computing using coupled oscillators.

Physical computing is also closely intertwined with neuroscience, particularly in the study of biological neural networks[18]. The rhythmic firing (or spiking) of neurons can be described by oscillations[19]. Neuron oscillations have been quantified through measurements where synchronized activity of a large number neurons give rise to macroscopic oscillations that can be observed in electroencephalogram (EEG)[20–22]. Oscillations can vary in amplitude and frequency but once synchronization occurs, oscillations converge and adapt to a common frequency mode[23,24]. Typically neuron activity is investigated as spike change over time such as changes in neuron membrane potential with time which are commonly emulated and studied in spiking neural networks[25–28]. In contrast, physical computing captures the *collective* neural activity through oscillations or the dynamics from a single neuron to their collective[29–31]. While spiking activities provide insight into individual neurons, oscillations reveal the dynamic nature of neural activity and give insights into how spike travels across the network and interacts with other neurons[32,33]. Such collective dynamics allow for the extraction of spatial and temporal patterns[34,35]. Understanding the evolution of oscillation dynamics and propagation are key to understanding cognitive states, particularly in the context of human memory[20,36,37]. Furthermore, describing neurons as oscillators offers a mathematically simple yet elegant formulation for investigating the dynamics of collective neurons[38]. This mathematical formalism avoids the need to simulate every voltage dependence of each ionic current in a neuron and how this moves from one neuron to the next.

Oscillatory neural networks (ONNs) hold promise for realizing both physical and neuromorphic computing while enabling continuous and analog information encoding through phase modulation[39]. These networks benefit from the diverse collective dynamics of coupled oscillators, ranging from chaotic behavior[40] to *synchronization*[41], for encoding information, sensing and processing. The phenomenon of synchronization unveils the stability and inherent memory of the network[30]. Synchronization—two or more events occurring simultaneously—is a ubiquitous phenomenon in nature[42,43]. The beauty of synchronization can be witnessed in various

[1]NanoComputing Research Lab, Integrated Circuits, Electrical Engineering Department, Eindhoven University of Technology, Eindhoven, The Netherlands.
[2]Department of Microelectronics, LIRMM, University of Montpellier, CNRS, Montpellier, France. ✉e-mail: a.todri.sanial@tue.nl
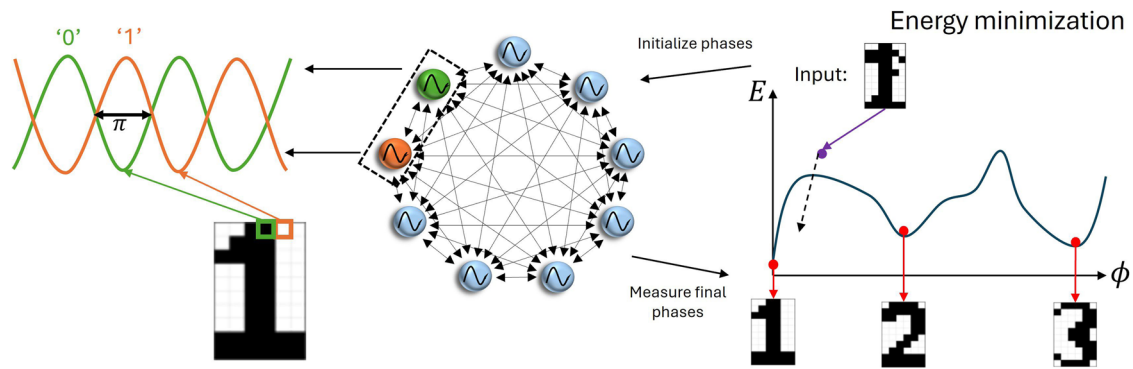
Fig. 1 | Illustration of oscillatory neural networks (ONNs) and computation involving the phase measurement between oscillators. ONNs are energy-based networks that minimize their energy as they evolve and stabilize towards ground minima.

domains, from sports like synchronized swimming and diving, to musical performances such as violinists in an orchestra playing perfectly in unison. Huygens was the first to observe synchronization between two coupled pendulums[44]. Since then, and particularly in the last century, researchers have delved into studying the computational capabilities of dynamical systems. Somehow, we are still unraveling the mystery of why Huygens pendulums swung with precision. Perhaps this essence of synchronization holds the key to our present-day computing needs. Based on their genesis, we provide a detailed review of computing with oscillators which can be identified into four main scientific branches that led to the physical computing with oscillators that we have today.

The first scientific branch is *statistical physics*, starting in the 1920s with contributions from Lenz and Ising[45]. They introduced a mathematical model describing the dynamics of magnetic spins on a lattice, establishing the groundwork for understanding how local interactions between particles can lead to global phenomena, particularly in spin systems and phase transition in magnets. Each spin represents a binary state (i.e., $+1$ or -1). In ONNs, the phase of each oscillator can be thought of as analogous to the spin in the Ising model. Oscillators tend to align their phases or synchronize depending on the coupling. In-phase synchronization can be seen as aligned spins. The other connection between oscillators and Ising model is on the interactions and energy landscape. The Ising model is a mathematical framework to describe phase transitions of interacting spins and can be understood in terms of an energy landscape, where energy depends in the configurations of spins. The system tends to evolve toward spin configurations that minimize energy. Similarly, coupled oscillators interact with each other and exhibit different phase transitions (in and out of phase) as a function of coupling strength. Such interactions can also be understood in terms of an energy-like function. ONNs evolve toward configurations where the interaction energy is minimized, leading to synchronized states. In both spin and oscillatory systems, the energy landscape determines the stable states. Progressing along this branch, Metropolis and Hastings devised methods for approximating complex probability distributions[46,47]. Further advancements emerged on probabilistic optimization algorithms, notably by Kirkpatrick and Ackley, who developed simulated annealing, a widely applied algorithm across various domains[48–50].

The second branch is on *neural networks* that eventually entwines with statistical physics. It all begins with McCullow and Pitts laying the foundation of neural networks through the mathematical neuron model and its basic operating principles[51]. The very first neural network architecture was the perceptron by Rosenblatt and later by Minsky and Papert to create the first computation model inspired by the human brain neural structure[52,53]. Hopfield introduced recurrent neural networks that can store and retrieve information also known as Hopfield neural networks (HNNs)[29]. Through continuous dynamics, HNNs minimize their energy as the network updates its states, serving as an inspiration to Hoppensteadt and Izhikevich for later developments of oscillatory neural networks[30,38,54].

The third branch is on *neuron models* that closely aligns with the progress between computational neuroscience and neural networks. Taking inspiration from biological neural networks, neuronal models have evolved to capture the excitatory and inhibitory behavior of neurons to mimick their action potential role. Several neuronal models have been developed over the years, such as van der Pol[55], Hodgkin and Huxley[56], and FitzHugh and Nagumo[57,58]. These models are biologically plausible and capture the non-linear action potential neuron behavior. Synapses also play an important role in learning and memory in neural networks. Similarly, in physical computing, coupling elements between oscillators emulate the synaptic potentiation and depression, which are mechanisms that modulate synaptic strengths such as weakly or strongly coupled oscillators. The merging of two branches, namely *neuron models* and *engineering computing with oscillators* is brought by the seminal works of Hoppensteadt et al.[30] and Izhikevich[38] that established the foundation for physical computing with oscillators.

The fourth branch is on *engineering of computing with oscillators*, with the first efforts reported to implement them for computing Boolean functions dating back to the 1960s[59,60]. However, oscillatory-based logic could not compete with more scalable transistor-based logic and faded away within a decade. At about the same time, Kuramoto et al.[61] developed the analytical formalism to analyze the dynamics of coupled oscillators. These four branches started to merge in the late 80s that led to the rebirth of the oscillatory-based physical computing. Figure 2 presents the evolution of the research field on oscillatory neural networks covering the progress on each of the four branches. Nowadays, the design of coupled oscillatory systems is driven by the development of novel oscillator circuits which promise higher scalability compared to the classical LC-based oscillators.

In recent years, researchers have started to investigate novel materials and devices to implement oscillators and coupling elements. There are ongoing efforts on phase transition and 2D transition metal-oxide devices (TMO)[62,63], spintronic devices[64], microelectromechanical systems (MEMS)[65], optical systems[66], or CMOS accelerators[67–69] as potentially scalable solutions[70,71] for applications like digital logic[72,73], classification[74–76], computing convolutions[62,77], or solving associative tasks[30,39,67,78,79] and combinatorial optimization problems (COP)[63,80–82].

This paper presents an in-depth view on the computational principles of coupled oscillatory systems used in various applications from Machine Learning (ML) to COPs covering aspects from oscillator to ONN architecture design and technology implementation. More concretely, it is divided into five parts. Starting of with "Oscillatory neuron", we introduce some of the various implementations of oscillatory neurons. Next, "ONN computing" presents the corresponding coupling elements and the resulting dynamics that can be tailored for specific applications. "ONN architectures and demonstrators" reports on the state-of-the-art ONN architectures and demonstrators implemented in various technologies as competitive solutions for the given applications, followed by some potential challenges in "Challenges". Finally, we discuss the possible future research directions.
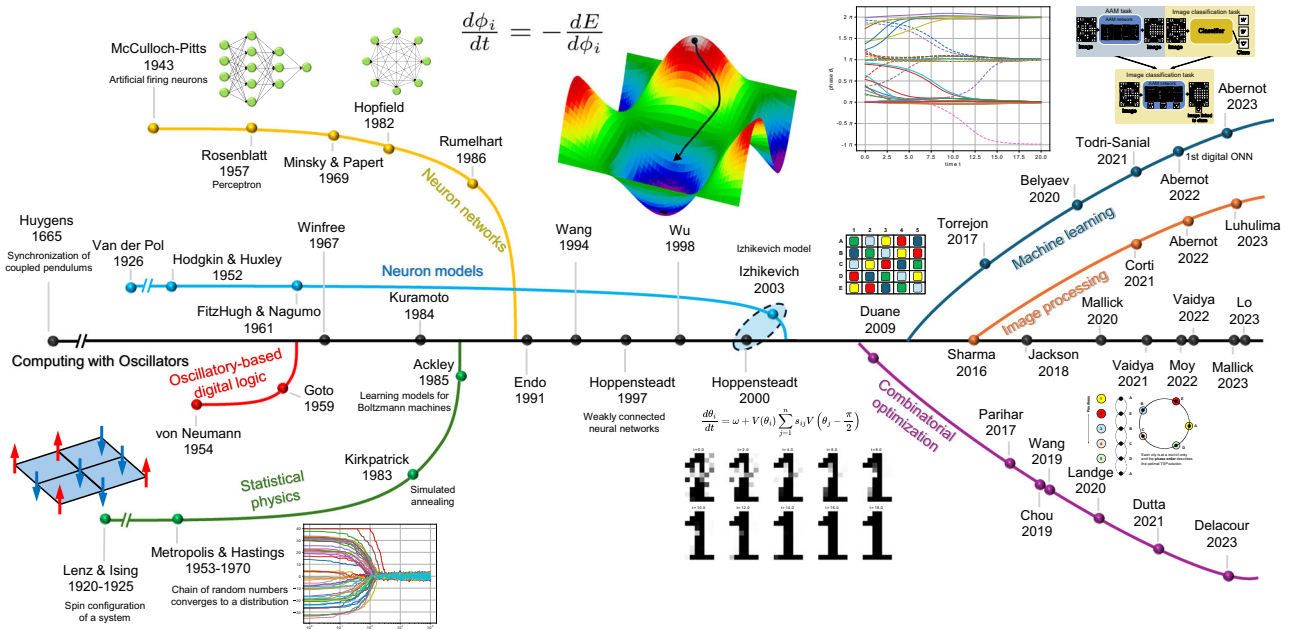
Fig. 2 | Timeline of the ONN computing paradigm. The first oscillatory-based computers were invented by von Neumann and Goto in the 1950s in parallel with the development of the perceptron by Rosenblatt[52]. Labeled parametron, the oscillatory circuit was used as a majority gate to solve Boolean logic, and machines with up to 9600 parametrons were used in the early 1960s. Note that this approach has been recently reconsidered using recent oscillator designs[72,73]. Inspired by neural networks and statistical physics, ONNs emerged in the 1990s as an analog approach to minimize some conceptual energy via gradient descent. For instance, seminal work from Hoppensteadt and Izhikevich[30] proposed to implement energy-based models like Hopfield's[29] in hardware using analog coupled oscillators that naturally converge to attractors, which is useful for pattern recognition. Currently, ONN design is driven by technological progress and is applied to machine learning, image processing, and combinatorial optimization.

Oscillatory neuron

Sustained oscillations can manifest in a wide array of dynamical systems such as mechanical, optical, electrical, biological, or chemical processes[61]. Despite the diverse nature of these oscillations, they are typically mathematically described as dynamical systems

$$\frac{dx}{dt} = f(x) \in \mathbb{R}^m$$

having a limit cycle attractor[83,84], i.e., an orbit $\gamma \subset \mathbb{R}^m$ with period T such that

$$\gamma(t) = \gamma(t + T)$$

For instance, van der Pol derived renowned oscillatory dynamics that correspond to an electrical harmonic oscillator with nonlinear damping[55]. In addition, Hodgkin-Huxley's neuron is recognized for generating oscillations when driven by a sufficient input current[56].

When computing with ONNs, the system dynamics are often reduced to frequency and phase dynamics by considering weak coupling between oscillators and the oscillation amplitude is ignored[30,83]. However, intrinsic parameters such as waveform shape and response to perturbations impact the collective ONN performances and can be tuned for specific applications[82,85,86], just like activation functions in neural networks.

Figure 3 shows various electrical oscillator implementations used in ONNs. One of the simplest, yet promising oscillators for ONN is the Ring Oscillator (RO) which consists of a closed loop of CMOS inverters in series. ROs benefit from advanced CMOS technologies and can be integrated at a very large scale with low energy consumption and high speed (1968 oscillators demonstrated @1 GHz[87]), although it is still unknown whether ROs provide any computational advantages over other types of oscillators.

Another interesting CMOS oscillator is the Schmitt-based relaxation oscillator where a Schmitt trigger alternately charges and discharges a load capacitor through a resistor. This oscillator is generally bulkier than RO but exhibits interesting properties when carefully tuned, such as phase binarization which is particularly useful for solving COPs[86,88].

Moreover, beyond-CMOS devices holding hysteresis behavior such as transition metal oxide devices (TMO)[62,63,89], volatile memristors[90], RRAM[91], and ferroelectric transistors[92] are intensively studied to replace Schmitt triggers and implement more compact relaxation oscillators. A promising TMO is vanadium dioxide (VO_2) which operates at room temperature and can transition from a semiconductor (insulating) to a metallic state by Joule effect[93,94]. When the VO_2 device is biased in its negative differential resistance region, the circuit is unstable and produces sustained oscillations.

Spin-torque nano oscillators (STNO) are promising for ONNs as they operate at radio frequencies from 100 MHz to the GHz range and could be integrated at a very large scale with CMOS electronics[64]. An STNO consists of a magnetic tunnel junction biased by a DC current which induces a magnetization and consequently voltage oscillations across the device[95]. STNOs are very versatile as they react to various magnetic or electrical perturbations, enabling various types of couplings and ranges[96]. In addition, STNOs also possess memory in the oscillation amplitude which can be harnessed for processing time series in real-time[75].

ONN computing

ONN computing is based on the synchronization of coupled oscillators, a phenomenon that naturally appears in biology, physics, or social interactions[97] and was first conceptualized by Winfree in the 1960s[98]. To illustrate oscillator synchronization, consider a crowd of people applauding during an event. The reader probably experienced a situation where the initial state is quite chaotic but after a sufficient time, most of the participants end up applauding in synchrony. From a coupled oscillator network perspective, one can model each individual as an oscillator adjusting its frequency, i.e., increasing or reducing the clapping rate to get synchronized with the neighboring crowd[99]. Such a phenomenon can be captured using the following ODE which expresses the time derivative of oscillators' phases
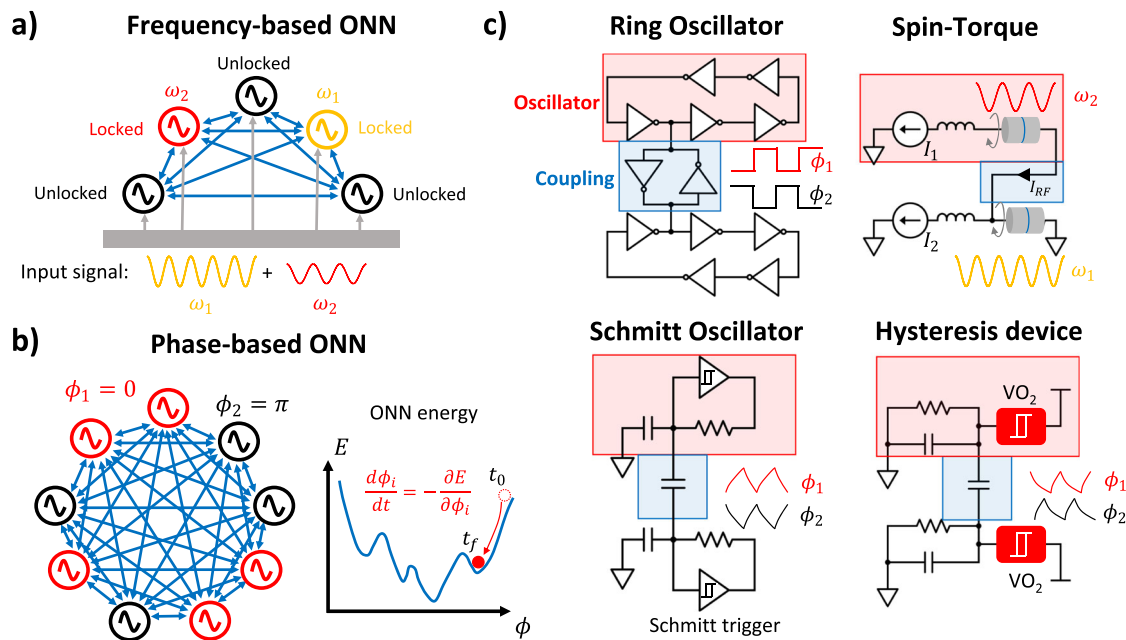
Fig. 3 | Illustration of ONN computing paradigm and its circuit implementations. a Illustration of a frequency-based ONN. The input consists of frequency-dependent signals fed into the ONN. The latter can have oscillators locking to a common frequency and providing the computational result. Labels of locked and unlocked are shown next to the oscillators. **b** Schematic of a phase-based ONN. The oscillators have a uniform frequency and evolve in the phase domain, minimizing a kind of Hopfield energy. **c** Two coupled oscillators implemented with various technologies. Ring oscillators can be coupled by back-to-back inverters[68] or using transmission gates[87]. Relaxation oscillators consist of a hysteresis device that charges/discharges a load capacitor, producing analog oscillations. The hysteresis component can be implemented by a Schmitt trigger[86,201] or beyond-CMOS devices that have a negative differential resistance region in their I–V characteristic. A partial list of potential devices includes VO_2[62,63,93,260], TaO_x and TiO_x oxide[89], NbO_x memristors[90], $PrMnO_3$ RRAM[91], or ferroelectric transistors[92].

(frequencies) ϕ_i generally under the presumption of weak coupling as[30,83]:

$$\frac{d\phi_i}{dt} = \omega_i + \sum_{j=1}^{n} K_{ij} H_{ij}\left(\phi_j - \phi_i\right) \tag{1}$$

with ω_i being the oscillator's free-running frequency, K_{ij} the coupling matrix and $H_{ij}(\chi)$ stands for the interaction between the oscillators and is neuron model dependent[83]. One of the most well-known models is the Kuramoto model ($H_{ij}(\chi) = \sin(\chi)$), where the ODE in Eq. (1) reduces to[61]:

$$\frac{d\phi_i}{dt} = \omega_i + \sum_{j=1}^{n} K_{ij} \sin\left(\phi_j - \phi_i\right) \tag{2}$$

The sinusoidal interaction terms are responsible for the frequency adjustment and model the adaptation of individual i to other clapping agents j in our example. Despite its simple expression, the Kuramoto model produces very complex dynamics depending on the connectivity (K_{ij}) and frequency distributions[100]. The model describes a large variety of rhythmic behaviors[101], including ONN dynamics.

The collective dynamics of coupled oscillators such as those described in Eq. (2) can be harnessed in many different ways to perform different tasks. However, most ONN developments fall into two classes depending on the type of input/output encoding:

- **Frequency-based ONN**: inputs are oscillator frequencies and outputs are the synchronization levels between oscillators.
- **Phase-based ONN**: oscillators have the same frequency and input/output are encoded in-phase between oscillators.

In a frequency-based ONN (Fig. 3a), frequency-dependent input signals are injected into the ONN that reacts to the input perturbations. The computation outcome consists of groups of oscillators that lock in frequency, i.e., are synchronized. This computation scheme has been used for image processing[77,78,102,103], associative memory tasks[79,104], or

spoken vowel classification[76,105]. In the latter application, input vowels are reduced to two frequencies f_A and f_B called "formants" injected into a 4-node ONN that assigns each input sample to a vowel by reading the synchronization state of the oscillators with inputs f_A and f_B. Interestingly, computing in the frequency domain can also skip the physical connections between oscillators. For that purpose, Hoppenstead and Izhikevich proposed an *Oscillatory Neurocomputer*[106] where a time-dependent injected signal emulates all-to-all connectivity by modulating the oscillator phases, and the output consists of the averaged phases after ONN convergence. Despite the advantageous physical scaling of the proposed ONN, generating the modulation signal which includes all pairwise oscillator interactions is not straightforward[107,108].

In the special case where oscillator frequencies are identical and coupling elements are symmetric ($K_{ij} = K_{ji}$), Hoppenstead and Izhikevich[30] have shown that the ONN is a gradient system that minimizes an energy function through time as:

$$\begin{cases} \frac{d\phi_i}{dt} = -\frac{\partial E}{\partial \phi_i} \\ E = -\frac{1}{2} \sum_i \sum_j K_{ij} \cos\left(\phi_i - \phi_j\right) \end{cases} \tag{3}$$

In this regime, the inputs and outputs are encoded in-phase between oscillators whose dynamics depend on the ONN energy and phase initialization. It turns out that the ONN energy is similar to Hopfield's energy[29] and to the Ising Hamiltonian[82], enabling a broad range of applications from machine learning and image processing to combinatorial optimization[17].

The memory of a system described in Eq. (1) is defined as its number of stable phase states[39] and is neuron model dependent. One can derive the memory of a system based on a specific neuron model by finding its phase transition function (PTF), which captures the final phase state transitions of a system composed of two oscillatory neurons given an initial phase delay and a coupling weight. More specifically, for a set of coupling weights (generally $\in [-1, 1]$) and an oscillator with initial delay of $\Delta\phi_{init} \in [0, \frac{T}{2}]$,

one can solve Eq. (1) and obtain the final phase differences between oscillators $\Delta\phi_{final}$. The weights represent the coupling strengths between oscillators. Similarly, as in Hopfield neural networks, the weights are generally in range $[-1, +1]$ to ensure weak coupling and due to the binary nature of neuron states and calculated via the Hebbian learning rule. Weights calculated via the Hebbian learning rule generally are within $[-1, +1]$ and shape the energy landscape to stable states corresponding to stored patterns[29]. In Fig. 4, we plot the phase state transitions of two coupled oscillators, representing the memory of the simplest ONN architecture. We show that depending on the coupling strength between two oscillators and initial phase delay (up to 50% of period) in one of the oscillators, the phase state transition can be obtained. The phase state transitions also represent the stored memory states of the system. Figure 4 shows the corresponding phase transition function (PTF) for the Kuramoto model simulated in software.

Clearly, the Kuramoto model has two stable phase states (0° and 180°), i.e., it has a bistate memory. The search is still on for neuron models with a larger memory. In order to obtain the PTF at hand in hardware, one would need to know the corresponding coupling resistances. Authors in ref. 39
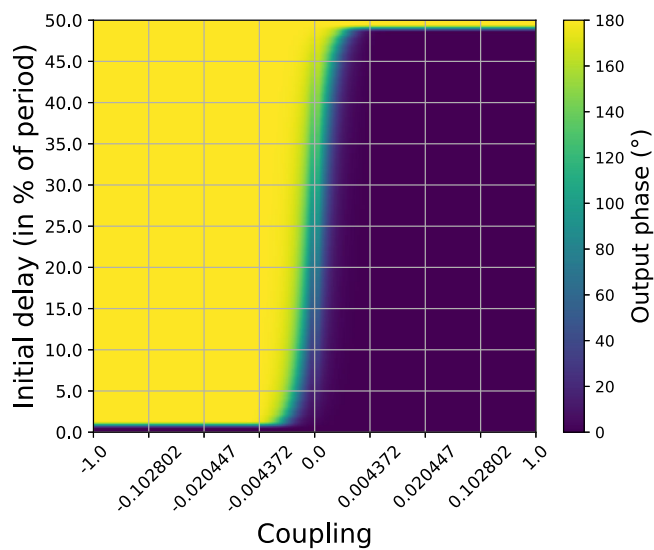
have developed a formalism to map weights computed by the Hebbian learning rule (software) to coupling resistances (hardware).

It is important to note that not all coupled oscillator networks can be fully understood or modeled by the described Kuramoto formalism. While the Kuramoto formalism is a very powerful, it also has its limitations, and it is applicable to only specific types of networks. Kuramoto formalism works particularly well for all-to-all and weakly coupled sinusoidal oscillators with information encoded in-phase while amplitude remains constant. Such formalism does not work well for networks with more complex topologies (i.e., sparce, hierarchical), oscillators with varying amplitude (i.e., both amplitude and phase), non-phase oscillators (i.e., chemical and biological), non-sinusoidal oscillators, time-dependent delay interactions (i.e., non-linear coupling) and high-dimensional interactions between oscillators. For example, memory states of the coupled oscillators are influenced by the time delay in the coupling between oscillators. Delay-coupled Kuramoto oscillators extend the classic Kuramoto model by introducing time-delay interactions. This time delay introduces a greater complexity in the system and also leading to a richer variety of synchronized states and behaviors.

ONN architectures and demonstrators

From the theory of computing with coupled oscillators, researchers investigated how to build ONN demonstrators for meaningful applications, starting with Wang and Terman in 1994[109] who introduced the LEGION ONN array for image segmentation, just before Hoppensteadt and Izhikevich[30,54] linked the theory of coupled oscillators to energy-based Hopfield Neural Networks[29] for associative memory applications. Another promising area proposes to build oscillatory Ising machines (OIM)[88] to solve combinatorial optimization problems (COPs). Finally, research has recently focused on ONNs as ML accelerators such as for convolutional neural networks (CNN), or providing nonlinearity in reservoir computing. Figure 5 summarizes the three classes of problems ONN can solve.

Fully coupled oscillators for associative memory

Hopfield's work[29,110] propelled novel energy-based models of neural networks using analog phase dynamics such as phasor neural networks[111] and ONNs[54,80]. These typically single-layered, fully connected architectures allow recurrent signal propagation with oscillating neurons capable of performing auto-associative memory tasks or pattern recognition. Although Hoppensteadt presented the first hardware solution to implement oscillatory Hopfield networks using phase-locked loops (PLLs)[30], many challenges were still limiting the large-scale implementation of such networks. In 2011, the first implemented ONN with 8 analog van der Pol oscillators performing phase-based pattern
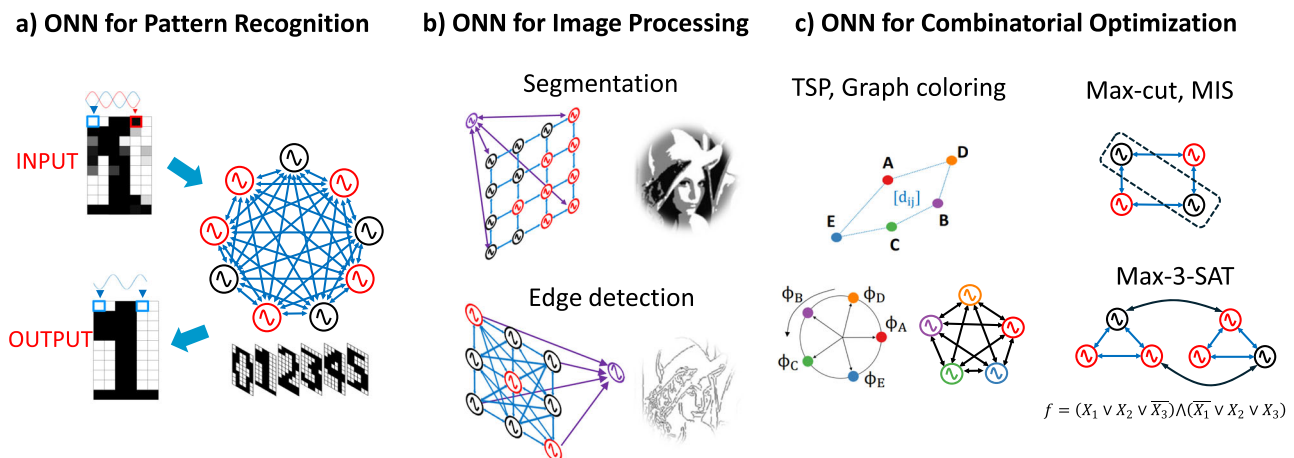

Fig. 4 | Phase transition function of the Kuramoto model as a function of coupling strengths between two oscillators and initial delay. The coupling strengths represent the weights.

Fig. 5 | Example of problems ONN can solve. a Similar to HNNs, ONNs have been extensively studied for pattern recognition[30,67,114–116]. **b** ONN synchronization can be harnessed in various ways to solve image processing problems such as segmentation[103,109,132] or edge detection[71,90,92,123]. **c** ONN phase ordering can also encode variable permutation and is useful to solve hard combinatorial optimization problems such as graph coloring[81,260–262] or TSP[163–165]. Focusing on binary phases, one can encode COPs to ONNs and solve graph partitioning[63,69,82,86] or Boolean Satisfiability problems[167–169].

Table 1 | Comparison of ANNs with ONNs for various problems

Problem	ANNs	Processing unit	ONNs
Pattern recognition	Developed by Wu and He[194]	GPU (GeForce GTX1080)	Developed by Zhang et al.[118], Biswas et al.[119], and Yun et al.[120]
Classification	Developed by Byerly et al.[195] (99.83% on MNIST)	GPU (GeForce GTX1080 Tis)	Developed by Abernot and Todri-Sanial[121] (~62% on MNIST), ClassONN[122] (~70% on MNIST), 8 × 8 MNIST with EP[179] (94.1%)
Edge detection	DexiNed[196]	GPU (Titan XP)	Developed by Kim et al.[123] and Abernot et al.[124]
Online learning	Developed by Nose et al.[197]	Raspberry Pi 3 Model B+	Developed by Abernot et al.[125] and Luhulima et al.[126]
Sensory processing	Developed by Jain et al.[198]	GPU (K40)	Developed by Yang et al.[127]

recognition was reported[107]. Later, building fully connected ONNs with spin-torque oscillators was suggested to solve pattern recognition with frequency-based computing[95].

However, oscillatory Hopfield networks raised more interest recently with novel hardware implementations. Jackson[112] and Shi[113] proposed to build large-scale oscillatory Hopfield networks using PLLs as neurons, and RRAM as synapses. Then, Jackson also showcased a mixed-signal PLL-free[67] ONN with 100 fully connected recurrent neurons, and lately, a fully digital implementation on FPGA was proposed, allowing up to 120 fully connected neurons[114]. In addition, the development of novel compact and low-power devices for neuromorphic computing offered novel solutions for efficient phase-computing oscillatory Hopfield networks[115–117].

Table 1 summarizes the various tasks, which can be solved using the auto-associative memory (AAM) capabilities of ONNs, comparing the coupled oscillators approaches to the most relevant algorithms based on artificial neural networks (ANNs). These problems include but are not limited to (multi-state) pattern recognition[118–120], pattern classification[121,122], edge detection[123,124], on-chip learning[125,126], and sensory processing[127].

Coupled oscillators array for image segmentation
Locally excitatory, globally inhibitory network (LEGION) was introduced in 1994 as a solution to perform image segmentation using an array topology of locally excitatory oscillators computing in-phase[109,128–130] with an additional global inhibitory neuron. The first analog implementation was proposed in 1999[131] and further improved with a neuromorphic analog image segmentation system in 2006[132]. Meanwhile, other works focused on adapted LEGION architectures with digital implementations[133–135]. Lately, LEGION motivated the development of phase-based and frequency-based ONNs for clustering and vision tasks[102,136–138] using oscillators' array topology.

Alternative architectures for AI edge applications
In the last decades, researchers also proposed alternative topologies to efficiently solve various AI tasks. For example, the star coupling topology was introduced to perform phase-based static or dynamic pattern recognition[84,139]. The star coupling was then derived in frequency-based computing to also perform image processing, like pattern recognition[79], image segmentation[103] and convolution operations[77]. Alternatively, convolution operations can also be solved using layered networks of oscillatory neurons[62,140]. The layered topology, often used in artificial neural networks shows interest to solve classification tasks using oscillatory neurons[64,74,105,141,142]. Dutta showcased another topology with ring architecture based on oscillatory neurons to emulate locomotion[143]. Finally, recently, the random and sparse topology of reservoir computing, whose neurons are typically created by time multiplexing[144–146], combined with the high nonlinearity of coupled oscillators highlighted low-power and low-density properties to solve complex AI tasks[75,147–153]. For example, reservoir computing is also a type of complex dynamical system which transforms the input into high-dimensional representation. By time multiplexing the system can simulate the activity of multiple neurons over time allowing for reduction in hardware or computational resources. The common link between ONNs and reservoir computing is on the concepts of nonlinear and temporal dynamics for processing complex information. They are both nonlinear dynamical

systems such as in ONNs, the collective dynamics and temporal sequences (past states) exhibits various nonlinear behaviors (phases) such as synchronization, multistability and chaos, whereas in reservoir computing, time multiplexing creates a sequence of states (temporal response) and helps to exploit nonlinearity to project input data into a high-dimensional space. Such systems are very attractive for processing and analyzing complex information and time-dependent data[154,155]. Tables 2 and 3 provide a list of notable ONN architectures and demonstrators along with their different specifications and application use cases.

Coupled oscillator graph for combinatorial optimization
Being based on the Ising model, ONNs posses a major potential in regards to solving hard combinatorial problems, including but not limited to NP-hard problems[17]. Informally, a problem is NP-hard, if its runtime explodes as the problem size increases. Graph partitioning and route planning are just some of these problems[156]. The author in[157] has shown the corresponding formulation of the Ising Hamiltonian for each NP-hard problem. Table 4 presents the (to the best of our knowledge) state-of-the-art exact and approximate algorithms for all 21 Karp's problems. In addition, the last column of that table presents attempts at solving the corresponding problem using ONNs.

Up until now, there have been a few attempts at solving NP-hard problems using a network of coupled oscillators. Due to its simple mapping to hardware, the Max-Cut problem has arguably attracted the most attention[158–161]. In addition, thanks to its wide range of applicability, the traveling salesman problem has been in the spotlight as well, as the authors in refs. 162–164 have explored three unique ways to solve it. The last one has been applied by the authors in ref. 165. Further notable instances of applying ONNs to NP-hard problems include but are not limited to graph coloring[81,158,166,167] and 3-SAT[167–169].

Challenges
Training algorithms
So far, ONNs have been trained using both existing unsupervised and supervised learning rules. The most commonly applied unsupervised learning rules include Hebbian, Storkey, Diederich-Opper I and II[170–172]. In addition, equilibrium propagation (EP) presents a promising supervised alternative for energy-based models[173]. In particular, authors in refs. 121,174 apply it for classification tasks. Furthermore, in classification tasks of the MNIST dataset[175] using the Diederich-Opper II learning rule, the best-reported results hover around 70% accuracy[122]. Still, this is far away from the accuracies typical for the state-of-the-art convolutional neural networks (CNNs)[176–178]. However, authors in[179] have recently trained ONNs with EP to classify reduced 8x8-pixel MNIST images with 94.1% accuracy. Given its local update rule, EP constitutes a promising method for future ONN training in hardware.

While EP is a supervised alternative for energy-based systems, there are only few supervised ONN-specific learning rules. To the best of our knowledge, there have been two instances of learning rules designed for ONNs up until now. First, the authors in ref. 180 have proposed a Hebbian rule adapted to oscillators called oscillatory Hebbian rule (OHR). Furthermore, an iterative random partial update symmetric Hebbian (IRPUSH)

Table 2 | List of notable ONN architectures and demonstrators using different types of oscillator circuit, network size and coupling weights

Year	References	Oscillator	Network size	Network type	Coupling weights
1959	60	Analog LC	9600	Nearest neighbor	Transformers
2000	30	Phase-lock loop	60	All-to-all	Direct PLL connections
2015	112	Voltage controlled oscillator	20	All-to-all	Resistors
2016	113	Phase-lock loop (PLL) and PPL-free	20	All-to-all	Resistors
2018	67	Digital	100	All-to-all	Resistors
2018	105	Spin-torque nano oscillator	4	All-to-all	Electrical microwave loop
2019	199	Analog LC	4	All-to-all	Resistors
2019	200	Analog LC	240	Tile coupling	Resistors
2021	201	Schmitt-trigger oscillators	30	All-to-all	Capacitors
2021	63	Analog relaxation	8	All-to-all	Capacitors, resistors
2021	68	Ring oscillators	560	Hexagonal	Inverter
2021	202	Schmitt-trigger oscillators	600	Intra-tile coupling	Capacitors
2021	114	Phase control digital oscillator	60	All-to-all	Registers
2021	62	Analog relaxation	4	All-to-all	Resistors
2022	87	Ring oscillators	1968	King's graph	Transmission gates
2022	69	Analog differential	400	King's graph	Current sources with DACs
2023	86	Analog relaxation	16	All-to-all	Capacitors
2023	203	Ring oscillators	48	All-to-all	Transmission gates
2023	126	Phase control digital oscillator	90	All-to-all	Registers
2023	123	Analog relaxation	9	All-to-all	Resistors
2024	204	Analog relaxation	9	All-to-all	Capacitors
2024	127	Analog relaxation	3	All-to-all	Capacitors
2024	120	Bistable resistor (Biristor)	4	Nearest neighbor	Capacitors

Table 3 | Complementary list of notable ONN architectures and demonstrators (as in Table 2) with respect to weight precision, application and technology choice for implementation

Year	References	Weight precision	Application	Technology choice
1959	60	1 bit, unsigned	Digital logic	Transformers
2000	30	1 bit, signed	Pattern recognition	Off the shelf PLLs by national semiconductor
2015	112	2-bit, signed	Pattern recognition	TMO i.e., TaO_x
2018	113	5 bit, signed	Pattern recognition	CMOS 28 nm
2018	67	5 bit, signed	Pattern recognition	Digital
2018	105	1 bit, unsigned	Vowel recognition	Magnetic tunnel junction
2019	199	5 bit, signed	COP	Discrete component, breadboard
2019	200	8 bit, signed	COP	Discrete components, printed circuit boards
2021	201	1 bit, unsigned	COP	CMOS 65 nm
2021	63	1 bit, unsigned	COP	VO_2 phase transition nano oscillators
2021	68	1 bit, unsigned	COP	CMOS 65 nm
2021	202	1 bit, unsigned	COP	CMOS 65 nm
2021	114	4 bit, signed	Pattern recognition	Digital, FPGA ZynQ board
2021	62	1 bit, unsigned	Pattern recognition, CNN filter	VO_2
2022	87	5 levels, signed	COP	CMOS 65 nm
2022	69	6 bit, signed	COP	CMOS 28 nm
2023	86	5 bit, signed	COP, image processing	CMOS 65 nm
2023	203	4 bit, signed	COP	CMOS 65 nm
2023	126	4 bit, signed	Pattern recognition	Digital, FPGA PYNQ board
2023	123	1 bit, unsigned	Edge detection	NbO_x material
2024	204	1 bit, unsigned	COP	VO_2
2024	127	1 bit, unsigned	Sensory processing	VO_2
2024	120	1 bit, unsigned	Pattern recognition	n–p diode and p-n diode on SOI

Table 4 | Best classical algorithms for Karp's 21 NP-complete problems[156] and attempts at solving them using ONNs

Problem	Exact algorithm	Approximate/heuristic algorithm	Attempts with ONNs
Satisfiability	Chaff[205], GRASP[206]	GSAT[207], WalkSAT[208]	Attempted by Bashar et al.[209] and Bybee et al.[210]
Binary integer programming	Cutting plane[211], Branch-and-bound[212]	Tabu search[213]	–
(Maximum) Clique	Developed by Robson[214]	Developed by Feige[215] and Boppana & Halldórsson[216]	Attempted by Mallick et al.[217,218] (both MIS) and Ferrari et al.[219]
Set packing	–	Developed by Halldórsson et al.[220]	–
Vertex cover	Developed by Chen et al.[221] (small vertex cover)	Developed by Papadimitriou & Steiglitz[222] and Karakostas[223]	Attempted by Al Beattie & Ochs[224] and Al Beattie et al.[225] and Mallick et al.[217] (MIS)
Set covering	–	Developed by Chvatal[226]	–
Feedback node set	Developed by Fomin and Villanger[227]	Developed by Becker & Geiger[228]	–
Feedback arc set	Developed by Lawler[229]	Developed by Even et al.[230]	–
(Un)directed Hamiltonian cycle	Developed by Bjorklund[231]	–	Attempted by Mallick et al.[218]
Traveling salesman problem	Concorde TSP solver[232]	Christofides algorithm[233]	Attempted by Duane[163], Landge et al.[164], and Mallick et al.[218]
3-SAT	Developed by Schoning[234]	–	Attempted by Maher et al.[167], Cílasun et al.[168], and Bashar et al.[169]
Graph coloring problem	Developed by Björklund et al.[235]	Developed by Halldórsson[236]	Attempted by Wu[81], Wang & Roychowdhury[158], and Mallick et al.[166]
Clique cover	Developed by Espelage et al.[237] (bounded clique-width)	Developed by Cerioli et al.[238] (3 maximum vertex degree) and Blanchette et al.[239] (planar graphs)	Attempted by Yamada & Inaba[240]
Exact cover	DLX[241]	–	–
Hitting set	–	Developed by Chvatal[226]	–
Steiner tree	Developed by Wang[242]	Developed by Wu et al.[243], Byrka et al.[244], and Chen and Hsieh[245]	–
3-dimensional matching	–	Developed by Cygan[246]	–
Knapsack	Dynamic programming in-advance algorithm[247], Meet-in-the-middle[248]	Developed by Dantzig[249], FPTAS[250]	Attempted by Watanabe et al.[251]
Job sequencing	–	Developed by Chen et al.[252] (online) and Hochbaum & Shmoys[253] (offline)	–
(multiway number) Partition	Pseudopolynomial time number partitioning[254], complete greedy algorithm[255], Complete Karmarkar-Karp algorithm[256]	Multifit algorithm[257]	Attempted by Nigg et al.[258]
Max-Cut	–	Goemans–Williamson algorithm[259]	Attempted by Wang et al.[158], Bashar et al.[159], Steinberger[160], and Bashar et al.[161]

Empty fields remark the absence of exact/approximate algorithms and/or ONN attempts to solve the problem.

rule has been developed by authors in ref. 181. Both proposed learning rules which expand on the classic Hebbian rule. However, there are other more powerful learning rules that have still not been redesigned for the ONNs. Hence, the development of training algorithms specific to ONNs will aid in exploring their potential advantages across various AI tasks.

Scalability

Scalability remains a significant concern in the physical design and implementation of ONNs for executing practical use cases. While each technology choice presents its own challenges, the primary limitation stems from the polynomial increase in coupling elements with the number of oscillators. As the size of the ONN network grows, implementing a fully connected network becomes challenging, primarily due to unwanted parasitics in the connections. For instance, advanced CMOS technology offers a viable platform for the design and implementation of large-scale ONNs. However, the current ONN architecture imposes certain symmetry constraints on the connectivity between oscillators to enable all-to-all coupling. Moreover, as the ONN network size increases, measuring and distinguishing the phase difference between oscillators become increasingly difficult. One potential approach being explored to scale up ONNs is to adopt a modular design, where a larger ONN is constructed from many smaller ONNs[69,182].

Benchmarking and comparison

Presently, much of the literature benchmarks ONNs primarily on MNIST classification for pattern retrieval and combinatorial problems like TSP and Max-Cut[121,122,163,164,183]. Although some heuristics have been developed for solving combinatorial problems, such as the traveling salesman problem (TSP)[157,163,164], and shared resources like the TSPLib library[184] provide sample instances for benchmarking and comparison against the best-reported solutions. We firmly believe that the ONN community will find it relevant to consolidate and develop open-source benchmarks and libraries for evaluating different problems, enabling community contributions and sharing. As the community expands, setting up benchmarking contests on various problem sets will become increasingly important. These contests will aid in advancing the scientific development in ONN computing and in understanding where ONN computing can offer a competitive advantage in comparison to the classical computing.

Variety of problems that ONN can solve

Considering the ONN's intrinsic feature as a dynamical system that exhibits rich dynamics from synchronization to non-stable states, studying a wider range of mathematical problems, that can exploit ONN dynamics, remains

relevant. While intrinsically meant for pattern retrieval[29,30], other problems can be investigated with ONNs such as:

- Artificial intelligence and edge AI—various machine learning algorithms running on the edge, such as those used in human-machine interfaces, demand real-time decision-making, reinforcement learning, and incremental learning. Exploring and comprehending how ONNs can empower such algorithms, while capitalizing on their parallel processing and low-resource utilization[71], could prove valuable for robotic applications and edge devices that operate with limited data for learning and decision-making.
- Secure communication—it is observed that depending on the coupling, an oscillator can exhibit chaotic behavior[185]. This characteristic can be utilized in ONNs to implement random number generators and secure communication protocols[186]. Furthermore, ONN's sensitivity to initial conditions and any perturbations can alter its stability, which can be intentionally exploited for secure communication.
- Complex computation problems in engineering—ONN dynamics are described through differential equations that represent their behavior[30]. There is interest in further adapting these differential equations to model various physical phenomena such as fluid flow, atom interactions (like Hubbard model[187]), heat transfer, motion and quantum-mechanical systems (such as wave functions).
- Complex networks—ONN's dynamics can also be exploited for modeling large, complex networks, such as decision-making processes in economics and finance, as well as social networks. An analysis of ONN's dynamics will allow to obtain statistical estimation and to perform data analysis to predict the behavior of time-varying data.

Perspectives
ONNs draw inspiration from oscillatory neural activity in biology and exhibit a strong connection with physics, enabling their application to various problems such as optimization[45,157,158]. However, there remains a significant potential for fruitful interactions between physics, machine learning, and ONNs, which could yield new insights and hardware solutions for implementing deep learning. For instance, employing the Ising formalism could enable massive parallel processing[188]. Ultimately, one can envision an ONN IP core coexisting alongside other accelerators and image sensors, facilitating accelerated, high-accuracy, and fast performance for specific machine learning algorithms on edge devices with limited power and computational resources. Here, we detail our vision on the perspectives of future research on computing with oscillators.

Synergy between ONN and neuroscience
The oscillatory neural dynamics observed in biological neural networks bear many similarities to ONN dynamics and offer valuable insights into learning and plasticity[189]. For instance, the synchronization of neural clusters and phase dynamics could provide insights into how oscillations propagate through large networks in distinct patterns, influencing memory creation and consolidation. Understanding how oscillatory propagation, such as spiral or in-plane dynamics, affects the stability of ONN networks and their memory states is crucial. Studies using electrophysiological recordings in insects and humans reveal that various conditions, including noise, non-uniformities (no two neurons are identical) and oscillation frequencies, impact short- and long-term memory formation and performance on cognitive tasks. Exploring whether insights from neuroscience can lead to new learning rules for ONNs remains a path worth investigating.

Synergy between ONN and physics
The synchronization phenomenon is extensively studied across various domains, with a substantial body of work in mathematics and physics delving into its fundamental nature[97]. The Kuramoto model stands out as one of the most studied models due to its simplicity and ability to exhibit rich dynamics in coupled complex networks[61]. From physics, the understandings of synchronization and the emergence of clusters in synchronization and chimera states can shed light not only on the stability of ONN

networks but also on how factors such as noise, variability, nonidealities, and synaptic schemes (from fully to sparsely coupled) impact synchronization and cluster appearance, ultimately influencing the network's memory and memory capacity[189,190]. There are numerous unexplored synergies stemming from the non-trivial physics of complex networks that could further enhance our understanding of ONN dynamics and their computing capabilities, benefiting from established rigorous mathematical formalisms.

Synergy between ONN and machine learning
Currently, classification, one of the most commonly used machine learning algorithms, is predominantly performed using convolutional neural networks (CNNs)[176–178], which typically involve several layers and thousands to millions of neuron nodes. In addition, they are normally trained using backpropagation which makes these neural networks even more energy inefficient[191]. Thus far, classification using ONNs has been accomplished with a perceptron-type network and a limited number of neurons corresponding to the size of the image pixels[121,183]. Although the accuracy of ONNs is not yet sufficiently high for practical use (~70%)[122], they demonstrate the simplicity and powerful computational capabilities of ONNs by performing classification with fewer neurons and a single fully connected layer. Recent studies have demonstrated the utility of ONNs by employing them as the first layer of CNNs, functioning as a filter layer for parallel feature extraction[62,124]. Furthermore, there is a potential to use ONNs for training of deep Boltzmann machines[192,193]. There should be additional synergies and opportunities for exploiting ONNs in machine learning that remain to be explored, and these will likely become apparent with further interdisciplinary research efforts.

Technology choice for ONN implementation
The physical implementation of ONNs remains challenging, despite numerous efforts to design them using advanced CMOS nodes, mature CMOS nodes, novel materials and devices, optically, spintronics, among others. Designing ONNs that are practical, useful, and capable of exploiting their phase dynamics necessitates a detailed analysis of the benefits and costs of each technology choice. While exploring multiple technology paths is constructive for further maturing ONN computing, envisioning ONNs as part of a larger system, where they can serve as either accelerator IP cores or be embedded in many ONN IP cores as massive parallel processors, is essential. To realize practical value from ONN computing, we believe that investing effort in CMOS-based ONN fabrication is worthy, enabling ONNs to be integrated into large system-on-chip designs. While exploring novel materials and devices represents a longer-term route to ONN implementation, valuable insights can be obtained from the silicon platform for ONN computing.

Conclusion
In this review, we explore research avenues that have shaped and marked the most promising frontiers in computing with oscillatory neural networks. Initially, considered for their associative memory as an intrinsic physical property of complex dynamical systems, the investigation of ONNs for computing has sparked significant interest in emulating quantum analogies using coupled oscillators. Recent years have witnessed the rapid growth of new techniques and fresh ideas for harnessing the rich nonlinear dynamics of ONNs.

From the perspective of diverse scientists across fields such as neuroscience, theoretical physics, computer scientists among others, there is a growing interest in solving practical, everyday intractable problems. This interest has been fueled, in part, by enthusiastic researchers like us in analog, digital and mixed-signal circuit design. They have taken a leap of faith, committing their efforts to expand the current body of knowledge within this research area to engineer such computational models. The goal is to implement complex theoretical concepts into hardware, addressing not only the energy efficiency challenge of computing but also providing alternative computing solutions for intractable problems such as NP-hard combinatorial optimization problems.

This review provides an overview of the current state of computing with ONNs. In addition, it highlights exciting methods and unresolved challenges. The use of coupled oscillators in computing holds promise as a platform to explore fundamental many-body physics, with practical implementations yet to be fully developed. What began as a curiosity-driven endeavor has evolved into an influential exploration of engineered dynamical computing systems that have the potential to unlock scientific discoveries that were previously out of reach.

Data availability
No datasets were generated or analyzed during this study.

References
1. von Neumann, J. First draft of a report on the edvac. *IEEE Ann. Hist. Comput.* **15**, 27–75 (1993).
2. Goralski, M. A. & Tan, T. K. Artificial intelligence and sustainable development. *Int. J. Manag. Educ.* **18**, 100330 (2020).
3. Toews, R. Deep learning's carbon emissions problem. *Forbes* https://www.forbes.com/sites/robtoews/2020/06/17/deep-learnings-climate-change-problem/ (2020).
4. Vincent, J. How much electricity does ai consume? *The Verge* https://www.theverge.com/24066646/ai-electricity-energy-watts-generative-consumption (2024).
5. Lang, R. Künstliche intelligenz entpuppt sich als enormer energiefresser. *Der Standard* https://www.derstandard.at/story/3000000213815/kuenstliche-intelligenz-entpuppt-sich-als-enormer-energiefresser (2024).
6. Wall, C., Hetherington, V. & Godfrey, A. Beyond the clinic: the rise of wearables and smartphones in decentralising healthcare. *NPJ Digit. Med.* **6**, 219 (2023).
7. Yee, L., Chui, M. & Roberts, R. McKinsey Technology Trends Outlook 2024. https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-top-trends-in-tech (2024).
8. PcSite. The computer revolution: how digital technology transformed our world. *PcSite* https://pcsite.medium.com/the-computer-revolution-how-digital-technology-transformed-our-world-fc7b271ae908 (2023).
9. Koulopoulos, T. The end of the digital revolution is coming: here's what's next. *Inc.* https://www.inc.com/thomas-koulopoulos/the-end-of-digital-revolution-is-coming-heres-whats-next.html (2019).
10. Finocchio, G. et al. Roadmap for unconventional computing with nanotechnology. *Nano Futures*. **8**, 012001 (2024).
11. Sebastian, A., Le Gallo, M., Khaddam-Aljameh, R. & Eleftheriou, E. Memory devices and applications for in-memory computing. *Nat. Nanotechnol.* **15**, 529–544 (2020).
12. Zou, X., Xu, S., Chen, X., Yan, L. & Han, Y. Breaking the von Neumann bottleneck: architecture-level processing-in-memory technology. *Sci. China Inf. Sci.* **64**, 160404 (2021).
13. Woodell, E. The hidden trap of gpu-based AI: the von Neumann bottleneck. *Linkedin* https://www.linkedin.com/pulse/hidden-trap-gpu-based-ai-von-neumann-bottleneck-dr-eric-woodell-n9m9e/ (2024).
14. Willard, J., Jia, X., Xu, S., Steinbach, M. & Kumar, V. Integrating Scientific Knowledge with Machine Learning for Engineering and Environmental Systems. ACM Comput. Surv. **55**, 4 (2022).
15. Fischer, E. New physics-based self-learning machines could replace current artificial neural networks and save energy. *TechExplore* https://techxplore.com/news/2023-09-physics-based-self-learning-machines-current-artificial.html (2023).
16. Willcox, K. E., Ghattas, O. & Heimbach, P. The imperative of physics-based modeling and inverse theory in computational science. *Nat. Comput. Sci.* **1**, 166–168 (2021).
17. Csaba, G. & Porod, W. Coupled oscillators for computing: a review and perspective. *Appl. Phys. Rev.* **7**, 1–19 (2020).
18. Jeon, I. & Kim, T. Distinctive properties of biological neural networks and recent advances in bottom-up approaches toward a better biologically plausible neural network. *Front. Comput. Neurosci.* **17**, 1092185 (2023).
19. Stiefel, K. M. & Ermentrout, B. Neurons as oscillators. *J. Neurophysiol.* **116**, 2950–2960 (2016).
20. Ward, L. M. Synchronous neural oscillations and cognitive processes. *Trends Cogn. Sci.* **7**, 553–559 (2003).
21. Tavano, A., Rimmele, J. M., Michalareas, G. & Poeppel, D. Neural oscillations in eeg and meg. In *Neuromethods* Vol. 202. 241–284 (Springer, 2023).
22. Yeung, N., Bogacz, R., Holroyd, C. B. & Cohen, J. D. Detection of synchronized oscillations in the electroencephalogram: an evaluation of methods. *Psychophysiology* **41**, 822–832 (2004).
23. Taylor, D., Ott, E. & Restrepo, J. G. Spontaneous synchronization of coupled oscillator systems with frequency adaptation. *Phys. Rev. E-Stat. Nonlinear Soft Matter Phys.* **81**, 046214 (2010).
24. Crook, S. M., Ermentrout, B. & Bower, J. M. Spike frequency adaptation affects the synchronization properties of networks of cortical oscillators. *Neural Comput.* **10**, 837–854 (1998).
25. Shmuel, A. & Leopold, D. A. Neuronal correlates of spontaneous fluctuations in fMRI signals in monkey visual cortex: implications for functional connectivity at rest. *Hum. Brain Mapp.* **29**, 751–761 (2008).
26. Schölvinck, M. L., Maier, A., Ye, F. Q., Duyn, J. H. & Leopold, D. A. Neural basis of global resting-state fmri activity. *Proc. Natl. Acad. Sci. USA* **107**, 10238–10243 (2010).
27. Shmuel, A. On the relationship between functional MRI signals and neuronal activity. In *Casting Light on the Dark Side of Brain Imaging*, 49–53 (Academic Press, 2019). https://doi.org/10.1016/B978-0-12-816179-1.00007-4, https://www.sciencedirect.com/science/article/pii/B9780128161791000074.
28. Maass, W. Networks of spiking neurons: the third generation of neural network models. *Neural Netw.* **10**, 1659–1671 (1997).
29. Hopfield, J. J. Neural networks and physical systems with emergent collective computational abilities. *Proc. Natl. Acad. Sci. USA* **79**, 2554–2558 (1982).
30. Hoppensteadt, F. C. & Izhikevich, E. M. Pattern recognition via synchronization in phase-locked loop neural networks. *IEEE Trans. Neural Netw.* **11**, 734–738 (2000).
31. Gerstner, W., Kistler, W. M., Naud, R. & Paninski, L. *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition* (Cambridge University Press, 2014).
32. Jacobs, J., Kahana, M. J., Ekstrom, A. D. & Fried, I. Brain oscillations control timing of single-neuron activity in humans. *J. Neurosci.* **27**, 3839–3844 (2007).
33. Gerstner, W. Population dynamics of spiking neurons: fast transients, asynchronous states, and locking. *Neural Comput.* **12**, 43–89 (2000).
34. Contreras, D., Destexhe, A., Sejnowski, T. J. & Steriade, M. Spatiotemporal patterns of spindle oscillations in cortex and thalamus. *J. Neurosci.* **17**, 1179–1196 (1997).
35. Borisyuk, R. & Hoppensteadt, F. Oscillatory models of the hippocampus: a study of spatio-temporal patterns of neural activity. *Biol. Cybern.* **81**, 359–371 (1999).
36. Kahana, M. J. The cognitive correlates of human brain oscillations. *J. Neurosci.* **26**, 1669–1672 (2006).
37. Schmidt, H., Avitabile, D., Montbrió, E. & Roxin, A. Network mechanisms underlying the role of oscillations in cognitive tasks. *PLoS Comput. Biol.* **14**, e1006430 (2018).
38. Izhikevich, E. M. Simple model of spiking neurons. *IEEE Trans. Neural Netw.* **14**, 1569–1572 (2003).
39. Delacour, C. & Todri-Sanial, A. Mapping Hebbian learning rules to coupling resistances for oscillatory neural networks. *Front. Neurosci.* **15**, 694549 (2021).

40. Nakagawa, N. & Kuramoto, Y. Collective chaos in a population of globally coupled oscillators. *Prog. Theor. Phys.* **89**, 313–323 (1993).

41. Ochab, J. & Góra, P. F. Synchronisation of Coupled Oscillators in a Local One-dimensional Kuramoto model. *Acta Phys. Pol. B Proc.* **3**, 453–462, (2010).

42. Blekhman, I. I. Synchronization in science and technology. *Moscow Izdatel Nauka*, 255-255, (ASME press,1988).

43. Strogatz, S. H. Spontaneous synchronization in nature. In *Proceedings of International Frequency Control Symposium*, 2–4 (IEEE, 1997).

44. Huygens, C. Oeuvres complètes de Christiaan Huygens. *Publiées par la Société hollandaise dessciences*, (La Haye, M. Nijhoff, 1888).

45. Ising, E. *Beitrag zur theorie des ferro-und paramagnetismus*. Ph.D. thesis, Grefe & Tiedemann Hamburg, Germany (1924).

46. Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. & Teller, E. Equation of state calculations by fast computing machines. *J. Chem. Phys.* **21**, 1087–1092 (1953).

47. Hastings, W. K. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* **57**, 97–109 (1970).

48. Kirkpatrick, S., Gelatt Jr, D. & Vecchi, M. P. Optimization by simulated annealing. *Science* **220**, 671–680 (1983).

49. Ackley, D. H., Hinton, G. E. & Sejnowski, T. J. A learning algorithm for Boltzmann machines. *Cogn. Sci.* **9**, 147–169 (1985).

50. Sibalija, T. Application of simulated annealing in process optimization: a review. *Simul. Anneal. Introd. Appl. Theory* **1**, 1–14 (2018).

51. McCulloch, W. S. & Pitts, W. A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* **5**, 115–133 (1943).

52. Rosenbaltt, F. *The Perceptron—a Perciving and Recognizing Automation* (Cornell Aeronautical Laboratory, 1957).

53. Minsky, M. & Papert, S. An introduction to computational geometry. *Camb. tiass. HIT* **479**, 104 (1969).

54. Hoppensteadt, F. C. & Izhikevich, E. M. eds. Weakly connected oscillators. In *Weakly Connected Neural Networks*, 247–293 (Springer, 1997).

55. van der Pol, B. Lxxxviii. on "relaxation-oscillations". *Lond. Edinb. Dublin Philos. Mag. J. Sci.* **2**, 978–992 (1926).

56. Hodgkin, A. L. & Huxley, A. F. A quantitative description of membrane current and its application to conduction and excitation in nerve. *J. Physiol.* **117**, 500 (1952).

57. FitzHugh, R. Impulses and physiological states in theoretical models of nerve membrane. *Biophys. J.* **1**, 445–466 (1961).

58. Nagumo, J., Arimoto, S. & Yoshizawa, S. An active pulse transmission line simulating nerve axon. *Proc. IRE* **50**, 2061–2070 (1962).

59. von Neumann, J. Non-linear capacitance or inductance switching, amplifying, and memory organs. US Patent 2,815,488 (1957).

60. Goto, E. The parametron, a digital computing element which utilizes parametric oscillation. *Proc. IRE* **47**, 1304–1316 (1959).

61. Kuramoto, Y. *Chemical Turbulence* (Springer, 1984).

62. Corti, E. et al. Coupled vo2 oscillators circuit as analog first layer filter in convolutional neural networks. *Front. Neurosci.* **15**, 628254 (2021).

63. Dutta, S. et al. An ising Hamiltonian solver based on coupled stochastic phase-transition nano-oscillators. *Nat. Electron.* **4**, 502–512 (2021).

64. Grollier, J. et al. Neuromorphic spintronics. *Nat. Electron.* **3**, 360–370 (2020).

65. Maffezzoni, P., Bahr, B., Zhang, Z. & Daniel, L. Analysis and design of boolean associative memories made of resonant oscillator arrays. *IEEE Trans. Circ. Syst. I: Regul. Pap.* **63**, 1964–1973 (2016).

66. Honjo, T. et al. 100,000-spin coherent ising machine. *Sci. Adv.* **7**, eabh0952 (2021).

67. Jackson, T., Pagliarini, S. & Pileggi, L. An oscillatory neural network with programmable resistive synapses in 28 nm CMOS. In *2018 IEEE International Conference on Rebooting Computing (ICRC)*, 1–7 (IEEE, 2018).

68. Ahmed, I., Chiu, P.-W., Moy, W. & Kim, C. H. A probabilistic compute fabric based on coupled ring oscillators for solving combinatorial optimization problems. *IEEE J. Solid-State Circ.* **56**, 2870–2880 (2021).

69. Graber, M. & Hofmann, K. A versatile & adjustable 400 node CMOS oscillator based ising machine to investigate and optimize the internal computing principle. In *2022 IEEE 35th International System-on-Chip Conference (SOCC)*, 1–6 (IEEE, 2022).

70. Nikonov, D. E. & Young, I. A. Benchmarking delay and energy of neural inference circuits. *IEEE J. Explor. Solid-State Comput. Devices Circ.* **5**, 75–84 (2019).

71. Delacour, C., Carapezzi, S., Abernot, M. & Todri-Sanial, A. Energy-performance assessment of oscillatory neural networks based on vo _2 devices for future edge ai computing. In *IEEE Transactions on Neural Networks and Learning Systems* (IEEE, 2023).

72. Roychowdhury, J. Boolean computation using self-sustaining nonlinear oscillators. *Proc. IEEE* **103**, 1958–1969 (2015).

73. Avedillo, M. J., Quintana, J. M. & Núñez, J. Phase transition device for phase storing. *IEEE Trans. Nanotechnol.* **19**, 107–112 (2020).

74. Vodenicarevic, D., Locatelli, N., Abreu Araujo, F., Grollier, J. & Querlioz, D. A nanotechnology-ready computing scheme based on a weakly coupled oscillator network. *Sci. Rep.* **7**, 44772 (2017).

75. Torrejon, J. et al. Neuromorphic computing with nanoscale spintronic oscillators. *Nature* **547**, 428–431 (2017).

76. Dutta, S. et al. Spoken vowel classification using synchronization of phase transition nano-oscillators. In *2019 Symposium on VLSI Technology*, T128–T129 (IEEE, 2019).

77. Nikonov, D. E. et al. Convolution inference via synchronization of a coupled CMOS oscillator array. *IEEE J. Explor. Solid-State Comput. Devices Circ.* **6**, 170–176 (2020).

78. Shukla, N. et al. Synchronized charge oscillations in correlated electron systems. *Sci. Rep.* **4**, 4964 (2014).

79. Nikonov, D. E. et al. Coupled-oscillator associative memory array operation for pattern recognition. *IEEE J. Explor. Solid-State Comput. Devices Circ.* **1**, 85–93 (2015).

80. Endo, T. & Takeyama, K. Neural network using oscillators. *Electron. Commun. Jpn. (Part III: Fundamental Electron. Sci.)* **75**, 51–59 (1992).

81. Wu, C. W. Graph coloring via synchronization of coupled oscillators. *IEEE Trans. Circ. Syst. I: Fundamental Theory Appl.* **45**, 974–978 (1998).

82. Wang, T., Wu, L., Nobel, P. & Roychowdhury, J. Solving combinatorial optimisation problems using oscillator based Ising machines. *Nat. Comput.* **20**, 287–306 (2021).

83. Izhikevich, E. M. & Kuramoto, Y. Weakly coupled oscillators. *Encycl. Math. Phys.* **5**, 448 (2006).

84. Corinto, F., Bonnin, M. & Gilli, M. Weakly connected oscillatory network models for associative and dynamic memories. *Int. J. Bifurc. Chaos* **17**, 4365–4379 (2007).

85. Andrawis, R. & Roy, K. A new oscillator coupling function for improving the solution of graph coloring problem. *Phys. D: Nonlinear Phenom.* **412**, 132617 (2020).

86. Delacour, C. et al. A mixed-signal oscillatory neural network for scalable analog computations in phase domain. *Neuromorphic Comput. Eng.* **3**, 034004 (2023).

87. Moy, W. et al. A 1,968-node coupled ring oscillator circuit for combinatorial optimization problem solving. *Nat. Electron.* **5**, 310–317 (2022).

88. Vaidya, J., Surya Kanthi, R. & Shukla, N. Creating electronic oscillator-based Ising machines without external injection locking. *Sci. Rep.* **12**, 981 (2022).

89. Sharma, A. A., Bain, J. A. & Weldon, J. A. Phase coupling and control of oxide-based oscillators for neuromorphic computing. *IEEE J. Explor. Solid-State Comput. Devices Circ.* **1**, 58–66 (2015).

90. Weiher, M. et al. Improved vertex coloring with nbox memristor-based oscillatory networks. *IEEE Trans. Circ. Syst. I: Regul. Pap.* **68**, 2082–2095 (2021).

91. Lashkare, S., Kumbhare, P., Saraswat, V. & Ganguly, U. Transient joule heating-based oscillator neuron for neuromorphic computing. *IEEE Electron Device Lett.* **39**, 1437–1440 (2018).

92. Eslahi, H., Hamilton, T. J. & Khandelwal, S. Energy-efficient ferroelectric field-effect transistor-based oscillators for neuromorphic system design. *IEEE J. Explor. Solid-State Comput. Devices Circ.* **6**, 122–129 (2020).

93. Carapezzi, S. et al. Advanced design methods from materials and devices to circuits for brain-inspired oscillatory neural networks for edge computing. *IEEE J. Emerg. Sel. Top. Circ. Syst.* **11**, 586–596 (2021).

94. Carapezzi, S. et al. Role of ambient temperature in modulation of behavior of vanadium dioxide volatile memristors and oscillators for neuromorphic applications. *Sci. Rep.* **12**, 19377 (2022).

95. Csaba, G. & Porod, W. Computational study of spin-torque oscillator interactions for non-boolean computing applications. *IEEE Trans. Magn.* **49**, 4447–4451 (2013).

96. Lebrun, R. et al. Mutual synchronization of spin torque nano-oscillators through a long-range and tunable electrical coupling scheme. *Nat. Commun.* **8**, 15825 (2017).

97. Dörfler, F. & Bullo, F. Synchronization in complex networks of phase oscillators: a survey. *Automatica* **50**, 1539–1564 (2014).

98. Winfree, A. T. Biological rhythms and the behavior of populations of coupled oscillators. *J. Theor. Biol.* **16**, 15–42 (1967).

99. Néda, Z., Ravasz, E., Vicsek, T., Brechet, Y. & Barabási, A.-L. Physics of the rhythmic applause. *Phys. Rev. E* **61**, 6987 (2000).

100. Wu, Y., Zheng, Z., Tang, L. & Xu, C. Synchronization dynamics of phase oscillator populations with generalized heterogeneous coupling. *Chaos Solitons Fractals* **164**, 112680 (2022).

101. Acebrón, J. A., Bonilla, L. L., Pérez Vicente, C. J., Ritort, F. & Spigler, R. The Kuramoto model: a simple paradigm for synchronization phenomena. *Rev. Mod. Phys.* **77**, 137–185 (2005).

102. Sharma, A. A. et al. Low-power, high-performance s-ndr oscillators for stereo (3d) vision using directly-coupled oscillator networks. In *2016 IEEE Symposium on VLSI Technology*, 1–2 (IEEE, 2016).

103. Shukla, N. et al. Ultra low power coupled oscillator arrays for computer vision applications. In *2016 IEEE Symposium on VLSI Technology*, 1–2 (IEEE, 2016).

104. Vassilieva, E., Pinto, G., de Barros, J. & Suppes, P. Learning pattern recognition through quasi-synchronization of phase oscillators. *IEEE Trans. Neural Netw.* **22**, 84–95 (2010).

105. Romera, M. et al. Vowel recognition with four coupled spin-torque nano-oscillators. *Nature* **563**, 230–234 (2018).

106. Hoppensteadt, F. C. & Izhikevich, E. M. Oscillatory neurocomputers with dynamic connectivity. *Phys. Rev. Lett.* **82**, 2983 (1999).

107. Hölzel, R. W. & Krischer, K. Pattern recognition with simple oscillating circuits. *N. J. Phys.* **13**, 073031 (2011).

108. Albertsson, D. I. & Rusu, A. Highly reconfigurable oscillator-based ising machine through quasiperiodic modulation of coupling strength. *Sci. Rep.* **13**, 4005 (2023).

109. Wang, D. & Terman, D. Locally excitatory globally inhibitory oscillator networks: theory and application to pattern segmentation. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, Vol. 2, 945–950 (IEEE, 1994).

110. Hopfield, J. J. Neurons with graded response have collective computational properties like those of two-state neurons. *Proc. Natl. Acad. Sci. USA* **81**, 3088–3092 (1984).

111. Noest, A. J. Associative memory in sparse phasor neural networks. *Europhys. Lett.* **6**, 469 (1988).

112. Jackson, T. C., Sharma, A. A., Bain, J. A., Weldon, J. A. & Pileggi, L. Oscillatory neural networks based on TMO nano-oscillators and multi-level RRAM cells. *IEEE J. Emerg. Sel. Top. Circuits Syst.* **5**, 230–241 (2015).

113. Shi, R., Jackson, T. C., Swenson, B., Kar, S. & Pileggi, L. On the design of phase locked loop oscillatory neural networks: mitigation of transmission delay effects. In *2016 International Joint Conference on Neural Networks (IJCNN)*, 2039–2046 (IEEE, 2016).

114. Abernot, M. et al. Digital implementation of oscillatory neural network for image recognition applications. *Front. Neurosci.* **15**, 713054 (2021).

115. Corti, E. et al. Resistive coupled vo 2 oscillators for image recognition. In *2018 IEEE International Conference on Rebooting Computing (ICRC),* 1–7 (IEEE, 2018).

116. Todri-Sanial, A. et al. How frequency injection locking can train oscillatory neural networks to compute in phase. *IEEE Trans. Neural Netw. Learn. Syst.* **33**, 1996–2009 (2021).

117. Núñez, J. et al. Oscillatory neural networks using vo2 based phase encoded logic. *Front. Neurosci.* **15**, 655823 (2021).

118. Zhang, T., Haider, M. R., Massoud, Y. & Alexander, J. I. D. An oscillatory neural network based local processing unit for pattern recognition applications. *Electronics* **8**, 64 (2019).

119. Biswas, D., Pallikkulath, S. & Chakravarthy, V. S. A complex-valued oscillatory neural network for storage and retrieval of multidimensional aperiodic signals. *Front. Comput. Neurosci.* **15**, 551111 (2021).

120. Yun, S.-Y., Han, J.-K. & Choi, Y.-K. A nanoscale bistable resistor for an oscillatory neural network. *Nano Lett.* **24**, 2751–2757 (2024).

121. Abernot, M. & Todri-Sanial, A. Training energy-based single-layer hopfield and oscillatory networks with unsupervised and supervised algorithms for image classification. *Neural Comput. Appl.* **35**, 18505–18518 (2023).

122. Sabo, F. & Todri-Sanial, A. Classonn: classification with oscillatory neural networks using the Kuramoto model. In *Design, Automation, Test in Europe (DATE)* 1–2 (IEEE, 2024).

123. Kim, H. et al. Understanding rhythmic synchronization of oscillatory neural networks based on nbox artificial neurons for edge detection. *IEEE Trans. Electron Devices* **70**, 3031–3036 (2023).

124. Abernot, M., Gauthier, S., Gonos, T. & Todri-Sanial, A. Sift-onn: sift feature detection algorithm employing onns for edge detection. In *Proceedings of the 2023 Annual Neuro-Inspired Computational Elements Conference (NICE '23)*, 100–107 (Association for Computing Machinery, New York, NY, USA, 2023). https://doi.org/10.1145/3584954.3584999.

125. Abernot, M., Gil, Th. & Todri-Sanial, A. On-chip learning with a 15-neuron digital oscillatory neural network implemented on Zynq processor. In *Proceedings of the International Conference on Neuromorphic Systems 2022 (ICONS '22)*, 1–4 (Association for Computing Machinery, New York, NY, USA, 2022). https://doi.org/10.1145/3546790.3546822.

126. Luhulima, E., Abernot, M., Corradi, F. & Todri-Sanial, A. Digital implementation of on-chip Hebbian learning for oscillatory neural network. In *2023 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED)*, 1–6 (IEEE, 2023).

127. Yang, K. et al. High-order sensory processing nanocircuit based on coupled vo2 oscillators. *Nat. Commun.* **15**, 1693 (2024).

128. Terman, D. & Wang, D. Global competition and local cooperation in a network of neural oscillators. *Phys. D: Nonlinear Phenom.* **81**, 148–176 (1995).

129. Campbell, S. R., Wang, D. L. & Jayaprakash, C. Synchrony and desynchrony in integrate-and-fire oscillators. *Neural Comput.* **11**, 1595–1619 (1999).

130. Chen, K. & Wang, D. L. Image segmentation based on a dynamically coupled neural oscillator network. In *IJCNN'99. International Joint Conference on Neural Networks. Proceedings (Cat. No. 99CH36339)*, Vol. 4, 2653–2658 (IEEE, 1999).

131. Cosp, J., Madrenas, J. & Cabestany, J. A VLSI implementation of a neuromorphic network for scene segmentation. In *Proceedings of the Seventh International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems*, 403–408 (IEEE, 1999).

132. Cosp, J., Madrenas, J. & Fernández, D. Design and basic blocks of a neuromorphic vlsi analogue vision system. *Neurocomputing* **69**, 1962–1970 (2006).

133. Fernandes, D. N., Stedile, J. P. & Navaux, P. O. A. Architecture of oscillatory neural network for image segmentation. In *14th Symposium on Computer Architecture and High Performance Computing, 2002. Proceedings,* 29–36 (IEEE, 2002).

134. Fernandes, D. N. & Navaux, P. O. A. A low complexity digital oscillatory neural network for image segmentation. In *Proceedings of the Fourth IEEE International Symposium on Signal Processing and Information Technology, 2004*, 365–368 (IEEE, 2004).

135. Girau, B. & Torres-Huitzil, C. Massively distributed digital implementation of an integrate-and-fire legion network for visual scene segmentation. *Neurocomputing* **70**, 1186–1197 (2007).

136. Yu, G. & Slotine, J.-J. Visual grouping by neural oscillator networks. *IEEE Trans. Neural Netw.* **20**, 1871–1884 (2009).

137. Tan, X., Dong, H., Yang, X. & Tan, X. A hierarchical image segmentation oscillator network based on shared contextual synchronization. In *2012 International Conference on Computer Science and Information Processing (CSIP)*, 113–116 (IEEE, 2012).

138. Cotter, M. J., Fang, Y., Levitan, S. P., Chiarulli, D. M. & Narayanan, V. Computational architectures based on coupled oscillators. In *2014 IEEE Computer Society Annual Symposium on VLSI*, 130–135 (IEEE, 2014).

139. Itoh, M. & Chua, L. O. Star cellular neural networks for associative and dynamic memories. *Int. J. Bifurc. Chaos* **14**, 1725–1772 (2004).

140. Velichko, A., Belyaev, M. & Boriskov, P. A model of an oscillatory neural network with multilevel neurons for pattern recognition and computing. *Electronics* **8**, 75 (2019).

141. Rao, A. R. An oscillatory neural network model that demonstrates the benefits of multisensory learning. *Cogn. Neurodyn.* **12**, 481–499 (2018).

142. Abernot, M. & Aida, T.-S. Simulation and implementation of two-layer oscillatory neural networks for image edge detection: bidirectional and feedforward architectures. *Neuromorphic Comput. Eng.* **3**, 014006 (2023).

143. Dutta, S. et al. Programmable coupled oscillators for synchronized locomotion. *Nat. Commun.* **10**, 3299 (2019).

144. Zhang, H. et al. Integrated photonic reservoir computing based on hierarchical time-multiplexing structure. *Opt. Express* **22**, 31356–31370 (2014).

145. Borghi, M., Biasi, S. & Pavesi, L. Reservoir computing based on a silicon microring and time multiplexing for binary and analog operations. *Sci. Rep.* **11**, 15642 (2021).

146. Goldmann, M., Mirasso, C. R., Fischer, I. & Soriano, M. C. Exploiting transient dynamics of a time-multiplexed reservoir to boost the system performance. In *2021 International Joint Conference on Neural Networks (IJCNN),* 1–8 (IEEE, 2021).

147. Yamane, T., Katayama, Y., Nakane, R., Tanaka, G. & Nakano, D. Wave-based reservoir computing by synchronization of coupled oscillators. In *Neural Information Processing: 22nd International Conference, ICONIP 2015, Istanbul, Turkey, November 9-12, 2015, Proceedings Part III 22*, 198–205 (Springer, 2015).

148. Coulombe, J. C., York, M. C. A. & Sylvestre, J. Computing with networks of nonlinear mechanical oscillators. *PLoS ONE* **12**, e0178663 (2017).

149. Tanaka, G. et al. Recent advances in physical reservoir computing: a review. *Neural Netw.* **115**, 100–123 (2019).

150. Riou, M. et al. Temporal pattern recognition with delayed-feedback spin-torque nano-oscillators. *Phys. Rev. Appl.* **12**, 024049 (2019).

151. Marković, D. et al. Reservoir computing with the frequency, phase, and amplitude of spin-torque nano-oscillators. *Appl. Phys. Lett.* **114**, 012409 (2019).

152. Velichko, A. A., Ryabokon, D. V., Khanin, S. D., Sidorenko, A. V. & Rikkiev, A. G. Reservoir computing using high order synchronization of coupled oscillators. In *IOP Conference Series: Materials Science and Engineering*, Vol. 862, 052062 (IOP Publishing, 2020).

153. Nokkala, J., Martínez-Peña, R., Zambrini, R. & Soriano, M. C. High-performance reservoir computing with fluctuations in linear networks. *IEEE Trans. Neural Netw. Learn. Syst.* **33**, 2664–2675 (2021).

154. Lukoševičius, M. & Jaeger, H. Reservoir computing approaches to recurrent neural network training. *Computer Sci. Rev.* **3**, 127–149 (2009).

155. Appeltant, L. et al. Information processing using a single dynamical node as complex system. *Nat. Commun.* **2**, 468 (2011).

156. Karp, R. M. *Reducibility Among Combinatorial Problems* (Springer, 2010).

157. Lucas, A. Ising formulations of many np problems. *Front. Phys.* **2**, 5 (2014).

158. Wang, T. & Roychowdhury, J. Oim: Oscillator-based Ising machines for solving combinatorial optimisation problems. In *Unconventional Computation and Natural Computation: 18th International Conference, UCNC 2019, Tokyo, Japan, June 3–7, 2019, Proceedings 18*, 232–256 (Springer, 2019).

159. Bashar, M. K. et al. Experimental demonstration of a reconfigurable coupled oscillator platform to solve the max-cut problem. *IEEE J. Explor. Solid-State Comput. Devices Circ.* **6**, 116–121 (2020).

160. Steinerberger, S. Max-cut via Kuramoto-type oscillators. *SIAM J. Appl. Dyn. Syst.* **22**, 730–743 (2023).

161. Bashar, M. K., Li, Z., Narayanan, V. & Shukla, N. An FPGA-based max-k-cut accelerator exploiting oscillator synchronization model. In *2024 25th International Symposium on Quality Electronic Design (ISQED),* 1–8 (IEEE, 2024).

162. Hopfield, J. J. & Tank, D. W. "neural" computation of decisions in optimization problems. *Biol. Cybern.* **52**, 141–152 (1985).

163. Duane, G. S. A "cellular neuronal" approach to optimization problems. *Chaos: Interdiscip. J. Nonlinear Sci.* **19**, 3 (2009).

164. Landge, S., Saraswat, V., Singh, S. F. & Ganguly, U. n-oscillator neural network based efficient cost function for n-city traveling salesman problem. In *2020 International Joint Conference on Neural Networks (IJCNN)*, 1–8 (IEEE, 2020).

165. Delacour, C. & Todri-Sanial, A. Solving the travelling salesman problem in continuous phase domain with neuromorphic oscillatory neural networks. In *Nature Conference: AI, Neuroscience and Hardware: From Neural to Artificial Systems and Back Again* (Bonn, Germany, 2022).

166. Mallick, A. et al. Graph coloring using coupled oscillator-based dynamical systems. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)* 1–5 (IEEE, 2021).

167. Maher, O. et al. A CMOS-compatible oscillation-based vo2 ising machine solver. *Nat. Commun.* **15**, 3334 (2024).

168. Cílasun, H. et al. 3sat on an all-to-all-connected CMOS Ising solver chip. *Sci. Rep.* **14**, 10757 (2024).

169. Bashar, M. K., Lin, Z. & Shukla, N. Oscillator-Inspired Dynamical Systems to Solve Boolean Satisfiability. In *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, Vol. 9, 12–20. https://doi.org/10.1109/JXCDC.2023.3241045 (2023).

170. Hebb, D. O. *The Organization of Behavior: A Neuropsychological Theory* (Psychology Press, 2005).

171. Storkey, A. Increasing the capacity of a hopfield network without sacrificing functionality. In *Artificial Neural Networks-ICANN'97: 7th International Conference Lausanne, Switzerland, October 8–10, 1997 Proceedings 7*, 451–456 (Springer, 1997).

172. Diederich, S. & Opper, M. Learning of correlated patterns in spin-glass networks by local learning rules. *Phys. Rev. Lett.* **58**, 949 (1987).

173. Scellier, B. & Bengio, Y. Equilibrium propagation: Bridging the gap between energy-based models and backpropagation. *Front. Comput. Neurosci.* **11**, 24 (2017).

174. Laydevant, J., Marković, D. & Grollier, J. Training an ising machine with equilibrium propagation. *Nat. Commun.* **15**, 3671 (2024).

175. Deng, L. The MNIST database of handwritten digit images for machine learning research [best of the web]. *IEEE Signal Process. Mag.* **29**, 141–142 (2012).

176. Kayed, M., Anter, A. & Mohamed, H. Classification of garments from fashion mnist dataset using cnn lenet-5 architecture. In *2020 international conference on innovative trends in communication and computer engineering (ITCE)*, 238–243 (IEEE, 2020).

177. Wang, Y. et al. Improvement of MNIST image recognition based on CNN. In *IOP Conference Series: Earth and Environmental Science*, Vol. 428, 012097 (IOP Publishing, 2020).

178. Kadam, S. S., Adamuthe, A. C. & Patil, A. B. Cnn model for image classification on mnist and fashion-mnist dataset. *J. Sci. Res.* **64**, 374–384 (2020).

179. Wang, Q. et al Training coupled phase oscillators as a neuromorphic platform using equilibrium propagation. *Neuromorph. Comput. Eng.* **4,** 034014 (2024).

180. Shamsi, J., Avedillo, M. J., Linares-Barranco, B. & Serrano-Gotarredona, T. Oscillatory Hebbian rule (ohr): an adaption of the Hebbian rule to oscillatory neural networks. In *2020 XXXV conference on design of circuits and integrated systems (DCIS)*, 1–6 (IEEE, 2020).

181. Jiménez, M., Avedillo, M. J., Linares-Barranco, B. & Núñez, J. Learning algorithms for oscillatory neural networks as associative memory for pattern recognition. *Front. Neurosci.* **17**, 1257611 (2023).

182. Graber, M. & Hofmann, K. An enhanced 1440 coupled CMOS oscillator network to solve combinatorial optimization problems. In *2023 IEEE 36th International System-on-Chip Conference (SOCC)*, 1–6 (IEEE, 2023).

183. Belyaev, M. A. & Velichko, A. A. Classification of handwritten digits using the Hopfield network. In *IOP Conference Series: Materials Science and Engineering*, Vol. 862, 052048 (IOP Publishing, 2020).

184. Reinelt, G. Tsplib-a traveling salesman problem library. *ORSA J. Comput.* **3**, 376–384 (1991).

185. Popovych, O. V., Maistrenko, Y. L. & Tass, P. A. Phase chaos in coupled oscillators. *Phys. Rev. E-Stat., Nonlinear, Soft Matter Phys.* **71**, 065201 (2005).

186. Phan, N.-T. et al. Unbiased random bitstream generation using injection-locked spin-torque nanooscillators. *Phys. Rev. Appl.* **21**, 1–14 (2024).

187. Hubbard, J. Electron correlations in narrow energy bands. *Proc. R. Soc. Lond. Ser. A. Math. Phys. Sci.* **276**, 238–257 (1963).

188. Aadit, N. A. et al. Massively parallel probabilistic computing with sparse Ising machines. *Nat. Electron.* **5**, 460–468 (2022).

189. Beggs, J. M. & Plenz, D. Neuronal avalanches in neocortical circuits. *J. Neurosci.* **23**, 11167–11177 (2003).

190. Al Beattie, B., Feketa, P., Ochs, K. & Kohlstedt, H. Criticality in Fitzhugh-Nagumo oscillator ensembles: design, robustness, and spatial invariance. *Commun. Phys.* **7**, 46 (2024).

191. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986).

192. Niazi, S. et al. Training deep Boltzmann networks with sparse Ising machines. *Nat. Electron.* **7**, 610–619 (2024).

193. Böhm, F., Alonso-Urquijo, D., Verschaffelt, G. & Van der Sande, G. Noise-injected analog ising machines enable ultrafast statistical sampling and machine learning. *Nat. Commun.* **13**, 5847 (2022).

194. Wu, Z. & He, S. Improvement of the Alexnet networks for large-scale recognition applications. *Iran. J. Sci. Technol. Trans. Electr. Eng.* **45**, 493–503 (2021).

195. Byerly, A., Kalganova, T. & Dear, I. No routing needed between capsules. *Neurocomputing* **463**, 545–553 (2021).

196. Poma, X. S., Riba, E. & Sappa, A. Dense extreme inception network: towards a robust cnn model for edge detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 1923–1932 (IEEE, 2020).

197. Nose, Y., Kojima, A., Kawabata, H. & Hironaka, T. A study on a lane keeping system using cnn for online learning of steering control from real time images. In *2019 34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, 1–4 (IEEE, 2019).

198. Jain, A., Singh, A., Koppula, H. S., Soh, S. & Saxena, A. Recurrent neural networks for driver activity anticipation via sensory-fusion architecture. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 3118–3125 (IEEE, 2016).

199. Chou, J., Bramhavar, S., Ghosh, S. & Herzog, W. Analog coupled oscillator based weighted Ising machine. *Sci. Rep.* **9**, 14786 (2019).

200. Wang, T., Wu, L. & Roychowdhury, J. New computational results and hardware prototypes for oscillator-based ISINg machines. In *Proceedings of the 56th Annual Design Automation Conference*, 2–6 (IEEE 2019).

201. Bashar, M. K., Mallick, A. & Shukla, N. Experimental investigation of the dynamics of coupled oscillators as Ising machines. *IEEE Access* **9**, 148184–148190 (2021).

202. Mallick, A., Bashar, M. K., Truesdell, D. S., Calhoun, B. H. & Shukla, N. Overcoming the accuracy vs. performance trade-off in oscillator Ising machines. In *2021 IEEE International Electron Devices Meeting (IEDM)*, 1–4 (IEEE, 2021).

203. Lo, H., Moy, W., Yu, H., Sapatnekar, S. & Kim, C. H. An ising solver chip based on coupled ring oscillators with a 48-node all-to-all connected array architecture. *Nat. Electron.* **6**, 771–778 (2023).

204. Maher, O. et al. Highly reproducible and CMOS-compatible vo2-based oscillators for brain-inspired computing. *Sci. Rep.* **14**, 11600 (2024).

205. Moskewicz, M. W., Madigan, C. F., Zhao, Y., Zhang, L. & Malik, S. Chaff: engineering an efficient sat solver. In *Proceedings of the Annual Design Automation Conference (DAC)*, 530–535 (2001).

206. Marques-Silva, J. P. & Sakallah, K. A. Grasp: A search algorithm for propositional satisfiability. *IEEE Trans. Computers* **48**, 506–521 (1999).

207. Selman, B., Levesque, H. & Mitchell, D. A new method for solving hard satisfiability problems. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, AAAI'92, 440–446 (AAAI Press, 1992).

208. Hoos, H. H. & Stützle, T. Local search algorithms for SAT: an empirical evaluation. *J. Automated Reason.* **24**, 421–481 (2000).

209. Bashar, M. K. et al. An oscillator-based maxsat solver. Preprint at https://arxiv.org/abs/2109.09897 (2021).

210. Bybee, C. et al. Efficient optimization with higher-order Ising machines. *Nat. Commun.* **14**, 6033 (2023).

211. Gilmore, P. C. & Gomory, R. E. A linear programming approach to the cutting-stock problem. *Oper. Res.* **9**, 849–859 (1961).

212. Land, A. H. & Doig, A. G. *An Automatic Method for Solving Discrete Programming Problems* (Springer, 2010).

213. Glover, F. Future paths for integer programming and links to artificial intelligence. *Comput. Oper. Res.* **13**, 533–549 (1986).

214. Robson, J. M. *Finding a Maximum Independent Set in Time o (2n/4)*. Tech. Rep., Technical Report 1251-01, LaBRI (Université Bordeaux I, 2001).

215. Feige, U. Approximating maximum clique by removing subgraphs. *SIAM J. Discret. Math.* **18**, 219–225 (2004).

216. Boppana, R. & Halldórsson, M. M. Approximating maximum independent sets by excluding subgraphs. *BIT Numer. Math.* **32**, 180–196 (1992).

217. Mallick, A. et al. Using synchronized oscillators to compute the maximum independent set. *Nat. Commun.* **11**, 4689 (2020).

218. Mallick, A., Bashar, M. K., Lin, Z. & Shukla, N. Computational models based on synchronized oscillators for solving combinatorial optimization problems. *Phys. Rev. Appl.* **17**, 064064 (2022).

219. Ferrari, L., Scaglione, A., Gentz, R. & Hong, Y.-W. P. Convergence results on pulse coupled oscillator protocols in locally connected networks. *IEEE/ACM Trans. Netw.* **25**, 1004–1019 (2016).

220. Halldórsson, M. M., Kratochvíl, J. & Telle, J. A. Independent sets with domination constraints. *Discret. Appl. Math.* **99**, 39–54 (2000).

221. Chen, J., Kanj, I. A. & Xia, G. Improved parameterized upper bounds for vertex cover. In *Mathematical Foundations of Computer Science 2006: 31st International Symposium, MFCS 2006, Stará Lesná, Slovakia, August 28-September 1, 2006. Proceedings 31*, 238–249 (Springer, 2006).

222. Papadimitriou, C. H. & Steiglitz, K. *Combinatorial Optimization: Algorithms and Complexity* (Courier Corporation, 1998).

223. Karakostas, G. A better approximation ratio for the vertex cover problem. *ACM Trans. Algorithms (TALG)* **5**, 1–8 (2009).

224. Corder, K., Monaco, J. V. & Vindiola, M. M. Solving Vertex Cover via Ising Model on a Neuromorphic. *Processor, IEEE International Symposium on Circuits and Systems (ISCAS)*, 1–5, (Florence, Italy, 2018). https://doi.org/10.1109/ISCAS.2018.8351248.

225. Al Beattie, B., Noll, M., Kohlstedt, H. & Ochs, K. Oscillator-based optimization: design, emulation, and implementation. *Eur. Phys. J. B* **97**, 7 (2024).

226. Chvatal, V. A greedy heuristic for the set-covering problem. *Math. Oper. Res.* **4**, 233–235 (1979).

227. Fomin, F. V. & Villanger, Y. Finding Induced Subgraphs via Minimal Triangulations, Leibniz International. *Proceedings in Informatics*, Vol. 5, 383–394 (Dagstuhl Publishing, 2010).

228. Becker, A. & Geiger, D. Optimization of Pearl's method of conditioning and greedy-like approximation algorithms for the vertex feedback set problem. *Artif. Intell.* **83**, 167–188 (1996).

229. Lawler, E. A comment on minimum feedback arc sets. *IEEE Trans. Circ. Theory* **11**, 296–297 (1964).

230. Even, G., Naor, J., Schieber, B. & Sudan, M. Approximating minimum feedback sets and multicuts in directed graphs. *Algorithmica* **20**, 151–174 (1998).

231. Bjorklund, A. Determinant sums for undirected hamiltonicity. *SIAM J. Comput.* **43**, 280–299 (2014).

232. Applegate, D. et al. On the solution of traveling salesman problems. *Documenta Mathematica*, 645–656 (1998).

233. Christofides, N. Worst-case analysis of a new heuristic for the travelling salesman problem. In SN *Operations Research Forum*, Vol. 3, 1–4 (Springer, 2022). https://doi.org/10.1007/s43069-021-00101-z.

234. Schoning, U. A probabilistic algorithm for k-sat and constraint satisfaction problems. In *40th Annual Symposium on Foundations of Computer Science (Cat. No. 99CB37039)*, 410–414 (IEEE, 1999).

235. Björklund, A., Husfeldt, T. & Koivisto, M. Set partitioning via inclusion-exclusion. *SIAM J. Comput.* **39**, 546–563 (2009).

236. Halldórsson, M. M. A still better performance guarantee for approximate graph coloring. *Inf. Process. Lett.* **45**, 19–23 (1993).

237. Espelage, W., Gurski, F. & Wanke, E. How to solve np-hard graph problems on clique-width bounded graphs in polynomial time. In *Graph-Theoretic Concepts in Computer Science: 27th InternationalWorkshop, WG 2001 Boltenhagen, Germany, June 14–16, 2001 Proceedings 27*, 117–128 (Springer, 2001).

238. Cerioli, M. R. et al. Partition into cliques for cubic graphs: planar case, complexity and approximation. *Discret. Appl. Math.* **156**, 2270–2278 (2008).

239. Blanchette, M., Kim, E. & Vetta, A. Clique cover on sparse networks. In *2012 Proceedings of the Fourteenth Workshop on Algorithm Engineering and Experiments (ALENEX)*, 93–102 (SIAM, 2012).

240. Yamada, Y. & Inaba, K. Detecting partial synchrony in a complex oscillatory network using pseudovortices. *Phys. Rev. E* **108**, 024307 (2023).

241. Knuth, D. E. Dancing links. *Millennial Perspect. Comput. Sci.* **187**, 1569 (2000).

242. Wang, X. *Exact Algorithms for the Steiner Tree Problem*. PhD thesis, University of Twente, Enschede, The Netherlands (2008).

243. Wu, Y.-F., Widmayer, P. & Wong, C.-K. A faster approximation algorithm for the Steiner problem in graphs. *Acta Inform.* **23**, 223–229 (1986).

244. Byrka, J., Grandoni, F., Rothvoß, T. & Sanita, L. An improved lp-based approximation for Steiner tree. In *Proceedings of the forty-second ACM symposium on Theory of computing*, 583–592 (ACM, 2010).

245. Chen, C.-Y. & Hsieh, S.-Y. An improved algorithm for the Steiner tree problem with bounded edge-length. *J. Comput. Syst. Sci.* **123**, 20–36 (2022).

246. Cygan, M. Improved approximation for 3-dimensional matching via bounded pathwidth local search. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, 509–518 (IEEE, 2013).

247. Andonov, R., Poirriez, V. & Rajopadhye, S. Unbounded knapsack problem: dynamic programming revisited. *Eur. J. Oper. Res.* **123**, 394–407 (2000).

248. Horowitz, E. & Sahni, S. Computing partitions with applications to the knapsack problem. *J. ACM (JACM)* **21**, 277–292 (1974).

249. Dantzig, G. B. Discrete-variable extremum problems. *Oper. Res.* **5**, 266–288 (1957).

250. Vazirani, V. V. *Approximation Algorithms*, Vol. 1 (Springer, 2001).

251. Watanabe, Y., Yoshino, K. & Kakeshita, T. Solving combinatorial optimization problems using the oscillatory neural network. *IEICE Trans. Inf. Syst.* **80**, 72–77 (1997).

252. Chen, X., Lan, Y., Benko, A., Dósa, G. & Han, X. Optimal algorithms for online scheduling with bounded rearrangement at the end. *Theor. Comput. Sci.* **412**, 6269–6278 (2011).

253. Hochbaum, D. S. & Shmoys, D. B. Using dual approximation algorithms for scheduling problems theoretical and practical results. *J. ACM (JACM)* **34**, 144–162 (1987).

254. Korf, R. E. Multi-way number partitioning. In *Proceedings of the 21st international joint conference on artificial intelligence* 538–543 (Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2009).

255. Martello, S. & Toth, P. *Knapsack Problems: Algorithms and Computer Implementations* (John Wiley & Sons, Inc., 1990).

256. Korf, R. E. A complete anytime algorithm for number partitioning. *Artif. Intell.* **106**, 181–203 (1998).

257. Coffman Jr E. G., Garey, M. R. & Johnson, D. S. An application of bin-packing to multiprocessor scheduling. *SIAM J. Comput.* **7**, 1–17 (1978).

258. Nigg, S. E., Lörch, N. & Tiwari, R. P. Robust quantum optimizer with full connectivity. *Sci. Adv.* **3**, e1602273 (2017).

259. Goemans, M. X. & Williamson, D. P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM (JACM)* **42**, 1115–1145 (1995).

260. Parihar, A., Shukla, N., Jerry, M., Datta, S. & Raychowdhury, A. Vertex coloring of graphs via phase dynamics of coupled oscillatory networks. *Sci. Rep.* **7**, 911 (2017).
261. Ascoli, A. et al. Control strategies to optimize graph coloring via M-CNNS with locally-active nbo x memristors. In *2021 10th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, 1–8 (IEEE, 2021).
262. Wu, J., Jiao, L., Li, R. & Chen, W. Clustering dynamics of nonlinear oscillator network: application to graph coloring problem. *Phys. D: Nonlinear Phenom.* **240**, 1972–1978 (2011).

Author contributions

A.T.S. was responsible for the conceptualization of the paper and methodology. C.D., F.S., M.A., and A.T.S. were responsible for data curation, illustrations, and writing. A.T.S. was responsible for funding acquisition, project administration, resources and supervision. All authors wrote the original draft, and reviewed and edited the manuscript. All authors have read and approved the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Aida Todri-Sanial.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.