

Fully Non-Linear Neuromorphic Computing with Linear Wave Scattering

Clara C. Wanjura and Florian Marquardt

Max Planck Institute for the Science of Light, Staudtstraße 2, 91058 Erlangen, Germany

(Dated: August 31, 2023)

The increasing complexity of neural networks and the energy consumption associated with training and inference create a need for alternative neuromorphic approaches, e.g. using optics. Current proposals and implementations rely on physical non-linearities or opto-electronic conversion to realise the required non-linear activation function. However, there are significant challenges with these approaches related to power levels, control, energy-efficiency, and delays.

Here, we present a scheme for a neuromorphic system that relies on linear wave scattering and yet achieves non-linear processing with a high expressivity. The key idea is to inject the input via physical parameters that affect the scattering processes. Moreover, we show that gradients needed for training can be directly measured in scattering experiments. We predict classification accuracies on par with results obtained by standard artificial neural networks. Our proposal can be readily implemented with existing state-of-the-art, scalable platforms, e.g. in optics, microwave and electrical circuits, and we propose an integrated-photonics implementation based on racetrack resonators that achieves high connectivity with a minimal number of waveguide crossings.

I. INTRODUCTION

The rapid growth in neural network complexity has led to an exponential increase in energy consumption and training costs. This has created a need for more energy and cost efficient alternatives sparking the rapidly developing field of neuromorphic computing [1] in which computations are performed with physical artificial neurons.

Typically, neural networks connect neurons in consecutive layers through linear maps and non-linear activation functions. So far, the prevalent approach has been to realise the linear maps with linear physical interactions and employ physical non-linearities (or approaches like optoelectronic conversion) to realise the non-linear activation function. Among the many neuromorphic computing platforms [2–5], optical platforms [2, 3, 6] are one of the most promising contenders for neuromorphic computing as they efficiently allow to implement the linear aspects of the neural network offering a high degree of parallelism, high computation speeds and scalability. Furthermore, linear computations can be performed passively [7, 8] while propagation losses can be very small making these devices potentially very energy efficient. Linear optical networks (in free space or integrated photonics) for implementing the linear aspects of neural networks, i.e., matrix-vector multiplication, are already well developed and have become the basis of commercial chips [9, 10].

On the other hand, physical non-linearities to realise the non-linear aspects of the neural network are still hard to implement, incurring substantial hardware overhead, fabrication challenges, and other possibly demanding requirements [11–13], such as high laser powers in an implementation with non-linear crystals. Furthermore, non-linearities can induce chaotic dynamics which makes it impossible to train the system. An alternative approach bypasses these challenges by applying the non-linearity opto-electronically [14–17]. However, proposals relying on optoelectronic conversion to realise the non-linear activation functions may be bulky, less energy-efficient and may suffer from delays.

Moreover, an efficient physics-based training in the presence of non-linearities is an open challenge, although

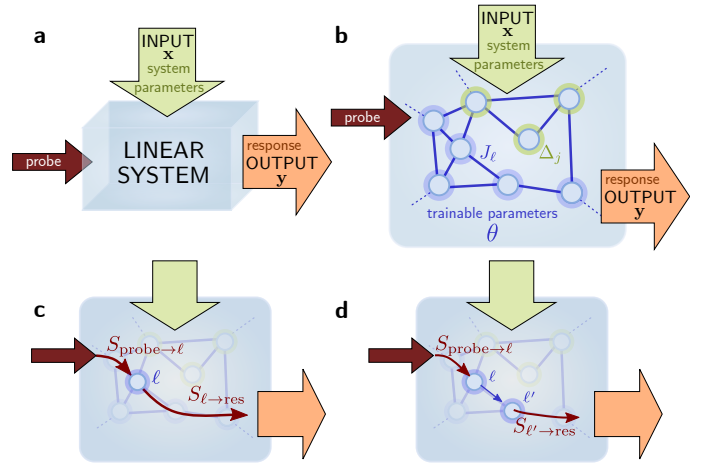


FIG. 1. **Fully non-linear neuromorphic system with linear wave propagation.** **a** The input to the linear, neuromorphic system is encoded in some of the tunable system parameters while the output is determined through the scattering response to a probe signal. Other controllable parameters serve as learnable parameters. **b** Example for an implementation using wave propagation through a network of coupled modes (e.g. optical resonators). Here, the detunings Δ_j of some of the optical modes are utilized as input x while other detunings and the coupling constants J_j between modes are learnable parameters θ . The output is a suitable set of scattering matrix elements, Eqs. (3) and 4, which we obtain by comparing the response to the probe field, Eq. (1). **c-d** Due to the linearity of these systems, we have access to the gradients w.r.t. system parameters which are required for training. The gradients are given by the products of scattering matrix elements, Eqs. (7) and (8).

some conceptual progress has been made, especially in the form of equilibrium propagation [18–20] and Hamiltonian-echo back-propagation [21]. In the presence of decay, a back-propagation algorithm was developed [22] and implemented [23] only for specific types of non-linearities. Another approach to physical training is to adjust parameters based on feedback [24] or random parameter shifts [10]. However, this approach may scale unfavourably with network

size [25]. Other neuromorphic systems can be trained in a computer simulation [26] which necessitate complete and accurate knowledge of the system but still offers a potential advantage in terms of speed and efficiency during inference. Finally, hybrid approaches [14, 16] perform physical back-propagation on the linear components of the neuromorphic system.

Here, we propose a completely new approach for a *fully non-linear* neuromorphic system which is only based on *linear* scattering. and therefore bypasses all of the challenges associated with realising and controlling physical non-linearities. The key idea lies in injecting the input of the neural network in the system parameters, Fig. 1 a: While linear physical systems *linearly* relate a probe signal to the response via the scattering matrix, the scattering matrix itself depends *non-linearly* on system parameters. Therefore, we interpret some of the system parameters as the input of our neural network and some as learnable parameters. The output of our neural network is given by suitable scattering matrix elements which we obtain from scattering experiments by comparing the response signal to the probe signal. Remarkably, this non-linear dependence enables us to implement a fully functional neural network capable of performing the same tasks as standard artificial neural networks. As a major advantage of working with a linear scattering system, gradients needed to perform gradient descent can be directly measured in scattering experiments requiring only a minimal number of scattering experiments. This does not require complete knowledge or control of the system; in fact, one can treat untrained parts of the system as black box. The neural network can be evaluated efficiently (inference) by performing a minimal number of scattering experiments promising high computation speeds. We simulate and train our system on hand-written digit recognition and achieve classification accuracies on par with the accuracies obtained by standard artificial neural networks.

Our proposal can be readily implemented with existing state-of-the-art scalable platforms, e.g., in photonics [7, 8, 27] and analog electrical circuits [28]. Concretely, we propose an implementation of our network with optical racetrack resonators coupled via microring resonators allowing for a densely connected network at a minimal number of waveguide crossings. The architecture of our racetrack resonator array may also find application in other neuromorphic computing applications that require a high connectivity.

Beyond that, our results are very general and apply to any type of linear system for which measurable quantities can be obtained from a Green's function. For instance, in analog electrical circuits, the scattering matrix can be replaced by the impedance matrix [8–10] which is obtained from the circuit's Green's function, and tunable resistances, capacitors or inductances can serve as input of the network.

II. NON-LINEAR NEUROMORPHIC COMPUTING WITH LINEAR SCATTERING

A. Concept

Waves propagating through a linear multi-mode systems establish a linear relation between probe signals $\mathbf{a}_{\text{probe}}$ and the response \mathbf{a}_{res} (both are vectors in a multi-port device) via the scattering matrix

$$\mathbf{a}_{\text{res}}(\omega) = S(\omega, \mathbf{x}, \theta) \mathbf{a}_{\text{probe}}(\omega). \quad (1)$$

This matrix explicitly depends on the frequency ω of the probe signal and the system parameters, some of which we label as \mathbf{x} and some as θ . Later, \mathbf{x} will represent the input vector to the neuromorphic system and θ the set of training parameters.

Concretely, the system could, for instance, consist of a number of coupled modes a_j , such as the modes of optical resonators, which we call *neuron modes*. The scattering matrix is determined by the linear evolution equations for the neuron modes a_j collected in the vector $\mathbf{a} \equiv (a_1, \dots, a_N)^T$. In the frequency domain, $\mathbf{a}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} dt e^{i\omega t} \mathbf{a}(t)$, the equations of motion take the form

$$-i\omega \mathbf{a}(\omega) = -iH(\mathbf{x}, \theta) \mathbf{a}(\omega) - \sqrt{\kappa} \mathbf{a}_{\text{probe}}(\omega). \quad (2)$$

H denotes the dynamic matrix which explicitly depends on the system parameters \mathbf{x} and θ , and $\kappa \equiv \text{diag}(\kappa_1, \dots, \kappa_N)$ are the external decay rates of the neuron modes to the probe waveguides. We obtain the scattering matrix from Eq. (2) through input-output boundary conditions, $a_{j,\text{res}} = a_{j,\text{probe}} + \sqrt{\kappa_j} a_j$ [32, 33],

$$\begin{aligned} S(\omega, \mathbf{x}, \theta) &= \mathbb{1} - i\sqrt{\kappa} G(\omega, \mathbf{x}, \theta) \sqrt{\kappa} \\ &= \mathbb{1} - i\sqrt{\kappa} [\omega \mathbb{1} - H(\mathbf{x}, \theta)]^{-1} \sqrt{\kappa}. \end{aligned} \quad (3)$$

with the system's Green's function $G(\omega, \mathbf{x}, \theta) \equiv -i(\omega \mathbb{1} - H(\omega, \mathbf{x}, \theta))^{-1}$.

Eq. (3) reveals the general idea behind our concept. While the relation between probe and response is *linear*, Eq. (1), the scattering matrix $S(\omega, \mathbf{x}, \theta)$, Eq. (3), is a *non-linear* function of the system parameters. Hence, interpreting some of the system parameters \mathbf{x} as inputs, and some as learnable parameters θ , see Fig. 1 a, we are now able to represent learnable non-linear functions of the input with the help of the scattering matrix.

The network's output is a suitable set of scattering matrix elements $S_{j,\ell}$ which are obtained as the ratio between the response and the probe signal. Since the scattering matrix is generally complex, one can consider real or imaginary part as output, or, more generally,

$$y_r = \text{Re}(e^{i\phi} S_{r,p}), \quad (4)$$

with a suitable set of p and r and a convenient ϕ . p here refers to a probe site and r to the sites at which the response is recorded. The number of response sites r equals the output dimension N_{out} . The phase ϕ allows to select a quadrature, e.g., $\phi = 0$ corresponds to $\text{Re} S_{r,p}$ whereas $\phi = \frac{\pi}{2}$ corresponds to

Im $S_{r,p}$ which we will use later on. In practise, the quadrature can be measured through homodyne detection. There are two obvious, convenient choices for p and r : (i) the probe site p is fixed and we consider different sites r at which we record the system response. This allows to record the output of the neuromorphic system (inference) with a single measurement. (ii) We set $p = r$. This requires N_{out} measurements to infer the output, but we found that in this case, the training converges more reliably.

B. Input replication for improved non-linear expressivity

The non-linearity of the scattering matrix (3) as function of the input \mathbf{x} stems from the calculation of the matrix inverse representing a specific type of non-linearity. For instance, considering a scalar input x , any scattering matrix element $S_{j,\ell}$ is of the form $S_{j,\ell} = (ax + b)/(cx + d)$ with a, b, c, d some quantities that depend on the other system parameters. This non-linearity stems from the fact that waves can propagate back and forth between modes and the scattering matrix (3) is a result of the interference between all these waves. We can see this by expressing the matrix inverse in Eq. (3) for a specific matrix element via the adjugate, $(H^{-1})_{j,\ell} = (-1)^{j+\ell} \frac{\det H_{j,\ell}}{\det H}$, in which $H_{j,\ell}$ denotes the matrix in which the j th row and the ℓ th line have been omitted. Expanding the determinant expressions with the Laplace expansion, it is straightforward to see that both $\det H_{j,\ell}$ and $\det H$ linearly depend on any matrix entry $H_{m,n}$; therefore, both denominator and numerator in the previous expression depend linearly on x .

Note, however, that the matrix determinant of $\det H$ (and similarly $\det H_{j,\ell}$) contains terms such as $H_{1,1}H_{2,2}H_{3,3}\dots$ and $H_{1,1}H_{2,3}H_{3,2}\dots$. This allows us to overcome the seeming limitation of the scattering matrix, namely, by letting the input value x explicitly enter in more than one system parameter. In this way, it is possible to show, that one can represent general non-linear functions with the scattering matrix in which the number of repetitions R of the input determines how well the target function can be approximated. Letting $R \rightarrow \infty$, the approximation approaches the target function. We prove this in the SI for one-dimensional inputs.

Most applications of machine learning, however, operate on high-dimensional input spaces. In this case, the situation becomes even more favourable: even at a single replication of the input, the scattering matrix allows to encode correlations between different elements of the input vector, i.e., the scattering matrix automatically includes terms such as $x_1x_2x_3\dots$. Therefore, in practise, it can be sufficient to replicate the input only once or twice. In particular, we will show later that for a digit recognition task, Fig. 2, it was sufficient to replicate the input once, $R = 1$.

C. Training

We will now show one particularly useful consequence of working with a linear scattering system: it is possible to per-

form gradient descent based on physically measurable gradients, which is rare in neuromorphic systems. Specifically, gradients w.r.t. θ are directly measurable as products of scattering matrix elements.

The aim of the training is to minimise a cost function \mathcal{C} , such as the square distance between the target output \mathbf{y}_{tar} and the output \mathbf{y} of the system, Eq. (4), $\mathcal{C} = |\mathbf{y}_{\text{tar}} - \mathbf{y}|^2$. The derivative of the cost function w.r.t. a learnable parameter θ_j is given by

$$\frac{\partial \mathcal{C}}{\partial \theta_j} = \nabla_{\mathbf{y}} \mathcal{C} \cdot \left(\frac{\partial \mathbf{y}}{\partial \theta_j} \right) \quad (5)$$

with \mathbf{y} defined by Eq. (4), so it only depends on the derivative of the scattering matrix.

Given the mathematical structure of the scattering matrix (3), we obtain a simple formula for its derivative

$$\frac{\partial S_{r,p}}{\partial \theta_j} = i\kappa_p \left(G \frac{\partial H}{\partial \theta_j} G \right)_{r,p} \quad (6)$$

with $G = \sqrt{\kappa}^{-1}(S - \mathbb{1})\sqrt{\kappa}^{-1}$ the Green's function (3), see the SI for the derivation. Since $\partial H/\partial \theta_j$ is local, this implies that gradients can be obtained from scattering experiments as combination of scattering matrix elements. Gradients can be physically extracted allowing for very efficient training.

We make this more concrete in the next section in which we consider an optical system whose learnable parameters are the mode detunings and coupling strengths between modes.

D. Implementation based on coupled modes

To make the previous considerations more concrete, we consider a system of coupled radiation modes, e.g., in the microwave or optical regime or in electrical circuits composed of coupled resonant modes, but stress that the concept is general and applies to a variety of systems and platforms. In an optical system, the neuron modes can be represented by cavity modes, Fig. 1 b, at frequencies ω_j at detunings $\Delta_j \equiv \omega_j - \omega_0$ from some suitable reference frequency ω_0 . They all experience intrinsic decay at rate κ'_j . The modes can, in full generality, be coupled via any form of bi-linear coherent interactions, such as beam-splitter interactions, single-mode or two-mode squeezing, or via linear engineered dissipators. For simplicity, we will focus on beam-splitter couplings at strength J_j as shown in Fig. 1 b. Furthermore, a probe $a_{j,\text{probe}}$ is injected via a waveguide coupled to the modes at rate κ_j which induces additional losses at this rate. The detunings of some of the neuron modes serve as input \mathbf{x} while other tunable parameters, such as detunings and coupling strengths, are the learnable parameters θ , see Fig. 1 b. We obtain the network's output (4) by performing a suitable homodyne measurement on the response fields $a_{j,\text{res}}$. Importantly, not all of the system parameters need to be tunable for our concept to work.

Coming back to training, we can now more explicitly present the gradients of the scattering matrix elements defining the output (6) for the parameters of this coupled-mode

structure. In particular, the derivative w.r.t. detuning at the j th site is given by the scattering path from the probe site r to site j , and from j to the response site r as illustrated in Fig. 1 c

$$\frac{\partial S_{r,p}}{\partial \Delta_j} = i \frac{1}{\kappa_j} G_{j,p} G_{r,j} \quad (7)$$

with the Green's function $G_{m,n} = (S_{m,n} - \delta_{m,n}) / \sqrt{\kappa_m \kappa_n}$. Similarly, the derivative w.r.t. the coupling between the j th and ℓ th site is the sum of the possible scattering paths from the probe site to j or ℓ and the response site, Fig. 1 d,

$$\frac{\partial S_{r,p}}{\partial J_{j,\ell}} = i \frac{1}{\sqrt{\kappa_j \kappa_\ell}} (G_{j,p} G_{r,\ell} + G_{\ell,p} G_{r,j}). \quad (8)$$

As we will show in more detail below, the physical extraction of gradients is most straightforward in integrated-photonics platforms based on coupled resonators [14], with probe waveguides attached to every resonator whose resonance frequency represents a tunable training parameter. Other, non-tunable linear components can be freely added to the setup and need not even be fully characterized or fabricated according to some specified design: they can be treated as a black box, while the training procedure will automatically take care of accounting for their effect on the wave propagation. Gradients can be efficiently measured requiring only N_{out} measurements with N_{out} the dimension of the network output. These measurements record the full scattering response to a probe at any of the probe sites p , so the network can be evaluated (inference) at the same time as computing the gradients.

During training, gradients can either be applied directly or be post-processed to average over a mini-batch or to perform more advanced adaptive gradient-descent optimization (like with the well-known Adam optimizer). Post-processing can be directly implemented with analog electronics, e.g., to perform the sum operation over multiple gradients [16].

Note that while we focus on photonic systems for the rest of this work, our results are very general and apply to any type of linear system for which measurable quantities can be obtained from a Green's function. In fact, linear equations of the form (2) appear in many different contexts so we expect our findings to have wide-ranging application. For instance, for electrical circuits, the scattering matrix can be replaced by the impedance matrix [8–10] which is obtained from the circuit's Green's function and tunable resistances can serve as input of the network, see the SI for more details.

III. LAYERED ARCHITECTURE

A. Recursive solution to the scattering problem

The principle of information processing in the neuromorphic platform introduced here is fundamentally different from standard neural networks since the waves representing the information will be scattered back and forth inside the device rather than propagating unidirectionally. Nevertheless, with

this in mind, we can choose an architecture that is at least inspired by the typical layer-wise structure of artificial neural networks. It allows us to gain analytic insight into the scattering matrix and to derive optimal layer sizes to make efficient use of the number of independent parameters available.

We consider a layered architecture as sketched in Fig. 2 a with L layers and N_ℓ neuron modes in the ℓ th layer. Neuron modes are only coupled to neuron modes in consecutive layers but not within a layer. Note that while we sketch fully connected layers in Fig. 2 a, the network does not, in principle, have to be fully connected. However, fully connected layers have the advantage that there is *a priori* no ordering relation between modes based on their proximity within a layer.

This architecture allows us to gain analytic insight into the mathematical structure of the scattering matrix. For a compact notation, we split the vector \mathbf{a} of neuron modes, Eq. (2), into vectors $\mathbf{a}_n \equiv (a_1^{(n)}, \dots, a_{N_n}^{(n)})$ collecting the neuron modes of the n th layer. Correspondingly, we define the detunings of the neuron modes in the n th layer $\Delta^{(n)} = \text{diag}(\Delta_1^{(n)}, \dots, \Delta_{N_n}^{(n)})$, the extrinsic decay rates to the waveguides $\kappa^{(n)} = \text{diag}(\kappa_1^{(n)}, \dots, \kappa_{N_n}^{(n)})$, the intrinsic decay rates $\kappa'^{(n)} = \text{diag}(\kappa'_1{}^{(n)}, \dots, \kappa'_{N_n}{}^{(n)})$, the total decay rate $\kappa_{\text{tot}}^{(n)} = \kappa'^{(n)} + \kappa^{(n)}$ and $J^{(n)}$ the coupling matrix between layer n and $(n+1)$ in which the element $J_{j,\ell}^{(n)}$ connects neuron mode j in layer n to neuron mode ℓ in layer $(n+1)$. Note that this accounts for the possibility of attaching waveguides to all modes in each layer to physically evaluate gradients according to Eqs. (7) and (8). In the frequency domain, we obtain for the equations of motion of the n th layer

$$-i\omega \mathbf{a}_n = \left(-\frac{\kappa_{\text{tot}}^{(n)}}{2} - i\Delta^{(n)} \right) \mathbf{a}_n - iJ^{(n)} \mathbf{a}_{n+1} - iJ^{(n-1)} \mathbf{a}_{n-1} - \sqrt{\kappa^{(n)}} \mathbf{a}_{n,\text{probe}}, \quad (9)$$

in which we omitted the frequency argument ω for clarity. Eq. (9) does not have the typical structure of a feed-forward neural network since neighbouring layers are coupled to the left and right—a consequence of the wave propagation through the system.

We are interested in calculating the scattering response at the last layer L (the output layer) which defines the output of the neuromorphic system, so we set $\mathbf{a}_{n,\text{probe}} = 0$ for $n \neq L$ in Eq. (9) and only consider the response to the probe fields $\mathbf{a}_{L,\text{probe}}$. The following procedure allows us to calculate the scattering matrix block S_{out} relating only $\mathbf{a}_{L,\text{probe}}$ to $\mathbf{a}_{L,\text{res}}$. A suitable set of matrix elements of S_{out} then defines the output of the neuromorphic system via Eq. (4).

Solving for $\mathbf{a}_1(\omega)$, then $\mathbf{a}_2(\omega)$ and subsequent layers up to $\mathbf{a}_L(\omega)$, we obtain a recursive formula for $\mathbf{a}_n(\omega)$

$$\mathbf{a}_n = i\mathcal{G}_n J^{(n)} \mathbf{a}_{n+1} \quad (10)$$

with

$$\mathcal{G}_n(\omega) \equiv \left[\frac{\kappa_{\text{tot}}^{(n)}}{2} + i(\Delta^{(n)} - \omega) + J^{(n-1)\dagger} \mathcal{G}_{n-1} J^{(n-1)} \right]^{-1} \quad (11)$$

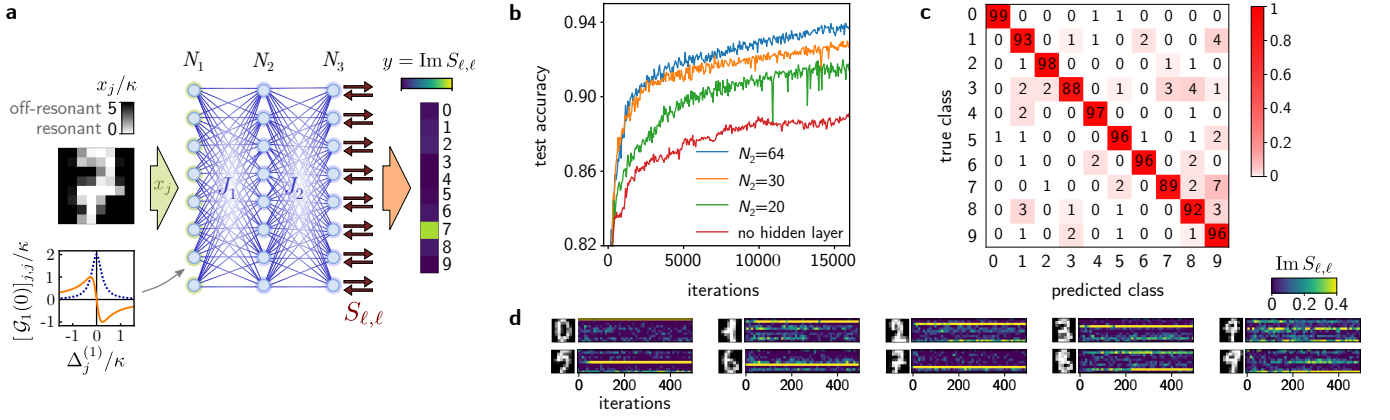


FIG. 2. Digit recognition using a scattering neuromorphic system with layered structure. **a** Scattering network used for digit recognition consisting of two or three fully connected layers with $N_1 = 128$, $N_3 = 10$ and either without hidden layer or with $N_2 \in \{20, 30, 64\}$. We consider equal decay rates κ , set the intrinsic decay to zero $\kappa' = 0$ and start from $J/\kappa = 2$ with random disorder on top. The input consisting of 64 greyscale pixel values is encoded in the detuning of the first layer to which we initially add a trainable offset which is initialised according to Eq. (14). A vector of pixel values serves as input in which we detune the background to $x_j = 5\kappa$ and make the foreground, the numerals, resonant $x_j = 0$. The inset illustrates the non-linear effect of the first layer showing real and imaginary part of $[\mathcal{G}_1(0)]_{j,j}/\kappa$, Eq. (11). The response to a probe signal at the third layer constitutes the output vector. The index of the maximal response constitutes the class. **b** Evolution of the test accuracy during training for different architectures: without hidden layer, or with $N_2 = \{20, 30, 64\}$. Here, an iteration over one mini-batch corresponds to evaluating 200 randomly chosen images while the shown test accuracy is evaluated on the entire test set. Increasing the size of the hidden layer improves both the convergence speed and the best accuracy. **c** The confusion matrix of the trained model with $N_2 = 64$ after 28,000 iterations. The overall test accuracy amounts to 94.66%. **d** Convergence for individual input pictures. The scattering matrix element with the largest imaginary part indicates the class. In most cases, the training rapidly converges towards the correct classification results. Digits of similar appearance, however, are frequently mistaken for the other, such as the digit 4 and 9, and only converge relatively late during training. For the training run shown here, we used Adam optimisation.

and $\mathcal{G}_0 = 0$ so that in the last layer, we have

$$\mathbf{a}_L = \mathcal{G}_L(\omega)\mathbf{a}_{L,\text{probe}}. \quad (12)$$

At the last layer, the matrix $\mathcal{G}_L(\omega)$ is equal to the system's

Green's function $G(\omega) = \mathcal{G}_L(\omega)$, Eq. (3).

Employing input-output boundary conditions [33], $\mathbf{a}_{L,\text{res}} = \mathbf{a}_{L,\text{probe}} + \sqrt{\kappa^{(L)}}\mathbf{a}_L$, we obtain the scattering matrix for the response at the last layer

$$S_{\text{out}}(\omega, x, \theta) = \mathbb{1} + \sqrt{\kappa^{(L)}} \left[\frac{\kappa_{\text{tot}}^{(L)}}{2} + i(\Delta^{(L)} - \omega) + J^{(L-1)\dagger} \mathcal{G}^{(L-1)} J^{(L-1)} \right]^{-1} \sqrt{\kappa^{(L)}} \quad (13)$$

The structure of Eqs. (13) and (11) is reminiscent of a generalised continued fraction, with the difference that scalar coefficients are replaced by matrices. We explore this analogy further in the SI where we also show that for scalar input and output, the scattering matrix (13) can approximate arbitrary analytic functions. Furthermore, the recursive structure defined by Eqs. (11) and (12) mimics that of a standard artificial neural network in which the weight matrix is replaced by the coupling matrix and the matrix inverse serves as activation function. However, in contrast to the standard activation function, which is applied element-wise for each neuron, the matrix inversion acts on the entire layer. To gain intuition for the effect of taking the matrix inverse, we plot a diagonal entry of $[\mathcal{G}_1(0)]_{j,j}/\kappa = [\frac{1}{2} + i\Delta_j^{(1)}/\kappa_j^{(1)}]^{-1}$ in Fig. 2 a. The real part follows a Lorentzian whereas the imaginary part is

reminiscent of a tapered sigmoid function.

The recursive structure of the scattering matrix (13) bears some resemblance to the variational quantum circuits of the quantum machine learning literature [34, 35] in which the subsequent application of discrete unitary operators allows to realise non-linear operations. In contrast, here, we consider the steady state scattering response which allows waves to propagate back and forth giving rise to yet more complicated non-linear maps (3).

B. Test case: digit recognition

To benchmark our model, we simulate and train a network with three layers, see Fig. 2 a, on handwritten digit classifica-

tion on a down-sized version of the MNIST dataset [36]. The set consists of 3,823 8×8 pixels training images of numerals between 0 and 9 as well as 1,797 test images. We choose an architecture in which the input is encoded in the detunings $\Delta_j^{(1)}$ of the first layer to which we add a trainable offset $\Delta_j^{(0)}$

$$\begin{aligned}\Delta_{2j}^{(1)} &= \Delta_{2j}^{(0)} + x_j \\ \Delta_{2j+1}^{(1)} &= -\Delta_{2j+1}^{(0)} + x_j\end{aligned}\quad (14)$$

which we initially set to $\Delta_{2j} = \Delta_{2j+1} = 4\kappa$ (for simplicity, we consider equal decay rates $\kappa_j^{(n)} \equiv \kappa$ from now on). In this way, the input enters the second layer in the form of $[G_1(0)]_{j,j}/\kappa$ once with positive, once with negative sign, see the plot of $[G_1(0)]_{j,j}/\kappa$ in Fig. 2 b. We found that, beyond improving the expressivity as explained above, this also leads to faster convergence of the training. This doubling of the input fixes the layer size to $N_1 = 2 \cdot 64$. The second layer (hidden layer) can be of variable size and we train a system (i) without the hidden layer, (ii) with $N_2 = 20$, (iii) with $N_2 = 30$, (iv) with $N_2 = 64$. We use one-hot encoding of the classes, so the output layer is fixed at $N_3 = 10$. We choose equal decay rates κ and express all other parameters in terms of κ . For simplicity, we set the intrinsic decay to zero $\kappa' = 0$.

We initialise the system by making the neuron modes in the second and third layer resonant and add a small amount of disorder, i.e., $\Delta_j^{(n)}/\kappa = w_\Delta(\xi_j^{(n)} - 1/2)$ with $w_\Delta = 0.002$ and $\xi \in [0, 1)$. Similarly, we set the coupling rates to 2κ and add a small amount of disorder, $J_{j,\ell}^{(n)}/\kappa = 2 + w_J\xi_{j,\ell}^{(n)}$ with $w_J = 0.2$. We found empirically that this initialisation leads to the fastest convergence of the training. Furthermore, we scale our input images such that the background is off-resonant at $\Delta/\kappa = 5$ and the numerals are resonant $\Delta/\kappa = 0$, see Fig. 2 a since, otherwise, the initial gradients are very small.

As output of the system, we consider the imaginary part of the diagonal entries of the scattering matrix (13) at the last layer at $\omega = 0$, i.e. $y_\ell \equiv \text{Im } S_{\ell,\ell}(0, x, \theta)$, Fig. 2 a. The goal is to minimise the cost function $\mathcal{C} = |\mathbf{y}_{\text{tar}} - \mathbf{y}|^2$ in which \mathbf{y}_{tar} is the target output vector of the system which is 1 at the index of the correct class and zero elsewhere. We train the system by performing gradient descent, i.e., for one mini-batch we select 200 random images, and compute the gradients of the cost function according to which we adjust the system parameters: the detunings $\Delta_j^{(n)}$ and the coupling rates $J_{j,\ell}^{(n)}$.

We show the accuracy evaluated on the test set at different stages during the training for four different architectures in Fig. 2 b. Both the convergence speed and the maximally attainable test accuracy depend on the size of the hidden layer with the system without hidden layer performing worst. This should not surprise us, since an increase in N_2 also increases the number of trainable parameters. We obtained the best classification accuracy on the test set of 94.7% for the system with $N_2 = 64$ after iterating over 28,000 mini-batches. We show the associated confusion matrix in Fig. 2 c. This accuracy is on par with the accuracy of 94.7% achieved by a standard artificial neural network. The artificial model we trained had an input layer of size 64, a hidden layer of size 64, an output

layer of size 10, and we used a sigmoid activation function.

We show the evolution of the output scattering matrix elements evaluated for a few specific images in Fig. 2 d. During training, the scattering matrix converges to the correct classification result quickly. Only for such images that look very similar, e.g., the image of the 4 and 9 shown in Fig. 2 d, the training oscillates between the two classes and takes a longer time to converge.

C. Hyperparameters

The architecture we propose has the following hyperparameters which influence the accuracy and the convergence speed: (i) the number of neuron modes per layer N_ℓ ; (ii) the number of layers L ; (iii) the number of input replications R ; (iv) The intrinsic decay rate of the system.

(i) The number of neuron modes in the first layer should match the product of the input dimension D and the number of repetitions of the input R . The number of neurons in the final layer is given by the output dimension, e.g., for classification tasks the dimension corresponds to the number of classes. The sizes of intermediate layers should be chosen to provide enough training parameters; as we can see from Fig. 2 b, both the convergence speed and the best accuracy rely on having a sufficient number of training parameters available, as for instance, a system with a second layer of only 20 neuron modes performs worse than a system with 64 neuron modes in the second layer. In particular, as we show in the SI, that independent of the architecture, the maximal number of independent couplings for an input of dimension D , input replication R and output dimension N_{out} is given by

$$N_{\text{opt}} = \frac{(RD + N_{\text{out}})(RD + N_{\text{out}} + 1)}{2}. \quad (15)$$

In addition, there are $RD + N_{\text{out}}$ local detunings that can be varied independently. For the layered structure of Fig. 2 a, this translates to an optimal size of the second layer $N_{2,\text{opt}} = [(N_1 + N_3 + 1)/2]$. For a system with $N_1 = 128$ and $N_3 = 10$ the optimal value $N_{2,\text{opt}}$ is given by $N_{2,\text{opt}} = 70$. Smaller N_2 leads to fewer independent parameters, while larger N_2 introduces redundant parameters. In our simulations, the system with $N_2 = 64$ already achieved the same classification accuracy as a standard artificial neural network, so it may not always be necessary to increase the layer size to $N_{2,\text{opt}}$. Further increasing the number of parameters and introducing redundant parameters can help avoid getting stuck during training. Similarly, even though a neuromorphic system with all-to-all couplings provides a sufficient number of independent parameters, considering a system with multiple hidden layers can be advantageous for the training.

(ii) Similar considerations apply to choosing the number of layers of the network. L should be large enough to provide a sufficient number of training parameters. At the same time, L should not be too large to minimise attenuation losses. For deep networks with $L \gtrsim 3$, localisation effects may become important [6, 7] which may hinder the training, so it is advisable to choose an architecture with sufficiently small L .

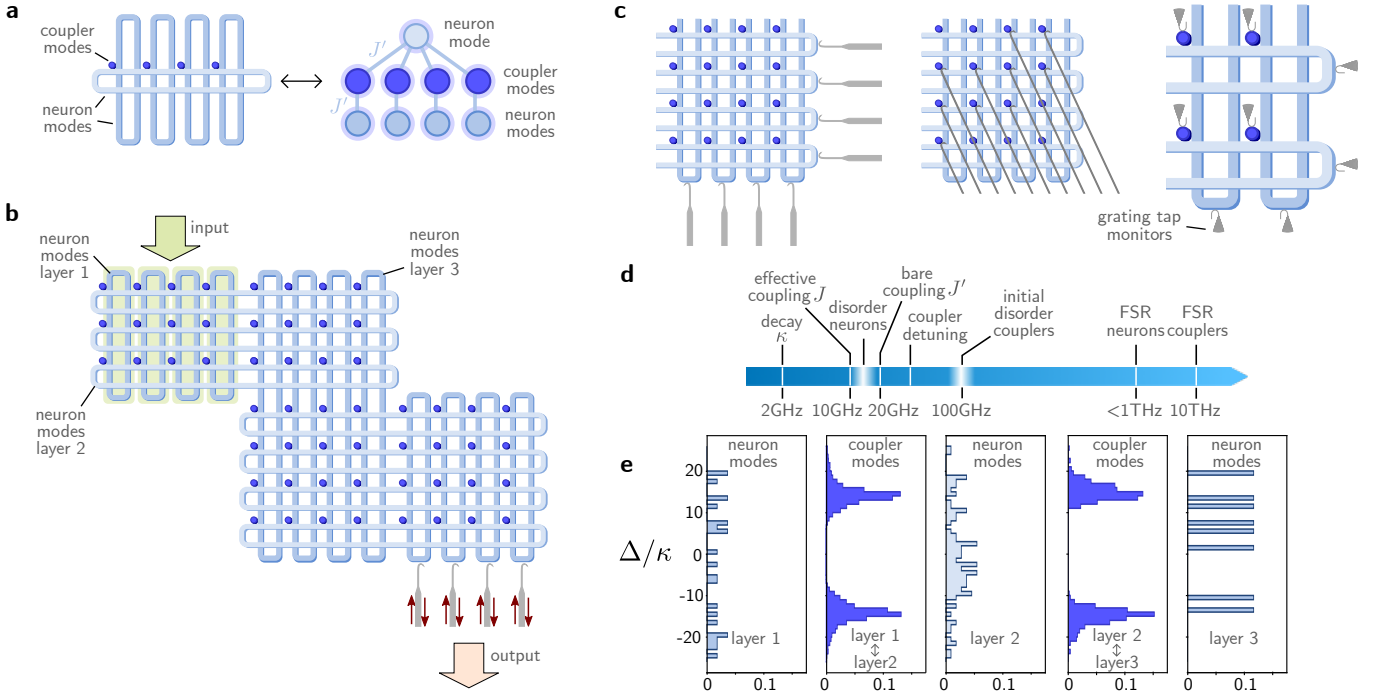


FIG. 3. Implementation with racetrack resonators. **a** Neuron modes are represented by racetrack resonators and racetrack resonators (light blue) of different layers in the neural network are crossed employing techniques to reduce the cross-talk between them. They are then coupled via microring resonators (dark blue)—the coupler modes. Changing the detuning of the coupler modes changes the effective coupling strength J between any two racetrack resonators that cross. This is illustrated in the ersatz image on the right. **b** The advantage of this design is that the system can be scaled up while requiring only a minimal number of waveguide crossing and the neuron modes can still be accessed with waveguides from outside. **c** Three possibilities to measure gradients: following the expression for the gradient in terms of scattering matrix elements, one can either measure the response at the racetrack resonators and use Eq. (8) or directly at the microring resonators and use Eq. (7) to update parameters. Alternative to coupling to waveguides to each resonator, optical grating tap monitors can be utilised which light up according to the output signal at that resonator which can be recorded by a camera [16, 37]. In commercial implementations, grating tap monitors can be complemented by integrated photodetectors for a faster readout. To be sensitive to a specific quadrature (4), the light coupled to the grating tap monitor can be combined with light from a local oscillator (not shown) to perform a homodyne measurement. **d** Scale of the relevant frequencies in an optical implementation. **e** Distribution of neuron modes and coupler mode detunings after training.

(iii) The choice of R depends on the complexity of the training set. For the digit recognition task, $R = 2$ was sufficient, but more complicated datasets can require larger R . We explore this question in the SI, where we show for scalar functions that R determines the approximation order and utilise our system to fit scalar functions. To fit quickly oscillating functions or functions with other sharp features, we require larger R .

Here, we only considered injecting the input at the first layer. However, it could be interesting to explore in the future whether spreading the (replicated) input over different layers holds an advantage, since this would allow to make subsequent layers more ‘non-linear’.

(iv) The intrinsic decay rate determines the sharpest feature that can be resolved, or, equivalently, a larger rate κ smoothens the output functions. It is straightforward to see from Eq. (7) that the derivative of the output w.r.t. a component of the input scales with κ . In the context of one-dimensional function fitting this is straightforward to picture and we provide some examples in the SI. However, the network does not lose its use since the larger decay rate can be compensated for by rescaling the range of the input according

to κ . For instance, for the digit recognition training, we set the image background to $\Delta/\kappa = 5$ and made the pixels storing the number resonant $\Delta/\kappa = 0$ which is still possible in lossy systems.

IV. PROPOSED OPTICAL IMPLEMENTATION

A. Racetrack resonator architecture

The experimental realisation of our proposed system has two main requirements: (i) a sufficiently large number of system parameters (not necessarily all) needs to be tunable; (ii) a large number of modes needs to be sufficiently densely connected. A simple geometry could consist of localised resonators (neuron modes) connected by waveguides (couplings). However, we find a more promising geometry in terms of tunability and spatial layout.

Here we propose an integrated-photonics design based on racetrack resonators as neuron modes that are coupled via microring resonators which realise an effective coupling between

the neuron modes, see Fig. 3 a. This alleviates the need to tune the coupling between the neuron modes directly, since the effective coupling strength can be controlled via the detuning of the coupling resonator. The detunings of the racetrack resonators and the microring resonators could in practice, for instance, be tuned electro-optically [40, 41] or by locally heating the resonator. Alternatively, the couplings can directly be controlled electro-optically [42].

To achieve high connectivity, we propose to cross racetrack resonators of different network layers as in Fig. 3 b and use techniques developed to reduce cross-talk between overlapping waveguides [43, 44] to isolate the racetrack resonators of different layers from one another. The coupling between these neuron modes is controlled via the detuning of the microring resonators placed at the intersection between racetrack resonators which we dub *coupler modes*. The effective coupling J induced by a coupler mode of detuning Δ_{coupler} coupled to two modes with strength J' is given by $J = |J'|^2 / (i\kappa_{\text{coupler}}/2 - (\Delta_{\text{coupler}} - \omega))$, see the SI. The coupler modes also induce a frequency shift on the neuron modes by the same amount which can be compensated by alternating positive and negative detuning in neighbouring coupler modes. A special feature of our proposed design is that the neuron modes are spatially extended whereas the coupling is facilitated by local microring resonators. In fact, the device layout of Fig. 3 c can be seen as a direct visualisation of the coupling matrix $J^{(n)}$ between consecutive layers. In practice, it is advantageous to operate in the regime $|\Delta_{\text{coupler}}| \gg \kappa_{\text{coupler}}$ in which the effective coupling is predominantly coherent. We outline further requirements on Δ_{coupler} and κ_{coupler} as well as a procedure to create a suitable initial configuration of the system in the SI.

Following our procedure to measure gradients based on the scattering matrix, Eqs. (8) and (7), there are three possibilities to obtain the gradients, Fig. 3 c: (i) One only measures the scattering matrix at the racetrack resonators which directly yields the gradients according to Eq. (7) w.r.t. the neuron mode detunings and uses Eq. (8) to infer the gradients for the coupler modes, see the SI. This requires knowledge of the coupling strength J' between coupler modes and neuron modes. (ii) Alternatively, one directly measures the scattering response at the coupler modes and uses Eq. (7) to update all parameters. This immediately yields the gradients for the detuning of the coupler modes while requiring a larger number of (crossed) waveguides. (iii) To reduce the number of waveguides required to measure the system, one can couple grating tap monitors to each resonator which can either be monitored with a camera [16] or integrated photodetectors for a faster readout. To detect a specific quadrature (4), the light coupled to the grating tap monitor can be combined with light from a local oscillator to perform a homodyne measurement.

We simulate the system of the previous section using the ersatz description illustrated in Fig. 3 a to represent the coupler modes. To make the simulation as realistic as possible, we initialised the system such that the detuning range of the initially random detunings was $|\Delta|/\kappa \sim 5$ for the neuron modes and $|\Delta_c|/\kappa \sim 10$ for the coupler modes. This is in-line with realistic values. We provide a complete overview of the rel-

evant frequencies in state-of-the-art electro-optically tunable resonators in Fig. 3 d. We further discuss the requirements for the experimental implementation of our neuromorphic system in more detail in the next section. We train the system for 11,300 iterations and achieve a test accuracy of 92.6% which is similar to the previous simplified setup. The slightly lower accuracy is due to the shorter training time. Fig. 3 e shows the distribution of the detunings of neuron modes and coupler modes after training. The final distributions spread over a range of approximately $|\Delta|/\kappa \sim |\Delta_c|/\kappa \sim 20$ which is well within the tunable range that can be realistically achieved in experiments [45, 46], see Fig. 3 d. We discuss requirements on the physical implementation in further detail in the next section.

We propose that this layout can also find application in the implementation of other neuromorphic systems since it presents a convenient way to achieve high connectivity while keeping the number of waveguide crossings minimal.

B. Experimental requirements

Fig. 3 d shows the relevant frequencies that are important for the implementation of our neuromorphic system. The approach proposed here relies on efficient tuneability of a linear system, both for injecting the input \mathbf{x} and for learning. During training (or device calibration), the detuning range needs to be sufficiently large to overcome any fabrication disorder. In optical systems, disorder in the resonance frequencies typically amounts to about one percent of the free spectral range (FSR), and this can easily be overcome via electrical thermo-optic tuners that have shown [47, 48] tuning by a full FSR in resonator structures of the scale considered here ($10 \mu\text{m}$ to $100 \mu\text{m}$), see Fig. 3 e, operating on time scales [48] of $10 \mu\text{sec}$. For input injection, it is desirable to have faster response times, but conversely the tuning range can be much smaller, on the order of the couplings or decay rates. This could be achieved via electro-optic tuning that has demonstrated resonator tuning by many linewidths at speeds of tens of MHz for microtoroids [49], tens of GHz for photonic-crystal resonators [50] and tens of GHz [45] for racetrack resonators. Tuning ranges even up to hundreds of GHz [46] were demonstrated for racetrack resonators which would be sufficient to compensate for initial frequency disorder. In an alternative approach to coupling the neuron modes via coupler modes, one can tune the couplings between racetrack resonators directly which has been demonstrated at multiples of the linewidth [51]. Without further optimization, one can expect tuning speeds of multiple GHz [45, 52] which with further optimization can be brought down to 100 GHz modulation frequency [53] allowing for fast training and inference. Assuming 5 GHz modulation frequency, one could complete all tuning steps of the entire training with 28,000 iterations over minibatches of 200 images reported above in only 1.1 ms.

We would like to stress that the ideas developed here are very general and apply to a range of linear systems. In particular, we discuss in the SI how to transfer these ideas to analog

electronic circuits.

V. DISCUSSION

We introduced a concept for neuromorphic computing that relies purely on linear wave scattering and does not require any physical non-linearities. By injecting the input in the system parameters and considering the scattering response to a probe signal as output of the neuromorphic system, Fig. 1, we obtain a non-linear function between input data and output. Replicating the input multiple times allows to realise more complex non-linear functions.

With our approach, training is most straight-forward since gradients can be physically measured as products of scattering matrix elements which can be efficiently obtained with a minimal number of measurements. Furthermore, our approach does not require full knowledge of the system. In fact, we can freely add modes and couplings to the systems, which are not trained, that can be treated like a black box and the training procedure remains the same.

We propose a layer architecture and gain analytic insight into its scattering matrix which has a recursive structure reminiscent of standard neural networks. Simulations of our neuromorphic system achieved a classification accuracy on par with the accuracy of standard artificial neural networks.

Furthermore, we propose its implementation in a system of crossed, tunable racetrack resonators which are coupled through tunable microring resonators that allow to control the coupling strength. We suggest that this architecture of crossed racetrack resonators can also find application in the implementation of other neuromorphic systems since it presents an

efficient way to achieve a high connectivity between modes at a minimal number of waveguide crossings.

Our approach solves two major challenges in the field of neuromorphic computing. We not only overcome the challenge of implementing physical non-linearities, our approach also allows to perform physical back-propagation with measured gradients. Moreover, our proposal can be implemented in state-of-the-art integrated and scalable, photonic circuits in which high resonator tuning speeds promise fast and reliable data processing.

In future work, we propose to explore different architectures and the possibility of employing a Floquet scheme for input replication or the use of multiple modes of the same cavity to reduce the hardware overhead. Furthermore, the framework developed here is very general and can be applied to a range of settings beyond optics, for instance, analog electronic circuits. Therefore, our work opens up new possibilities for neuromorphic devices with physical back-propagation over a broad range of platforms.

During the completion of our manuscript, two related works appeared as preprints [54, 55]. In contrast to these works which rely on free-space light propagation, our approach is based on the (steady state) scattering response which has potential advantages in terms of the realisable non-linear functions and allowed us to formulate a simple technique for physically extracting gradients via scattering experiments.

Acknowledgements

We would like to thank Jason Herrmann for sharing useful insight into the tuning speeds of state-of-the-art electro-optical resonators. CCW would like to thank Henning Schomerus for helpful discussion.

-
- [1] D. Marković, A. Mizrahi, D. Querlioz, and J. Grollier, Physics for neuromorphic computing, *Nature Reviews Physics* **2**, 499 (2020).
- [2] G. Wetzstein, A. Ozcan, S. Gigan, S. Fan, D. Englund, M. Soljačić, C. Denz, D. A. Miller, and D. Psaltis, Inference in artificial intelligence with deep optics and photonics, *Nature* **588**, 39 (2020).
- [3] B. J. Shastri, A. N. Tait, T. Ferreira de Lima, W. H. Pernice, H. Bhaskaran, C. D. Wright, and P. R. Prucnal, Photonics for artificial intelligence and neuromorphic computing, *Nature Photonics* **15**, 102 (2021).
- [4] J. Grollier, D. Querlioz, K. Camsari, K. Everschor-Sitte, S. Fukami, and M. D. Stiles, Neuromorphic spintronics, *Nature electronics* **3**, 360 (2020).
- [5] M. Schneider, E. Toomey, G. Rowlands, J. Shainline, P. Tschirhart, and K. Segall, Supermind: a survey of the potential of superconducting electronics for neuromorphic computing, *Superconductor Science and Technology* **35**, 053001 (2022).
- [6] K. Wagner and D. Psaltis, Multilayer optical learning networks, *Applied Optics* **26**, 5061 (1987).
- [7] W. Bogaerts, D. Pérez, J. Capmany, D. A. Miller, J. Poon, D. Englund, F. Morichetti, and A. Melloni, Programmable photonic circuits, *Nature* **586**, 207 (2020).
- [8] N. C. Harris, J. Carolan, D. Bunandar, M. Prabhu, M. Hochberg, T. Baehr-Jones, M. L. Fanto, A. M. Smith, C. C. Tison, P. M. Alsing, and D. Englund, Linear programmable nanophotonic processors, *Optica* **5**, 1623 (2018).
- [9] M. Davies, N. Srinivasa, T.-H. Lin, G. Chinya, Y. Cao, S. H. Choday, G. Dimou, P. Joshi, N. Imam, S. Jain, Y. Liao, C.-K. Lin, A. Lines, R. Liu, D. Mathaikutty, S. McCoy, A. Paul, J. Tse, G. Venkataramanan, Y.-H. Weng, A. Wild, Y. Yang, and H. Wang, Loihi: A neuromorphic manycore processor with on-chip learning, *IEEE Micro* **38**, 82 (2018).
- [10] S. Bandyopadhyay, A. Sludds, S. Krastanov, R. Hamerly, N. Harris, D. Bunandar, M. Streshinsky, M. Hochberg, and D. Englund, Single chip photonic deep neural network with accelerated training, (2022), [arXiv:2208.01623 \[cs.ET\]](https://arxiv.org/abs/2208.01623).
- [11] J. Feldmann, N. Youngblood, C. D. Wright, H. Bhaskaran, and W. H. Pernice, All-optical spiking neurosynaptic networks with self-learning capabilities, *Nature* **569**, 208 (2019).
- [12] J. Feldmann, N. Youngblood, M. Karpov, H. Gehring, X. Li, M. Stappers, M. Le Gallo, X. Fu, A. Lukashchuk, A. S. Raja, *et al.*, Parallel convolutional processing using an integrated photonic tensor core, *Nature* **589**, 52 (2021).
- [13] Y. Zuo, B. Li, Y. Zhao, Y. Jiang, Y.-C. Chen, P. Chen, G.-B. Jo, J. Liu, and S. Du, All-optical neural network with nonlinear activation functions, *Optica* **6**, 1132 (2019).

- [14] T. W. Hughes, M. Minkov, Y. Shi, and S. Fan, Training of photonic neural networks through in situ backpropagation and gradient measurement, *Optica* **5**, 864 (2018).
- [15] R. Hamerly, L. Bernstein, A. Sludds, M. Soljačić, and D. Englund, Large-scale optical neural networks based on photoelectric multiplication, *Physical Review X* **9**, 021032 (2019).
- [16] S. Pai, Z. Sun, T. W. Hughes, T. Park, B. Bartlett, I. A. Williamson, M. Minkov, M. Milanizadeh, N. Abebe, F. Morichetti, *et al.*, Experimentally realized in situ backpropagation for deep learning in photonic neural networks, *Science* **380**, 398 (2023).
- [17] Z. Chen, A. Sludds, R. Davis III, I. Christen, L. Bernstein, L. Ateshian, T. Heuser, N. Heermeier, J. A. Lott, S. Reitzenstein, *et al.*, Deep learning with coherent vcsel neural networks, *Nature Photonics*, 1 (2023).
- [18] B. Scellier and Y. Bengio, Equilibrium propagation: Bridging the gap between energy-based models and backpropagation, *Frontiers in computational neuroscience* **11**, 24 (2017).
- [19] E. Martin, M. Ernoult, J. Laydevant, S. Li, D. Querlioz, T. Petrisor, and J. Grollier, Eqspike: spike-driven equilibrium propagation for neuromorphic implementations, *Iscience* **24** (2021).
- [20] M. Stern, D. Hexner, J. W. Rocks, and A. J. Liu, Supervised learning in physical networks: From machine learning to learning machines, *Physical Review X* **11**, 021045 (2021).
- [21] V. Lopez-Pastor and F. Marquardt, Self-learning machines based on hamiltonian echo backpropagation, arXiv preprint arXiv:2103.04992 (2021).
- [22] X. Guo, T. D. Barrett, Z. M. Wang, and A. Lvovsky, Backpropagation through nonlinear units for the all-optical training of neural networks, *Photonics Research* **9**, B71 (2021).
- [23] J. Spall, X. Guo, and A. I. Lvovsky, Training neural networks with end-to-end optical backpropagation (2023), arXiv:2308.05226 [physics.optics].
- [24] M. J. Filipovich, Z. Guo, M. Al-Qadasi, B. A. Marquez, H. D. Morison, V. J. Sorger, P. R. Prucnal, S. Shekhar, and B. J. Shastri, Silicon photonic architecture for training deep neural networks with direct feedback alignment, *Optica* **9**, 1323 (2022).
- [25] S. Bartunov, A. Santoro, B. A. Richards, L. Marris, G. E. Hinton, and T. Lillicrap, Assessing the scalability of biologically-motivated deep learning algorithms and architectures (2018), arXiv:1807.04587 [cs.LG].
- [26] L. G. Wright, T. Onodera, M. M. Stein, T. Wang, D. T. Schachter, Z. Hu, and P. L. McMahon, Deep physical neural networks trained with backpropagation, *Nature* **601**, 549 (2022).
- [27] E. Pelucchi, G. Fagas, I. Aharonovich, D. Englund, E. Figueroa, Q. Gong, H. Hannes, J. Liu, C.-Y. Lu, N. Matsuda, J.-W. Pan, F. Schreck, F. Sciarrino, C. Silberhorn, J. Wang, and K. D. Jöns, The potential and global outlook of integrated photonics for quantum technologies, *Nature Reviews Physics* **4**, 194 (2022).
- [28] G. Indiveri, B. Linares-Barranco, T. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S.-C. Liu, P. Dudek, P. H. äfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen, Neuro-morphic silicon neuron circuits, *Frontiers in Neuroscience* **5**, 10.3389/fnins.2011.00073 (2011).
- [8] F. Y. Wu, Theory of resistor networks: the two-point resistance, *Journal of Physics A: Mathematical and General* **37**, 6653 (2004).
- [9] V. Čerňanová, J. Brenkuš, and V. Stopjaková, Non-symmetric finite networks: The two-point resistance, *Journal of Electrical Engineering* **65**, 283 (2014).
- [10] J. Cserti, G. Széchenyi, and G. Dávid, Uniform tiling with electrical resistors, *Journal of Physics A: Mathematical and Theoretical* **44**, 215201 (2011).
- [32] C. W. Gardiner and M. J. Collett, Input and output in damped quantum systems: Quantum stochastic differential equations and the master equation, *Phys. Rev. A* **31**, 3761 (1985).
- [33] A. A. Clerk, M. H. Devoret, S. M. Girvin, F. Marquardt, and R. J. Schoelkopf, Introduction to quantum noise, measurement, and amplification, *Rev. Mod. Phys.* **82**, 1155 (2010).
- [34] D. P. García, J. Cruz-Benito, and F. J. García-Peñalvo, Systematic literature review: Quantum machine learning and its applications (2022), arXiv:2201.04093 [quant-ph].
- [35] X. L. Q. Z. W. R. X. L. . Yiming Huang, Hang Lei, Quantum generative model with variable-depth circuit, *Computers, Materials & Continua* **65**, 445 (2020).
- [36] E. Alpaydin and C. Kaynak, Optical Recognition of Handwritten Digits, UCI Machine Learning Repository (1998), DOI: https://doi.org/10.24432/C50P49.
- [37] C. Scarcella, J. S. Lee, C. Eason, M. Antier, J. Bourderionnet, C. Larat, E. Lallier, A. Brignon, T. Spuesens, P. Verheyen, P. Absil, R. Baets, and P. A. O'Brien, Plat4m: Progressing silicon photonics in europe, *Photonics* **3** (2016).
- [6] S. Iida, H. A. Weidenmüller, and J. A. Zuk, Wave propagation through disordered media and universal conductance fluctuations, *Phys. Rev. Lett.* **64**, 583 (1990).
- [7] S. Iida, H. Weidenmüller, and J. Zuk, Statistical scattering theory, the supersymmetry method and universal conductance fluctuations, *Annals of Physics* **200**, 219 (1990).
- [40] A. Guarino, G. Poberaj, D. Rezzonico, R. Degl'Innocenti, and P. Günter, Electro-optically tunable microring resonators in lithium niobate, *Nature Photonics* **1**, 407 (2007).
- [41] Y. Hu, M. Yu, D. Zhu, N. Sinclair, A. Shams-Ansari, L. Shao, J. Holzgrafe, E. Puma, M. Zhang, and M. Lončar, On-chip electro-optic frequency shifters and beam splitters, *Nature* **599**, 587 (2021).
- [42] D. Jia, R. Zhang, C. Yang, Z. Hao, X. Yu, F. Gao, F. Bo, G. Zhang, and J. Xu, Electrically tuned coupling of lithium niobate microresonators, *Opt. Lett.* **48**, 2744 (2023).
- [43] A. M. Jones, C. T. DeRose, A. L. Lentine, D. C. Trotter, A. L. Starbuck, and R. A. Norwood, Ultra-low crosstalk, cmos compatible waveguide crossings for densely integrated photonic interconnection networks, *Opt. Express* **21**, 12002 (2013).
- [44] M. Johnson, M. G. Thompson, and D. Sahin, Low-loss, low-crosstalk waveguide crossing for scalable integrated silicon photonics applications, *Opt. Express* **28**, 12498 (2020).
- [45] J. F. Herrmann, D. J. Dean, C. J. Sarabalis, V. Ansari, K. Multani, E. A. Wollack, T. P. McKenna, J. D. Witmer, and A. H. Safavi-Naeini, Arbitrary electro-optic bandwidth and frequency control in lithium niobate optical resonators (2023), arXiv:2307.16397 [physics.optics].
- [46] Z. Zhou and S. Zhang, Electro-optically tunable racetrack dual microring resonator with a high quality factor based on a lithium niobate-on-insulator, *Optics Communications* **458**, 124718 (2020).
- [47] D. Armani, B. Min, A. Martin, and K. J. Vahala, Electrical thermo-optic tuning of ultrahigh-q microtoroid resonators, *Applied physics letters* **85**, 5439 (2004).
- [48] M. Popovic, M. Dahlem, C. Holzwarth, P. Rakich, H. Smith, E. Ippen, and F. Kartner, Maximizing the thermo-optic tuning range of silicon photonic structures, in *2007 Photonics in Switching* (IEEE, 2007) pp. 67–68.
- [49] C. G. Baker, C. Bekker, D. L. McAuslan, E. Sheridan, and W. P. Bowen, High bandwidth on-chip capacitive tuning of microtoroid resonators, *Optics Express* **24**, 20400 (2016).

- [50] M. Li, J. Ling, Y. He, U. A. Javid, S. Xue, and Q. Lin, Lithium niobate photonic-crystal electro-optic modulator, *Nature Communications* **11**, 4123 (2020).
- [51] W. D. Sacher, W. M. J. Green, S. Assefa, T. Barwicz, H. Pan, S. M. Shank, Y. A. Vlasov, and J. K. S. Poon, Coupling modulation of microrings at rates beyond the linewidth limit, *Opt. Express* **21**, 9722 (2013).
- [52] J. F. Herrmann, V. Ansari, J. Wang, J. D. Witmer, S. Fan, and A. H. Safavi-Naeini, Mirror symmetric on-chip frequency circulation of light, *Nature Photonics* **16**, 603 (2022).
- [53] C. Wang, M. Zhang, X. Chen, M. Bertrand, A. Shams-Ansari, S. Chandrasekhar, P. Winzer, and M. Lončar, Integrated lithium niobate electro-optic modulators operating at cmos-compatible voltages, *Nature* **562**, 101 (2018).
- [54] M. Yildirim, N. U. Dinc, I. Oguz, D. Psaltis, and C. Moser, Nonlinear processing with linear optics (2023), [arXiv:2307.08533](https://arxiv.org/abs/2307.08533) [physics.optics].
- [55] F. Xia, K. Kim, Y. Eliezer, L. Shaughnessy, S. Gigan, and H. Cao, Deep learning with passive optical nonlinear mapping (2023), [arXiv:2307.08558](https://arxiv.org/abs/2307.08558) [physics.optics].

Fully Non-Linear Neuromorphic Computing with Linear Wave Scattering Supplementary Information

Clara C. Wanjura, Florian Marquardt

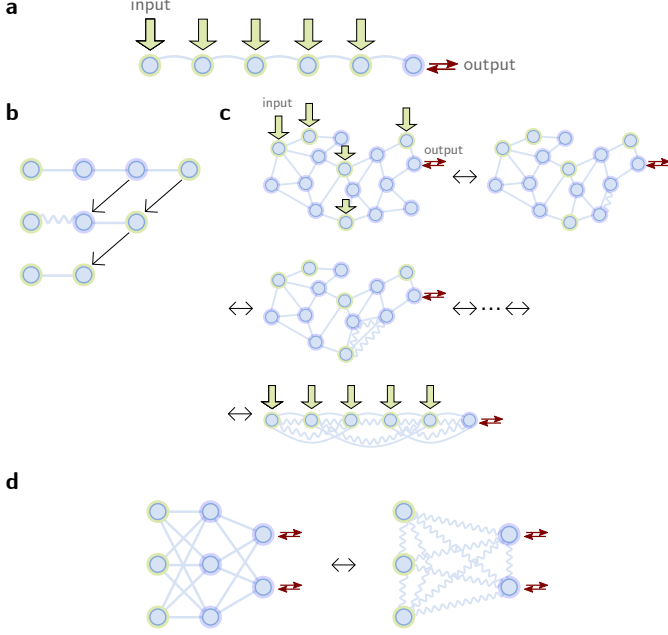


FIG. S1. Mapping of a multimode system to a one-dimensional chain. **a** The scattering matrix of a one-dimensional chain with nearest-neighbour couplings in which the input enters in the detuning of each mode has the form of a generalised continued fraction which can be used to approximate any analytic function. In the limit of an infinite chain length, this approximation approaches the exact result. **b** Reduction rules to map between systems of coupled modes [1]. Eliminating an intermediate mode in a chain of modes coupled via beamsplitter couplings results in effective dissipative coupling with added local decay, Eq. (S3), a further elimination step yields effective coherent beamsplitter coupling with added decay, Eq. (S4). This technique is discussed in more detail and generalised to combinations of beamsplitter and two-mode-squeezing couplings in Ref. [1]. **c** Employing the reduction rules of **b**, one can map *any* system of arbitrarily coupled modes to a one-dimensional chain with couplings of varying range. For clarity, not all resulting effective couplings are shown. **d** Any neuromorphic system with input dimension D and input replication R , e.g., the three-layer system of Fig. 2 **a**, can be mapped to a system consisting only of the input and output modes with all-to-all dissipative couplings. The number of these dissipative couplings equals the number of couplings we can at most control independently in a neuromorphic system with this input dimension and input replication.

I. THE SCATTERING MATRIX CAN APPROXIMATE ARBITRARY NON-LINEAR FUNCTIONS

Here, we prove for the case of scalar input and output that the scattering matrix of a system of coupled modes can be used to approximate arbitrary analytic functions $x \rightarrow f(x) \in \mathbb{R}$. The scalar input $x \in \mathbb{R}$ is encoded in the detunings of each mode and the function is approximated by optimizing (some of) the other system parameters. In the limit of an infinitely large system, the approximation becomes exact.

1. One-dimensional chain with nearest-neighbour couplings

The Green's function of a one-dimensional chain with nearest-neighbour couplings is particularly straightforward to compute. We consider the setup for a chain of length $N = R + 1$, with R the number of input replications, shown in Fig. S1 **a**. The detuning at the j th mode is given by $\Delta_j + x$ for $j \leq N = R + 1$ and nearest-neighbour modes j and $j + 1$ are coupled at strength J_j . The system is probed at the N th site where the detuning is simply Δ_N . Similar to the derivation of the scattering matrix (13) for the layered architecture, we consider the equation of motions in the frequency domain

$$\begin{aligned} \omega a_j(\omega) &= \left(-i \frac{\kappa_j}{2} + (\Delta_j + x) \right) a_j + J_{j-1} a_{j-1} + J_{j+1} a_{j+1}, \\ \omega a_N(\omega) &= \left(-i \frac{\kappa_N + \kappa'_N}{2} + (\Delta_N + x) \right) a_N + J_{N-1} a_{N-1} \\ &\quad - \sqrt{\kappa'_N} a_{N,\text{probe}} \end{aligned} \quad (\text{S1})$$

and recursively solve for $a_j(\omega)$ starting from $a_1(\omega)$ up to $a_N(\omega)$. Together with the input-output boundary conditions $a_{N,\text{res}} = \sqrt{\kappa} a_N + a_{N,\text{probe}}$, we can compute the scattering matrix element $S_{N,N}$

$$S_{N,N}(\omega, x, \theta) = 1 + \frac{\kappa_N}{i \frac{\kappa_N}{2} - (\Delta_N - \omega) + \frac{\kappa_N}{i \frac{\kappa_{N-1}}{2} - (\Delta_{N-1} + x - \omega) + \frac{\kappa_{N-1}}{i \frac{\kappa_{N-2}}{2} - (\Delta_{N-2} + x - \omega) + \frac{\kappa_{N-2}}{i \frac{\kappa_{N-3}}{2} - (\Delta_{N-3} + x - \omega) + \frac{\kappa_{N-3}}{i \frac{\kappa_{N-4}}{2} - (\Delta_{N-4} + x - \omega) + \frac{\kappa_{N-4}}{i \frac{\kappa_{N-5}}{2} - (\Delta_{N-5} + x - \omega) + \frac{\kappa_{N-5}}{i \frac{\kappa_{N-6}}{2} - (\Delta_{N-6} + x - \omega) + \frac{\kappa_{N-6}}{i \frac{\kappa_{N-7}}{2} - (\Delta_{N-7} + x - \omega) + \frac{\kappa_{N-7}}{i \frac{\kappa_{N-8}}{2} - (\Delta_{N-8} + x - \omega) + \frac{\kappa_{N-8}}{i \frac{\kappa_{N-9}}{2} - (\Delta_{N-9} + x - \omega) + \frac{\kappa_{N-9}}{i \frac{\kappa_{N-10}}{2} - (\Delta_{N-10} + x - \omega) + \frac{\kappa_{N-10}}{i \frac{\kappa_{N-11}}{2} - (\Delta_{N-11} + x - \omega) + \frac{\kappa_{N-11}}{i \frac{\kappa_{N-12}}{2} - (\Delta_{N-12} + x - \omega) + \frac{\kappa_{N-12}}{i \frac{\kappa_{N-13}}{2} - (\Delta_{N-13} + x - \omega) + \frac{\kappa_{N-13}}{i \frac{\kappa_{N-14}}{2} - (\Delta_{N-14} + x - \omega) + \frac{\kappa_{N-14}}{i \frac{\kappa_{N-15}}{2} - (\Delta_{N-15} + x - \omega) + \frac{\kappa_{N-15}}{i \frac{\kappa_{N-16}}{2} - (\Delta_{N-16} + x - \omega) + \frac{\kappa_{N-16}}{i \frac{\kappa_{N-17}}{2} - (\Delta_{N-17} + x - \omega) + \frac{\kappa_{N-17}}{i \frac{\kappa_{N-18}}{2} - (\Delta_{N-18} + x - \omega) + \frac{\kappa_{N-18}}{i \frac{\kappa_{N-19}}{2} - (\Delta_{N-19} + x - \omega) + \frac{\kappa_{N-19}}{i \frac{\kappa_{N-20}}{2} - (\Delta_{N-20} + x - \omega) + \frac{\kappa_{N-20}}{i \frac{\kappa_{N-21}}{2} - (\Delta_{N-21} + x - \omega) + \frac{\kappa_{N-21}}{i \frac{\kappa_{N-22}}{2} - (\Delta_{N-22} + x - \omega) + \frac{\kappa_{N-22}}{i \frac{\kappa_{N-23}}{2} - (\Delta_{N-23} + x - \omega) + \frac{\kappa_{N-23}}{i \frac{\kappa_{N-24}}{2} - (\Delta_{N-24} + x - \omega) + \frac{\kappa_{N-24}}{i \frac{\kappa_{N-25}}{2} - (\Delta_{N-25} + x - \omega) + \frac{\kappa_{N-25}}{i \frac{\kappa_{N-26}}{2} - (\Delta_{N-26} + x - \omega) + \frac{\kappa_{N-26}}{i \frac{\kappa_{N-27}}{2} - (\Delta_{N-27} + x - \omega) + \frac{\kappa_{N-27}}{i \frac{\kappa_{N-28}}{2} - (\Delta_{N-28} + x - \omega) + \frac{\kappa_{N-28}}{i \frac{\kappa_{N-29}}{2} - (\Delta_{N-29} + x - \omega) + \frac{\kappa_{N-29}}{i \frac{\kappa_{N-30}}{2} - (\Delta_{N-30} + x - \omega) + \frac{\kappa_{N-30}}{i \frac{\kappa_{N-31}}{2} - (\Delta_{N-31} + x - \omega) + \frac{\kappa_{N-31}}{i \frac{\kappa_{N-32}}{2} - (\Delta_{N-32} + x - \omega) + \frac{\kappa_{N-32}}{i \frac{\kappa_{N-33}}{2} - (\Delta_{N-33} + x - \omega) + \frac{\kappa_{N-33}}{i \frac{\kappa_{N-34}}{2} - (\Delta_{N-34} + x - \omega) + \frac{\kappa_{N-34}}{i \frac{\kappa_{N-35}}{2} - (\Delta_{N-35} + x - \omega) + \frac{\kappa_{N-35}}{i \frac{\kappa_{N-36}}{2} - (\Delta_{N-36} + x - \omega) + \frac{\kappa_{N-36}}{i \frac{\kappa_{N-37}}{2} - (\Delta_{N-37} + x - \omega) + \frac{\kappa_{N-37}}{i \frac{\kappa_{N-38}}{2} - (\Delta_{N-38} + x - \omega) + \frac{\kappa_{N-38}}{i \frac{\kappa_{N-39}}{2} - (\Delta_{N-39} + x - \omega) + \frac{\kappa_{N-39}}{i \frac{\kappa_{N-40}}{2} - (\Delta_{N-40} + x - \omega) + \frac{\kappa_{N-40}}{i \frac{\kappa_{N-41}}{2} - (\Delta_{N-41} + x - \omega) + \frac{\kappa_{N-41}}{i \frac{\kappa_{N-42}}{2} - (\Delta_{N-42} + x - \omega) + \frac{\kappa_{N-42}}{i \frac{\kappa_{N-43}}{2} - (\Delta_{N-43} + x - \omega) + \frac{\kappa_{N-43}}{i \frac{\kappa_{N-44}}{2} - (\Delta_{N-44} + x - \omega) + \frac{\kappa_{N-44}}{i \frac{\kappa_{N-45}}{2} - (\Delta_{N-45} + x - \omega) + \frac{\kappa_{N-45}}{i \frac{\kappa_{N-46}}{2} - (\Delta_{N-46} + x - \omega) + \frac{\kappa_{N-46}}{i \frac{\kappa_{N-47}}{2} - (\Delta_{N-47} + x - \omega) + \frac{\kappa_{N-47}}{i \frac{\kappa_{N-48}}{2} - (\Delta_{N-48} + x - \omega) + \frac{\kappa_{N-48}}{i \frac{\kappa_{N-49}}{2} - (\Delta_{N-49} + x - \omega) + \frac{\kappa_{N-49}}{i \frac{\kappa_{N-50}}{2} - (\Delta_{N-50} + x - \omega) + \frac{\kappa_{N-50}}{i \frac{\kappa_{N-51}}{2} - (\Delta_{N-51} + x - \omega) + \frac{\kappa_{N-51}}{i \frac{\kappa_{N-52}}{2} - (\Delta_{N-52} + x - \omega) + \frac{\kappa_{N-52}}{i \frac{\kappa_{N-53}}{2} - (\Delta_{N-53} + x - \omega) + \frac{\kappa_{N-53}}{i \frac{\kappa_{N-54}}{2} - (\Delta_{N-54} + x - \omega) + \frac{\kappa_{N-54}}{i \frac{\kappa_{N-55}}{2} - (\Delta_{N-55} + x - \omega) + \frac{\kappa_{N-55}}{i \frac{\kappa_{N-56}}{2} - (\Delta_{N-56} + x - \omega) + \frac{\kappa_{N-56}}{i \frac{\kappa_{N-57}}{2} - (\Delta_{N-57} + x - \omega) + \frac{\kappa_{N-57}}{i \frac{\kappa_{N-58}}{2} - (\Delta_{N-58} + x - \omega) + \frac{\kappa_{N-58}}{i \frac{\kappa_{N-59}}{2} - (\Delta_{N-59} + x - \omega) + \frac{\kappa_{N-59}}{i \frac{\kappa_{N-60}}{2} - (\Delta_{N-60} + x - \omega) + \frac{\kappa_{N-60}}{i \frac{\kappa_{N-61}}{2} - (\Delta_{N-61} + x - \omega) + \frac{\kappa_{N-61}}{i \frac{\kappa_{N-62}}{2} - (\Delta_{N-62} + x - \omega) + \frac{\kappa_{N-62}}{i \frac{\kappa_{N-63}}{2} - (\Delta_{N-63} + x - \omega) + \frac{\kappa_{N-63}}{i \frac{\kappa_{N-64}}{2} - (\Delta_{N-64} + x - \omega) + \frac{\kappa_{N-64}}{i \frac{\kappa_{N-65}}{2} - (\Delta_{N-65} + x - \omega) + \frac{\kappa_{N-65}}{i \frac{\kappa_{N-66}}{2} - (\Delta_{N-66} + x - \omega) + \frac{\kappa_{N-66}}{i \frac{\kappa_{N-67}}{2} - (\Delta_{N-67} + x - \omega) + \frac{\kappa_{N-67}}{i \frac{\kappa_{N-68}}{2} - (\Delta_{N-68} + x - \omega) + \frac{\kappa_{N-68}}{i \frac{\kappa_{N-69}}{2} - (\Delta_{N-69} + x - \omega) + \frac{\kappa_{N-69}}{i \frac{\kappa_{N-70}}{2} - (\Delta_{N-70} + x - \omega) + \frac{\kappa_{N-70}}{i \frac{\kappa_{N-71}}{2} - (\Delta_{N-71} + x - \omega) + \frac{\kappa_{N-71}}{i \frac{\kappa_{N-72}}{2} - (\Delta_{N-72} + x - \omega) + \frac{\kappa_{N-72}}{i \frac{\kappa_{N-73}}{2} - (\Delta_{N-73} + x - \omega) + \frac{\kappa_{N-73}}{i \frac{\kappa_{N-74}}{2} - (\Delta_{N-74} + x - \omega) + \frac{\kappa_{N-74}}{i \frac{\kappa_{N-75}}{2} - (\Delta_{N-75} + x - \omega) + \frac{\kappa_{N-75}}{i \frac{\kappa_{N-76}}{2} - (\Delta_{N-76} + x - \omega) + \frac{\kappa_{N-76}}{i \frac{\kappa_{N-77}}{2} - (\Delta_{N-77} + x - \omega) + \frac{\kappa_{N-77}}{i \frac{\kappa_{N-78}}{2} - (\Delta_{N-78} + x - \omega) + \frac{\kappa_{N-78}}{i \frac{\kappa_{N-79}}{2} - (\Delta_{N-79} + x - \omega) + \frac{\kappa_{N-79}}{i \frac{\kappa_{N-80}}{2} - (\Delta_{N-80} + x - \omega) + \frac{\kappa_{N-80}}{i \frac{\kappa_{N-81}}{2} - (\Delta_{N-81} + x - \omega) + \frac{\kappa_{N-81}}{i \frac{\kappa_{N-82}}{2} - (\Delta_{N-82} + x - \omega) + \frac{\kappa_{N-82}}{i \frac{\kappa_{N-83}}{2} - (\Delta_{N-83} + x - \omega) + \frac{\kappa_{N-83}}{i \frac{\kappa_{N-84}}{2} - (\Delta_{N-84} + x - \omega) + \frac{\kappa_{N-84}}{i \frac{\kappa_{N-85}}{2} - (\Delta_{N-85} + x - \omega) + \frac{\kappa_{N-85}}{i \frac{\kappa_{N-86}}{2} - (\Delta_{N-86} + x - \omega) + \frac{\kappa_{N-86}}{i \frac{\kappa_{N-87}}{2} - (\Delta_{N-87} + x - \omega) + \frac{\kappa_{N-87}}{i \frac{\kappa_{N-88}}{2} - (\Delta_{N-88} + x - \omega) + \frac{\kappa_{N-88}}{i \frac{\kappa_{N-89}}{2} - (\Delta_{N-89} + x - \omega) + \frac{\kappa_{N-89}}{i \frac{\kappa_{N-90}}{2} - (\Delta_{N-90} + x - \omega) + \frac{\kappa_{N-90}}{i \frac{\kappa_{N-91}}{2} - (\Delta_{N-91} + x - \omega) + \frac{\kappa_{N-91}}{i \frac{\kappa_{N-92}}{2} - (\Delta_{N-92} + x - \omega) + \frac{\kappa_{N-92}}{i \frac{\kappa_{N-93}}{2} - (\Delta_{N-93} + x - \omega) + \frac{\kappa_{N-93}}{i \frac{\kappa_{N-94}}{2} - (\Delta_{N-94} + x - \omega) + \frac{\kappa_{N-94}}{i \frac{\kappa_{N-95}}{2} - (\Delta_{N-95} + x - \omega) + \frac{\kappa_{N-95}}{i \frac{\kappa_{N-96}}{2} - (\Delta_{N-96} + x - \omega) + \frac{\kappa_{N-96}}{i \frac{\kappa_{N-97}}{2} - (\Delta_{N-97} + x - \omega) + \frac{\kappa_{N-97}}{i \frac{\kappa_{N-98}}{2} - (\Delta_{N-98} + x - \omega) + \frac{\kappa_{N-98}}{i \frac{\kappa_{N-99}}{2} - (\Delta_{N-99} + x - \omega) + \frac{\kappa_{N-99}}{i \frac{\kappa_{N-100}}{2} - (\Delta_{N-100} + x - \omega) + \frac{\kappa_{N-100}}{i \frac{\kappa_{N-101}}{2} - (\Delta_{N-101} + x - \omega) + \frac{\kappa_{N-101}}{i \frac{\kappa_{N-102}}{2} - (\Delta_{N-102} + x - \omega) + \frac{\kappa_{N-102}}{i \frac{\kappa_{N-103}}{2} - (\Delta_{N-103} + x - \omega) + \frac{\kappa_{N-103}}{i \frac{\kappa_{N-104}}{2} - (\Delta_{N-104} + x - \omega) + \frac{\kappa_{N-104}}{i \frac{\kappa_{N-105}}{2} - (\Delta_{N-105} + x - \omega) + \frac{\kappa_{N-105}}{i \frac{\kappa_{N-106}}{2} - (\Delta_{N-106} + x - \omega) + \frac{\kappa_{N-106}}{i \frac{\kappa_{N-107}}{2} - (\Delta_{N-107} + x - \omega) + \frac{\kappa_{N-107}}{i \frac{\kappa_{N-108}}{2} - (\Delta_{N-108} + x - \omega) + \frac{\kappa_{N-108}}{i \frac{\kappa_{N-109}}{2} - (\Delta_{N-109} + x - \omega) + \frac{\kappa_{N-109}}{i \frac{\kappa_{N-110}}{2} - (\Delta_{N-110} + x - \omega) + \frac{\kappa_{N-110}}{i \frac{\kappa_{N-111}}{2} - (\Delta_{N-111} + x - \omega) + \frac{\kappa_{N-111}}{i \frac{\kappa_{N-112}}{2} - (\Delta_{N-112} + x - \omega) + \frac{\kappa_{N-112}}{i \frac{\kappa_{N-113}}{2} - (\Delta_{N-113} + x - \omega) + \frac{\kappa_{N-113}}{i \frac{\kappa_{N-114}}{2} - (\Delta_{N-114} + x - \omega) + \frac{\kappa_{N-114}}{i \frac{\kappa_{N-115}}{2} - (\Delta_{N-115} + x - \omega) + \frac{\kappa_{N-115}}{i \frac{\kappa_{N-116}}{2} - (\Delta_{N-116} + x - \omega) + \frac{\kappa_{N-116}}{i \frac{\kappa_{N-117}}{2} - (\Delta_{N-117} + x - \omega) + \frac{\kappa_{N-117}}{i \frac{\kappa_{N-118}}{2} - (\Delta_{N-118} + x - \omega) + \frac{\kappa_{N-118}}{i \frac{\kappa_{N-119}}{2} - (\Delta_{N-119} + x - \omega) + \frac{\kappa_{N-119}}{i \frac{\kappa_{N-120}}{2} - (\Delta_{N-120} + x - \omega) + \frac{\kappa_{N-120}}{i \frac{\kappa_{N-121}}{2} - (\Delta_{N-121} + x - \omega) + \frac{\kappa_{N-121}}{i \frac{\kappa_{N-122}}{2} - (\Delta_{N-122} + x - \omega) + \frac{\kappa_{N-122}}{i \frac{\kappa_{N-123}}{2} - (\Delta_{N-123} + x - \omega) + \frac{\kappa_{N-123}}{i \frac{\kappa_{N-124}}{2} - (\Delta_{N-124} + x - \omega) + \frac{\kappa_{N-124}}{i \frac{\kappa_{N-125}}{2} - (\Delta_{N-125} + x - \omega) + \frac{\kappa_{N-125}}{i \frac{\kappa_{N-126}}{2} - (\Delta_{N-126} + x - \omega) + \frac{\kappa_{N-126}}{i \frac{\kappa_{N-127}}{2} - (\Delta_{N-127} + x - \omega) + \frac{\kappa_{N-127}}{i \frac{\kappa_{N-128}}{2} - (\Delta_{N-128} + x - \omega) + \frac{\kappa_{N-128}}{i \frac{\kappa_{N-129}}{2} - (\Delta_{N-129} + x - \omega) + \frac{\kappa_{N-129}}{i \frac{\kappa_{N-130}}{2} - (\Delta_{N-130} + x - \omega) + \frac{\kappa_{N-130}}{i \frac{\kappa_{N-131}}{2} - (\Delta_{N-131} + x - \omega) + \frac{\kappa_{N-131}}{i \frac{\kappa_{N-132}}{2} - (\Delta_{N-132} + x - \omega) + \frac{\kappa_{N-132}}{i \frac{\kappa_{N-133}}{2} - (\Delta_{N-133} + x - \omega) + \frac{\kappa_{N-133}}{i \frac{\kappa_{N-134}}{2} - (\Delta_{N-134} + x - \omega) + \frac{\kappa_{N-134}}{i \frac{\kappa_{N-135}}{2} - (\Delta_{N-135} + x - \omega) + \frac{\kappa_{N-135}}{i \frac{\kappa_{N-136}}{2} - (\Delta_{N-136} + x - \omega) + \frac{\kappa_{N-136}}{i \frac{\kappa_{N-137}}{2} - (\Delta_{N-137} + x - \omega) + \frac{\kappa_{N-137}}{i \frac{\kappa_{N-138}}{2} - (\Delta_{N-138} + x - \omega) + \frac{\kappa_{N-138}}{i \frac{\kappa_{N-139}}{2} - (\Delta_{N-139} + x - \omega) + \frac{\kappa_{N-139}}{i \frac{\kappa_{N-140}}{2} - (\Delta_{N-140} + x - \omega) + \frac{\kappa_{N-140}}{i \frac{\kappa_{N-141}}{2} - (\Delta_{N-141} + x - \omega) + \frac{\kappa_{N-141}}{i \frac{\kappa_{N-142}}{2} - (\Delta_{N-142} + x - \omega) + \frac{\kappa_{N-142}}{i \frac{\kappa_{N-143}}{2} - (\Delta_{N-143} + x - \omega) + \frac{\kappa_{N-143}}{i \frac{\kappa_{N-144}}{2} - (\Delta_{N-144} + x - \omega) + \frac{\kappa_{N-144}}{i \frac{\kappa_{N-145}}{2} - (\Delta_{N-145} + x - \omega) + \frac{\kappa_{N-145}}{i \frac{\kappa_{N-146}}{2} - (\Delta_{N-146} + x - \omega) + \frac{\kappa_{N-146}}{i \frac{\kappa_{N-147}}{2} - (\Delta_{N-147} + x - \omega) + \frac{\kappa_{N-147}}{i \frac{\kappa_{N-148}}{2} - (\Delta_{N-148} + x - \omega) + \frac{\kappa_{N-148}}{i \frac{\kappa_{N-149}}{2} - (\Delta_{N-149} + x - \omega) + \frac{\kappa_{N-149}}{i \frac{\kappa_{N-150}}{2} - (\Delta_{N-150} + x - \omega) + \frac{\kappa_{N-150}}{i \frac{\kappa_{N-151}}{2} - (\Delta_{N-151} + x - \omega) + \frac{\kappa_{N-151}}{i \frac{\kappa_{N-152}}{2} - (\Delta_{N-152} + x - \omega) + \frac{\kappa_{N-152}}{i \frac{\kappa_{N-153}}{2} - (\Delta_{N-153} + x - \omega) + \frac{\kappa_{N-153}}{i \frac{\kappa_{N-154}}{2} - (\Delta_{N-154} + x - \omega) + \frac{\kappa_{N-154}}{i \frac{\kappa_{N-155}}{2} - (\Delta_{N-155} + x - \omega) + \frac{\kappa_{N-155}}{i \frac{\kappa_{N-156}}{2} - (\Delta_{N-156} + x - \omega) + \frac{\kappa_{N-156}}{i \frac{\kappa_{N-157}}{2} - (\Delta_{N-157} + x - \omega) + \frac{\kappa_{N-157}}{i \frac{\kappa_{N-158}}{2} - (\Delta_{N-158} + x - \omega) + \frac{\kappa_{N-158}}{i \frac{\kappa_{N-159}}{2} - (\Delta_{N-159} + x - \omega) + \frac{\kappa_{N-159}}{i \frac{\kappa_{N-160}}{2} - (\Delta_{N-160} + x - \omega) + \frac{\kappa_{N-160}}{i \frac{\kappa_{N-161}}{2} - (\Delta_{N-161} + x - \omega) + \frac{\kappa_{N-161}}{i \frac{\kappa_{N-162}}{2} - (\Delta_{N-162} + x - \omega) + \frac{\kappa_{N-162}}{i \frac{\kappa_{N-163}}{2} - (\Delta_{N-163} + x - \omega) + \frac{\kappa_{N-163}}{i \frac{\kappa_{N-164}}{2} - (\Delta_{N-164} + x - \omega) + \frac{\kappa_{N-164}}{i \frac{\kappa_{N-165}}{2} - (\Delta_{N-165} + x - \omega) + \frac{\kappa_{N-165}}{i \frac{\kappa_{N-166}}{2} - (\Delta_{N-166} + x - \omega) + \frac{\kappa_{N-166}}{i \frac{\kappa_{N-167}}{2} - (\Delta_{N-167} + x - \omega) + \frac{\kappa_{N-167}}{i \frac{\kappa_{N-168}}{2} - (\Delta_{N-168} + x - \omega) + \frac{\kappa_{N-168}}{i \frac{\kappa_{N-169}}{2} - (\Delta_{N-169} + x - \omega) + \frac{\kappa_{N-169}}{i \frac{\kappa_{N-170}}{2} - (\Delta_{N-170} + x - \omega) + \frac{\kappa_{N-170}}{i \frac{\kappa_{N-171}}{2} - (\Delta_{N-171} + x - \omega) + \frac{\kappa_{N-171}}{i \frac{\kappa_{N-172}}{2} - (\Delta_{N-172} + x - \omega) + \frac{\kappa_{N-172}}{i \frac{\kappa_{N-173}}{2} - (\Delta_{N-173} + x - \omega) + \frac{\kappa_{N-173}}{i \frac{\kappa_{N-174}}{2} - (\Delta_{N-174} + x - \omega) + \frac{\kappa_{N-174}}{i \frac{\kappa_{N-175}}{2} - (\Delta_{N-175} + x - \omega) + \frac{\kappa_{N-175}}{i \frac{\kappa_{N-176}}{2} - (\Delta_{N-176} + x - \omega) + \frac{\kappa_{N-176}}{i \frac{\kappa_{N-177}}{2} - (\Delta_{N-177} + x - \omega) + \frac{\kappa_{N-177}}{i \frac{\kappa_{N-178}}{2} - (\Delta_{N-178} + x - \omega) + \frac{\kappa_{N-178}}{i \frac{\kappa_{N-179}}{2} - (\Delta_{N-179} + x - \omega) + \frac{\kappa_{N-179}}{i \frac{\kappa_{N-180}}{2} - (\Delta_{N-180} + x - \omega) + \frac{\kappa_{N-180}}{i \frac{\kappa_{N-181}}{2} - (\Delta_{N-181} + x - \omega) + \frac{\kappa_{N-181}}{i \frac{\kappa_{N-182}}{2} - (\Delta_{N-182} + x - \omega) + \frac{\kappa_{N-182}}{i \frac{\kappa_{N-183}}{2} - (\Delta_{N-183} + x - \omega) + \frac{\kappa_{N-183}}{i \frac{\kappa_{N-184}}{2} - (\Delta_{N-184} + x - \omega) + \frac{\kappa_{N-184}}{i \frac{\kappa_{N-185}}{2} - (\Delta_{N-185} + x - \omega) + \frac{\kappa_{N-185}}{i \frac{\kappa_{N-186}}{2} - (\Delta_{N-186} + x - \omega) + \frac{\kappa_{N-186}}{i \frac{\kappa_{N-187}}{2} - (\Delta_{N-187} + x - \omega) + \frac{\kappa_{N-187}}{i \frac{\kappa_{N-188}}{2} - (\Delta_{N-188} + x - \omega) + \frac{\kappa_{N-188}}{i \frac{\kappa_{N-189}}{2} - (\Delta_{N-189} + x - \omega) + \frac{\kappa_{N-189}}{i \frac{\kappa_{N-190}}{2} - (\Delta_{N-190} + x - \omega) + \frac{\kappa_{N-190}}{i \frac{\kappa_{N-191}}{2} - (\Delta_{N-191} + x - \omega) + \frac{\kappa_{N-191}}{i \frac{\kappa_{N-192}}{2} - (\Delta_{N-192} + x - \omega) + \frac{\kappa_{N-192}}{i \frac{\kappa_{N-193}}{2} - (\Delta_{N-193} + x - \omega) + \frac{\kappa_{N-193}}{i \frac{\kappa_{N-194}}{2} - (\Delta_{N-194} + x - \omega) + \frac{\kappa_{N-194}}{i \frac{\kappa_{N-195}}{2} - (\Delta_{N-195} + x - \omega) + \frac{\kappa_{N-195}}{i \frac{\kappa_{N-196}}{2} - (\Delta_{N-196} + x - \omega) + \frac{\kappa_{N-196}}{i \frac{\kappa_{N-197}}{2} - (\Delta_{N-197} + x - \omega) + \frac{\kappa_{N-197}}{i \frac{\kappa_{N-198}}{2} - (\Delta_{N-198} + x - \omega) + \frac{\kappa_{N-198}}{i \frac{\kappa_{N-199}}{2} - (\Delta_{N-199} + x - \omega) + \frac{\kappa_{N-199}}{i \frac{\kappa_{N-200}}{2} - (\Delta_{N-200} + x - \omega) + \frac{\kappa_{N-200}}{i \frac{\kappa_{N-201}}{2} - (\Delta_{N-201} + x - \omega) + \frac{\kappa_{N-201}}{i \frac{\kappa_{N-202}}{2} - (\Delta_{N-202} + x - \omega) + \frac{\kappa_{N-202}}{i \frac{\kappa_{N-203}}{2} - (\Delta_{N-203} + x - \omega) + \frac{\kappa_{N-203}}{i \frac{\kappa_{N-204}}{2} - (\Delta_{N-204} + x - \omega) + \frac{\kappa_{N-204}}{i \frac{\kappa_{N-205}}{2} - (\Delta_{N-205} + x - \omega) + \frac{\kappa_{N-205}}{i \frac{\kappa_{N-206}}{2} - (\Delta_{N-206} + x - \omega) + \frac{\kappa_{N-206}}{i \frac{\kappa_{N-207}}{2} - (\Delta_{N-207} + x - \omega) + \frac{\kappa_{N-207}}{i \frac{\kappa_{N-208}}{2} - (\Delta_{N-208} + x - \omega) + \frac{\kappa_{N-208}}{i \frac{\kappa_{N-209}}{2} - (\Delta_{N-209} + x - \omega) + \frac{\kappa_{N-209}}{i \frac{\kappa_{N-210}}{2} - (\Delta_{N-210} + x - \omega) + \frac{\kappa_{N-210}}{i \frac{\kappa_{N-211}}{2} - (\Delta_{N-211} + x - \omega) + \frac{\kappa_{N-211}}{i \frac{\kappa_{N-212}}{2} - (\Delta_{N-212} + x - \omega) + \frac{\kappa_{N-212}}{i \frac{\kappa_{N-213}}{2} - (\Delta_{N-213} + x - \omega) + \frac{\kappa_{N-213}}{i \frac{\kappa_{N-214}}{2} - (\Delta_{N-214} + x - \omega) + \frac{\kappa_{N-214}}{i \frac{\kappa_{N-215}}{2} - (\Delta_{N-215} + x - \omega) + \frac{\kappa_{N-215}}{i \frac{\kappa_{N-216}}{2} - (\Delta_{N-216} + x - \omega) + \frac{\kappa_{N-216}}{i \frac{\kappa_{N-217}}{2} - (\Delta_{N-217} + x - \omega) + \frac{\kappa_{N-217}}{i \frac{\kappa_{N-218}}{2} - (\Delta_{N-218} + x - \omega) + \frac{\kappa_{N-218}}{i \frac{\kappa_{N-219}}{2} - (\Delta_{N-219} + x - \omega) + \frac{\kappa_{N-219}}{i \frac{\kappa_{N-220}}{2} - (\Delta_{N-220} + x - \omega) + \frac{\kappa_{N-220}}{i \frac{\kappa_{N-221}}{2} - (\Delta_{N-221} + x - \omega) + \frac{\kappa_{N-221}}{i \frac{\kappa_{N-222}}{2} - (\Delta_{N-222} + x - \omega) + \frac{\kappa_{N-222}}{i \frac{\kappa_{N-223}}{2} - (\Delta_{N-223} + x - \omega) + \frac{\kappa_{N-223}}{i \frac{\kappa_{N-224}}{2} - (\Delta_{N-224} + x - \omega) + \frac{\kappa_{N-224}}{i \frac{\kappa_{N-225}}{2} - (\Delta_{N-225} + x - \omega) + \frac{\kappa_{N-225}}{i \frac{\kappa_{N-226}}{2} - (\Delta_{N-226} + x - \omega) + \frac{\kappa_{N-226}}{i \frac{\kappa_{N-227}}{2} - (\Delta_{N-227} + x - \omega) + \frac{\kappa_{N-227}}{i \frac{\kappa_{N-228}}{2} - (\Delta_{N-228} + x - \omega) + \frac{\kappa_{N-228}}{i \frac{\kappa_{N-229}}{2} - (\Delta_{N-229} + x - \omega) + \frac{\kappa_{N-229}}{i \frac{\kappa_{N-230}}{2} - (\Delta_{N-230} + x - \omega) + \frac{\kappa_{N-230}}{i \frac{\kappa_{N-231}}{2} - (\Delta_{N-231} + x - \omega) + \frac{\kappa_{N-231}}{i \frac{\kappa_{N-232}}{2} - (\Delta_{N-232} + x - \omega) + \frac{\kappa_{N-232}}{i \frac{\kappa_{N-233}}{2} - (\Delta_{N-233} + x - \omega) + \frac{\kappa_{N-233}}{i \frac{\kappa_{N-234}}{2} - (\Delta_{N-234} + x - \omega) + \frac{\kappa_{N-234}}{i \frac{\kappa_{N-235}}{2} - (\Delta_{N-235} + x - \omega) + \frac{\kappa_{N-235}}{i \frac{\kappa_{N-236}}{2} - (\Delta_{N-236} + x - \omega) + \frac{\kappa_{N-236}}{i \frac{\kappa_{N-237}}{2} - (\Delta_{N-237} + x - \omega) + \frac{\kappa_{N-237}}{i \frac{\kappa_{N-238}}{2} - (\Delta_{N-238} + x - \omega) + \frac{\kappa_{N-238}}{i \frac{\kappa_{N-239}}{2} - (\Delta_{N-239} + x - \omega) + \frac{\kappa_{N-239}}{i \frac{\kappa_{N-240}}{2} - (\Delta_{N-240} + x - \omega) + \frac{\kappa_{N-240}}{i \frac{\kappa_{N-241}}{2} - (\Delta$$

ous fraction with the tunable parameters J_j and Δ_j . For this type of continued fraction, it has been shown that any analytic function can be expanded into a corresponding type continued fraction, short type continued fraction, [2] which in the limit $N \rightarrow \infty$ becomes exact. Therefore, assuming that J_j and Δ_j can be arbitrarily tunable, $S_{N,N}(\omega, x, \theta)$ can approximate arbitrary analytic functions with N dictating the accuracy of the expansion.

This expansion has a number of advantages over expansions such as power series, since it converges much faster than, for instance, a Taylor series, and it converges for any physical, dynamically stable system such as the ones we are concerned with in this work. In particular, the generalised continued fractions (S2) of stable physical systems are bounded at each order. In contrast to power series the series behaves well for large $|x|$ at finite N since at each order, the approximated function $y(x) \rightarrow 0$ for $|x| \rightarrow \infty$.

2. General system of coupled modes

Let us now consider a general systems of coupled modes such as the one shown in the first panel of Fig. S1 c. The

scalar input enters in the detunings $\Delta_j + x$ of R sites while the other sites have detunings Δ_ℓ which serve as parameters. The system is again probed at a specific site p and we derive the scattering matrix element $S_{p,p}$ of that site.

Here, we can employ a technique to map this system of coupled modes to a linear chain by employing a reduction technique that eliminates individual modes to map a smaller physical system with different couplings and added decay [1]. Specifically, the rules of this reduction technique are shown in Fig. S1 b. In the first row, four modes are coupled via beam-splitter couplings

$$\begin{aligned} -i\omega a_1 &= \left(-\frac{\kappa_1}{2} - i\Delta_1\right) a_1 - iJ_1 a_2 \\ -i\omega a_2 &= \left(-\frac{\kappa_2}{2} - i\Delta_2\right) a_2 - iJ_1^* a_1 - iJ_2 a_3 \\ -i\omega a_3 &= \left(-\frac{\kappa_3}{2} - i\Delta_3\right) a_3 - iJ_2^* a_2 - iJ_3 a_4 \\ -i\omega a_4 &= \left(-\frac{\kappa_4}{2} - i\Delta_4\right) a_4 - iJ_3^* a_3. \end{aligned}$$

Eliminating the third mode induces effective dissipative coupling between the second and third mode with additional local decay and a detuning shift

$$\begin{aligned} -i\omega a_2 &= \left(-\frac{\kappa_2}{2} - i\Delta_2 - \frac{|J_2|^2}{\frac{\kappa_3}{2} + i(\Delta_3 - \omega)}\right) a_2 - iJ_1^* a_1 - \frac{J_2 J_3}{\frac{\kappa_3}{2} + i(\Delta_3 + \omega)} a_4 \\ -i\omega a_4 &= \left(-\frac{\kappa_4}{2} - i\Delta_4 \frac{|J_3|^2}{\frac{\kappa_3}{2} + i(\Delta_3 - \omega)}\right) a_4 - \frac{J_3^* J_2^*}{\frac{\kappa_3}{2} + i(\Delta_3 + \omega)} a_2. \end{aligned} \quad (\text{S3})$$

Subsequently eliminating the second mode a_2 yields again

beamsplitter coupling with additional local decay and a detuning shift [1]

$$\begin{aligned} -i\omega a_1 &= \left(-\frac{\kappa_1}{2} - i\Delta_1 - \frac{|J_1|^2}{-\frac{\kappa_2}{2} - i(\Delta_2 - \omega) - \frac{|J_2|^2}{\frac{\kappa_3}{2} + i(\Delta_3 - \omega)}}\right) a_1 + i\frac{J_1 J_2 J_3}{\frac{\kappa_3}{2} + i(\Delta_3 + \omega)} a_4 \\ -i\omega a_4 &= \left(-\frac{\kappa_4}{2} - i\Delta_4 - \frac{|J_3|^2}{\frac{\kappa_3}{2} + i(\Delta_3 + \omega)} + \frac{|J_2 J_3|^2}{\left(\frac{\kappa_3}{2} + i(\Delta_3 + \omega)\right)^2}\right) a_4 + i\frac{J_3^* J_2^* J_1^*}{\frac{\kappa_3}{2} + i(\Delta_3 - \omega)} a_1. \end{aligned} \quad (\text{S4})$$

The reduction approach can be applied in large networks of modes in which the elimination step of a mode involves replacing all couplings this mode was connected to by the appropriate coherent or dissipative coupling as described above. This technique is discussed in more detail and generality in the Supplementary Information of Ref. [1] where also the generality of the argument to two-mode and single-mode squeezing is derived; a related technique is employed in Ref. [3, 4].

pling strength, effective local decay and detuning shifts are not relevant. We merely employ this reduction technique to map one physical system to one that is easier to analyse. In particular, we map the generic system of multiple modes, such as the first panel in Fig. S1 c, to a chain with arbitrary, possibly long-range couplings, such as the chain in the last panel of Fig. S1 c, consisting only of the modes whose detunings encode the input and the mode p which is used to probe the system which is the $(L+1)$ th modes of the chain. This is pos-

For the following arguments, the values of the effective cou-

sible by performing multiple reduction steps eliminating all other modes as is sketching in the other panels of Fig. S1 c. The couplings between the modes of this reduced model are now a combination of coherent beamsplitter coupling and dissipative coupling, with coupling constants which may even be complex. We introduce the effective coupling strengths $g_{j,\ell}$ for the effective coupling between modes j and ℓ in which we indexed modes from left to right. We note that $g_{j,\ell}$ need not be symmetric, i.e., $g_{j,\ell} \neq g_{\ell,j}^*$. The on-site terms, i.e., decay rate and detuning, were also modified by the reduction to the chain and we define $\mu_j \equiv -\frac{\kappa_j}{2} - i\Delta_j$ as the effective on-site term of the j th site in the equations of motion for the reduced chain. We can then proceed as before for the chain with nearest-neighbour couplings to derive the scattering matrix element at the $(R+1)$ th site, $S_{L+1,L+1} = S_{p,p}$ by eliminating mode by mode along the chain

$$S_{R+1,R+1}(\omega, x, \theta) = 1 + \frac{\kappa_{R+1}}{\tilde{\mu}_{R+1}} \quad (\text{S5})$$

with

$$\tilde{\mu}_j = \mu_j + \frac{g_{j,j-1}g_{j-1,j}}{\tilde{\mu}_{j-1}} + \sum_{\substack{\text{paths of} \\ \text{length } \ell: \\ 3 \leq \ell \leq 2^j}} (-i)^\ell \frac{g_{j,m_1}g_{m_1,m_2} \cdots g_{m_\ell,j}}{\tilde{\mu}_{m_1}\tilde{\mu}_{m_2} \cdots \tilde{\mu}_{m_\ell}} \quad (\text{S6})$$

with $\tilde{\mu}_1 = \mu_1$. Here, the sum goes over all paths from mode j to modes $m_1, m_2, \dots, m_\ell < j$ and come back to mode j that have a length smaller than 2^j and involve at least one coupling of a range longer than one, i.e., next-nearest neighbour coupling and beyond. Note that the scalar input x enters as summand in each μ_j .

As result of this exercise, we see that Eq. (S5) reduces to Eq. (S2) in the case of only symmetric nearest-neighbour couplings in which case we know that $S_{R+1,R+1}(\omega, x, \theta)$ can represent arbitrary, non-linear analytic functions. In the general case, the expression for $S_{R+1,R+1}(\omega, x, \theta)$, Eq. (S5), contains sums of generalised continued fractions in which the coefficients are again generalised continued fractions. So Eq. (S5) has both more parameters and contains more general terms, e.g., powers of x , so is by no means more restrictive than expression (S2) which already allowed to approximate arbitrary analytic functions. We therefore conclude that a scattering matrix of any linear system in which the input enters R times can be used to approximate arbitrary analytic functions. The number of input replications L determines the recursive depth of expression (S5) and therefore plays a crucial role for the accuracy of the approximation.

To exemplify the expressive power of the scattering matrix, we employ the system sketched in Fig. S2 a to fit different non-linear functions ranging from a combination of hyperbolic tangents to oscillating functions. The system we consider consists of a first layer with R modes in which the input is replicated R times in the detunings $\Delta_j^{(1)} + x$ of the first layer, then a second layer of variable size and an output layer with only one mode. The decay rate κ is considered equal at all sites.

We note that functions with sharper features or many oscillations require larger R than smoother functions. Furthermore, we selected for each plot the largest value of κ that still yielded a good approximation of the function. As we would expect from Eq. (7) of the main text, a larger derivative requires a smaller decay rate κ .

II. EXPRESSIBILITY

1. Number of independent parameters

In general, a scattering matrix element is of the form $S_{j,\ell}(0) = 1 + i\kappa_j(-1)^{j+\ell} \frac{\det H^{(j,\ell)}}{\det H}$. For a neuromorphic system with multi-dimensional input and output, both numerator and denominator automatically contains terms such as x_1x_2 , x_2x_3 and $x_1x_2x_3$ even at $R = 1$ with some coefficients defined through the other system parameters. In fact, with D the input dimension and R the number of input replications, the number of these terms grows as

$$\sum_{j=1}^{DR} \binom{DR}{j} = 2^{DR} - 1. \quad (\text{S7})$$

However, while, in general, the number of terms grows exponentially with the input dimension and input replication, not all of these coefficients are independent parameters that can be controlled by adjusting the other system parameters. The number of independent parameters is typically much smaller.

To find the actual number of independent parameters we can maximally achieve for a neuromorphic system with input dimension D , input replication R and output dimension N_{out} , including systems with the three-layer architecture of Fig. 2 a, we consider the map to the minimal reduced system which still contains all input sites and probe sites, see Fig. S1 d. The number of couplings entering in this system is equal to the maximal number of independent parameters we can maximally achieve for this given input dimension D and input replication R . We find that the minimal system consists only of input and output modes with all-to-all dissipative couplings between them, see Fig. S1 d. The total number of couplings in this system is

$$N_{\text{max}} = \frac{(DR + N_{\text{out}})(DR + N_{\text{out}} + 1)}{2}. \quad (\text{S8})$$

In the three-layer architecture, $N_1 = DR$ and $N_3 = N_{\text{out}}$. In addition to that, we have $DR + N_{\text{out}}$ local terms (decay rates, detunings). Expression (S8) is the maximal number of couplings we can at most control independently in any system at given D , R and N_{out} independent of the architecture.

Next, we come back to the three-layer architecture, Fig. 2 a, and we ask how large the second layer N_2 has to be to have enough independent parameters to control all of these effective couplings independently. (The $N_1 + N_3$ local parameters are automatically independent parameters since they also exist in the original system.) The total number of couplings

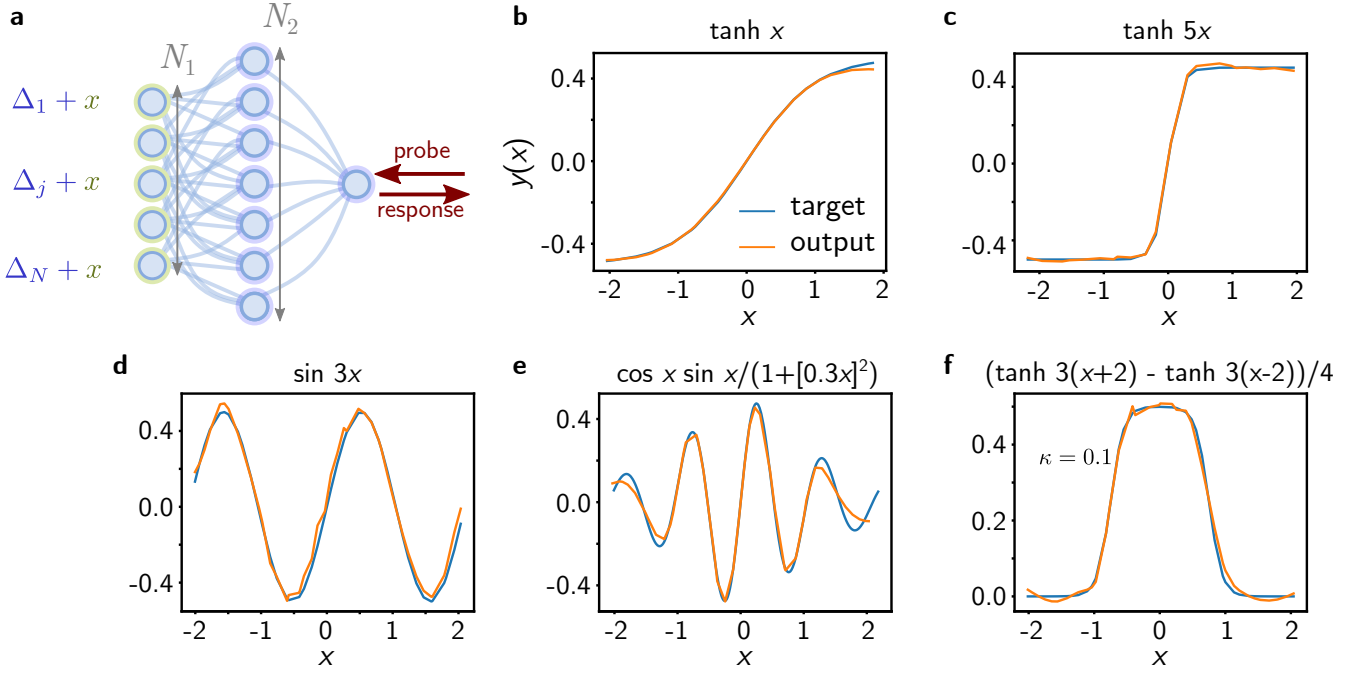


FIG. S2. **One-dimensional function fitting.** Examples for one-dimensional functions fitted with the network shown in **a**. The input enters as offset to the detuning of the first layer and enters $R = N_1$ number of times. A second layer of variable size is connected to a single site. The imaginary part of the scattering response at that site serves as output. The parameters used in the fits were **b** $N_1 = 20, N_2 = 20, \kappa = 1$; **c** $N_1 = 60, N_2 = 30, \kappa = 0.2$; **d-f** $N_1 = 60, N_2 = 30, \kappa = 0.1$. Note that functions with sharper features require larger R and larger N_2 . Typically, κ has to be small to capture sharp features, although, this can to some extent be compensated by rescaling the range of input values by $1/\kappa$.

between layers of sizes N_1, N_2 and N_3 is given by

$$N_{\text{control}} = N_1 N_2 + N_2 N_3. \quad (\text{S9})$$

This is the number of couplings, we can actually control. When $N_{\text{control}} < N_{\text{max}}$ we do not utilise the full potential of the architecture, whereas when $N_{\text{control}} > N_{\text{max}}$, some of the couplings are, at least in principle, redundant and the system has a hardware overhead.

The optimal case is when $N_{\text{control}} = N_{\text{max}}$. Equating expressions (S8) and (S9), we find that to be able to independently control all parameters, N_2 should be larger or equal to

$$N_2 \geq \left\lceil \frac{N_1 + N_3 + 1}{2} \right\rceil. \quad (\text{S10})$$

For the system we consider in the main text with $N_1 = 128$ and $N_3 = 10$, we find $N_2 \geq 70$. This explains why the classification accuracy is limited for smaller N_2 and why we achieve the highest classification accuracy for $N_2 = 64$.

Depending on the training data, it may also be worth to explore larger N_2 than this bound, even though the system is then over determined, since this may have advantages during training.

2. Remarks about systems encoding functions with multi-dimensional input and output

In the main text, we derive the expression for the scattering matrix (13) of a neuromorphic system with a layered architecture, Fig. 2 a. The mathematical form of this equation resembles that of the generalised continued fraction (S2) but in matrix form.

In the literature of condensed matter physics, expressions such as Eq. (13) are known as recursive Green's functions [5] and have been employed to study various condensed matter systems. It is expected that for larger depth L , these networks display localisation when subject to disorder in the couplings and detunings [6, 7]—this may possibly impede the training of such networks (correlations between input and output layer are expected to decrease with L in which case the gradients will also become small). For most applications, $L = 3$ or 4 may already be sufficient. If the application requires deeper networks, introducing gain, e.g., in the form of two-mode or single-mode squeezing, or, alternatively, introducing a few strategic couplings of longer range between layers that are further apart, may resolve this issue of localisation.

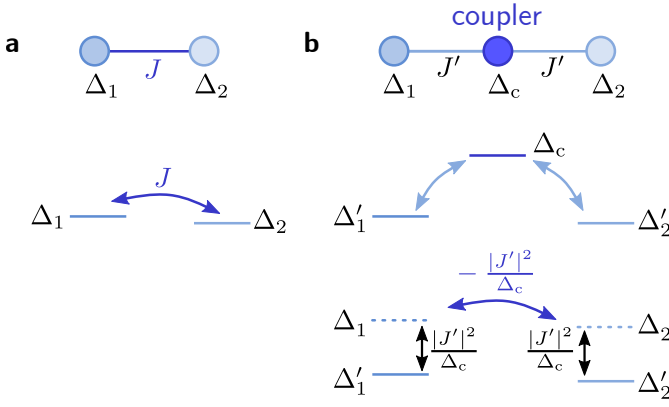


FIG. S3. **Effective coupling mediated by coupler modes.** The aim of the implementation, Fig. 3 a, is to replace a the tunable couplings of strength J by b a coupler mode at tunable detuning Δ_c coupled to two neuron modes at fixed coupling strength J' . This gives rise to an effective coupling between the two neuron modes at strength $J = |J'|^2 / (i\frac{\kappa_c}{2} + \omega - \Delta_c)$, Eq. (S16), so the effective coupling strength J can be tuned by tuning Δ_c . At the same time, the detuning of each mode is shifted by the same amount. To map back to exactly the same situation as in a, this shift needs to be compensated when tuning Δ'_j , Eq. (S17).

III. DERIVATION OF THE GRADIENT FORMULAS

Here, we derive the derivative of the scattering matrix w.r.t. a parameter θ_j , Eq. (6),

$$\frac{\partial S(\omega, \mathbf{x}, \theta)}{\partial \theta_j} = -i\sqrt{\kappa} \frac{\partial}{\partial \theta_j} [\omega \mathbb{1} - H(\mathbf{x}, \theta)]^{-1} \sqrt{\kappa}. \quad (\text{S11})$$

The derivative of the matrix inverse M^{-1} w.r.t. a parameter follows from

$$\begin{aligned} 0 &= \frac{\partial}{\partial \theta_j} \mathbb{1} = \frac{\partial}{\partial \theta_j} (M^{-1}(\theta)M(\theta)) \\ &= \left(\frac{\partial}{\partial \theta_j} M^{-1}(\theta) \right) M(\theta) + M^{-1}(\theta) \left(\frac{\partial}{\partial \theta_j} M(\theta) \right). \end{aligned} \quad (\text{S12})$$

Solving for $\frac{\partial}{\partial \theta_j} M^{-1}(\theta)$, we obtain

$$\frac{\partial}{\partial \theta_j} M^{-1}(\theta) = M^{-1}(\theta) \left(\frac{\partial}{\partial \theta_j} M(\theta) \right) M^{-1}(\theta). \quad (\text{S13})$$

Using this expression for Eq. (S11), we obtain Eq. (6) of the main text.

IV. FURTHER DETAILS ABOUT THE IMPLEMENTATION

A. Derivation of the effective coupling

In Fig. 3 we introduced coupler modes mediating the coupling between two neuron modes. Here, we derive how a

change in the detuning of the coupler mode changes the effective coupling between two neuron modes. The aim is to achieve an effective, predominantly coherent coupling between two neuron modes which can be tuned by adjusting the detuning of the coupler mode, see Fig. S3. This allows to transfer all results obtained for a network with tunable coherent coupling strengths to this system with coupler modes.

For simplicity consider the two neuron modes a_1 and a_2 at detunings Δ_1 and Δ_2 , respectively, coupled via coupler mode c at detuning Δ_c and experiencing decay at κ_c to which both a_1 and a_2 are coupled at strengths $J'/2$. The corresponding, Fourier transformed equations of motion are given by

$$\begin{aligned} -i\omega a_1 &= -\left(\frac{\kappa}{2} + i\Delta'_1\right) a_1 - iJ'c \\ -i\omega a_2 &= -\left(\frac{\kappa}{2} + i\Delta'_2\right) a_2 - i(J')^*c \\ -i\omega c &= -\left(\frac{\kappa_c}{2} + i\Delta_c\right) c - i(J')^*a_1 - iJ'a_2. \end{aligned} \quad (\text{S14})$$

Solving for $c(\omega)$, we find

$$c = \frac{J'}{i\frac{\kappa_c}{2} + \omega - \Delta_c} (a_1 + a_2) \quad (\text{S15})$$

which we insert in Eqs. (S14)

$$\begin{aligned} -i\omega a_1 &= -\left(\frac{\kappa}{2} + i\Delta'_1\right) a_1 - i\frac{|J'|^2}{i\frac{\kappa_c}{2} + \omega - \Delta_c} (a_1 + a_2) \\ -i\omega a_2 &= -\left(\frac{\kappa}{2} + i\Delta'_2\right) a_2 - i\frac{|J'|^2}{i\frac{\kappa_c}{2} + \omega - \Delta_c} (a_1 + a_2). \end{aligned} \quad (\text{S16})$$

There are two key points to notice: (i) the coupler mode induces effective coupling between the two modes in which the coupling strength is complex, i.e., it results in both coherent and dissipative coupling. To come as close as possible to an effectively coherent coupling, we require $\kappa_c \ll \Delta_c$, in which case the coherent coupling strength is approximately $-|J'|^2 / (\Delta_c - \omega)$. (ii) the coupler modes lead to an on-site shift in detuning and contributes towards the dissipation. To compensate for this detuning shift, we offset Δ'_j by the same amount, i.e.,

$$\Delta'_j = \Delta_j + |J'|^2 / \Delta_c. \quad (\text{S17})$$

This gives rise to

$$\begin{aligned} -i\omega a_1 &= -\left(\frac{\kappa}{2} + i\Delta_1\right) a_1 + i\frac{|J'|^2}{\Delta_c} a_2 \\ -i\omega a_2 &= -\left(\frac{\kappa}{2} + i\Delta_2\right) a_2 + i\frac{|J'|^2}{\Delta_c} a_1 \end{aligned} \quad (\text{S18})$$

which are the equations describing the scenario depicted in Fig. S3 a with $J = -|J'|^2 / \Delta_c$. Note that if the neuron modes are coupled to multiple coupler modes, the detuning shifts from all coupler modes have to be compensated, so, more generally,

$$\Delta'_j = \Delta_j + \sum_{\ell} |J'_{\ell}|^2 / \Delta_{\ell,c} \quad (\text{S19})$$

in which ℓ sums over all coupler modes that mode j is connected to.

In practise, a neuron mode can be connected to a large number of coupler modes, so to avoid large detuning shifts (which may then even be outside the tuning range), one can alternate positive and negative detuning of the coupler modes. In that case, most of the terms in Eq. (S19) balance out and the net shift is very small.

B. Creating the initial configuration in the proposed implementation

To start training the neuromorphic system, it is first necessary to calibrate the setup, so any fabrication disorder should not be larger than the tunable range, see Fig. 3 d. Once the setup is calibrated, one can fine tune the detunings to a suitable initial configuration. We found in our simulations that starting from configurations in which the neuron modes are effectively approximately resonant (up to some small disorder), i.e., taking into account the detuning shifts (S19) due to the coupler modes, leads to a faster and reliable convergence of the training. The detunings of the coupler modes should be large enough to enable sufficiently strong coupling between neuron modes. In particular, $\Delta_c/\kappa_c \gg 1$ ensures that the coherent coupling dominates over the dissipative coupling induced by the coupler mode, Eq. (S14). In addition, some small disorder in the effective coupling strengths helps to avoid getting stuck early in the training.

Concretely, in numerical experiments, it has proven helpful to generate a random initial configuration in the ersatz description, i.e., without the coupler modes and to translate this back to a description in terms of coupler modes, i.e., solving for Δ_c . Subsequently, one chooses alternating sign for the coupler mode detunings and calculates the resulting on-site shift (S19) which is then compensated for by changing the neuron mode detuning which effectively makes the neuron modes resonant again. This serves as initial configuration for the training. Subsequent training steps can then be performed *in situ* based purely on the measured scattering matrix.

V. ANALOG ELECTRICAL CIRCUITS

The general idea of exploiting the non-linear structure of the Green's function of a linear system transfers to other systems such as analog electrical circuits. Here, we sketch the idea for a circuit network of resistors, but the same principle holds for more general electric circuits that can, for instance, also contain inductors and capacitors.

Consider an electrical circuit in which resistors $\rho_{j,\ell}$ connect nodes j and ℓ in a resistor network. Kirchhoff's law establishes a relation between the current I_n at each node n and the voltage V_n via the connectivity matrix L or Laplacian of the network. Concretely,

$$-I_n = \sum_m L_{n,m} V_m. \quad (\text{S20})$$

with

$$L_{m,n} = \begin{cases} \frac{1}{\rho_{m,n}} & : m \neq n \\ -\sum_m \frac{1}{\rho_{m,n}} & : m = n \end{cases}. \quad (\text{S21})$$

Our ultimate goal is to calculate the resistance between nodes j and ℓ . To that purpose, we solve the above equation for the voltages, by calculating the circuit's Green's function [8–10]

$$G = (L - \mathbf{v} \cdot \mathbf{v}^T)^{-1} \quad (\text{S22})$$

with $\mathbf{v} = (1, 1, \dots, 1)^T$. This matrix of ones $\mathbf{v} \cdot \mathbf{v}^T$ is the outer product of the eigenvector corresponding to the zero eigenvalue of L which accounts for the fact that adding a constant, uniform potential to every node yields the same circuit with the same relative voltages.

The two-point resistance $R_{j,\ell}$ between nodes j and ℓ is then given through elements of this Green's function

$$R_{j,\ell} = G_{j,j} + G_{\ell,\ell} - (G_{j,\ell} + G_{\ell,j}). \quad (\text{S23})$$

The elements $R_{j,\ell}$ define the *impedance matrix* of the system. Similar as for the scattering matrix, the impedance matrix is a non-linear function of the resistors $\rho_{m,n}$, so we can interpret a set of tunable $\rho_{m,n}$ to be the input \mathbf{x} of our neuromorphic system and some other tunable resistors $\rho_{m',n'}$ to be the learning parameters θ . The output is the measured two-point resistance $R_{j,\ell}$ between a suitable set of nodes. If the ground of the circuit is fixed at a certain node, e.g. node j , the output is proportional to the measured voltage difference to nodes ℓ such that the output can be recorded with a single measurement step. As before, the linear nature of the system enables us to directly measure gradients via the impedance matrix.

[1] C. C. Wanjura, J. J. Slim, J. del Pino, M. Brunelli, E. Verhagen, and A. Nunnenkamp, Quadrature nonreciprocity in bosonic networks without breaking time-reversal symmetry, *Nature Physics* [10.1038/s41567-023-02128-x](https://doi.org/10.1038/s41567-023-02128-x) (2023).
 [2] E. Frank, Corresponding type continued fractions, *American Journal of Mathematics* **68**, 89 (1946).

[3] L. Ranzani and J. Aumentado, Graph-based analysis of nonreciprocity in coupled-mode systems, *New Journal of Physics* **17**, 023024 (2015).
 [4] O. Naaman and J. Aumentado, Synthesis of parametrically coupled networks, *PRX Quantum* **3**, 020201 (2022).
 [5] F. Sols, Recursive tight-binding green's function method: Ap-

- plication to ballistic and dissipative transport in semiconductor nanostructures, in *Quantum Transport in Ultrasmall Devices: Proceedings of a NATO Advanced Study Institute on Quantum Transport in Ultrasmall Devices, held July 17–30, 1994, in II Ciocco, Italy*, edited by D. K. Ferry, H. L. Grubin, C. Jacoboni, and A.-P. Jauho (Springer US, Boston, MA, 1995) pp. 329–338.
- [6] S. Iida, H. A. Weidenmüller, and J. A. Zuk, Wave propagation through disordered media and universal conductance fluctuations, *Phys. Rev. Lett.* **64**, 583 (1990).
- [7] S. Iida, H. Weidenmüller, and J. Zuk, Statistical scattering theory, the supersymmetry method and universal conductance fluctuations, *Annals of Physics* **200**, 219 (1990).
- [8] F. Y. Wu, Theory of resistor networks: the two-point resistance, *Journal of Physics A: Mathematical and General* **37**, 6653 (2004).
- [9] V. Čerňanová, J. Brenkuš, and V. Stopjaková, Non-symmetric finite networks: The two-point resistance, *Journal of Electrical Engineering* **65**, 283 (2014).
- [10] J. Cserti, G. Széchenyi, and G. Dávid, Uniform tiling with electrical resistors, *Journal of Physics A: Mathematical and Theoretical* **44**, 215201 (2011).