Theory of Neuromorphic Computing by Waves:
Machine Learning by Rogue Waves, Dispersive Shocks, and Solitons

Giulia Marcucci[ ], Davide Pierangeli, and Claudio Conti[*]

*Institute for Complex Systems, National Research Council (ISC-CNR), Via dei Taurini 19, 00185 Rome, Italy
and Department of Physics, Sapienza University, Piazzale Aldo Moro 2, 00185 Rome, Italy*

We study artificial neural networks with nonlinear waves as a computing reservoir. We discuss universality and the conditions to learn a dataset in terms of output channels and nonlinearity. A feed-forward three-layered model, with an encoding input layer, a wave layer, and a decoding readout, behaves as a conventional neural network in approximating mathematical functions, real-world datasets, and universal Boolean gates. The rank of the transmission matrix has a fundamental role in assessing the learning abilities of the wave. For a given set of training points, a threshold nonlinearity for universal interpolation exists. When considering the nonlinear Schrödinger equation, the use of highly nonlinear regimes implies that solitons, rogue, and shock waves do have a leading role in training and computing. Our results may enable the realization of novel machine learning devices by using diverse physical systems, as nonlinear optics, hydrodynamics, polaritonics, and Bose-Einstein condensates. The application of these concepts to photonics opens the way to a large class of accelerators and new computational paradigms. In complex wave systems, as multimodal fibers, integrated optical circuits, random, topological devices, and metasurfaces, nonlinear waves can be employed to perform computation and solve complex combinatorial optimization.

Deep artificial neural networks (DNN) have unprecedented success in learning large datasets for, e.g., image classification or speech synthesis [1,2]. However, when the number of weights grows, optimization becomes hard. In recent years, many groups proposed new computational models, which are less demanding with respect to DNN in terms of the number of weights to be trained, and more versatile for their physical implementation. These models include neuromorphic computing and random neural networks, which require the training of only a subset of nodes [3–10]. This fact opens the possibility of using many physical systems for large-scale computing [11]. Following earlier investigations [12,13], various groups reported on computing machines with propagating waves, like Wi-Fi waves [14], polaritons [15,16], and lasers [17–26]. The photonic accelerators speed up large-scale neural networks [27–30] or Ising machines [31–37]. Photonic architectures for machine learning might also pave the way to the solutions of complex combinatorial problems. For example, the recent article on Ising recurrent systems in integrated photonics [19] extends the coherent hardware for machine learning to combinatorial problems, as the works on the spatial photonic Ising machine [35,38–40] expand the hardware previously used for random neural networks and reservoir computing.

Despite these many investigations, fundamental questions remain open. What kind of computation can waves do? Are linear or nonlinear waves universal computing machines? Which parameter quantifies the learning process? We need a general theory that links the physics of nonlinear waves with the maths of machine learning and reservoir computing [41]. Also, no theoretical work addresses highly nonlinear processes, such as nonlinear gases, shocks, and solitons (recently subject of intense research [42–48]), to perform computation at a large scale. In this Letter, we study computational machines in which the nonlinear waves replace the internal layers of the neural network. We discuss the conditions for learning, and demonstrate function interpolation, datasets, and Boolean operations.

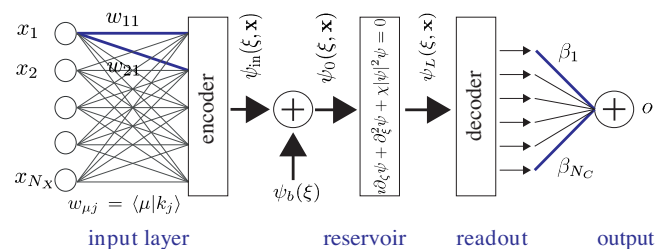*The wave-layer model.*—Figure 1 shows our model, a "single wave-layer feed-forward network" (SWFN), in

FIG. 1. Single wave-layer feed forward neural network. The input vector $\mathbf{x}$ is encoded in the input wave ψ_0, including a bias wave function ψ_b. The wave evolves according to a nonlinear partial differential equation. The readout layer decodes by sampling the modulus square $|\psi_L|^2$ of the final wave in N_C readout channels, linearly combined to give the output o.

analogy with the "single layer feed-forward neural network" (SLFN) [2,4,5]. An input vector $\mathbf{x}$ with components $x_1, x_2, \ldots, x_{N_X}$ seeds the network. N_C output channels g_j with $j = 1, 2, \ldots, N_C$ form the output vector $\mathbf{g}$. The channels are linearly combined at the readout $o = \sum_{j=1}^{N_C} \beta_j g_j = \boldsymbol{\beta} \cdot \mathbf{g}$. $\boldsymbol{\beta}$ with components β_j is the vector of the weights determined by the training dataset.

The link between $\mathbf{g}$ and $\mathbf{x}$ is given by the complex field $\psi_L(\xi, \mathbf{x})$ emerging from the propagation of the initial state $\psi_0(\xi, \mathbf{x})$, with ξ the transverse coordinate. The readout functions are samples $g_j = |\psi_L(\xi_j, \mathbf{x})|^2$ such that

$$o(\mathbf{x}) = \sum_{j=1}^{N_C} \beta_j g_j(\mathbf{x}) = \boldsymbol{\beta} \cdot \mathbf{g}(\mathbf{x}). \quad (1)$$

The goal is determining the weights $\boldsymbol{\beta}$ to approximate a target function $f(\mathbf{x})$, or learn a finite dataset. The final state $\psi_L(\xi, \mathbf{x})$ depends on the input vector $\mathbf{x}$, because $\mathbf{x}$ is encoded in the initial condition $\psi_0(\xi, \mathbf{x})$. We report further details in the Supplemental Material [49].

Using Dirac brackets to simplify the notation, we have $\psi_0(\xi, \mathbf{x}) = \langle \xi | 0, \mathbf{x} \rangle$, and the final state is $\psi_L(\xi, \mathbf{x}) = \langle \xi | L, \mathbf{x} \rangle$. The evolution from $\psi_0(\xi, \mathbf{x})$ to $\psi_L(\xi, \mathbf{x})$ follows a nonlinear partial differential equation. We use the non-linear Schrödinger equation

$$\iota \partial_\zeta \psi + \partial_\xi^2 \psi + \chi |\psi|^2 \psi = 0, \quad (2)$$

with $\psi(\xi, \zeta = 0) = \psi_0(\xi, \mathbf{x})$, and $\psi(\xi, \zeta = L) = \psi_L(\xi, \mathbf{x})$, and ζ the evolution coordinate. In (2), χ measures the strength of the nonlinearity, and $\chi = 0$ corresponds to linear propagation. We write the input as $|0, \mathbf{x}\rangle = |\text{in}, \mathbf{x}\rangle + |b\rangle$, where $|b\rangle$ is the bias (see Fig. 1). The encoder maps $\mathbf{x}$ into $|\text{in}, \mathbf{x}\rangle$ in the Fourier space. For N_X plane waves $|k_q\rangle$, $\psi_{\text{in}}(\xi, \mathbf{x}) = \langle \xi | \text{in}, \mathbf{x} \rangle$, and

$$|\text{in}, \mathbf{x}\rangle = \sum_{q=1}^{N_X} x_q |k_q\rangle. \quad (3)$$

We adopt a generalized basis $|\mu\rangle$, being the initial state $\psi_0(\xi, \mathbf{x}) = \langle \xi | 0, \mathbf{x} \rangle$ and $|0, \mathbf{x}\rangle = \sum_\mu a_\mu |\mu\rangle$, with

$$a_\mu = \sum_{q=1}^{N_X} w_{\mu q} x_q + b_\mu, \quad (4)$$

$w_{\mu q} = \langle \mu | k_q \rangle$ the input weights, and $b_\mu = \langle \mu | b \rangle$ components of the bias vector $\boldsymbol{b}$. In the finite-bandwidth case, μ is a discrete index, and the wave function is determined by a set of samples, following the Nyquist-Shannon theorem with ξ_μ sampling points, $|\mu\rangle = |\xi_\mu\rangle$, and $b_\mu = \langle \xi_\mu | b \rangle$. In analogy to neural networks, the bias separates the nonlinear response with respect to the data. Without the bias, low amplitude inputs will experience different nonlinearity. The

bias input strengthens and tailors the nonlinear response by exciting background nonlinear waves, to which the input signal is superimposed.

Training.—The model maps the SWFN training to conventional reservoir computing protocols. Given a finite number N_T of training points $\mathbf{x}^{(t)}$, with $t = 1, 2, \ldots, N_T$, and corresponding targets $T^{(t)}$, the wave "learns" the training set when $o^{(t)} = o[\mathbf{x}^{(t)}] = T^{(t)}$ for any t.

To establish the conditions for learning, we evaluate the $N_T \times N_C$ transmission matrix $H_{tj} = g_j[\mathbf{x}^{(t)}] = |\psi_L(\xi_j, \mathbf{w} \cdot \mathbf{x}^{(t)} + \mathbf{b})|^2$ by evolving the wave on all the N_T inputs $\psi_0[\xi, \mathbf{x}^{(t)}]$, and solving the linear system

$$o[\mathbf{x}^{(t)}] = \sum_{j=1}^{N_C} H_{tj} \beta_j = \sum_{j=1}^{N_C} g_j[\mathbf{x}^{(t)}] \beta_j = f[\mathbf{x}^{(t)}] = T^{(t)}, \quad (5)$$

that is $\boldsymbol{H} \cdot \boldsymbol{\beta} = \boldsymbol{T}$, with $\boldsymbol{T} = (T^{(1)}, T^{(2)}, \ldots, T^{(N_T)})^T$.

The machine can learn the entire dataset with zero error only if $N_C = N_T$ and $\boldsymbol{H}$ has rank N_C. In this case, $\boldsymbol{\beta}$ exists such that $||\boldsymbol{H} \cdot \boldsymbol{\beta} - \boldsymbol{T}|| = 0$. In the general case $N_C \leq N_T$, the minimum-norm least-squares solution is found by the Moore-Penrose pseudoinverse of $\boldsymbol{H}$ [5].

However, even when $N_C = N_T$, the matrix $\boldsymbol{H}$ is not always invertible, and hence the model cannot learn the dataset with arbitrary precision. We find that there are two conditions for learning: (i) the function $|\psi_L(\xi, \mathbf{w} \cdot \mathbf{x}^{(t)} + \mathbf{b})|^2$ must be nonpolynomial in $\mathbf{b}$ and (ii) in ξ.

Following the theorem we demonstrate in the Supplemental Material [49], the need for condition (i) arises from Eq. (5), since an arbitrary $f(\mathbf{x})$ can only be represented by nonpolynomial functions $g_j(\mathbf{x})$ [50,51]. For example, for a bias vector b with identical components $b_\mu = c$ as input in the linear modulational instability, $|\psi_L|^2$ is a quadratic polynomial in c, and the SWFN cannot represent any arbitrary $f(\mathbf{x})$ [42]. In the recently studied nonlinear modulational instability, the output $|\psi_L|^2$ is a transcendental function of c [46–48]; in this highly non-linear regime, the SWFN can represent arbitrary functions.

The need for condition (ii) can be demonstrated following Refs. [5,52]: the vector $\mathbf{h}(\xi)$ with components $h_t(\xi) = |\psi_L[\xi, \mathbf{w} \cdot \mathbf{x}^{(t)} + \mathbf{b}]|^2$ spans $\mathbb{R}^{N_T}$ if $|\psi_L[\xi, \mathbf{w} \cdot \mathbf{x}^{(t)} + \mathbf{b}]|^2$ and all its derivatives with respect to ξ are nonvanishing. Correspondingly, N_T points ξ_j exist, such that $\mathbf{h}(\xi_j)$ is a basis, and $\boldsymbol{H}$ has rank $r = N_T$. Condition (ii) is commonly satisfied in practical implementations, as $\psi_L[\xi, \mathbf{w} \cdot \mathbf{x}^{(t)} + \mathbf{b}]$ is typically a normalizable function (finite energy) that is regularized upon evolution even in the presence of discontinuities in the initial conditions. It results that all the derivatives of $|\psi_L[\xi, \mathbf{w} \cdot \mathbf{x}^{(t)} + \mathbf{b}]|^2$ with respect to ξ are nonvanishing, as in the box problem here considered.

Conditions (i) and (ii) have two significant consequences: (i) if the wave evolution is linear, the wave is not a universal approximator, indeed, $o(\mathbf{x})$ is only a

quadratic function of **b**; (ii) not all nonlinear evolutions act as universal approximators. For example, a wave that undergoes only self-phase modulation, i.e., $\psi_L(\xi) = \psi_0(\xi) \exp[\iota |\psi_0(\xi)|^2 L]$, will not result in a nonpolynomial output function (it will be only a quadratic function of the bias b).

Also, in the presence of weak polynomial nonlinearities (as in conventional nonlinear optics), to have universal approximators, one needs that the final wave is a strongly nonlinear function of the input, which means that all the derivatives with respect to the bias wave parameters (amplitude, width, etc.) must be nonvanishing. This condition is not satisfied in a perturbative regime. Starting from a linear propagation [$\chi = 0$ in Eq. (2)], when augmenting the nonlinearity $|\chi|$, one observes an increasing ability to learn, corresponding to a growing rank r. For a given training set, there is a threshold nonlinearity for the training error to be zero.

These arguments suggest choosing the bias as the simplest function that triggers the generation of solitons and highly nonlinear waves (as breathers) and provides a nonpolynomial input or output relation. We adopt as bias a finite-energy flat beam. We test numerically the SWFN and we show in the following: (i) the SWFN can approximate arbitrary functions and learn datasets as conventional reservoir computing only above a critical nonlinearity, (ii) linear propagation does not act as a universal approximator, (iii) the SWFN can implement universal Boolean logic gates, as the NAND, and (iv) the SWFN can perform with binary and real valued inputs.

Example 1, $\sin(\pi x)/(\pi x)$ with binary encoding.—We first show learning input or output functions. We consider $y = \sin(\pi x)/(\pi x)$, and adopt a binary encoding of the real variable x in the range $[-\pi, \pi]$. We quantize x by $N_X = 12$ bits, such that $x = \sum_{q=1}^{N_X} \tilde{x}_q 2^{q-1}$ defines the binary string $\tilde{\boldsymbol{x}} = (\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_{N_X})^T$ with $\tilde{x}_q \in \{0, 1\}$. We obtain the input vector $\boldsymbol{x}$ by $x_q = \exp(\iota \phi_q)$ and the one-to-one correspondence $\phi_q = 0 \leftrightarrow \tilde{x}_q = 1$, $\phi_q = \pi \leftrightarrow \tilde{x}_q = 0$. Therefore $|\text{in}, \boldsymbol{x}\rangle$ in Eq. (3) is given by N_X plane waves $|k_q\rangle$ with unitary amplitude and phase ϕ_q. In the simulation of Fig. 2, ξ is discretized with 512 points in the domain $[-150, 150]$, and $L = 1$. The N_C channels are linearly distributed in the range $[-100, 100]$. The bias is a rectangular wave with amplitude $a_b = 1$ and half width $w_b = 100$ [Fig. 2(a)]. Figure 2(b) shows the evolution of a representative $\mathbf{x}$. We study the learning when increasing N_C, at $N_T = 200$ and $\chi = 25$. Figure 2(c) illustrates the training data compared with the SWFN output for $N_C = 20$ and $\chi = 25$; Fig. 2(d) shows the case for $N_C = N_T = 200$. In the latter case, all the training points are learned with zero error within numerical precision. Exact learning occurs at $N_C = N_T$, when the error abruptly drops of several orders of magnitudes, this is evident in Fig. 2(e), showing the training error versus N_C. As shown in Fig. 2(c), using
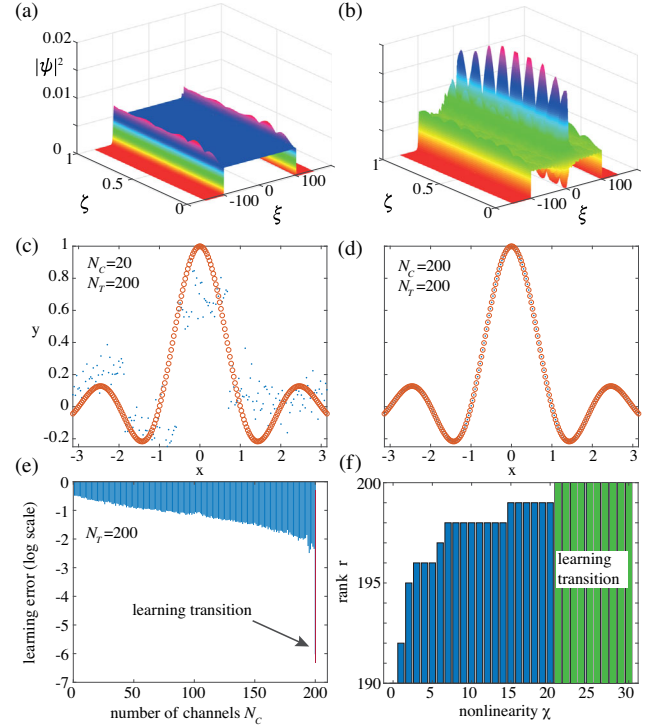

FIG. 2. Learning the function $y = \sin(\pi x)/(\pi x)$. (a) Bias evolution. (b) Wave evolution for a representative point $x = -\pi$. (c) Training points (red circles) and SWFN output (dots) after training with $N_T = 200$ and $N_C = 20$, (d) as in (c) for $N_T = 200$ and $N_C = 200$. (e) Training error (log scale) when increasing N_C at fixed nonlinearity $\chi = 25$, the learning threshold at $N_C = N_T$ is evident. (f) Rank r of the transmission matrix $\boldsymbol{H}$ varying the nonlinearity χ for $N_C = N_T = 200$, the threshold χ for learning corresponds to $r = N_T$.

$N_C = 20$ channels for $N_T = 200$ provides a poor representation. Figure 2(d) demonstrates that a nearly exact representation is obtained for $N_C = N_T = 200$, when the error drops of several order of magnitudes [Fig. 2(e)]. The role of nonlinearity is studied in Fig. 2(f). Learning requires that the rank r of $\boldsymbol{H}$ is equal to N_C. Figure 2(f) shows r versus χ for $N_C = N_T = 200$. Only above a threshold value for χ, the learning condition $r = N_T$ is achieved. The Supplemental Material [49] reports further results of the performance of the SWFN, with training and testing datasets.

Example 2, abalone dataset with amplitude encoding.— We test learning conventional datasets for neural networks. We consider the "abalone dataset" [53], which is one of the mostly used benchmarks for machine learning, and concerns the classification of sea snails in terms of age and physical parameters. Each training point in this dataset has a high-dimensional input with $N_X = 8$, identically encoded in the input vector $\boldsymbol{x}$. $|\text{in}, \boldsymbol{x}\rangle$ in Eq. (3) is then defined by N_X plane waves $|k_i\rangle$ with amplitude x_i and zero phase. Figures 3(a) and 3(b) show the bias wave function and a representative $\mathbf{x}$ propagation. Figure 3(c) shows values
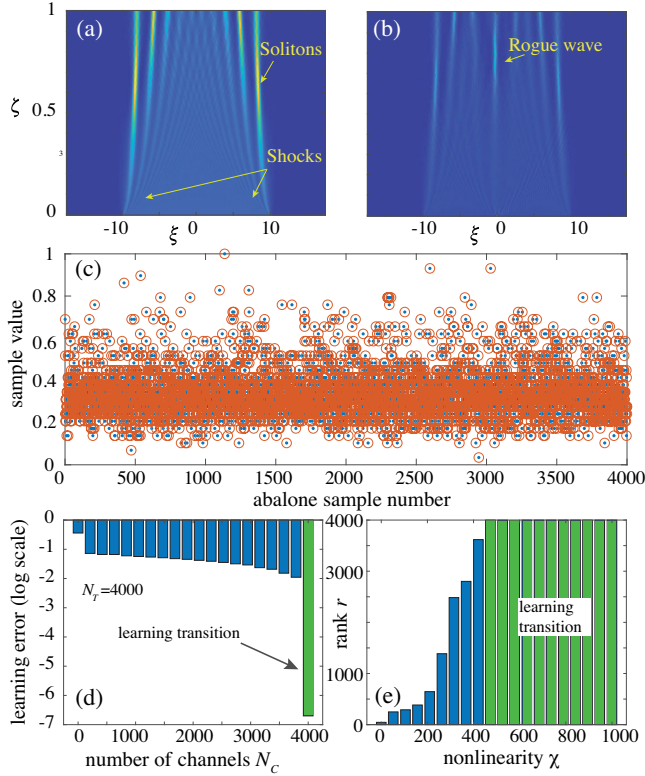
FIG. 3. Learning of the abalone dataset. (a) Bias function encompassing two counterpropagating shock waves and soliton generation. (b) Propagation of one representative point of the dataset with the generation of a rogue wave. (c) Comparison between training data (circle) and SWFN outputs (dots), with $N_T = N_C = 4000$, and $\chi = 500$, showing that the SWFN has learned the data. (d) Training error for $N_T = 4000$ and $\chi = 500$ vs N_C. (e) Rank r vs nonlinearity χ for $N_T = N_C = 4000$ showing the learning transition.

from the training set (red circles) and the SWFN output (dots) for $N_C = 4000$ and $\chi = 500$. The learning transition is shown when varying N_C for $\chi = 500$ in Fig. 3(d) and varying χ in Fig. 3(d) for $N_C = N_T = 4000$. The strength of the nonlinearity to interpolate a dataset grows with the size. The various inputs **x** generate different ensembles of nonlinear waves, as shocks, solitons, and rogue waves, resembling recent experimental results [46]. The Supplemental Material [49] reports further results.

Example 3, Boolean logic gate.—The SWFN can also realize universal classical Boolean logic gates. We consider, e.g., the NAND gate, which has two binary inputs and one binary output (truth table in Fig. 4). We encode the two inputs in the phases of two plane waves in the k space, using the binary encoding above with $N_X = 2$. Figure 4 shows the resulting trained gate. The SWFN output is given in the truth table and obtained at the machine precision within an error of 10^{-15}. The rectangular bias evolution is shown in Fig. 4(a). The sampling points $\xi_{1,2,3,4}$ at the readout are also indicated. The input 00 superimposed to
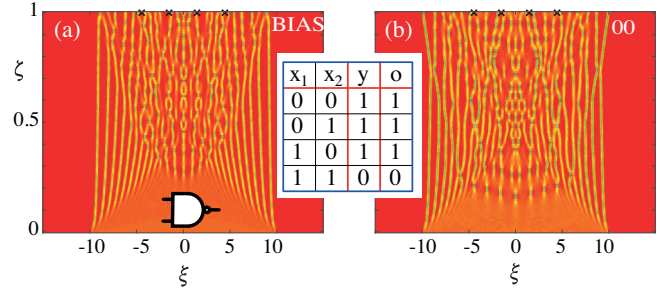

FIG. 4. Training a universal logic gates by soliton gases ($\chi = 1000$). (a) Bias function with two counterpropagating shocks in the initial dynamics evolving into solitons and rogue waves. (b) Propagation of the input 00 in the NAND gate. The crosses at $\zeta = 1$ correspond to the readout sampling points ($N_C = N_T = 4$). The inset shows the NAND gate truth table for training, and the SWFN output o.

the highly nonlinear bias produces the needed output [Fig. 4(b)]. Many solitons and rogue waves are visible. The neuromorphic nonlinear wave device learns the truth table and performs the logical operation. Similar results are obtained for other Boolean logic gates.

Conclusions.—We studied theoretically artificial neural networks with a nonlinear wave as a reservoir layer. We found that they are universal approximators, and developed a new computing model driven by nonlinear partial differential equations. Following general theorems in neural networks, we discovered that a threshold nonlinearity exists for the learning process. When the dataset is large, the threshold implies the generation of highly nonlinear processes as dispersive shocks, rogue waves, and soliton gases. The rank of the transmission matrix is the relevant parameter to assess the learning transition when varying the channels and the degree of nonlinearity. Linear or weakly nonlinear evolution may only learn small datasets. The SWFN performance for commonly adopted datasets in fitting and generalizing is comparable to current conventional models. It is also robust with respect to noise in the initial conditions and at the detection. Moreover, encoding the input state in the initial condition of a nonlinear wave equation enables significant scalability, as is known in optics, where recent engineering of Ising machines has demonstrated the computation with thousands of bits [35]. The robustness and convexity of the optimization arise from the general properties of the extreme learning machines, and related computational paradigms [3–10]. Other recent neural ordinary differential equations networks [54], also proposed to wave physics [55], need fine tuning and inhomogeneous media, e.g., very accurate control of the refractive index. This poses severe limitations to the experimental realization, and on reprogramming different functions. On the contrary, our model does not require strict control of the propagation medium. This fact makes quite feasible experimental tests into homogeneous nonlinear systems.

Our results may stimulate further theoretical work to determine the best nonlinear phenomena for learning and even their ability to generalize. The roles of external potentials, randomness, and noise in quantum and turbulent regimes are unexplored. Our framework holds in optics, polaritonics, hydrodynamics, Bose-Einstein condensates, and all the fields encompassing nonlinearity, also with models different from the nonlinear Schrödinger equation. Our analysis is the starting point for many other developments, as cascading wave layers and conventional nodes. If building heterogeneous deep computational systems with standard neural networks and waves provides computing advantages is still an open question, but photonics speed up electronic systems for machine learning. For these reasons, our theoretical results may foster new ultrafast computing hardware.

---

[*]claudio.conti@uniroma1.it

[1] Y. LeCun, Y. Bengio, and G. Hinton, Nature (London) **521**, 436 (2015).

[2] J. Schmidhuber, Neural Netw. **61**, 85 (2015).

[3] Y.-H. Pao, G.-H. Park, and D. J. Sobajic, Neurocomputing; Variable Star Bulletin **6**, 163 (1994).

[4] H. Jaeger and H. Haas, Science **304**, 78 (2004).

[5] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, Neurocomputing; Variable Star Bulletin **70**, 489 (2006).

[6] D. Verstraeten, B. Schrauwen, M. D'Haene, and D. Stroobandt, Neural Netw. **20**, 391 (2007).

[7] B. Widrow, A. Greenblatt, Y. Kim, and D. Park, Neural Netw. **37**, 182 (2013).

[8] D. Wang and M. Li, IEEE Trans. Syst. Man Cybern. **47**, 3466 (2017).

[9] S. Scardapane and D. Wang, WIREs Data Mining Knowl. Discovery **7**, e1200 (2017).

[10] C. Gallicchio, A. Micheli, and L. Pedrelli, Neural Netw. **108**, 33 (2018).

[11] J. Pathak, B. Hunt, M. Girvan, Z. Lu, and E. Ott, Phys. Rev. Lett. **120**, 024102 (2018).

[12] D. Psaltis and N. Farhat, Opt. Lett. **10**, 98 (1985).

[13] C. Denz, *Optical Neural Networks* (Springer, Berlin, 1998).

[14] P. del Hougne and G. Lerosey, Phys. Rev. X **8**, 041037 (2018).

[15] D. Ballarini, A. Gianfrate, R. Panico, A. Opala, S. Ghosh, L. Dominici, V. Ardizzone, M. D. Giorgi, G. Lerario, G. Gigli, T. C. H. Liew, M. Matuszewski, and D. Sanvitto, arXiv:1911.02923.

[16] A. Opala, S. Ghosh, T. C. H. Liew, and M. Matuszewski, Phys. Rev. Applied **11**, 064029 (2019).

[17] F. Duport, B. Schneider, A. Smerieri, M. Haelterman, and S. Massar, Opt. Express **20**, 22783 (2012).

[18] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman, Nat. Commun. **5**, 3541 (2014).

[19] M. Prabhu, C. Roques-Carmes, Y. Shen, N. Harris, L. Jing, J. Carolan, R. Hamerly, T. Baehr-Jones, M. Hochberg, V. Čeperić, J. D. Joannopoulos, D. R. Englund, and M. Soljačić, Optica **7**, 551 (2020).

[20] G. Van der Sande, D. Brunner, and M. C. Soriano, Nanophotonics **6**, 561 (2017).

[21] J. Bueno, S. Maktoobi, L. Froehly, I. Fischer, M. Jacquot, L. Larger, and D. Brunner, Optica **5**, 756 (2018).

[22] G. Favraud, J. S. T. Gongora, and A. Fratalocchi, Laser Photonics Rev. **12**, 1870047 (2018).

[23] P. Zhao, S. Li, X. Feng, S. M. Barnett, W. Zhang, K. Cui, F. Liu, and Y. Huang, J. Opt. **21**, 104003 (2019).

[24] N. Mohammadi Estakhri, B. Edwards, and N. Engheta, Science **363**, 1333 (2019).

[25] J. Dong, M. Rafayelyan, F. Krzakala, and S. Gigan, IEEE J. Sel. Top. Quantum Electron. **26**, 1 (2020).

[26] A. Röhm, L. Jaurigue, and K. Lüdge, IEEE J. Sel. Top. Quantum Electron. **26**, 1 (2020).

[27] D. Brunner, M. C. Soriano, C. R. Mirasso, and I. Fischer, Nat. Commun. **4**, 1364 (2013).

[28] Y. Shen, N. C. Harris, S. Skirlo, M. Prabhu, T. Baehr-Jones, M. Hochberg, X. Sun, S. Zhao, H. Larochelle, D. Englund, and M. Soljačić, Nat. Photonics **11**, 441 (2017).

[29] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan, Science **361**, 1004 (2018).

[30] G. Mourgias–Alexandris, A. R. Totović, A. Tsakyridis, N. Passalis, K. Vyrsokinos, A. Tefas, and N. Pleros, J. Lightwave Technol. **38**, 811 (2019).

[31] K. Wu, J. García de Abajo, C. Soci, P. Ping Shum, and N. I. Zheludev, Light Sci. Appl. **3**, e147 (2014).

[32] P. L. McMahon, A. Marandi, Y. Haribara, R. Hamerly, C. Langrock, S. Tamate, T. Inagaki, H. Takesue, S. Utsunomiya, K. Aihara, R. L. Byer, M. M. Fejer, H. Mabuchi, and Y. Yamamoto, Science **354**, 614 (2016).

[33] C. Roques-Carmes, Y. Shen, C. Zanoci, M. Prabhu, F. Atieh, L. Jing, T. Dubcek, V. Ceperic, J. D. Joannopoulos, D. Englund, and M. Soljacic, Nat. Commun. **11**, 249 (2020).

[34] K. P. Kalinin and N. G. Berloff, Sci. Rep. **8**, 17791 (2018).

[35] D. Pierangeli, G. Marcucci, and C. Conti, Phys. Rev. Lett. **122**, 213902 (2019).

[36] F. Böhm, G. Verschaffelt, and G. Van der Sande, Nat. Commun. **10**, 3538 (2019).

[37] H. Takesue, T. Inagaki, K. Inaba, T. Ikuta, and T. Honjo, J. Phys. Soc. Jpn. **88**, 061014 (2019).

[38] S. Kumar, H. Zhang, and Y.-P. Huang, arXiv:2001.05680.

[39] D. Pierangeli, G. Marcucci, D. Brunner, and C. Conti, Nanophotonics **2020**, 0119 (2020).

[40] D. Pierangeli, G. Marcucci, and C. Conti, arXiv:2005.08690.

[41] S. H. Rudy, S. L. Brunton, J. L. Proctor, and J. N. Kutz, Sci. Adv. **3**, e1602614 (2017).

[42] Y. Kivshar and G. P. Agrawal, *Optical Solitons* (Academic Press, New York, 2003).

[43] S. Barland, J. R. Tredicce, M. Brambilla, A. L. Lugiato, S. Balle, M. Giudici, T. Maggipinto, L. Spinelli, G. Tissoni, T. Knodl, M. Millar, and R. Jager, Nature (London) **419**, 699 (2002).

[44] M. Närhi, L. Salmela, J. Toivonen, C. Billet, J. M. Dudley, and G. Genty, Nat. Commun. **9**, 4923 (2018).

[45] A. Kokhanovskiy, A. Ivanenko, S. Kobtsev, S. Smirnov, and S. Turitsyn, Sci. Rep. **9**, 2916 (2019).

[46] G. Marcucci, D. Pierangeli, A. J. Agranat, R.-K. Lee, E. DelRe, and C. Conti, Nat. Commun. **10**, 5090 (2019).

[47] F. Bonnefoy, A. Tikan, F. Copie, P. Suret, G. Ducrozet, G. Pradehusai, G. Michel, A. Cazaubiel, E. Falcon, G. El, and S. Randoux, Phys. Rev. Fluids **5**, 034802 (2020).

[48] C. Naveau, P. Szriftgiser, A. Kudlinski, M. Conforti, S. Trillo, and A. Mussot, Opt. Lett. **44**, 5426 (2019).

[49] See the Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.125.093901 for further numerical results and details in the model.

[50] K. Hornik, M. Stinchcombe, and H. White, Neural Netw. **3**, 551 (1990).

[51] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken, Neural Netw. **6**, 861 (1993).

[52] S. Tamura and M. Tateishi, IEEE Trans. Neural Networks **8**, 251 (1997).

[53] https://archive.ics.uci.edu/ml/datasets/Abalone.

[54] T. Q. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud, in *Advances in Neural Information Processing Systems* (Neural Information Processing Systems Foundation (NIPS), 2018), pp. 6571–6583.

[55] T. W. Hughes, I. A. Williamson, M. Minkov, and S. Fan, Sci. Adv. **5**, eaay6946 (2019).