



Oscillator-based optimization: design, emulation, and implementation

Bakr Al Beattie^{1,a} , Maximiliane Noll^{2,b}, Hermann Kohlstedt^{2,c}, and Karlheinz Ochs^{1,d}

¹ Chair of Digital Communication Systems, Faculty of Electrical Engineering and Information Technology, Ruhr-University Bochum, 44801 Bochum, Nordrhein-Westfalen, Germany

² Chair of Nanoelectronics, Technical Faculty Kiel, 24143 Kiel, Schleswig-Holstein, Germany

Received 24 October 2023 / Accepted 28 November 2023 / Published online 15 January 2024
© The Author(s) 2024

Abstract. The emergence of digital computers has profoundly reshaped our interactions with technology and the processing of information. Despite excelling in data processing and arithmetics, these computers face limitations in tackling complex nondeterministic-polynomial (NP) problems. In response, researchers have started searching for new computational paradigms that possess the natural tendency of solving these problems. Oscillator-based optimizers are one such paradigm, where the idea is to exploit the parallelism of oscillators networks in order to efficiently solve NP problems. This involves a process of mapping a given optimization task to a quadratic unconstrained binary optimization program and then mapping the resulting program onto an inter-oscillator coupling circuit encoding its coefficients. This paper presents a comprehensive approach to constructing oscillator-based optimizers, offering both the rationale for employing oscillator networks and formulas for linking optimization coefficients to inter-oscillator coupling. Here, we cover most aspects of oscillator-based optimization starting from the design of the network up to its technical implementation. Moreover, we provide a platform-independent wave digital algorithm, which allows for emulating our network's behavior in a highly parallel fashion.

1 Introduction

Digital computers, based on the von Neumann architecture [1], excel at many things such as complex calculations and data processing. However, when it comes to optimization there remain some challenges when dealing with certain problems, such as nondeterministic-polynomial (NP) problems. These are optimization problem that can not be efficiently solved in polynomial time using classical algorithmic approaches [2]. As a result, researchers have explored alternative computational paradigms, such as quantum computers [3, 4] and optical computers [5, 6], which hold promise in tackling these computationally difficult tasks.

A more recent approach for dealing with NP problems, lies in mapping them onto “the energy” of a network of electrical oscillators and allowing the network to run and naturally solve the problem. Here, the underlying assumption is that the network has the natural tendency of minimizing its “energy” consumption, although the term energy is not clearly defined but is to be understood in terms of a “generalized energy” such as the one appearing in Lyapunov's direct

method [7]. An optimization program based on a network of electrical oscillators is what we refer to as oscillator-based optimization in this paper. In literature, these types of machines are usually referred to as oscillator-based Ising machines for reasons that we explain later on. To our utmost knowledge, the notion of oscillator-based optimizers has been first presented in the preprint of Wang and Roychowdury [8]. In their published works [9, 10], the authors discuss the functionality of LC-oscillator networks as optimizers using the theory of phase macromodels [11], which, in the case of LC-oscillators, degenerate to the well-established Kuramoto model [12]. This led to a series of publications, where different authors have attempted to explain the functionalities of oscillator-based optimizers in more detail using different phase models [13–15]. Interestingly, it has also been shown that phase macromodels can be reliably used for simulated annealing [14–17]. Over the years, many hardware implementations of oscillator-based optimizers have emerged of which we can only list a few [13, 18–21]. Most implementations are based on ring oscillators due to their simplicity and CMOS-compatibility [22, 23].

The oscillator-based optimization that we deal with in this work can be explained in three steps:

^a e-mail: bakr.albeattie@rub.de (corresponding author)

^b e-mail: mno@tf.uni-kiel.de

^c e-mail: hko@tf.uni-kiel.de

^d e-mail: karlheinz.ochs@rub.de

1. A given optimization task is mapped onto a quadratic unconstrained binary optimization problem (QUBO):

$$\min_x \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{r}^T \mathbf{x}, \quad \text{with } \mathbf{Q} = \mathbf{Q}^T. \quad (1a)$$

Here $x_\nu \in \{0, 1\}$ are binary optimization variables resulting from the solution of the problem, $\mathbf{Q} \in \mathbb{R}^{n \times n}$ and $\mathbf{r} \in \mathbb{R}^n$ represent a matrix and vector of optimization coefficients, respectively.

2. The QUBO is reformulated in terms of the so-called Ising Hamiltonian:

$$H(\mathbf{s}) = -\mathbf{s}^T \mathbf{J} \mathbf{s} - \mathbf{h}^T \mathbf{s}, \quad \text{with } \mathbf{J} = \mathbf{J}^T, \quad (1b)$$

where $s_\nu \in \{\pm 1\}$ are equivalent binary optimization variables that are usually referred to as spins. Here, the bijective mapping,

$$\mathbf{s} = 2\mathbf{x} - \mathbb{1} \quad \Leftrightarrow \quad \mathbf{x} = [\mathbf{s} - \mathbb{1}]/2, \quad (1c)$$

where $\mathbb{1}$ denotes a vector of ones, allows for reformulating every QUBO (1a) as a minimization of (1b) with:

$$\mathbf{J} = -\mathbf{Q} \quad \text{and} \quad \mathbf{h} = 2\mathbf{Q}\mathbb{1} - 2\mathbf{r}. \quad (1d)$$

In the following, we refer to the entries of \mathbf{Q} as quadratic optimization coefficients and the entries of \mathbf{h} as linear optimization coefficients.

3. The optimization coefficients are mapped onto the couplings of an oscillator network. The optimizer then runs and solves the problem. The solution can be decoded from the phase configuration that the network settles to. A (relative) phase shift of 0 (π) corresponds to $s = 1$ ($s = -1$).

In a previous work [15], we have discussed how general QUBOs, whose formulations are far more sophisticated than (1a), can be mapped onto the Ising Hamiltonian (1b). Together with the works of [24–27], which discuss the mapping of NP-problems onto QUBOs, steps 1 and 2 of the oscillator-based optimization program can be achieved. The third step assumes the existence of an oscillator network with the natural tendency of minimizing the Ising Hamiltonian (1b); this is also the reason why oscillator-based optimizers are referred to as Ising machines [9, 10].

The aim of this paper is to extensively discuss the third step of oscillator-based optimization. This involves the design of an oscillator network, which naturally tends to minimize the Ising Hamiltonian (1b). Contrary to most works, which attempt to explain the functionality of oscillator-based optimizers based on phase models, we explain their functionality based on the concepts of synchronization and power minimiza-

tion. Our approach is motivated by the fact that phase models only represent an approximation of the network's actual dynamics [28], and while they enhance our understanding of network dynamics, they do not explain the functionality of oscillator-based optimizers on a circuit level. Furthermore, our approach will allow us to draw a clear relation between the optimization coefficients of a QUBO problem and the inter-oscillator coupling circuit that is to be used. Another major advantage of our study is that we cover the entire third step of oscillator-based optimization. Most works on literature only partially cover this step by either discussing the modeling, hardware implementation, or mapping, separately. Therefore a clear relation between theory and practice has never been drawn in one work.

Besides modeling and implementation, a large portion of our work deals with the emulation of the proposed oscillator-based optimizer. Here, we make use of the wave digital (WD) concept as a powerful emulation tool [29–31]. A major benefit of WD algorithms is the fact that they can be massively parallelized, when dealing with structurally identical circuits [16, 32–36]. This is especially beneficial for oscillator-based optimizers, as they are composed of many oscillators of the same topology. Thus, we can exploit the concept of vector-valued wave digital models in order to efficiently emulate large networks on any platform in a highly parallel fashion.

In total, the main contributions of this paper can be summarized as follows:

- I We give a constructive guide for building oscillator-based optimizers;
- II We give a physical justification for why oscillator networks can be utilized as optimizers;
- III We explain the relationship between the Ising Hamiltonian (1b) and the synchronization tendencies of oscillator networks. Here, we also present a formula for mapping the coefficients of the Ising Hamiltonian onto the inter-oscillator coupling;
- IV We provide a compact electrical model for our oscillator-based optimizers;
- V We derive a massively parallel WD algorithm for emulating our oscillator-based optimizer with an arbitrary number of oscillators and inter-oscillator couplings; and
- VI We provide a technical blueprint of our in-house optimizer, based on the theoretical model discussed in this work.

The remainder of this paper is divided into three sections. Section 2 discusses the design of oscillator-based optimizers (I–IV). Section 3 derives a wave digital algorithm for our proposed oscillator-based optimizer (V). Section 4 presents the technical implementation of our in-house optimizer (VI). Finally Sect. 5 summarizes the contributions of our work and provides an outlook on future research in this context.

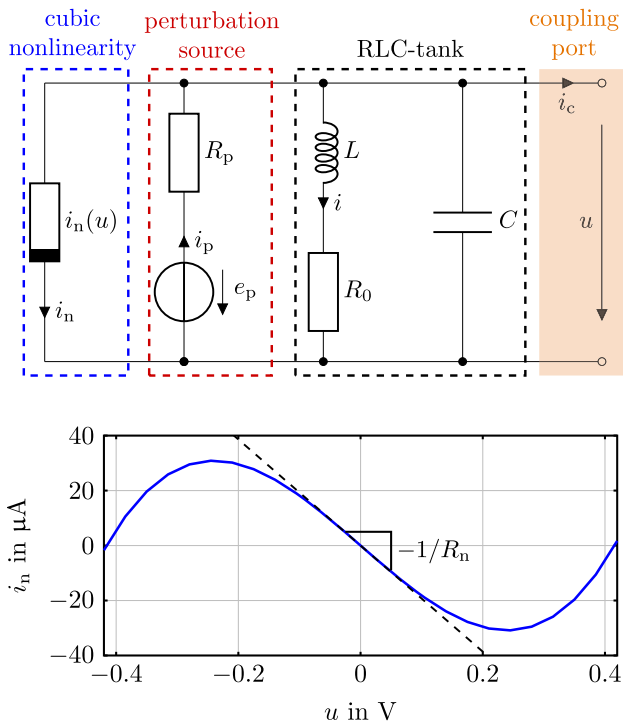


Fig. 1 Top: circuit schematic of the FitzHugh-Nagumo oscillator. Bottom: (i, u) -curve of the cubic nonlinearity $i_n(u)$ in the blue dashed box

2 Designing an oscillator-based optimizer

This section establishes the theoretical foundation for the oscillator-based optimizer, whose implementation is discussed in Sect. 4. We start by introducing our oscillator model of choice, which constitutes the core of our network. To understand how optimization problems can be mapped to an oscillator network, we then discuss general synchronization tendencies of coupled oscillators. Furthermore, we explain how (necessary) optimization constraints can be set. Finally, we present an electrical model that serves as a compact description for our oscillator-based optimizer.

2.1 Oscillator model

Our oscillator of choice is the FitzHugh-Nagumo oscillator (FNO) [37] depicted in Fig. 1. From a theoretical point of view, this oscillator has the advantage that it can be implemented in hardware. Furthermore, it only contains one active nonlinear component that is responsible for generating self-sustaining oscillations, which makes the overall oscillator accessible from a mathematical point of view.

The FNO can be described by the following set of algebraic-differential equations:

$$C \frac{du}{dt} = i_p(u) - i - i_n(u) - i_c, \quad (2a)$$

Table 1 FitzHugh-Nagumo oscillator simulation parameters

Oscillator parameters		
$R_0 = 4.7 \text{ k}\Omega$	$L = 23.5 \text{ H}$	$C = 100 \text{ nF}$
$R_p = 47 \text{ k}\Omega$	$R_n = 5.2 \text{ k}\Omega$	$U_0 = 0.24 \text{ V}$

$$L \frac{di}{dt} = u - R_0 i, \quad (2b)$$

$$R_n i_n(u) = u^3 / [3U_0^2] - u, \quad (2c)$$

$$R_p i_p(u) = e_p - u. \quad (2d)$$

Here, u and i denote the capacitor voltage and inductor current, respectively. The electrical components R_0 , L , and C constitute the oscillator’s RLC-tank. The current i_c is an external coupling current that results from the oscillator’s interaction with other circuits/oscillators. The cubic nonlinearity $i_n(u)$, with the fitting parameters R_n and U_0 , represents an active electrical component with a negative differential resistance $-R_n < 0$ that is used to maintain a constant energy landscape in order to generate self-sustaining oscillations. The activeness of the component can be verified from its (i, u) -curve, which is depicted on the bottom of Fig. 1. Finally, the perturbation voltage source e_p with the internal resistance R_p is used to set optimization constraints, as we explain in Sect. 2.4. Note, the oscillator parameters given in Table 1 lead to an oscillation period of $T_0 \approx 11.7 \text{ ms}$.

2.2 Synchronization tendency of coupled oscillators

Now that we have covered our oscillator model, we make use of it to demonstrate a general tendency observed in resistively coupled oscillator networks. To this end, consider two oscillators \mathcal{N}_1 and \mathcal{N}_2 with the outputs voltages u_1 and u_2 , respectively. Let us assume the oscillators to be coupled over a resistor G_{12}^- , as depicted on the left side of Fig. 2. In general, the conductance G_{12}^- is proportional to the coupling strength between \mathcal{N}_1 and \mathcal{N}_2 . If the coupling strength is chosen to be sufficiently strong, then the oscillators will always interact and have the tendency to synchronize, see the bottom left plot in Fig. 2. This general tendency is always present between coupled nonlinear oscillators regardless of the number of oscillators within the network. To our utmost knowledge, there is no formal mathematical proof showing the necessity of this behavior. However, we can justify it from a physical perspective: Synchronization minimizes power dissipation. To understand this statement, let us write out the equations governing the electrical coupling depicted on the left side of Fig. 2:

$$i_{12}^- = G_{12}^- v_{12}^-, \quad \text{with} \quad v_{12}^- = u_1 - u_2. \quad (3a)$$

The instantaneous (dissipated) power reads:

$$p_{12}(t) = v_{12}^- i_{12}^- = G_{12}^- [u_1 - u_2]^2. \quad (3b)$$

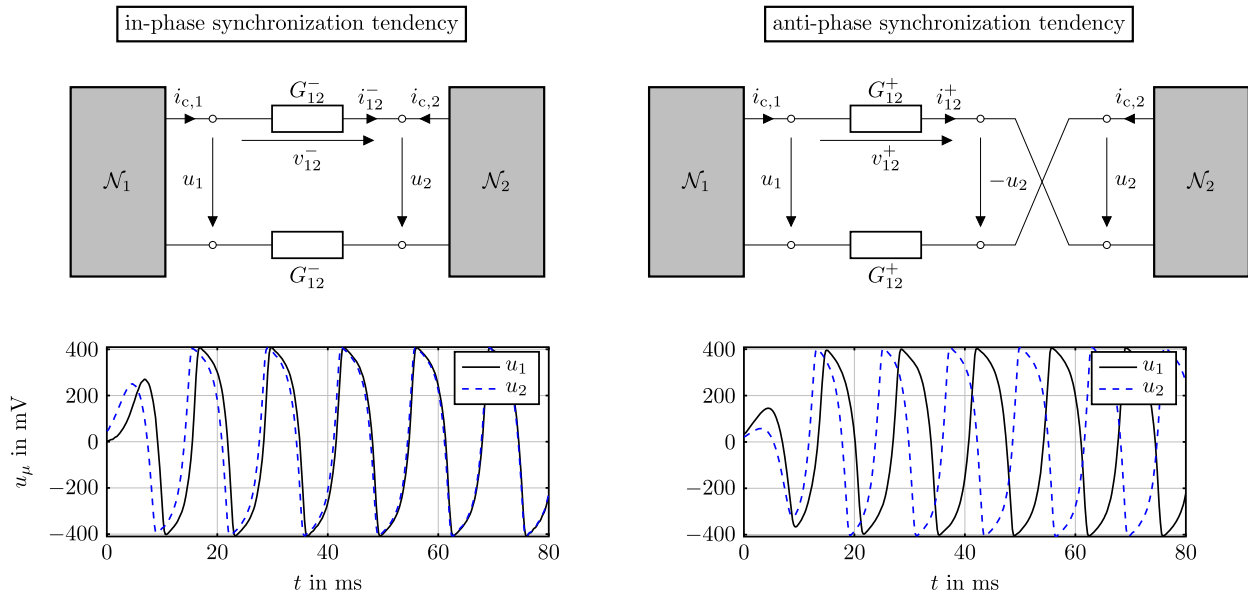


Fig. 2 Graphical representation of two coupled oscillators, which are represented by the gray dashed boxes titled \mathcal{N}_1 and \mathcal{N}_2 . Left: two oscillators coupled over a resistor with the conductance G_{12}^- , tend to synchronize in-phase through their interaction. Right: using the same type of coupling as the left side but inverting one oscillator output leads to anti-phase synchronization

As can be seen from (3b), the instantaneous power $p(t)$ is proportional to the squared voltage difference $[v_{12}^-]^2$. Therefore, we conclude that $p(t)$ is minimized when $[v_{12}^-]^2$ is minimized, i.e. for $u_1 = u_2$, the case where the both oscillators are synchronized. Overall, this lead us to the hypothesis that our oscillator network has the natural tendency of minimizing dissipated power by minimizing the amount of interaction (interchanged power) between oscillators. In physics, such tendencies have been hinted at for a long time. The tendency of a complex network to minimize a certain energy functional is referred to as the principle of least energy dissipation [38–40], which was first postulated by Onsager [41].

It is noteworthy to state that the network’s tendency to minimize power dissipation greatly depends on the magnitude of the coupling conductance G_{12}^- . For small values of G_{12}^- , the oscillator network may not minimize power dissipation due to the (already) minimal interaction caused by the high resistance. Hence, our hypothesis only holds, when G_{12}^- is chosen in a suitable manner. We cover this aspect again later on.

2.3 Mapping quadratic unconstrained binary optimization problems to synchronization tendencies

The state of synchronization only encodes the statement that two oscillators have reached a state of "agreement". To encode binary optimization problems, such as a general QUBO (1a), we also require the opposite, a state of "disagreement". The simplest way to encode the latter is by letting one oscillator experience a perspective change. If, for example, \mathcal{N}_1 only perceives

the inverted output of \mathcal{N}_2 , then \mathcal{N}_1 would synchronize with the inverted signal, again, due to principle of least energy dissipation [41]. This idea is depicted on the right side of Fig. 2. The bottom plot shows the oscillators asymptotically assuming an anti-phase configuration, which corresponds to a state of "disagreement". With reference to equations (1a) 1b, we associate agreement with $x = 1$ ($s = 1$) and disagreement with $x = 0$ ($s = -1$).

2.3.1 Quadratic optimization coefficients

Up to this point, we have only discussed how oscillators can be brought to a phase of agreement or disagreement. However, when mapping optimization problems onto oscillator networks, we have no prior knowledge about the states resulting from the solution of the encoded problem. Instead, we must draw a relation between the coefficients of the problem and the tendencies of the network. To this end, let us consider a version of (1b) with two coupled spins and no linear optimization coefficients ($\mathbf{h} = \mathbf{0}$):

$$H(\mathbf{s}) = - [s_1 \ s_2] \begin{bmatrix} 0 & J_{12} \\ J_{12} & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}. \tag{4a}$$

By varying the coupling coefficient J_{12} , we obtain the two implications:

$$\begin{aligned} J_{12} < 0 &\Rightarrow \arg \left\{ \min_s \{H(\mathbf{s})\} \right\} = \pm [1, 1]^T, \tag{4b} \\ J_{12} > 0 &\Rightarrow \arg \left\{ \min_s \{H(\mathbf{s})\} \right\} = \pm [1, -1]^T. \tag{4c} \end{aligned}$$

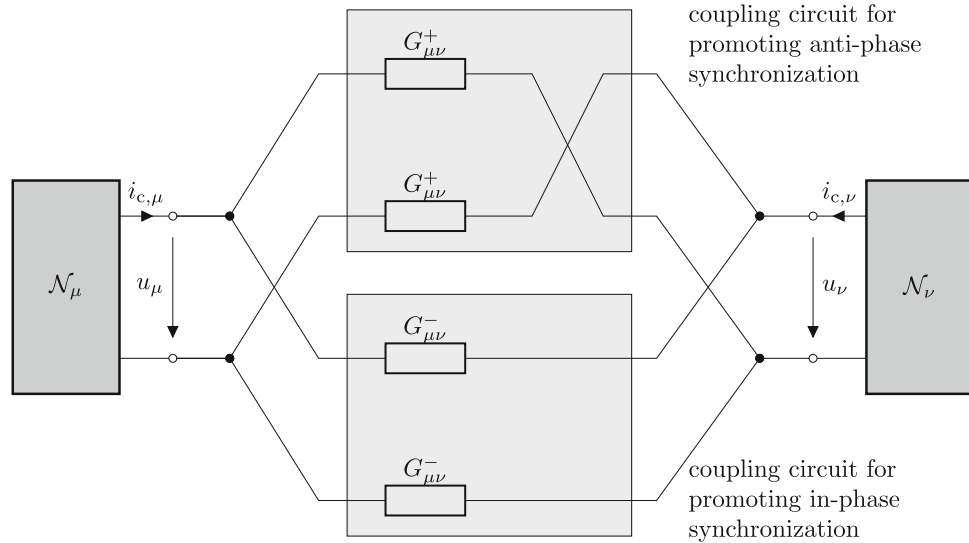


Fig. 3 Coupling circuit for encoding optimization coefficients. The dashed gray boxes \mathcal{N}_μ and \mathcal{N}_ν represent two arbitrary adjacent oscillators with the output voltages u_μ and u_ν , respectively. The top coupling circuit can be used to give a preference for anti-phase synchronization. The bottom coupling circuit can be used to give a preference for in-phase synchronization

Hence, a negative coupling coefficient $J_{12} < 0$ implies that the minimum of $H(\mathbf{s})$ is achieved, when the spins s_1 and s_2 have the same value. Meanwhile, a positive coupling coefficient $J_{12} > 0$ implies that the minimum of $H(\mathbf{s})$ is achieved, when the spins have different values. In the context of coupled oscillators, this means that $J_{12} > 0$ and $J_{12} < 0$ should ideally correspond to anti- and in-phase synchronization, respectively. To promote this behavior in our oscillator network, we can actively pick between the circuits depicted in Fig. 2.

To map the value of the coefficients in addition to their polarity, we start by normalizing all coefficients $J_{\mu\nu}$, so they are in the interval $[-1, 1]$. Thus, we assume in the following that $J_{\mu\nu} \in [-1; 1]$, where $J_{\mu\nu} \in \mathbb{R}$. With reference to our earlier discussion, $J_{\mu\nu} = 1$ should promote in-phase synchronization, hence we would utilize the coupling circuit on the left side of Fig. 2. Meanwhile, the opposite case $J_{\mu\nu} = -1$ should promote anti-phase synchronization, such that we would use the coupling circuit on the right side of Fig. 2. We can map coefficients in the interval $J_{\mu\nu}^2 < 1$ by considering the superposition of the two coupling schemes depicted in Fig. 2, i.e. by utilizing the parallel connection of both circuits as presented in Fig. 3. The electrical relation governing this coupling reads:

$$i_c = [G^+ + G^-]u, \quad i_c = \begin{bmatrix} i_{c,\mu} \\ i_{c,\nu} \end{bmatrix}, \quad u = \begin{bmatrix} u_\mu \\ u_\nu \end{bmatrix}, \tag{5a}$$

$$G^+ = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix} \frac{G_{\mu\nu}^+}{2}, \quad G^- = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \frac{G_{\mu\nu}^-}{2}. \tag{5b}$$

Here, the relation between the coupling conductances $G_{\mu\nu}^\pm$ and the optimization coefficients $J_{\mu\nu}$ is given by

$$G_{\mu\nu}^\pm = \frac{G_b}{2} \begin{cases} 1 \mp J_{\mu\nu}, & J_{\mu\nu} \neq 0 \\ 0, & J_{\mu\nu} = 0 \end{cases}, \quad R_b G_b = 1, \tag{5c}$$

where $G_b = 1/R_b$ is a multiplicative bias with the physical unit of a conductance. It can be used to adjust the global coupling strength between oscillators, refer to the discussion at the end of Sect. 2.2. As can be seen, the extreme cases $J_{\mu\nu} = 1$ and $J_{\mu\nu} = -1$ lead to $G_{\mu\nu}^+ = 0$ and $G_{\mu\nu}^- = 0$, respectively. In the first case, we promote in-phase synchronization, since we effectively obtain the coupling circuit on the left side of Fig. 2. In the second case, we are promoting an anti-phase tendency, as we effectively have the coupling circuit on depicted on the right side of Fig. 2. For all values of $J_{\mu\nu}^2 < 1$, we are giving a preference for one of the two synchronization states. For $J_{\mu\nu} > 0$ we are encoding a preference for in-phase synchronization:

$$J_{\mu\nu} > 0 \Rightarrow G_{\mu\nu}^- > G_{\mu\nu}^+.$$

This allows more current to flow through the coupling conductances $G_{\mu\nu}^-$, hence giving a preference for in-phase synchronization. Evidently, in the other case, we are giving a preference for anti-phase synchronization. Thus, we have now developed a concept for mapping optimization coefficients onto the coupling circuit of our network.

2.3.2 Linear optimization coefficients

The Ising Hamiltonian (1b) contains two terms. In Sect. 2.3 we have covered how the quadratic term $\mathbf{s}^T \mathbf{J} \mathbf{s}$ can be mapped onto the oscillator network at hand.

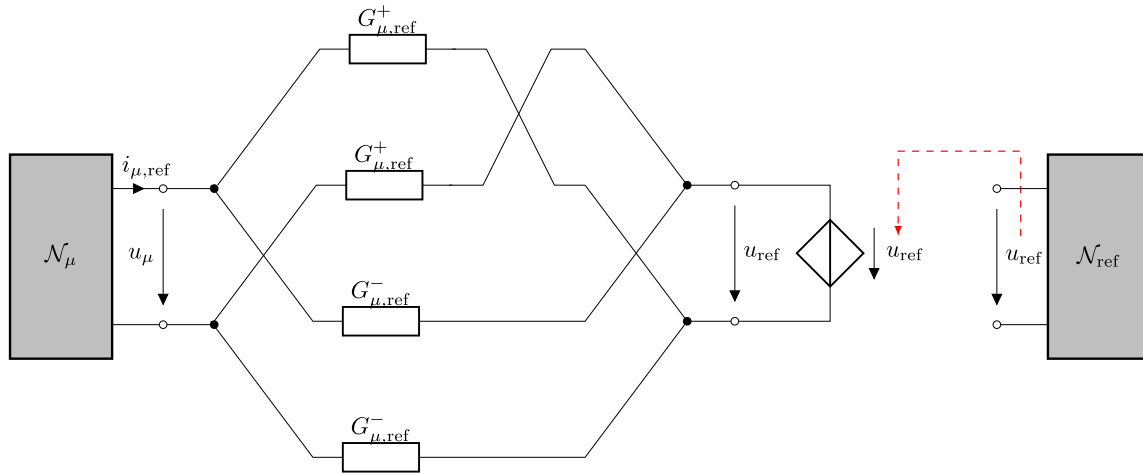


Fig. 4 Circuit for encoding linear optimization coefficients. The voltage u_{ref} denotes the voltage of a reference oscillator \mathcal{N}_{ref} . The latter is (unidirectionally) resistively coupled to every oscillator \mathcal{N}_μ via a controlled voltage source

In this section, we discuss, how the second term $\mathbf{h}^T \mathbf{s}$, sometimes referred to as the Zeeman term [24], can be mapped to the network as well. To this end, we start out by rewriting the Ising Hamiltonian (1b) as:

$$H(\mathbf{s}) = \mathbf{s}_e^T \mathbf{J}_e \mathbf{s}_e, \quad \mathbf{J}_e = \begin{bmatrix} \mathbf{J} & \mathbf{h}/2 \\ \mathbf{h}^T/2 & 0 \end{bmatrix}, \quad \mathbf{s}_e = \begin{bmatrix} \mathbf{s} \\ 1 \end{bmatrix}. \tag{6}$$

Interpreting this equation, we see that a linear term $\mathbf{h}^T \mathbf{s}$ can be treated just like a quadratic one in case a slack variable with a fixed value $s_{\text{ref}} = 1$ is appended to the vector of spins \mathbf{s} such that:

$$\mathbf{s}_e = [\mathbf{s}^T \ s_{\text{ref}}]^T, \quad \text{where } s_{\text{ref}} = 1.$$

In terms of our networks this means that we must expand it by one oscillator \mathcal{N}_{ref} with a fixed phase and couple that oscillator to the μ -th oscillator in case $h_\mu \neq 0$. The sign of h_μ determines whether we must give a preference to in- or anti-phase synchronization. This type of coupling can be realized by the coupling circuit depicted in Fig. 4, where we make use of a controlled voltage source to couple \mathcal{N}_{ref} to the μ -th oscillator \mathcal{N}_μ . The advantage of this coupling scheme is that \mathcal{N}_{ref} does not perceive any feedback or load from its coupling with the other oscillators \mathcal{N}_μ . Removing feedback is important so that the phase of \mathcal{N}_{ref} is kept fixed. Besides the controlled voltage source, the coupling is resistive and identical to the one in Fig. 3. Here, the coupling current $i_{\mu,\text{ref}}$ in Fig. 4 is given by

$$i_{\mu,\text{ref}} = \frac{G_{\mu,\text{ref}}^-}{2} [u_\mu - u_{\text{ref}}] + \frac{G_{\mu,\text{ref}}^+}{2} [u_\mu + u_{\text{ref}}], \tag{7a}$$

which is essentially identical to the coupling current in (5). The coupling conductances $G_{\mu,\text{ref}}^\pm$ can be calculated in a similar fashion as (5c):

$$G_{\mu,\text{ref}}^\pm = \frac{G_b}{2} \begin{cases} 1 \mp h_\mu/2, & h_\mu \neq 0 \\ 0, & h_\mu = 0 \end{cases}. \tag{7b}$$

Here, we are implicitly working with normalized coefficients $h_\mu \in [-1; 1]$ similar to Sect. 2.3. Note, it is important to normalize both the entries \mathbf{J} and \mathbf{h} by the same value. In other words, we are normalizing the entries of \mathbf{J}_e from (6). Thus, we are now able to fully map a QUBO problem of the form (1) to our oscillator network.

2.4 Optimization constraints

The most fundamental property of QUBOs, that makes them especially difficult to deal with, is their discrete solution space. In other words, our optimization variables x_ν are only allowed to assume two discrete values $x_\nu \in \{0, 1\}$. Although we have shown that two oscillators can be brought to a state of agreement or disagreement, which resembles the two discrete states of a QUBO problem, we have yet to discuss whether a network composed of more than two oscillators would also naturally (only) assume these two discrete states. In fact, since our hypothesis only states that coupled oscillators attempt to minimize dissipated power, there may be other phase configurations besides the two discrete states of agreement/disagreement that minimize the dissipated power even more.

To understand this statement, consider the network depicted on the left of Fig. 5, which presents an oscillator network composed of three oscillators. Here, the oscillators are coupled via the circuit presented in Fig. 3. The optimization task is assumed to have the coefficients $J_{\mu\nu} = -1$. This way, the coupling circuit of Fig. 3 degenerates to the one presented on the right side of Fig. 2. This type of coupling imposes all three oscillators to assume an anti-phase configuration, which, in this case, leads to a contradiction. The solution of the underlying NP problem is for two (arbitrary) oscillators to be in-phase, while the last one is in anti-phase w.r.t.

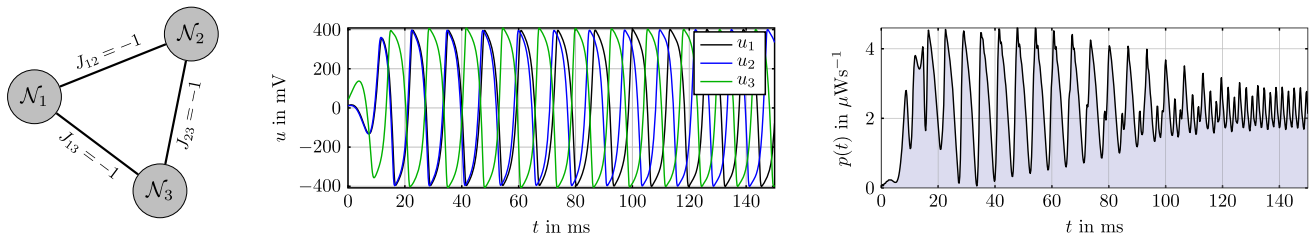


Fig. 5 Left: three coupled oscillators solving an optimization problem with the coefficients $J_{\mu\nu} = -1$. Center: output voltage u_μ of the oscillators over time. Right: dissipated power $p(t)$ over time

the other two. The oscillator voltages u_μ are presented in the center of Fig. 5. Here, we see that the network is able to find the actual solution of the problem for $20\text{ ms} < t < 80\text{ ms}$. However, it eventually leaves this state for $t > 80\text{ ms}$ and assumes a three-phase configuration. To interpret this behavior, we have plotted the (overall) power,

$$p(t) = \sum_{\mu < \nu} p_{\mu\nu}(t), \quad p_{\mu\nu}(t) = G_{\mu\nu}^- [u_\mu(t) + u_\nu(t)]^2, \tag{8}$$

dissipated by the coupling network on the right side of Fig. 5. Here, we see that the three-phase configuration ($t > 110\text{ms}$) leads to even lower power dissipation than the solution of the underlying NP problem.

This simple example shows us one important property of oscillator-based optimizers. Even though they tend to naturally minimize the dissipated energy $p(t)$, there is no guarantee that the minimum of $p(t)$ coincides with the minimum of (1b). In the following subsection, we will demonstrate how we can enforce binary constraints such that the oscillators are forced to achieve either in-phase or anti-phase synchronization.

Since we have shown the general problem with performing oscillator-based optimization without binary constraints, we now shed some light on how oscillator phases can be binarized. Here, we present the general methodology for introducing this kind of constraint and present a method that can be used to check, whether the constraints are enforced effectively.

The basic idea for applying a binarization constraint is to perturb the oscillator by an external sinusoidal signal, whose frequency is approximately double that of the oscillator’s eigenfrequency. The eigenfrequency f_0 is defined as the inverse of the oscillation’s period T_0 , i.e. $f_0 = 1/T_0$. This type of perturbation is referred to as subharmonic injection locking (SHIL) [42, 43] and has, for example, been shown to lead to bi-stable phase behavior in coupled metronomes [44] or even in general LC oscillators [43]. For relaxation-type oscillators, whose output voltages are generated by charging and discharging a capacitor, a perturbation can be achieved by injecting a source current at the node of the capacitor. In this work, we have already pre-built an external perturbation source e_p into our oscillator, see Fig. 1, which can be used to inject the SHIL signal defined as:

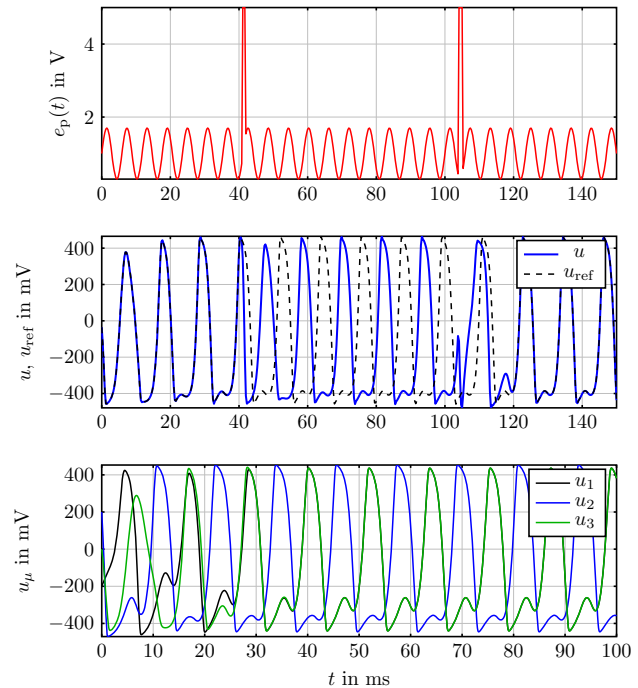


Fig. 6 Top: perturbation voltage $e_p(t)$, which is composed of SHIL and short voltage pulses. Center: oscillator output voltage u after being perturbed by the top signal (dashed black) and a reference oscillation (blue). Bottom: output voltages u_μ when solving the problem depicted in Fig. 5 but with SHIL

Table 2 SHIL parameters that have been used for obtaining the results in Fig. 6

Network parameters			
$\hat{e}_{\text{SHIL}} = 700\text{ mV}$	$\hat{e}_b = 1\text{ V}$	$T_0 = 11.7\text{ ms}$	$R_b = 70\text{ k}\Omega$

$$e_{\text{SHIL}}(t) = \hat{e}_{\text{SHIL}} \sin(2\Omega_0 t) + \hat{e}_b, \quad \Omega_0 = 2\pi/T_0. \tag{9}$$

Here, \hat{e}_{SHIL} denotes the amplitude of the SHIL signal, while \hat{e}_b is a bias voltage relating to the strength of the SHIL signal. Both these values must be chosen in a suitable manner in order to observe any effect.

Let us now consider the results depicted in Fig. 6 based on the parameters in Table 2. The top plot shows the perturbation signal $e_p(t)$. Here, $e_p(t)$ is by a composition of the SHIL signal e_{SHIL} defined in (9) and two short voltage pulses that are intended to demonstrate the bi-stable phase behavior. The central plot presents the oscillator’s output voltage (dashed black) against a reference signal (blue) that is perturbed by SHIL but not by the two voltage pulses. This plot illustrates the influence of SHIL on the oscillator’s output:

1. $t < 40$ ms: When an oscillator is perturbed by SHIL but not by any other external signal, we can observe a slightly distorted version of the oscillator’s output compared to the case when it is not perturbed by SHIL, compare Figs. 2 and 6.
2. $40 \text{ ms} < t < 110$ ms: When the oscillator is perturbed by an external signal, in this case a short voltage pulse, we see a phase jump of 180° .
3. $t > 110$ ms: When the oscillator is, again, disturbed by a short voltage pulse, the phase of the voltage again jumps by 180° , such that it effectively returns to case 1.

We would like to highlight that this experiment also serves as a guideline for testing SHIL in hardware. Essentially, we can try to replicate these three cases in hardware in order to calibrate the SHIL signal for our specific setup.

Using the SHIL signal, we solved the same problem depicted in Fig. 5. Our results are presented at the bottom of Fig. 6. Here, we see that SHIL forces the oscillator network to seek out a phase configuration from the discrete solution space of the QUBO problem. Indeed, we obtain the optimal solution for the mapped problem. Thus, we conclude that SHIL can be used to enforce binary constraints on the oscillator network.

2.5 Compact electrical model

In this section, we provide a compact electrical model describing a network composed of $n + 1$ oscillators solving QUBO problems of the form (1). This model serves as the fundamental for the wave digital model derived in the next section and can also be used to simulate the network with standard numerical integrators. To this end, we start by introducing the state voltage and current vector

$$\mathbf{u} = [u_1 \dots u_n u_{\text{ref}}]^T \quad \text{and} \quad \mathbf{i} = [i_1 \dots i_n i_{\text{ref}}]^T \tag{10a}$$

comprising the capacitor voltages and inductor current of every oscillator, respectively. The $[n + 1]$ -th entry of each vector represents the quantities of the reference oscillator \mathcal{N}_{ref} . Hence, our network is composed of n oscillators representing the problem and one reference oscillator for implementing linear optimization coefficients. Similar to the two state vectors, we also intro-

duce a vector of perturbation voltages and a vector of perturbation currents, given by:

$$\mathbf{e}_p = e_p \mathbf{1} \quad \text{and} \quad \mathbf{i}_p = [i_{p,1} \dots i_{p,n} i_{p,\text{ref}}]^T. \tag{10b}$$

The vector \mathbf{e}_p is a scaled vector of ones, because we are assuming that only one voltage source is used to perturb with an SHIL signal for the sake of efficiency. Furthermore, we introduce a vector-valued function $\mathbf{i}_n(\mathbf{u}) : \mathbb{R}^{n+1} \mapsto \mathbb{R}^{n+1}$, defined as:

$$\mathbf{i}_n(\mathbf{u}) = [i_n(u_1) \dots i_n(u_n) i_n(u_{\text{ref}})]^T. \tag{10c}$$

This function evaluates the cubic nonlinearity (2c) for every entry of \mathbf{u} and returns a corresponding $[n + 1]$ -dimensional vector. To accommodate for all the introduced vectors, we now introduce the following parameter matrices:

$$\mathbf{C} = \text{diag}(C_1, \dots, C_n, C_{\text{ref}}), \tag{10d}$$

$$\mathbf{L} = \text{diag}(L_1, \dots, L_n, L_{\text{ref}}), \tag{10e}$$

$$\mathbf{R}_0 = \text{diag}(R_{0,1}, \dots, R_{0,n}, R_{0,\text{ref}}), \tag{10f}$$

$$\mathbf{R}_p = \text{diag}(R_{p,1}, \dots, R_{p,n}, R_{p,\text{ref}}), \tag{10g}$$

where $\text{diag}(\cdot)$ denotes the diag operator. The μ -th diagonal entry of each of these matrices represents the electrical parameter of the μ -th oscillator. Thus, if all oscillators are assumed to be identical, they simplify to

$$\mathbf{C} = C\mathbf{1}, \quad \mathbf{L} = L\mathbf{1}, \quad \mathbf{R}_0 = R_0\mathbf{1}, \quad \mathbf{R}_p = R_p\mathbf{1}, \tag{10h}$$

where $\mathbf{1}$ denotes the unit matrix. Although not necessary, we will work with the assumption of identical oscillators in the following section. Using all introduced quantities, we can now rewrite (2) for the case of $n + 1$ coupled oscillators:

$$\mathbf{C} \frac{d\mathbf{u}}{dt} = \mathbf{i}_p(\mathbf{u}) - \mathbf{i} - \mathbf{i}_n(\mathbf{u}) - \mathbf{i}_c, \tag{11a}$$

$$\mathbf{L} \frac{d\mathbf{i}}{dt} = \mathbf{u} - \mathbf{R}_0 \mathbf{i}, \tag{11b}$$

$$\mathbf{R}_p \mathbf{i}_p(\mathbf{u}) = \mathbf{e}_p - \mathbf{u}. \tag{11c}$$

Here, we make use of a vector of coupling currents

$$\mathbf{i}_c = [i_{c,1} \dots i_{c,n} i_{c,\text{ref}}]^T, \tag{11d}$$

whose entries are given by

$$i_{c,\mu} = \frac{1}{2} \sum_{\mu,\nu} G_{\mu\nu}^+ [u_\mu + u_\nu] + G_{\mu\nu}^- [u_\mu - u_\nu], \tag{11e}$$

for every pair of coupled oscillator with the index μ and ν . Using the two matrices $\mathbf{G}^\pm = [\mathbf{G}^\pm]^T \in \mathbb{R}^{n \times n}$,

with the entries $G_{\mu\nu}^{\pm}$ defined in (5c), and the vectors $\mathbf{g}_{\text{ref}}^{\pm} \in \mathbb{R}^n$ with the entries $G_{\mu,\text{ref}}^{\pm}$ defined in (7b), we introduce the coupling matrices

$$\mathbf{G}_c^{\pm} = \frac{1}{2} \begin{bmatrix} \mathbf{G}^{\pm} & \mathbf{g}_{\text{ref}}^{\pm} \\ \mathbf{0}^T & 0 \end{bmatrix}, \tag{11f}$$

where $\mathbf{0}$ denotes a vector of zeros. Here, the unidirectional coupling depicted in Fig. 4 is reflected by the last zero row. Using these coupling matrices, we can now give a compact formulation for the coupling relation:

$$\begin{aligned} \mathbf{i}_c &= \mathbf{G}_c \mathbf{u}, \\ \mathbf{G}_c &= [\text{diag}(\mathbf{G}_c^+ \mathbb{1}) + \mathbf{G}_c^+] + [\text{diag}(\mathbf{G}_c^- \mathbb{1}) - \mathbf{G}_c^-]. \end{aligned} \tag{11g}$$

3 Wave digital emulation

This section utilizes the compact electrical model presented in Sect. 2.5 to derive a real-time capable wave digital (WD) algorithm for emulation purposes. This algorithm serves as a tool for deriving parameters for the optimizer’s technical implementation, which we discuss in the next section. It can therefore be used for designing and testing the optimizer prior to its actual production in order to seek out potential problems or derive suitable parameters. This section is divided into three subsections. The first subsection briefly recapitulates the WD concept. The second subsection derives a highly parallel WD algorithm for emulation purposes. Finally, the third subsection presents emulation results, where we have solved NP problems using the derived WD algorithm.

3.1 Wave digital concept

The WD concept is a powerful tool for the emulation of electrical circuits [29,31]. The idea lies in mapping a reference circuit onto a recursive digital structure known as the WD model. The platform-independent WD algorithm is then obtained from iteratively emulating the latter. To derive the WD model, we start by decomposing the reference circuit into a set of one- and multi-ports. A port is defined as a pair of terminals, where the in-flowing current is equal to the out-flowing current. It is characterized by three quantities, the port voltage u , the port current i , and the port resistance $R > 0$. A general m -port is assigned m port resistances $R_{\mu} > 0$ and governed by a constitutive relationship relating the port voltage u_{μ} to the port current i_{μ} at the μ -th port, where $\mu = 1, \dots, m$. If we comprise all port voltages u_{μ} , port currents i_{μ} , and port resistances R_{μ} into a voltage vector $\mathbf{u} \in \mathbb{R}^m$, a current vector $\mathbf{i} \in \mathbb{R}^m$, and a port resistance matrix $\mathbf{R} = \text{diag}(R_{\mu}) \in \mathbb{R}^{m \times m}$, respectively, we can translate any m -port into the WD domain by applying the bijective transformation:

$$\begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix} = \begin{bmatrix} \mathbf{1} & \mathbf{R} \\ \mathbf{1} & -\mathbf{R} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{i} \end{bmatrix} \Leftrightarrow \begin{bmatrix} \mathbf{u} \\ \mathbf{i} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \mathbf{1} & \mathbf{1} \\ \mathbf{G} & -\mathbf{G} \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \tag{12}$$

where $\mathbf{R}\mathbf{G} = \mathbf{1}$. In the WD domain, the port-wise structure of a circuit is retained. However, electrical m -ports translate to WD structures, where voltages u_{μ} and currents i_{μ} are replaced by incident waves a_{μ} and reflected waves b_{μ} . Here, a suitable choice of the port resistance R_{μ} can greatly simplify the resulting WD structure. Now, we comprise all wave quantities a_{μ} and b_{μ} into a vector of incident wave $\mathbf{a} \in \mathbb{R}^m$ and a vector of reflected waves $\mathbf{b} \in \mathbb{R}^m$, respectively. The constitutive equation relating \mathbf{u} and \mathbf{i} in the electrical domain translates to a scattering equation,

$$\mathbf{b} = \mathbf{S}\mathbf{a}, \quad \text{with } \mathbf{S} \in \mathbb{R}^{m \times m}, \tag{13}$$

relating the reflected wave vector \mathbf{b} to the incident wave vector \mathbf{a} in the case of a memoryless m -port. Electrical components with memory, such as capacitors or inductors, are governed by differential equations. These elements must first be discretized via a numerical integration method [29] before the bijective transformation (12) is applied. Here, we make use of the trapezoidal rule [29].

3.2 Wave digital model of oscillator-based optimizer

The electrical model derived in Sect. 2.5 can be equivalently represented by the port-wise decomposed circuit on the left side of Fig. 7. Let us briefly assume $\mathbf{i}_c = \mathbf{0}$. In this case, the circuit on the left side of Fig. 7 would describe n uncoupled FNOs, see Fig. 1. Hence, this circuit can be (virtually) thought of as a vector of FNOs and we generally refer to it as a vector-valued circuit. To allow for an interaction between the entries of this virtual vector, we incorporate the coupling relation defined in (11g), which leads to the multi-port resistor associated with the conductance matrix \mathbf{G}_c (highlighted in orange on the left side of Fig. 7). Translating our vector-valued circuit into the WD domain will also result in a vector-valued WD model, which allows for a massive parallelization of the associated WD algorithm. We refer the interested reader to [32] for a great exposition on the parallelization of WD algorithms.

In order to obtain a WD model, the general procedure is to individually translate each electrical component into the WD domain and interconnect them according to the topology of the reference circuit via so-called WD adaptors. These are signal flow diagrams that result from translating Kirchhoff interconnections into the WD domain. For example, a series (parallel) connection translates to a series (parallel) adaptor, which is symbolized by the squared box containing the \dashv (\vDash) symbol in Fig. 7.

A common phenomenon occurring in WD algorithms are so-called directed delay-free loops. These are implicit relationships, which hinder the evaluation of

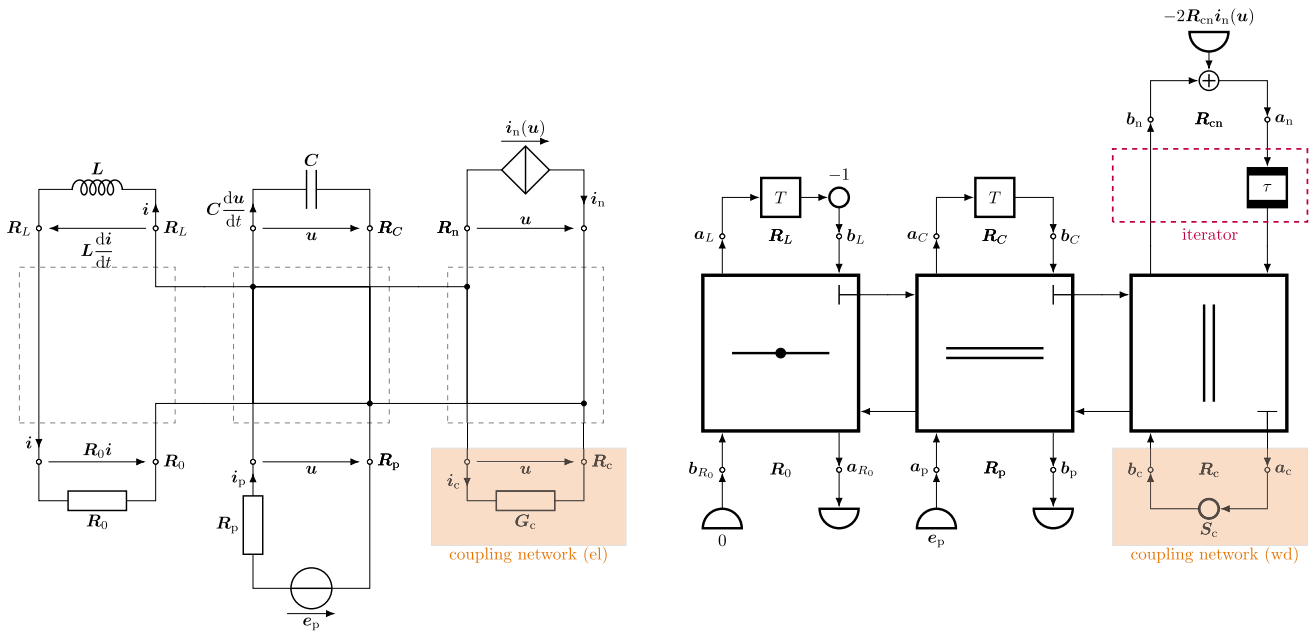


Fig. 7 Left: a port-wise decomposed equivalent circuit for the electrical model given in (11). For $i_c = \mathbf{0}$, the circuit represents n uncoupled FNOs, see Fig. 1. A multi-port resistor associated with the conductance matrix \mathbf{G}_c is used to represent the resistive coupling between oscillators (orange box). Right: corresponding wave digital model. The coupling network translates to a scattering relation in the wave digital domain (orange box). The block with the letter τ (purple dashed box) represents an iterator for solving (algebraic) directed delay-free loops

the WD model. There are generally two ways to deal with such loops depending on their type:

1. Topological loops occur when a reflected wave depends on its corresponding incident wave without any delay in between. To resolve such loops, we can make use of reflection-free ports [29]. Every WD adaptor is allowed to have exactly one reflection-free port, which is indicated by a T-shaped symbol in Fig. 7. This essentially eliminates the relationship between the incident and reflected wave (at the reflection-free port) but has the restriction that the corresponding port resistance is determined as a function of the other port resistances.
2. Algebraic loops occur due to the presence of algebraic dependencies that make the incident wave depend on its corresponding reflected wave with no delay separating the two; usually this type of loop occurs when dealing with nonlinear elements [45]. Here, we can make use of iterative methods, such as a fixed-point iteration, to solve such implicit relationships in the WD model, cf. [45].

Although, we have yet to discuss our WD model, we would like to mention that we have already optimally distributed reflection-free ports in Fig. 7 in order to avoid topological loops. To achieve this, we have made every port interconnecting two adaptors reflection-free from left to right. Moreover, we have inserted an iterator element, symbolized by the box containing the letter τ , in order to deal with an algebraic loop that we cover at the end of this subsection.

Let us now individually translate all electrical components into the WD domain, cf. [29].

1. The inductors and capacitors associated with the inductance matrix \mathbf{L} and the capacitance matrix \mathbf{C} translate to a delay element with and without a sign inversion, respectively. Their port resistance matrices are given by:

$$\mathbf{R}_C = \frac{T}{2} \mathbf{C}^{-1} \quad \text{and} \quad \mathbf{R}_L = \frac{2}{T} \mathbf{L}, \quad (14)$$

- where T denotes the sampling period of the WD algorithm.
2. Real voltage sources generally translate to reflective wave sources. However, in case their port resistances can be chosen freely, they can be made non-reflective. We have exploited this degree of freedom for the perturbation source with the supply voltage e_p . Hence, it translates to a wave source supplying with the incident wave $\mathbf{a}_p = \mathbf{e}_p$ in Fig. 7.
3. Resistances can be thought of as degenerate cases of real voltage sources, where the supply voltage is zero. Thus, they also generally translate to reflective wave sources in the WD domain. In the case of the resistances associated with the resistance matrix \mathbf{R}_0 , we have also matched their port resistances matrix to their resistance value, such that they translates to a wave source supplying the reflected wave $\mathbf{b}_{R_0} = \mathbf{0}$.
4. The coupling relation of the multi-port resistor (11g) is associated with the conductance matrix \mathbf{G}_c , see the orange box on the left side of Fig. 7. The consti-

tutive relation of a general multi-port resistor translates to a simple scattering equation (13). In this case, we have:

$$\mathbf{b}_c = \mathbf{S}_c \mathbf{a}_c, \quad \text{with} \quad (15a)$$

$$\mathbf{S}_c = [\mathbf{1} + \mathbf{G}_c \mathbf{R}_c]^{-1} [\mathbf{1} - \mathbf{G}_c \mathbf{R}_c] \quad \text{and} \quad (15b)$$

$$\mathbf{R}_c = [\mathbf{R}_n^{-1} + [\mathbf{R}_p^{-1} + \mathbf{R}_C^{-1}]]^{-1}. \quad (15c)$$

Here, (15c) is a consequence of our choice of reflection-free ports.

It remains to translate the cubic nonlinearity $i_n(u)$ into the WD domain. Here, it is modeled as an ideal controlled current source, see the left side of Fig. 7. This is necessary, because the nonlinearity is formulated in terms of an (i, u) -curve and not in terms of a varying resistance. Ideal current sources translate to reflective wave sources in the WD domain, which, in this case, is governed by the following relation:

$$\mathbf{a}_n = \mathbf{b}_n - 2\mathbf{R}_n \mathbf{i}_n(\mathbf{u}), \quad \text{with} \quad \mathbf{u} = \frac{\mathbf{a}_n + \mathbf{b}_n}{2}. \quad (16)$$

This relationship is implicit, because the calculation of \mathbf{a}_n depends on the voltage vector \mathbf{u} , which is a function of \mathbf{a}_n itself. This type of implicit relationship is what is referred to as an algebraic loop [45]. To resolve this implicitness, we make use of a fixed-point iteration to approximate the value of \mathbf{a}_n at every time instant. The use of an iteration method is indicated by the iterator symbol on the right handside of Fig. 7.

3.3 Emulation results and discussion

In this subsection, we present emulation results where we have solved two optimization problems utilizing the wave digital model derived in the previous subsection. All our computations were performed on a lab computer in MATLAB. The first problem is the so-called max-cut problem, which asks us to partition a given graph into two subgraphs, such that the number of edges connecting the two graphs is maximal. According to [9, 15, 24], the max-cut problem can be mapped to the Ising Hamiltonian by using the coefficients:

$$\mathbf{J} = -\mathbf{A} \quad \text{and} \quad \mathbf{h} = \mathbf{0}, \quad (17)$$

where \mathbf{A} denotes the adjacency matrix of the underlying graph. We will not cover how these coefficients can be derived from the baseline formulation of the optimization problem, but refer the interested reader to [15, 24] for a good exposition on this topic. Besides the max-cut problem, we solve the so-called minimal vertex cover problem. This problem asks us to find the minimal number of vertices that need to be marked, such that every edge in the graph is incident to one of the marked vertices.

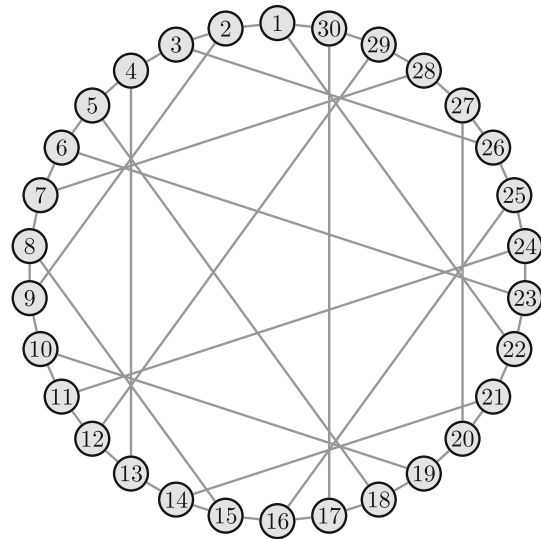


Fig. 8 The Tutte–Coxeter graph [46]

To map this problem onto the Ising machine, we choose the coefficients:

$$\mathbf{J} = -\mathbf{A} \quad \text{and} \quad \mathbf{h} = [2\mathbf{A} - 4\mathbf{1}] \mathbf{1}. \quad (18)$$

Contrary to the max-cut problem, the minimal vertex cover problem also requires a vector of linear optimization coefficients \mathbf{h} . We make use of this problem to prove the validity of our mapping approach, when it comes to linear optimization coefficients. Note, both problems are based on the graph depicted in Fig. 8. Also, we have used the oscillator parameters provided in Table 1 and the network parameters provided in Table 3 for both problems.

Our emulation results are depicted in Fig. 9. The top row presents our results for the max-cut problem, while the bottom row presents our results for the minimal vertex cover problem. Let us first consider the max-cut problem and hence the first row of Fig. 9. The left plot depicts the oscillator voltages u_μ over time. Here, we see that the oscillator voltages quickly converge to steady-state phase configuration encoding the solution of the problem after just about 20 ms. In the center, we have attempted to extract the oscillation phases denoted by φ_μ . This plot clearly demonstrates the influence of the SHIL signal, which enforces a bi-stable phase behavior with a phase shift of π . The right plot uses the phases φ_μ to approximate the cut size over time. Here, a phase of $\varphi_\mu = 0$ is associated with a spin value of $s_\mu = 1$, while a phase of $\varphi_\mu = \pi$ is associated with a spin value of $s_\mu = -1$. The graph in Fig. 8 has a maximal cut of 45 edges. As can be seen, the max-cut is successfully determined by the optimizer after just about 10 ms.

The Ising Hamiltonian encoding the minimal vertex cover problem consists of two terms [15, 24]:

$$H = H_0 + H_1. \quad (19)$$

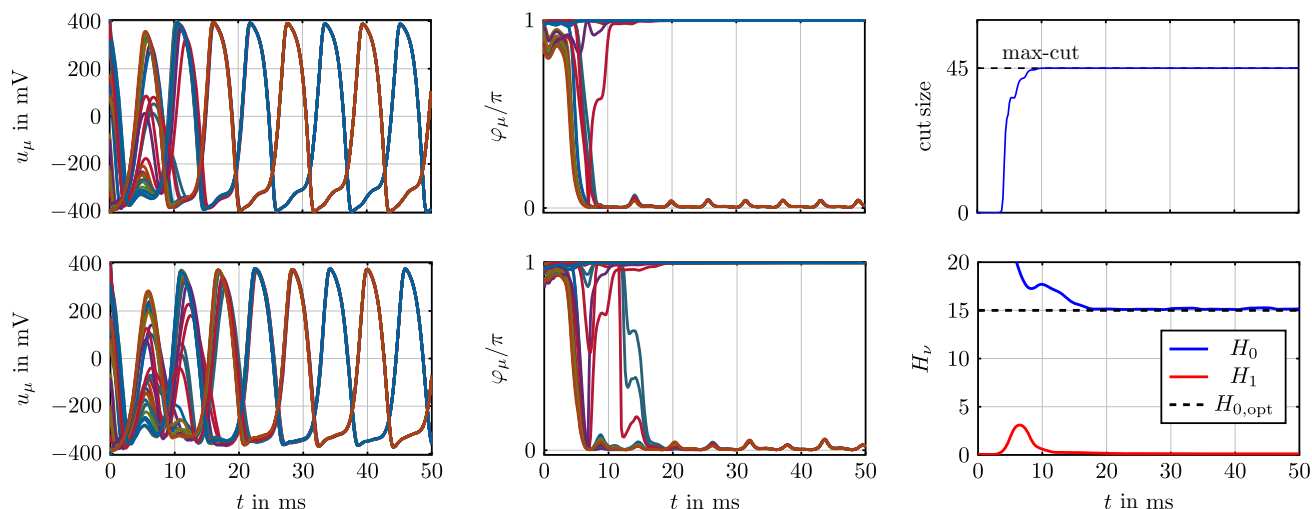


Fig. 9 Emulation results of the max-cut (top) and vertex cover (bottom) problem belonging to the graph in Fig. 8. Left column: oscillation voltages u_μ over time. Central column: approximation of oscillation phases φ_μ over time. Right column: cut size over time with reference to the max-cut (top) and Hamiltonian belonging to vertex cover problem over time (bottom)

The first term $H_0 \geq 0$ represents a counter that counts the number of vertices that have been marked by the optimizer. The second term $H_1 \geq 0$ is a penalty term. This term can only reach zero if all edges are incident to a vertex from the solution set. Thus, H is only minimized, when a valid vertex cover is found (imposed by H_1) that is also minimal (imposed by H_0). Looking at the bottom row of Fig. 9, we see that the oscillator voltages, again, quickly converge to a steady-state phase configuration after just about 20 ms. By extracting the phase values φ_μ out of the oscillation voltage u_μ , we were able to depict the values of H_0 and H_1 over time in the right plot. Here, we also mark the size of the minimal vertex cover using a black dashed line, which, in this case, is 15. Both the Hamiltonian H_0 and H_1 converge to a constant value once the solution is found. Since a valid vertex cover is found, the penalty term H_1 converges to zero. Meanwhile, the counter H_0 converges to its optimal/minimal value $H_{0,\text{opt}} = 15$, which is the size of the minimal vertex cover.

Overall, we have shown that the oscillator network designed in this section indeed serves as an optimizer for QUBO problems. How different problems can be mapped to this optimizer is a question that has been extensively discussed in [15, 24]. Note that the wave digital model presented in the previous section can be used to simulate arbitrarily large network. In Appendix A, we demonstrate the runtime of the associated wave digital algorithm for different network sizes.

We would now like to briefly comment on one important aspect of our oscillator-based optimizer. Like every other optimizer, our device has a set of parameters that must be suitably chosen in order to obtain good solutions:

1. The multiplicative bias $G_b = 1/R_b$, appearing in (5c) 7b, needs to be chosen adequately, being neither too high nor too low. In the first case, the inter-

action is too strong, such that the oscillators are usually unable to settle to one phase configuration. In the second, case, the oscillators are effectively uncoupled, such that the power exchange between them is too weak for optimization purposes. Here, the network usually converges to a suboptimal solution. These two cases are illustrated in Fig. 10a, b.

2. The SHIL signal has two parameters, namely the amplitude \hat{e}_{SHIL} and the bias \hat{e}_b , see (9). Both these parameters need to be chosen adequately. When the amplitude and bias are too high, the oscillators tend to lock (or even get stuck) to their current phases. These parameters must be chosen so a perturbation can have the chance to influence the oscillator's phase, as shown in Sect. 2.4. The consequences of strong and weak SHIL are illustrated in Fig. 10c, d, respectively.
3. A good balance between the multiplicative bias and the SHIL signal must be found. At first glance, these parameters seem independent of each other. However, if, for example, the SHIL signal is too strong, then this effect can be mitigated by increasing G_b , as this would lead to a greater perturbation by adjacent oscillators that may be enough for the oscillator to escape its current phase. Conversely, if the multiplicative bias is too high, the interaction between the oscillators may be too strong such that the bi-stable phase behavior is lost. In this case, increasing SHIL may restore bi-stability.

The emulation parameters in Table 3, which have been used to solve the optimization tasks presented in this section, have been chosen on the basis of this discussion. Here, the wave digital algorithm was used as a flexible tool to test and find the correct parameters. In Sect. 4, this tool is also used to determine the parameters of the implemented circuit.

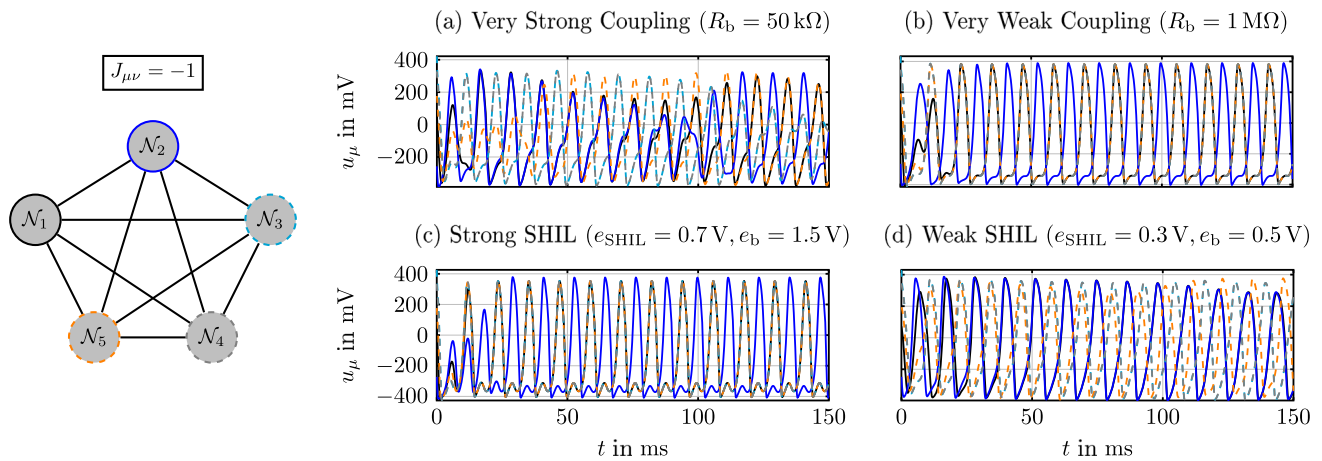


Fig. 10 Simulations of an oscillator-based optimizer solving the max-cut problem depicted on the left side. The solution of the problem is given by two partitions composed of two and three arbitrary oscillators, respectively. Plots **a**, **b** showcase the effects of using inadequate optimization parameters. **a** High values of G_b lead to a strong coupling, which does not allow the network to settle to one phase configuration. **b** Low values of G_b lead to the oscillators getting stuck in a certain phase configuration, usually determined by their initial values. **c** Similar to **b**, a strong SHIL signal leads to the oscillators getting stuck in a certain phase configuration and not correctly solving the problem. **d** Weak SHIL does not lead to bi-stable phase behavior, which (often) does not allow the network to settle to one binary phase configuration

Table 3 Emulation parameters that have been used for the results in Fig. 9

Emulation Parameters			
$\hat{e}_{SHIL} = 400 \text{ mV}$	$\hat{e}_b = 1 \text{ V}$	$T_0 = 11.7 \text{ ms}$	$R_b = 47 \text{ k}\Omega$

4 Technical implementation

The third and final part of this work discusses the technical implementation of our oscillator-based optimizer. This section is divided into three subsections. The first subsection discusses the technical implementation of the FNO. Here, we also present a method for enhancing the wave digital model of the oscillator, so it can emulate the real oscillator in a more accurate fashion. The second subsection discusses the different circuits used for problem mapping. This also includes the circuits required for setting optimization constraints. Finally, the third subsection provides measurements from our in-house oscillator-based optimizer, where we solve different small-sized optimization problems to validate the findings of this manuscript.

4.1 Oscillator circuit

Our FNO implementation is depicted in Fig. 11. The circuit is composed of the same two elements as in Fig. 1: the cubic nonlinearity and the RLC-tank. For the sake of efficiency, the perturbation source e_p is not embedded into the oscillator itself. Instead, we make use of one common signal generator for all oscillators in the network. We cover this aspect in more detail in the next section. The RLC-tank is composed of the

resistor R_0 , the capacitor C and inductor L just like in Fig. 1. However, the inductor is realized with the help of a capacitive gyrator. A gyrator is an ideal lossless two-port with the ability of inverting current-voltage characteristics [47]. With reference to Fig. 12, a capacitive gyrator is able to emulate an inductor described by:

$$L \frac{di}{dt} = u, \quad \text{with } L = R_g^2 C_g, \quad (20a)$$

where R_g is the so-called gyration resistance. Thus, a gyrator allows for emulating large inductances with small capacitors by adequately adjusting the gyration resistance. This is important for our oscillator implementation, because the oscillation frequency of the unperturbed FNO is mainly determined by the inductance L of its RLC-tank. The larger L is, the slower the frequency will be. Slow frequencies, on the other hand, are helpful for circuit testing, as they eliminate most parasitic high frequency effects. The capacitive gyrator in Fig. 11 has two resistances $R_{g,1}$ and $R_{g,2}$. These resistances relate to the gyration resistance in (20a) as follows [48]:

$$R_g = \sqrt{R_{g,1} R_{g,2}} \quad (20b)$$

Thus, for a desired inductance L , we require a capacitor

$$C_g = \frac{L}{R_g^2} = \frac{L}{R_{g,1} R_{g,2}}, \quad (20c)$$

where the latter equality is a direct consequence of (20b).

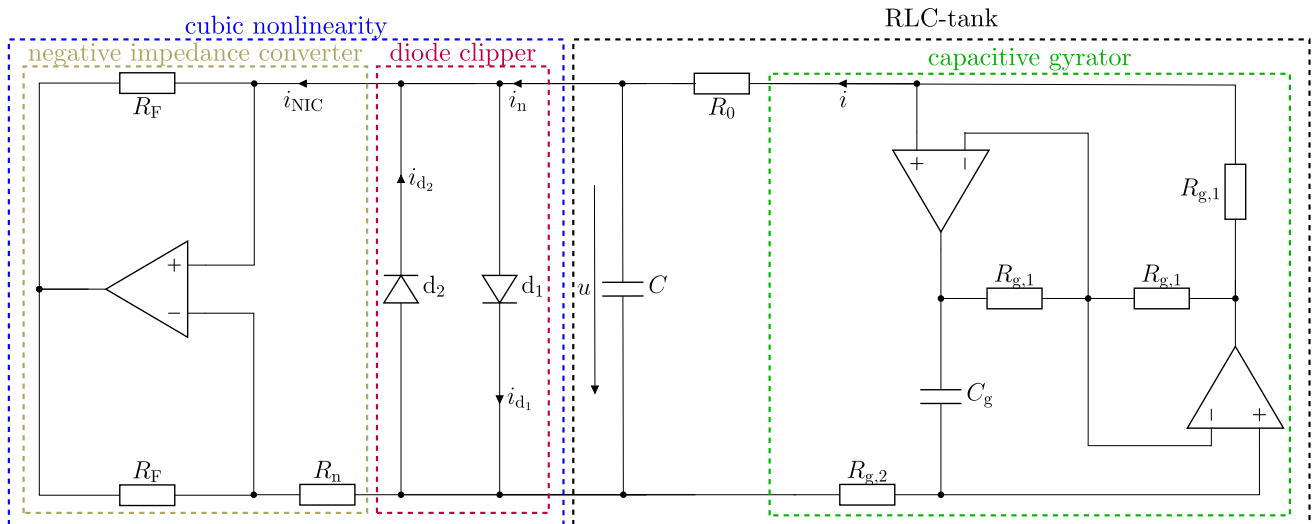


Fig. 11 Technical implementation of the FNO depicted in Fig. 1

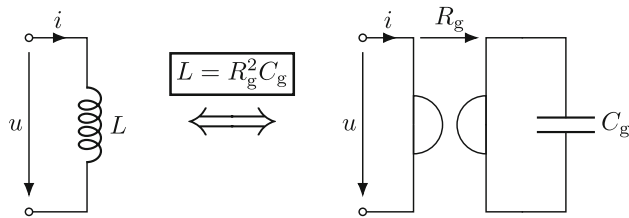


Fig. 12 An inductor can be realized by a capacitive gyrator

The cubic nonlinearity is realized by the parallel interconnection of a negative impedance converter (NIC) and a diode clipper, see the blue box in Fig. 11. The negative impedance converter is used to produce a negative resistance:

$$i_{NIC} = -G_n u, \quad \text{with } G_n R_n = 1. \quad (21a)$$

In combination with the diode clipper, we obtain

$$i_n(u) = i_{d_1}(u) - i_{d_2}(u) + i_{NIC}, \quad (21b)$$

where i_{d_ν} denotes the currents flowing through the diodes d_ν . The diode clipper effectively limits the amount of voltage that is perceived by the NIC. Combining the negative resistance of the latter with the rectification of the diode clipper, we obtain an N-shaped (i, u) -curve with the negative differential resistance R_n , see Fig. 14. Here, we see that the N-shaped nonlinearity is able to approximate the characteristics of the original cubic nonlinearity (2c).

To enhance our wave digital model, we suggest modeling the diodes by the Shockley equation

$$i_{d_{1/2}}(u) = I_s [\exp(\pm u/U_T) - 1], \quad (21c)$$

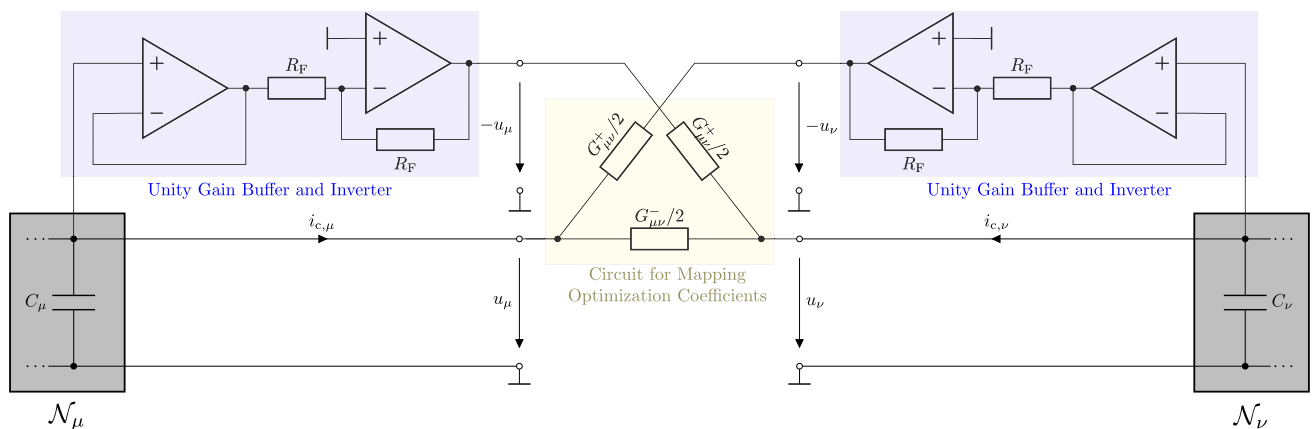


Fig. 13 Realization of the coupling depicted in Fig. 3 in hardware. The gray boxes titled \mathcal{N}_μ and \mathcal{N}_ν are used to compactly represent two FNOs just as in Fig. 11. Here, we only draw the capacitors to clearly demonstrate the coupling node. Both oscillators are equipped with a unity gain buffer and an inverter. The coupling circuit used in Fig. 3 can be found in the yellow box

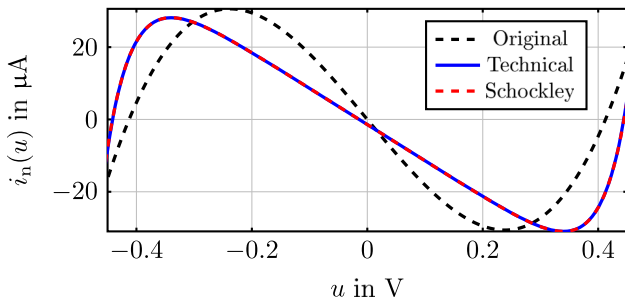


Fig. 14 Original cubic nonlinearity (2c) (dashed black), N-shaped nonlinearity produced by the circuit in Fig. 11 (solid blue), and the (i, u) -curve of the model (21) based on the fitted Shockley equation (dashed red)

Table 4 Parameters of the FNO circuit in Fig. 11

Oscillator circuit parameters		
$R_0 = 4.7 \text{ k}\Omega$	$C = 100 \text{ nF}$	$C_g = 4.7 \text{ }\mu\text{F}$
$R_n = 10 \text{ k}\Omega$	$R_{g,1} = 1 \text{ k}\Omega$	$R_{g,2} = 4.7 \text{ k}\Omega$
$R_F = 1 \text{ k}\Omega$		

with the scale current $I_s = 20\text{nA}$ and the thermal voltage constant $U_T = 51.8\text{mV}$. By fitting the latter, we were able to obtain good fit for our model (21) using the parameters in Table 4, see Fig. 14. Hence, for achieving more accurate results with the wave digital model of Sect. 3.2, we recommend replacing (2c) with (21) in the electrical model (11), which effectively modifies (16). For the sake of completeness, we would like to mention that our technical implementation uses the op-amp model TL071 and the diode model 1N4148.

4.2 Problem mapping

In this subsection, we demonstrate how the circuits presented in Sect. 2 translate to hardware in order to build an oscillator-based optimizer. The first part of this section discusses the mapping of optimization coefficients $J_{\mu\nu}$. The second part explains the injection of SHIL. The third and final part discusses the mapping of linear optimization coefficients.

4.2.1 Quadratic optimization coefficients

In order to map optimization coefficients onto our in-house optimizer, we make use of the circuit depicted in Fig. 13. Here, we see a coupling between two arbitrary oscillators \mathcal{N}_μ and \mathcal{N}_ν , which are (compactly) represented by the gray filled boxes with the capacitances C_μ and C_ν . Essentially, every oscillator can be considered to have two outputs ports, the first being the one parallel to the capacitor and the second being an inverted output. The latter is realized by interconnecting the upper capacitor node to a unity gain buffer and inverter, which invert the output voltages u_μ and u_ν , see the blue shaded boxes. In practice, these components are necessary, as all lower capacitor nodes are connected

to the ground, such that a signal inversion, like in Fig. 3, via a cross-over circuit is not possible. Using these two outputs, we can now interconnect the two oscillators in the same way as in Fig. 3, i.e. the non-inverted output of \mathcal{N}_μ is connected to the non-inverted output of \mathcal{N}_ν via a resistor with the conductance $G_{\mu\nu}^-$, while the inverted outputs of \mathcal{N}_μ and \mathcal{N}_ν are each cross-coupled to the non-inverted output via a resistor with the conductance $G_{\mu\nu}^+$. Contrary to the coupling in Fig. 3, the coupling conductances in Fig. 13 are halved in their value. This is because, the lower capacitor nodes in Fig. 3 are assumed to have different potentials, while the nodes in Fig. 13 are connected to the ground. This way, the two coupling schemes can only be identical, when the conductances are halved, such that the coupling currents $i_{c,\mu}$ and $i_{c,\nu}$ are the same as in (5).

4.2.2 Linear optimization coefficients

As we have seen in Fig. 4, the linear constraints can be essentially implemented in the same way as all other optimization coefficients. However, as explained in Sect. 2.3.2, the coupling must be unidirectional, i.e. the reference oscillator \mathcal{N}_{ref} must perceive no load/feedback from the rest of the circuit in order to retain a fixed phase. The simplest solution to eliminate feedback is to make use of a unity gain buffer, whose output voltage is then used to couple the reference oscillator to all other oscillators. We have depicted this type of coupling in Fig. 15 on the example of our reference oscillator \mathcal{N}_{ref} being coupled to one arbitrary oscillator \mathcal{N}_μ . The unity gain buffer is highlighted in red. As can be seen in this figure, both outputs of the reference oscillator are connected to a unity gain buffer, which decoupled it from the rest of the circuit. This way, \mathcal{N}_{ref} perceives no load allowing its output voltage u_{ref} to retain a fixed oscillation phase.

4.2.3 Optimization constraints

In order to implement binary constraints, we have used SHIL in Sect. 2.4. Here, SHIL current was injected using the perturbation source that we have pre-built into the oscillator model depicted in Fig. 1. However, from a practical standpoint, this type of implementation is inefficient, as it necessitates one signal generator per oscillator. To reduce the number of components, we suggest injecting SHIL by using a common signal generator, see Fig. 16. Here, one voltage source e_p is used to perturb all oscillators at the same time. The currents $i_{p,\nu}$ then result as a result of the current divider law.

4.3 Results and discussion

The set up for the measurements can be seen in Fig. 17. The printed circuit board (PCB) consists of 20 oscillators of which only the necessary amount is used for each measurement. The PCB is screwed onto an aluminum plate that is grounded to reduce the noise. To couple the oscillators additional bread boards are used with

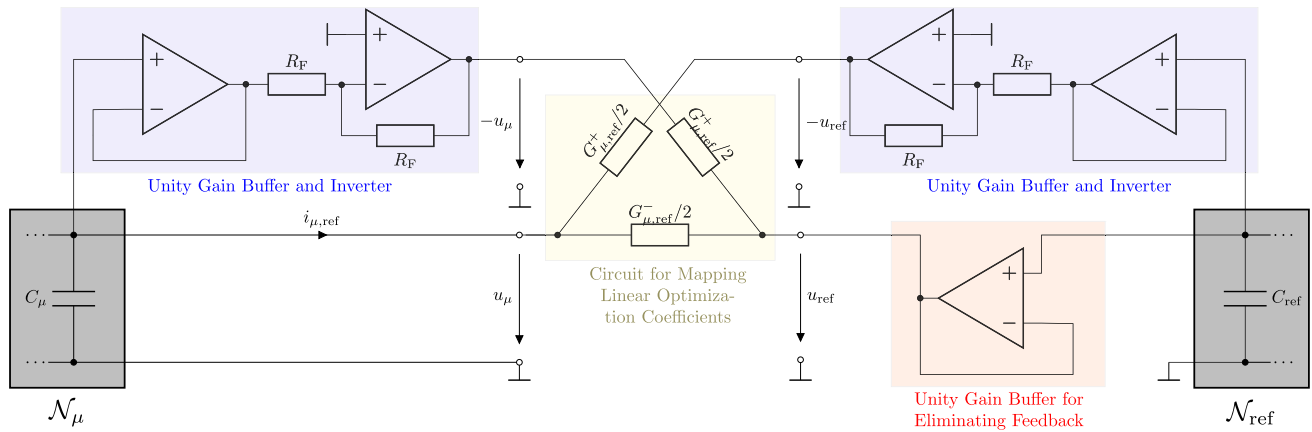


Fig. 15 Realization of the coupling depicted in Fig. 4 in hardware. The gray boxes titled \mathcal{N}_μ and \mathcal{N}_{ref} are used to compactly represent an arbitrary oscillator \mathcal{N}_μ and the reference oscillator \mathcal{N}_{ref} , respectively, just like in Fig. 4. The controlled voltage source in Fig. 4 is realized by the unity gain buffer highlighted in red, while the resistive coupling circuit is highlighted in yellow

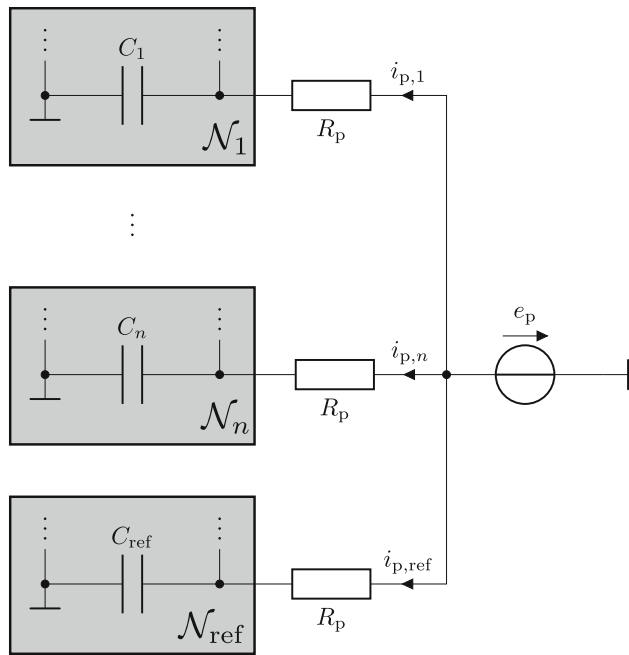


Fig. 16 Circuit schematic demonstrating SHIL injection at the capacitor node. The voltage source e_p represents a signal generator supplying (9). The gray dashed boxes entitled \mathcal{N}_μ are a compact version of the oscillators depicted in Fig. 11

variable resistances. The SHIL signal is supplied to the oscillators via a function generator (Keysight 33500B) and the oscillator voltages are measured with a Pico Scope and the corresponding computer program (Fig. 17).

With the laboratory setup in Fig. 17 and the parameters in Table 5, we have solved the four optimization problems depicted in the left column in Fig. 18. Here, every node \mathcal{N}_μ represents a FNO, as depicted in Fig. 11, while every edge represents a resistive connection as

depicted in Fig. 13. Problems (a) and (b) are used as simple benchmarks to reaffirm the discussions of Sects. 2.2 and 2.3. Problems (c) and (d), on the other hand, are small-sized examples for demonstrating the behavior of our oscillator-based optimizer. The middle column in Fig. 18 depicts the output voltages u_μ over time. Here, a red line is used to indicate the time instant at which SHIL is turned on. The right column in Fig. 18 shows the instantaneous power $p(t)$ over time. This signal is not measured but rather reconstructed by using the known coupling matrix G_c , the measured voltage vector $\mathbf{u}(t)$, and the coupling relation (11g):

$$p(t) = \mathbf{u}^T(t)\mathbf{i}_c(t) = \mathbf{u}^T(t)\mathbf{G}_c\mathbf{u}(t). \quad (22)$$

Note that all results in the central column of Fig. 18 represent extracts of our measurements, i.e. the time instant $t = 0$ only denotes the time at which the recording has been started.

Let us first consider problem (a) in Fig. 18, which is a simple max-cut problem with two coupled oscillators. According to (17), we have $J_{12} = -1$. Therefore, we use the coupling circuit depicted on the right side of Fig. 2 in order to promote anti-phase synchronization. Looking at the voltages u_μ in Fig. 18a, we observe that the oscillators achieve anti-phase synchronization even without the use of SHIL. According to our hypothesis, this outcome is natural, since anti-phase synchronization minimizes power dissipation, which is evident from the right plot in Fig. 18a. Once SHIL is turned on for $t > 60$ ms, we see an interesting phenomenon. Prior to turning on the SHIL signal, the voltage signals are symmetrical about the t -axis. This allows for a destructive superposition of the two signals, such that $u_1 + u_2 = 0$, when anti-phase synchronization is achieved; this also implies no power dissipation such that $p(t) = 0$. Once SHIL is turned on, we can observe a slight distortion in the voltage signals; they are now asymmetric about the t -axis. In this case, the two signals can not sum to

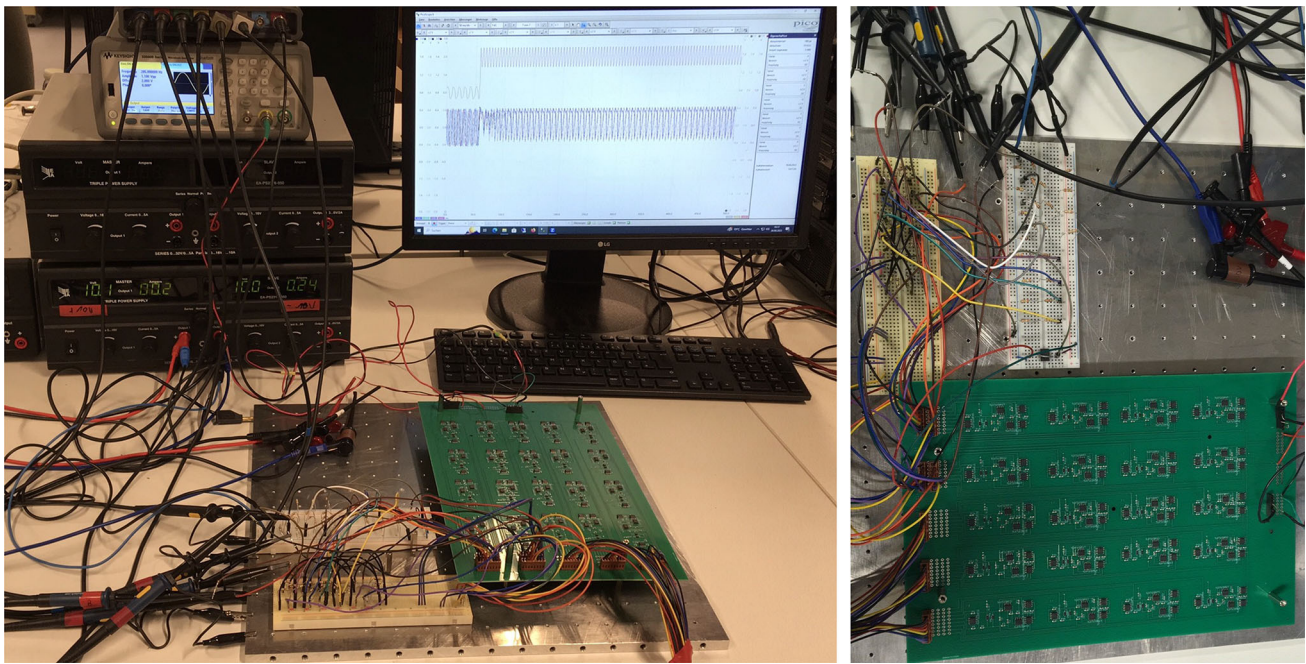


Fig. 17 Photograph of the experimental set up. On the left side the complete set up is shown with the printed circuit board (PCB) and bread boards placed on a grounding plate. The PCB is connected to voltage sources and a function generator (used for SHIL). For data logging a Pico Scope is used with the according computer software. The PCB shown from above on the right side contains 20 oscillators which get coupled via resistors on separate bread boards. On the breadboard is an additional unity gain buffer placed that is necessary to decouple the reference oscillator in the Vertex Cover Problem

Table 5 Parameters of the laboratory setup in Fig. 17

Circuit parameters	
$\hat{e}_{SHIL} = 700 \text{ mV}$	$\hat{e}_b = 2 \text{ V}$ $T_0 = 10.7 \text{ ms}$ $R_b = 100 \text{ k}\Omega$

zero, such that $p(t) = 0$ is not possible anymore. Therefore, we observe higher power dissipation, when SHIL is used. However, since anti-phase synchronization still minimizes power dissipation, we see that the oscillators assume this states nonetheless. This phenomenon is very important and has not been pointed out in literature so far. Depending on the problem, we see that SHIL can greatly influence the tendencies of coupled oscillators. This can even mean that the optimal solution of the considered problem does not map to the phase configuration minimizing the dissipated power $p(t)$ anymore, when SHIL is active. Also, this analysis reveals that it may be beneficial to use oscillators with symmetric oscillations for oscillator-based optimization, since this guarantees that the optimal solution of the considered problem is mapped to the minimum of $p(t)$, when the mapping of Sect. 2.3 is used.

Let us now consider problem (b) in Fig. 18. Here, we attempted to recreate the theoretical scenario depicted in Fig. 5. Again, we have $J_{12} = J_{13} = J_{23} = -1$, as in Fig. 5. As has been pointed out in that section, this type of coupling leads to a contradiction, as there is no binary phase configuration that can lead to $p(t) = 0$. Therefore, the oscillators actively search for

other phase configurations that minimize $p(t)$, which are not binary. The same results depicted in Fig. 5 can be clearly observed in our voltage measurements in Fig. 18b. The oscillators assume a three-phase configuration, when SHIL is turned off ($t < 60 \text{ ms}$). However, due to parameter mismatches (leading to a frequency mismatch between the oscillators), the oscillators do not fully lock to this state, but rather lock to it for short time intervals and oscillate around it otherwise. Once SHIL is turned on, the optimizer is now forced to search for binary solutions, and it successfully finds the same solution as in Fig. 5, which indeed corresponds to the problem's solution. In the right plot of Fig. 18b, we again see that the three-phase configuration minimizes the dissipated power $p(t)$ even more than the solution of the mapped problem, just like in Fig. 5.

The third problem depicted in Fig. 18c is an unweighted max-cut problem with a maximal cut size of 6 edges. There are multiple partitions that can lead to an optimal solution and we were able to achieve every one of these partitions using our oscillator-based optimizer. In the graph of Fig. 18c, we have depicted one of the optimal partitions that have been found by our optimizer. We would like to point out that the edge connecting \mathcal{N}_5 and \mathcal{N}_6 is grayed out, because it does not belong to solution of the problem, since we only count the edges that connect the two partitions, refer to Sect. 3.3. Looking at the central plot in Fig. 18c, we see that the optimizer finds this optimal partition even before the SHIL signal is turned on. In other words, we can clearly see that the optimal partition has been

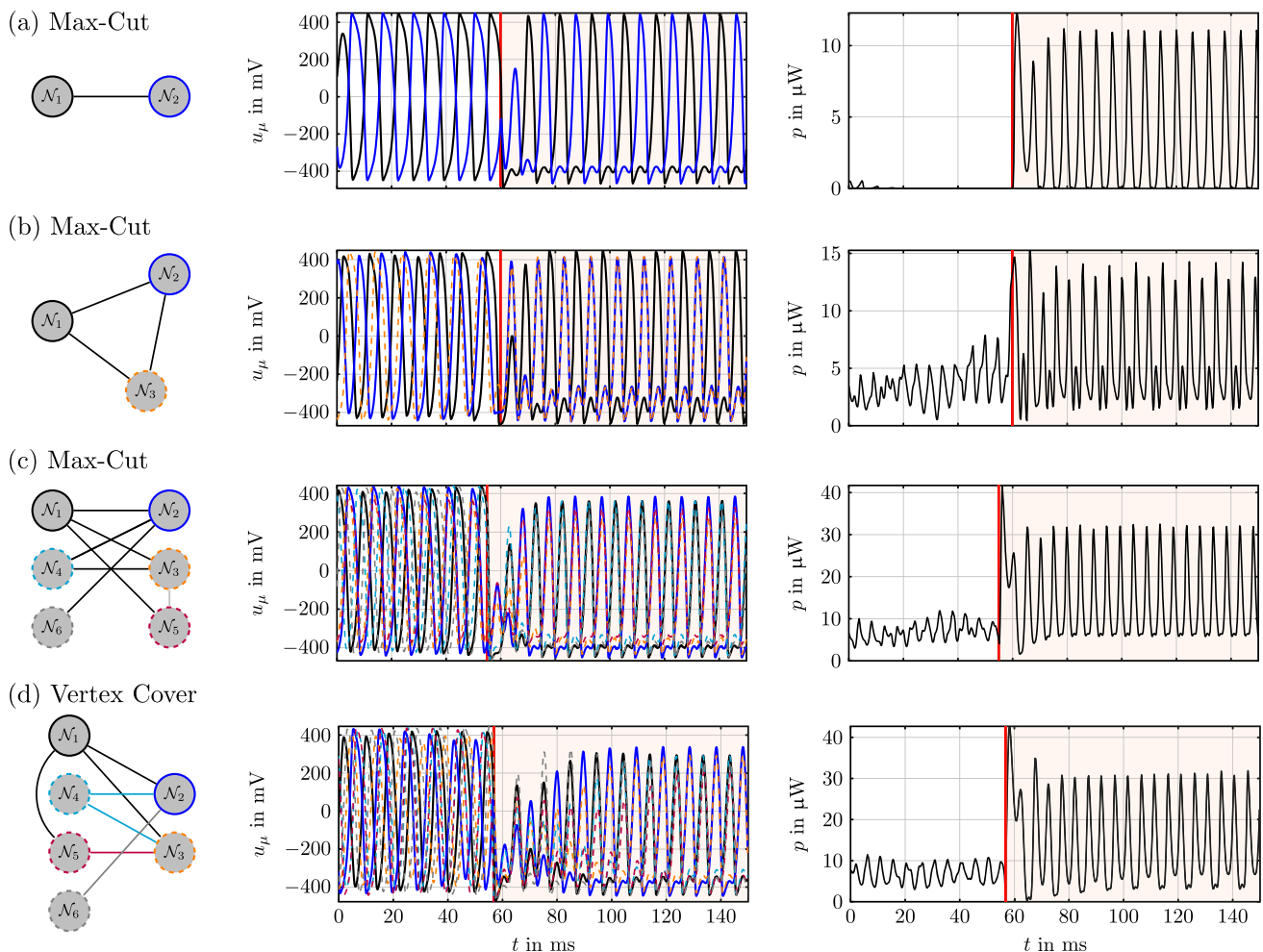


Fig. 18 Measurement results of four different optimization problems (**a–d**). The left column depicts the graph associated with the optimization task. The central column depicts the measured oscillator voltages u_μ over time, where the signal colors are correlated to the boundary colors of the nodes of the graphs in the left column. For all time instants preceding the red line, no SHIL signal is used. Afterwards SHIL is turned on. The right column depicts the dissipated power $p(t)$ over time

successfully minimized to the phase configuration minimizing the dissipated power. In order to clearly decode the solution, we turn on SHIL at $t = 60$ ms. Here, we see that the optimizer simply reassumes the solution of the problem. However, due to SHIL, the oscillation phases are exactly locked to 0 or π , which allows us to easily decode the solution. The right plot in Fig. 18c shows the same phenomenon observed in Fig. 18a: turning on SHIL leads to more power dissipation due to the distorted voltage signals.

Our final measurement is depicted in Fig. 18d. Here, we attempted to solve a minimal vertex cover problem with 6 nodes. As pointed out in Sect. 3.3 (see equation (18)), the minimal vertex cover problem contains a Zeeman term, hence we require a seventh reference oscillator in order to implement the coefficients h_μ . This problem is used as a proof-of-concept in order to show that our oscillator-based optimizer can also deal with problems containing the Zeeman term. We would like to point out that the graphs depicted in Fig. 18c, d are

isomorphic. The minimal vertex cover problem, however, has a different solution than the max-cut problem. Specifically, we require at least four nodes in order to color every edge in the graph, while the max-cut problem requires two or three nodes in order to achieve the largest possible cut size. The four nodes representing the solution of the problem ($\mathcal{N}_1, \mathcal{N}_4, \mathcal{N}_5, \mathcal{N}_6$) are drawn as a partition in the graph of Fig. 18d. We would like to point out that we have colored the edges in the same color as the incident nodes belonging to the solution of the problem. Prior to turning on the SHIL signal, the solution of the problem can not be identified from the output voltages u_μ depicted in the center of Fig. 18d. However, once the SHIL signal is turned on, the optimizer successfully finds the optimal solution, which translates to the graphical solution depicted on the left of Fig. 18d. The right plot in Fig. 18d, again, shows that the binary solution corresponds to a state of higher power dissipation, when compared against the state the oscillators assumed prior to turning on the

SHIL signal. If we compare the right plots in Fig. 18c, d, however, we see that the solution of the vertex cover problem dissipates less power than that of the max-cut problem. This proves that adding a seventh reference oscillator with adequate coupling weights indeed reformulates the original max-cut problem as a minimal vertex cover problem. This observation is very important since the matrix \mathbf{J} is identical in problems (c) and (d), compare equations (17, 18).

5 Conclusion and outlook

In this work, we have covered the fundamental aspects of oscillator-based optimization. We started by explaining the theoretical design of oscillator-based optimizers with the goal of solving quadratic unconstrained binary optimization (QUBO) problems. Here, we propose to build an oscillator-based optimizer using resistively coupled FitzHugh-Nagumo oscillators. The resistive coupling results as a natural consequence of the synchronization tendencies of coupled oscillators. The main hypothesis supporting it is that a network of coupled oscillators has the natural tendency of minimizing interactions, as this minimizes power dissipation. In addition to explaining how QUBOs can be mapped onto the resistive couplings of our network, we have also discussed the enforcement of optimization constraints. In the second part of our work, we derived a real-time capable wave digital algorithm, which allows for emulating our proposed optimizer. This algorithm serves as a testing tool for seeking out potential problems prior to implementing the network. The third and final part of this work presents our own in-house oscillator-based optimizer. Here, the theoretical circuitry presented in the first part of this work is fully implemented in analog hardware. The resulting circuit is then tested on four different optimization problems, which validate our proposed methodology for building oscillator-based optimizers.

In future works, we plan to enhance the circuit implementation presented in this work by reducing the number of active components both in the coupling as well as in the oscillator itself. Besides that, we aim to explore the use of memristors for dynamic topology formation [49]. Memristors are resistors with memory that remember the last state they are programmed to. Usually, these devices are built, so they switch between a high- and low-ohmic state. By coupling the oscillators using memristors instead of resistors, we can thus implement a dynamic connection, which allows us to "rewire" the network when dealing with new problems. When the memristor is in the high-ohmic state, the oscillators are effectively uncoupled. Otherwise, if the memristor is in the low-ohmic state, then the oscillators are coupled.

Acknowledgements. This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) "Project-ID 434434223" SFB 1461.

Author contributions

BAB conceptualized the work, wrote the manuscript, implemented the simulation framework, conducted numerical experiments analyzed the results, revised, and reviewed the manuscript. MN implemented the circuit, conducted the experiments, provided measurement results, partially wrote the manuscript, and revised the final version. KO and HK actively supported the conceptualization, writing, and revision process.

Funding Information Open Access funding enabled and organized by Projekt DEAL.

Data availability statement The data that support the findings of this study are available on request from the authors Bakr Al Beattie and Maximiliane Noll.

Declarations

Conflict of interest The authors have no relevant financial or non-financial interests to disclose.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

A Simulation time

In Fig. 19, we recorded the simulation runtime as a function of the network size. For every network size, we construct a complete network, in which every oscillator is connected to every other oscillator. The resulting network maps a max-cut problem of a complete graph for a given network size. All simulations were carried out on a lab computer (in MATLAB) with an Intel Core i7 processor running at 2.9 GHz, using 32GB of RAM and Windows 10.

As can be seen from Fig. 19, the wave digital concept allows for simulating large networks with up to 4500 oscillators with a reasonable runtime of about 3 min. At this point, we would like to stress that the wave digital concept is a platform-independent tool. By programming the wave digital algorithm in low-level programming languages such as C or C++, a significant improvement can be made. In other words, the simulation runtime of a wave digital algorithm is strongly dependent on the platform that it is programmed on.

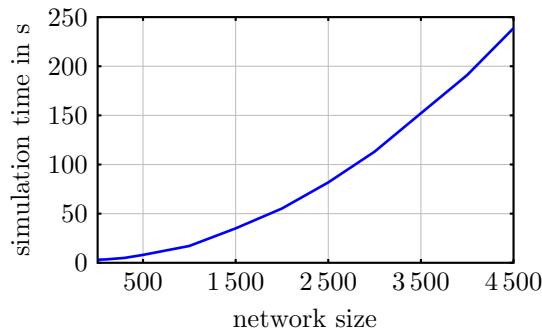


Fig. 19 Simulation runtime of the wave digital algorithm as a function of the network size. In every simulation, a complete graph is considered, where every oscillator is connected to every other oscillator

References

1. J. von Neumann, First draft of a report on the EDVAC. *IEEE Annals of the History of Computing*. **15**(4), 27–75 (1993). <https://doi.org/10.1109/85.238389>
2. R. M. Karp, In: Miller RE, Thatcher JW, Bohlinger JD, editors. *Reducibility among Combinatorial Problems*. Boston, MA: Springer US; (1972). p. 85–103. Available from: https://doi.org/10.1007/978-1-4684-2001-2_9
3. S. Harwood, C. Gambella, D. Trenev, A. Simonetto, D. Bernal, D. Greenberg, Formulating and Solving Routing Problems on Quantum Computers. *IEEE Transactions on Quantum Engineering*. **2**, 1–17 (2021). <https://doi.org/10.1109/TQE.2021.3049230>
4. R. Au-Yeung, N. Chancellor, P. Halffmann, NP-hard but no longer hard to solve? Using quantum computing to tackle optimization problems. *Frontiers in Quantum Science and Technology*. 2023;2. <https://doi.org/10.3389/frqst.2023.1128576>
5. Li J, Liu T, Hou T, Wang X, Liu C, Dai P, et al. Experimental verification of NP-complete problems via linear optics. In: 2021 13th International Conference on Wireless Communications and Signal Processing (WCSP); 2021. p. 1–6
6. Honjo T, Inaba K, Inagaki T, Ikuta T, Yamada Y, Takesue H. A coherent Ising machine based on a network of 100,000 degenerate optical parametric oscillator pulses. In: 2022 IEEE 22nd International Conference on Nanotechnology (NANO); 2022. p. 405–408
7. LYAPUNOV AM. The general problem of the stability of motion. *International Journal of Control*. 1992;55(3):531–534. <https://doi.org/10.1080/00207179208934253>
8. Wang T, Roychowdhury J. Oscillator-based Ising Machine. *CoRR*. 2017;abs/1709.08102
9. T. Wang, J. Roychowdhury, OIM: Oscillator-Based Ising Machines for Solving Combinatorial Optimisation Problems, in *Unconventional Computation and Natural Computation*. ed. by I. McQuillan, S. Seki (Springer International Publishing, Cham, 2019), pp.232–256
10. T. Wang, L. Wu, P. Nobel, J. Roychowdhury, Solving combinatorial optimisation problems using oscillator based Ising machines. *Natural Computing*. **20**(2), 287–306 (2021). <https://doi.org/10.1007/s11047-021-09845-3>
11. Srivastava S, Lai X, Roychowdhury J. Nonlinear Phase Macromodel Based Simulation/Design of PLLs with Superharmonically Locked Dividers. In: *IEEE Custom Integrated Circuits Conference 2006*; 2006. p. 349–352
12. Kuramoto Y. Self-entrainment of a population of coupled non-linear oscillators. *International Symposium on Mathematical Problems in Theoretical Physics*. 1975;p. 420–422
13. J. Chou, S. Bramhavar, S. Ghosh, W. Herzog, Analog Coupled Oscillator Based Weighted Ising Machine. *Scientific Reports*. **9**(1), 14786 (2019). <https://doi.org/10.1038/s41598-019-49699-5>
14. S. Dutta, A. Khanna, S. Datta, Understanding the Continuous-Time Dynamics of Phase-Transition Nano-Oscillator-Based Ising Hamiltonian Solver. *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*. **6**(2), 155–163 (2020). <https://doi.org/10.1109/JXCDC.2020.3045074>
15. B. Al Beattie, K. Ochs, A network-theoretical perspective on oscillator-based Ising machines. *International Journal of Circuit Theory and Applications*. **51**(6), 2499–2517 (2023). <https://doi.org/10.1002/cta.3553>
16. Ochs K, Al Beattie B, Jenderny S. An Ising Machine Solving Max-Cut Problems based on the Circuit Synthesis of the Phase Dynamics of a Modified Kuramoto Model. In: 2021 IEEE International Midwest Symposium on Circuits and Systems (MWSCAS); 2021. p. 982–985
17. Zhang T, Tao Q, Liu B, Han J. A Review of Simulation Algorithms of Classical Ising Machines for Combinatorial optimization. In: 2022 IEEE International Symposium on Circuits and Systems (ISCAS); 2022. p. 1877–1881
18. J. Vaidya, R.S. Surya Kanthi, N. Shukla, Creating electronic oscillator-based Ising machines without external injection locking. *Scientific Reports*. **12**(1), 981 (2022). <https://doi.org/10.1038/s41598-021-04057-2>
19. R. Afoakwa, Y. Zhang, U.K.R. Vengalam, Z. Ignjatovic, M. Huang, BRIM: Bistable Resistively-Coupled Ising Machine. *IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. **2021**, 749–760 (2021). <https://doi.org/10.1109/HPCA51647.2021.00068>
20. Csaba G, Prood W. Coupled oscillators for computing: A review and perspective. *Applied Physics Reviews*. 2020 01;7(1):011302. <https://doi.org/10.1063/1.5120412>. https://arxiv.org/abs/https://pubs.aip.org/aip/apr/article-pdf/doi/10.1063/1.5120412/14575376/011302_1_online.pdf
21. S. Dutta, A. Khanna, A.S. Assoa, H. Paik, D.G. Schlom, Z. Toroczka et al., An Ising Hamiltonian solver based on coupled stochastic phase-transition nano-oscillators. *Nature Electronics*. **4**(7), 502–512 (2021). <https://doi.org/10.1038/s41928-021-00616-7>
22. I. Ahmed, P.W. Chiu, W. Moy, C.H. Kim, A Probabilistic Compute Fabric Based on Coupled Ring Oscillators for Solving Combinatorial Optimization Problems. *IEEE Journal of Solid-State Circuits*. **56**(9), 2870–2880 (2021). <https://doi.org/10.1109/JSSC.2021.3062821>
23. Graber M, Angeli N, Hofmann K. An Efficient Modeling Approach for Large Ring Oscillator Based Ising Machines. In: *SMACD / PRIME 2021*; International Conference on SMACD and 16th Conference on PRIME; 2021. p. 1–4

24. Lucas A. Ising formulations of many NP problems. *Frontiers in Physics*. 2014;2. <https://doi.org/10.3389/fphy.2014.00005>
25. Yoshimura N, Tawada M, Tanaka S, Arai J, Yagi S, Uchiyama H, et al. Efficient Ising Model Mapping for Induced Subgraph Isomorphism Problems Using Ising Machines. 2019 IEEE 9th International Conference on Consumer Electronics (ICCE-Berlin). 2019;p. 227–232. <https://doi.org/10.1109/ICCE-Berlin47944.2019.8966218>
26. N. Matsumoto, Y. Hamakawa, K. Tatsumura, K. Kudo, Distance-based clustering using QUBO formulations. *Scientific Reports*. **12**(1), 2669 (2022). <https://doi.org/10.1038/s41598-022-06559-z>
27. A. Mahasinghe, V. Fernando, P. Samarawickrama, QUBO formulations of three NP problems. *Journal of Information and Optimization Sciences*. **42**(7), 1625–1648 (2021). <https://doi.org/10.1080/02522667.2021.1930657>
28. W. Kurebayashi, T. Yamamoto, S. Shirasaka, H. Nakao, Phase reduction of strongly coupled limit-cycle oscillators. *Physical Review Research*. **12**, 4 (2022). <https://doi.org/10.1103/PhysRevResearch.4.043176>
29. A. Fettweis, Wave Digital Filters: Theory and Practice. *Proceedings of the IEEE*. **74**(2), 270–327 (1986). <https://doi.org/10.1109/PROC.1986.13458>
30. A. Fettweis, Robust Numerical Integration Using Wave-Digital Concepts. *Multidimensional Systems and Signal Processing*. **17**(1), 7–25 (2006). <https://doi.org/10.1007/s11045-005-6236-3>
31. K. Meerkötter, On the Passivity of Wave Digital Networks. *IEEE Circuits and Systems Magazine*. **18**(4), 40–57 (2018). <https://doi.org/10.1109/MCAS.2018.2872664>
32. G. Hetmanczyk, Exploiting the parallelism of multidimensional wave digital algorithms on multicore computers. *Multidimensional Systems and Signal Processing*. **21**(1), 45–58 (2010). <https://doi.org/10.1007/s11045-009-0090-7>
33. B. Al Beattie, K. Ochs, Solving a one-dimensional moving boundary problem based on wave digital principles. *Multidimensional Systems and Signal Processing*. (2023). <https://doi.org/10.1007/s11045-023-00881-z>
34. Michaelis D, Ochs K, Beattie BA, Jenderny S. Towards A Self-Organizing Neuronal Network Based on Guided Axon-Growth. In: 2022 IEEE 65th International Midwest Symposium on Circuits and Systems (MWSCAS); 2022. p. 1–4
35. Ochs K, Beattie BA. Towards Wave Digital Modeling of Neural Pathways Using Two-Port Coupling Networks. In: 2022 IEEE International Symposium on Circuits and Systems (ISCAS); 2022. p. 809–812
36. Jenderny S, Ochs K. Wave Digital Emulation of a Bio-Inspired Circuit for Axon Growth. In: 2022 IEEE Biomedical Circuits and Systems Conference (BioCAS); 2022. p. 260–264
37. J. Nagumo, S. Arimoto, S. Yoshizawa, An Active Pulse Transmission Line Simulating Nerve Axon. *Proceedings of the IRE*. **50**(10), 2061–2070 (1962). <https://doi.org/10.1109/JRPROC.1962.288235>
38. Gyarmati I. In: *The Principle of Least Dissipation of Energy*. Berlin, Heidelberg: Springer Berlin Heidelberg; 1970. p. 89–113. Available from: https://doi.org/10.1007/978-3-642-51067-0_5
39. S.K. Vadlamani, T.P. Xiao, E. Yablonovitch, Physics successfully implements Lagrange multiplier optimization. *Proceedings of the National Academy of Sciences*. **117**(43), 26639–26650 (2020). <https://doi.org/10.1073/pnas.2015192117>
40. Vadlamani SK, Xiao TP, Yablonovitch E. Equivalence of Coupled Parametric Oscillator Dynamics to Lagrange Multiplier Optimization. In: 2022 IEEE International Conference on Rebooting Computing (ICRC); 2022. p. 45–50
41. L. Onsager, Reciprocal Relations in Irreversible Processes. I. *Phys Rev*. **37**, 405–426 (1931). <https://doi.org/10.1103/PhysRev.37.405>
42. X. Zhang, X. Zhou, B. Aliener, A.S. Daryoush, A study of subharmonic injection locking for local oscillators. *IEEE Microwave and Guided Wave Letters*. **2**(3), 97–99 (1992). <https://doi.org/10.1109/75.124911>
43. A. Neogy, Roychowdhury J. Analysis, design of subharmonically injection locked oscillators. In: *Design, Automation and Test in Europe Conference and Exhibition (DATE) 2012*, 1209–1214 (2012)
44. Wang T.: Sub-harmonic Injection Locking in Metronomes
45. T. Schwerdtfeger, A. Kummert, Nonlinear Circuit Simulation by Means of Alfred Fettweis' Wave Digital Principles. *IEEE Circuits and Systems Magazine*. **19**(1), 55–C3 (2019). <https://doi.org/10.1109/MCAS.2018.2872666>
46. H.S.M. Coxeter, Twelve points in $\langle i \rangle PG \langle /i \rangle (5, 3)$ with 95040 self-transformations. *Proceedings of the Royal Society of London Series A Mathematical and Physical Sciences*. **247**(1250), 279–293 (1958). <https://doi.org/10.1098/rspa.1958.0184>
47. K.K. Nair, M.N.S. Swamy, On Gyrator Networks. *IETE Journal of Research*. **7**(3), 111–116 (1961). <https://doi.org/10.1080/03772063.1961.11486279>
48. A. Antoniou, K. Naidu, Modeling of a gyrator circuit. *IEEE Transactions on Circuit Theory*. **20**(5), 533–540 (1973). <https://doi.org/10.1109/TCT.1973.1083731>
49. L. Chua, Memristor-The missing circuit element. *IEEE Transactions on Circuit Theory*. **18**(5), 507–519 (1971). <https://doi.org/10.1109/TCT.1971.1083337>