# Resonance for analog recurrent neural network

**Yurui Qu**
University of Wisconsin-Madison https://orcid.org/0000-0002-8837-2663
**Ming Zhou**
University of Wisconsin Madison https://orcid.org/0000-0003-2368-8883
**Erfan Khoram**
University of Wisconsin-Madison
**Nanfang Yu**
Columbia University
**Zongfu Yu** ( zyu54@wisc.edu )
University of Wisconsin–Madison https://orcid.org/0000-0002-4536-1526

**Article**

**Keywords:** optics, artificial neural computing, resonators, analog recurrent neural network

**DOI:** https://doi.org/10.21203/rs.3.rs-480598/v1

# Resonance for analog recurrent neural network

Yurui Qu[1], Ming Zhou[1], Erfan Khoram[1], Nanfang Yu[2], Zongfu Yu[1*]

[1]Department of Electrical and Computer Engineering, University of Wisconsin, Madison, WI 53706, USA

[2]Department of Applied Physics and Applied Mathematics, Columbia University, New York, NY 10027, USA

*email: zyu54@wisc.edu

## Abstract

There is a strong interest in using physical waves for artificial neural computing because of their unique advantages in fast speed and intrinsic parallelism. Resonance, as a ubiquitous feature across many wave systems, is a natural candidate for analog computing in temporal signals. We demonstrate that resonance can be used to construct stable and scalable recurrent neural networks. By including resonators with different lifetimes, the computing system develops both short-term and long-term memory simultaneously.

Since 2012, data centers around the world have started to see a new and fast-growing task: artificial intelligence. The amount of computing used to train models doubles every 3.4 months [1]. By comparison, Moore's Law has an approximately 2-year period for doubling the number of transistors on a computer chip. Artificial neural networks have been enormously successful in image recognition [2], natural language processing [3], gaming [4], and even scientific discovery [5-16]. These applications necessitate larger and deeper neural networks, which in turn require new and high-efficiency computing architectures. The search for alternative computing has recently intensified. One interesting candidate is to use physical waves such as light to perform analog computing [17-23]. Such analogy computing enjoys benefits of intrinsic parallelism and it can be extremely energy efficient. The great promise also comes with great challenges. The very fact that gives rise to the lightning speed of wave computing also leads to one major challenge: the transient nature of propagating waves makes it difficult to construct memory in the wave domain. Since memory is indispensable for computing in the temporal data, today researchers have to resort to other means to realize the effect of memory such as optoelectronic conversion [24], routing through long waveguides [25], and random internal feedback [26].

As a natural memory, resonance is well suited as the building block for scalable recurrent neural networks (RNNs). RNNs have been widely used for speech recognition [27] and synthesis [28], and machine translation [29]. In digital RNNs, memory has been realized by constructing digital feedbacks [30], and the same feedback can be realized by wave reflection at the boundaries of resonators. Different types of analog RNNs have been previously investigated. One strategy is to faithfully follow digital feedbacks in RNNs by routing the output of a computing device back to its input. For example, one can feed the output of optical networks back to the input via long optical waveguides or electro-optical conversion [24, 25]. By merely functioning as a mirror copy of digital RNNs, this strategy does not take advantage of intrinsic feedback mechanisms in wave physics. Recently, Khoram *et al*. proposed to use the intrinsic feedback in disorder structures to increase the expressive power and reduce the size of

analog neural computing [21]. Hughes *et al*. showed that such internal feedback mechanism can be used to realize RNNs [26]. For a long time, reservoir computing in the optical domain has also explored complex temporal dynamics to realize the memory effect [31]. In all these works, the memory is implicitly built into the complex structures. One cannot easily identify the spatial distribution or the temporal characteristics of memory. While in principle these analog systems can perform certain temporal computation, physical intuition and interpretation are lacking.

RNNs have the vanishing gradient problem in nature: the gradient of the loss function decays exponentially with time [32]. As a result, it is difficult for vanilla RNNs to keep track of arbitrary long-term dependencies in the input sequences. Long short-term memory (LSTM) is designed to solve the problem of learning long-term temporal dependencies [33, 34]. It incorporates cell states and gates such that events from the remote past can have current impacts. It is difficult to construct and interpret LSTM in existing wave computing systems [24, 26] because their temporal characteristics are buried in a black box. However, in the resonance system, we can include resonators with different lifetimes to realize short-term and long-term memory. This explicit form of memory makes it easy to construct scalable RNNs and advanced recurrent models such as LSTM. As the quest for the new computing intensifies, resonance could play an important role in emerging wave-based analog computing in general.

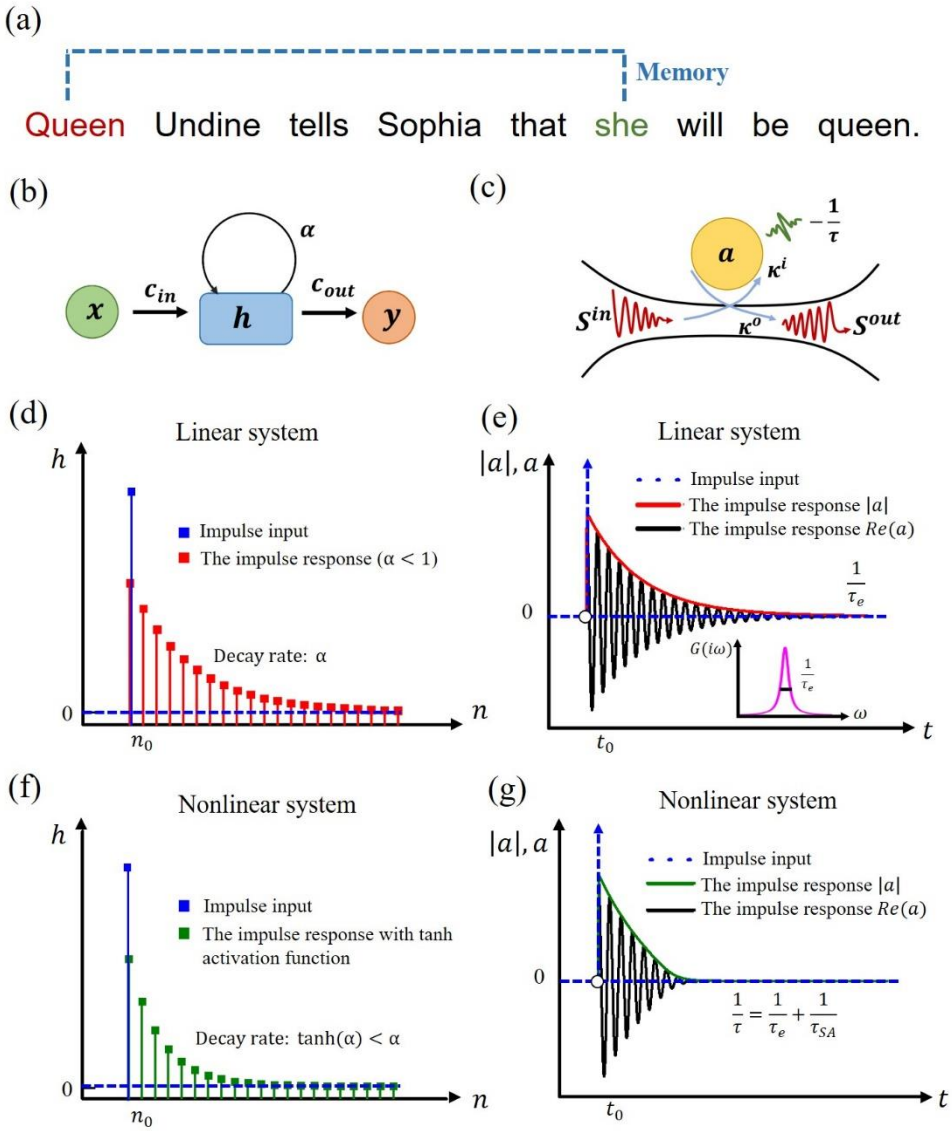


Fig. 1. Comparing the impulse responses of a RNN unit and a resonator. (a) RNN requires internal memory to process a sentence. (b) A RNN unit consists of recurrent hidden state $h$, trainable parameters $c_{in}$, $c_{out}$ and $\alpha$. (c) A resonator as the unit of analog memory with a lifetime of $\tau$, coupling to input and output ports with a coefficient $\kappa$. (d) Impulse response of a RNN unit with linear activation function. The input impulse is the Kronecker delta function. (e) Impulse response of a linear resonator. The input impulse is the Dirac delta function. Inset is resonance amplitude in frequency domain. (f) Impulse response of a RNN unit with a nonlinear activation function $tanh(\cdot)$. (g) Impulse response of a nonlinear resonator that includes saturation absorption. The decay rate is $\frac{1}{\tau} = \frac{1}{\tau_e} + \frac{1}{\tau_{SA}}$, where $\tau_{SA,0}(1 + \frac{|a|^2}{W_{sat}})$, $|a|^2$ represents the energy inside the cavity.

We start by discussing the role of memory in temporal computing. We take the example of natural language processing. When we see “queen” in a sentence, we expect

to encounter “she” rather than “he” soon [Fig. 1(a)]. RNNs are designed to have the internal memory to remember information in the past. A RNN can be represented as a discrete-time system. Its internal state *h*, often named hidden state, at the $n^{\text{th}}$ time step can be described as

$$h[n] = \sigma(\alpha \cdot h[n-1] + c_{in} \cdot x[n]). \quad (1.)$$

It is determined by the hidden state $h[n-1]$ at the previous time step and the input $x[n]$ at the current time step. $\sigma(\cdot)$ is an activation function, such as a sigmoid or a tanh function. $c_{in}$ and $\alpha$ are trainable parameters. By updating the state *h* using the state of the previous time step, the memory is built into the hidden state by recurrent circulation as illustrated in Fig. 1(b).

The memory effect can be directly visualized by the impulse response, i.e., the output of the RNN when excited by the Kronecker delta function $x[n] = \delta[n-n_0]$. The pulse is located at time step $n_0$. The impulse response of the hidden neuron can be shown as

$$R[n] = c_{in} \cdot \big(\sigma(\alpha)\big)^{n-n_0}. \quad (2.)$$

The impulse response obeys exponential decay for linear systems [Fig. 1(d)], for example when $\sigma(\alpha) = \alpha$, and for nonlinear systems [Fig. 1(f)], for example when $\sigma(\alpha) = tanh(\alpha)$. The memory effect is illustrated by the fact that the impact of input at time step $n_0$ extends into certain time horizon.

We now discuss the impulse response of a resonator. The dynamic equation of a resonator coupled with input and output ports can be written as [35]

$$\frac{da}{dt} = \left(i\omega_0 - \frac{1}{\tau}\right) a + \kappa \cdot S^{in}, \quad (3.)$$

where $a$ and $|a|^2$ represent the resonance field and energy, respectively [Fig. 1(c)]. $\omega_0$, $\frac{1}{\tau}$ and $\kappa$ represent resonance frequency, total decay rate and coupling coefficient, respectively. For a continuous-time dynamic system, the impulse is modeled as the

Dirac delta function $S^{in}(t) = \delta(t - t_0)$. The impulse response (resonance amplitude $|a|$) of a resonator is

$$R(t) = |a| = \kappa \cdot e^{-\frac{1}{\tau}(t-t_0)}, \quad (4.)$$

which obeys exponential decay for both linear and nonlinear resonance systems [Fig. 1(e) and 1(g)]. Similar to RNNs, the dynamics of a resonance system carries the memory of past information. Here we model the nonlinearity by having saturable absorption in the decay rate: $\frac{1}{\tau} = \frac{1}{\tau_e} + \frac{1}{\tau_{SA}}$ [36], where $\tau_e$ and $\tau_{SA}$ are the photon resonator lifetimes corresponding to the coupling and resistive losses, respectively. The variable lifetime $\tau_{SA}$ depends on the energy in the cavity: $\tau_{SA} = \tau_{SA,0}(1 + \frac{|a|^2}{W_{sat}})$, where $W_{sat}$ is saturation energy. Nonlinearity can also be provided by other physical mechanisms, including gain saturation [37], Kerr effect [38], or acoustical nonlinearity in bubbly fluids and soft materials [39].

A digital RNN consists of many artificial neurons with memories. One neuron often connects to many others. Similarly, one resonator can couple to many, providing a scalable way to construct large-scale analog computing system with memory. The coupling between resonators can be mediated through free space or waveguides. The coupling coefficients (e.g., connection weights) determine the function of computing. They will be trained in a similar way that neural networks are trained. The trained coefficients can be physically implemented, for example, by adjusting the distance between resonators.

Figure 2a shows a representative RNN with multiple artificial neurons. The memory of RNN is encoded into the hidden state $h_t$, which is related to the past state $h_{t-1}$ by

$$h_t = \sigma\left(\boldsymbol{W^h} \cdot h_{t-1} + \boldsymbol{W^x} \cdot x_t\right) \quad (5.)$$

It is similar to Eq. 1 except that now $\boldsymbol{W^x}$ and $\boldsymbol{W^h}$ are dense matrices that couple different artificial neurons. The output of the RNN is $y_t = \sigma\left(\boldsymbol{W^y} \cdot h_t + b_y\right)$, where

$\boldsymbol{W^y}$ is a weight matrix to be trained together with $\boldsymbol{W^x}$ and $\boldsymbol{W^h}$. $\sigma(\cdot)$ are activation functions. The update equations are diagrammed in Fig. 2(c).

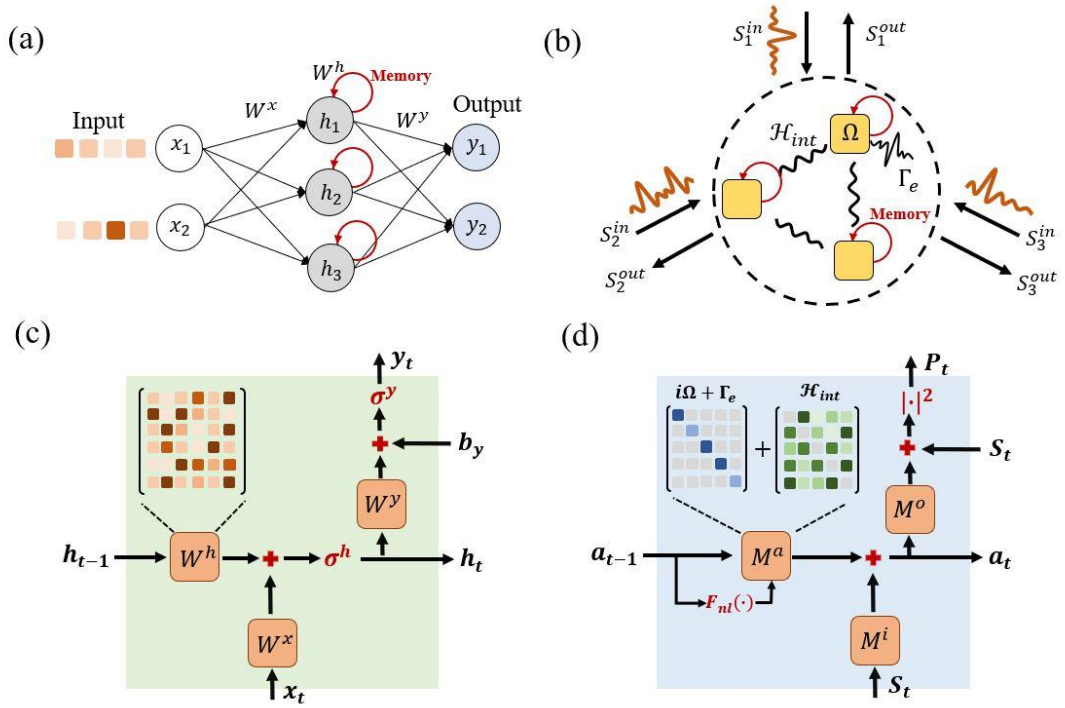


Fig. 2. Conceptual equivalence between of a RNN and a resonance system. (a) General RNN architecture consisting of an input layer, hidden states with feedback loops and an output layer. The RNN can operate on discrete-time sequence. (b) Layout of a resonant recurrent network that can operate on continuous-time signal. $\Omega$, $\Gamma_e$, and $\mathcal{H}_{int}$ represent resonance frequency, decay rate into channels, and coupling between resonators, respectively. (c) Diagram of a RNN cell composed of trainable parameters $W^x$, $W^h$, and $W^y$. (d) Diagram of the recurrence relation for the resonance system composed of trainable parameters $M^i$, $M^a$, and $M^o$.

Similarly, many resonators can couple together to construct a scalable RNN [Fig. 2(b)]. A system of coupled resonators can be described by a temporal coupled mode theory equation as $\frac{d\boldsymbol{a}}{dt} = (i\Omega - \Gamma_e - \mathcal{H}_{int})\boldsymbol{a} + K^T \cdot S_t$, where the matrices $\Omega$, $\Gamma_e$, and $\mathcal{H}_{int}$ represent the resonance frequency, the decay rate into channels, and the coupling between resonators, respectively. The matrix $K$ characterizes the coupling between resonators and ports. $S_t$ is a continuous input sequence. The resonant amplitudes $\boldsymbol{a}$ carry the memory of past information. To make the analogy to RNNs more transparent, we can discretize the differential equations using the backward Euler method as

$$a_t = \boldsymbol{M^a} \cdot a_{t-1} + \boldsymbol{M^{in}} \cdot S_t, \quad (6.)$$

where $M^a$ describes the connection between the present states and the previous states. Coupling between resonators leads to a dense matrix $\boldsymbol{M^a} = (I - i\Delta t \cdot \boldsymbol{\Omega} + \Delta t \cdot \boldsymbol{\Gamma_e} + \Delta t \cdot \boldsymbol{\mathcal{H}_{int}})^{-1}$. Training parameters are the physical quantities of the systems, including the resonant frequency $\boldsymbol{\Omega}$, the decay rate $\boldsymbol{\Gamma_e}$, and the inter-resonator coupling $\boldsymbol{\mathcal{H}_{int}}$. The input coupling is given by $M^{in} = \Delta t \cdot K \cdot M^a$. The output power of the system is given by $P_t = |M^o \cdot a_t + S_t|^2$, where the output coupling $M^o = -K^*$. Additional information on the construction of these matrices is given in Supplementary Materials S1. The resonant recurrent network can be scaled up just like RNNs by including more channels (input/output neurons) and resonators (hidden neurons). As shown in Fig. 2(d), the updating equations of the coupled resonators go in parallel with those of RNNs, allowing them to scale up for large-scale recurrent neural computing. An additional bias term can be implemented by a constant input in one waveguide and an additional complex-valued variable to make the bias trainable. More discussions on the bias term in the resonant recurrent network are included in Supplementary Materials S2 and S3.

Next, we demonstrate a specific example where we train acoustic resonators to recognize the vowels spoken by human. All computing is in the acoustic domain. The computing device [Fig. 3(a)] contains three parallel acoustic waveguides that couple to two rows of whispering gallery resonators [40]. Each row contains 60 resonators. The coupling between resonators is mediated through the waveguides. Depending on the identity of the sound, the computing device will route the energy of three different vowel sounds to three corresponding output waveguides.

The input to the acoustic system is the replay of audio recording of 10 vowel classes from 45 different male speakers and 48 different female speakers. We use a subset of the original data in Ref [26], which includes 279 recordings corresponding to three vowel classes, represented by ae, ei, and iy. The training is performed by using 80% of these recording while the rest 20% is used to test the trained system. In the training, we can adjust geometric parameters, including the resonator positions $l$, the

resonator sizes $r$, and the distances between resonators and waveguides $d$. The training goal is to minimize a loss function that is defined by the output of the three waveguides and ground truth label. We use adjoint method to perform gradient calculation and also back-propagation [41]. More details about the vowel signal, definition of loss function, and the neural network training are included in Supplementary Materials.

Figure 3 (b) and (c) show the results of the cross-entropy loss and the prediction accuracy, respectively. The trained resonance system can achieve an accuracy of 81.7% for the training dataset and 83.3% accuracy for the test dataset over 50 training epochs. The confusion matrix for the test data indicates that the resonance system can indeed perform vowel recognition [inset of Fig. 3(c)]. The time-integrated power at each waveguide demonstrates that the optimized resonant architecture can routes most of the energy of the object vowel class to the correct channel [Fig. 3(d)]. The physical configuration based on the actual geometric parameters is discussed in Supplementary Materials S4. We further clarified how to implement nonlinearity in the resonant recurrent system and present a version of the resonance-based RNN with saturable absorption and the corresponding performances, as shown in Supplementary Materials S5 and S6.

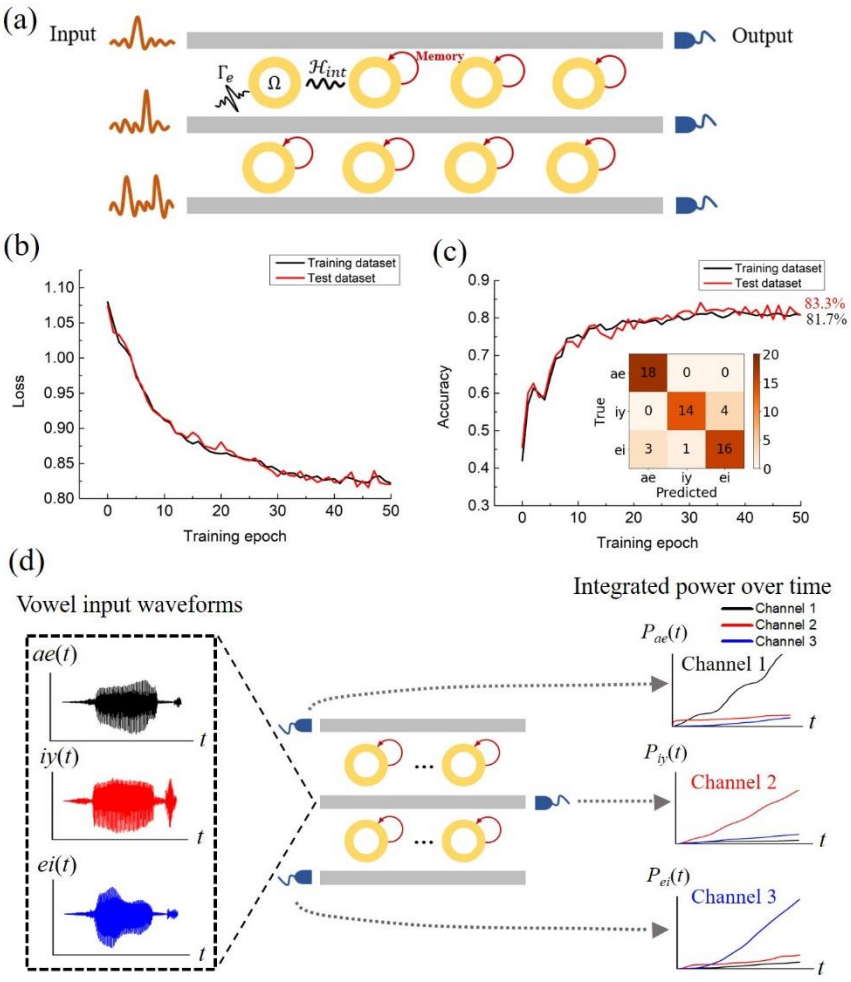


Fig. 3. Acoustic resonators to recognize vowel sound in the wave domain. (a) Schematic of the vowel-recognition resonant recurrent network with three waveguides and two rows of resonators. (b) Cross-entropy loss and (c) prediction accuracy over 50 training epochs. Inset is confusion matrix for the test dataset. The correct predictions are located in the diagonal of the table and prediction errors are located outside the diagonal. (d) Raw audio waveforms of three spoken vowel classes and time-integrated power at each output channel.

The classification of vowel requires memory in temporal domain. In previous work [24], the memory has been built in by converting waves to digital signals. Recently, Hughes *et al*. [26] has used random internal reflection to build implicit memory. In contrast, resonance provides an explicit form of analog memory, making it possible to directly interpret long and short-term memory effects. To demonstrate that, next we consider a challenging case in RNNs: LSTM network. LSTM is explicitly designed to include both short-term and long-term memory. A common LSTM architecture is shown in Fig. 4(a). It incorporates cell states and gates such that events

from the remote past can have current impacts. In the resonance system, we can include resonators with different lifetimes to accomplish short-term and long-term memory as shown in Fig. 4(b).

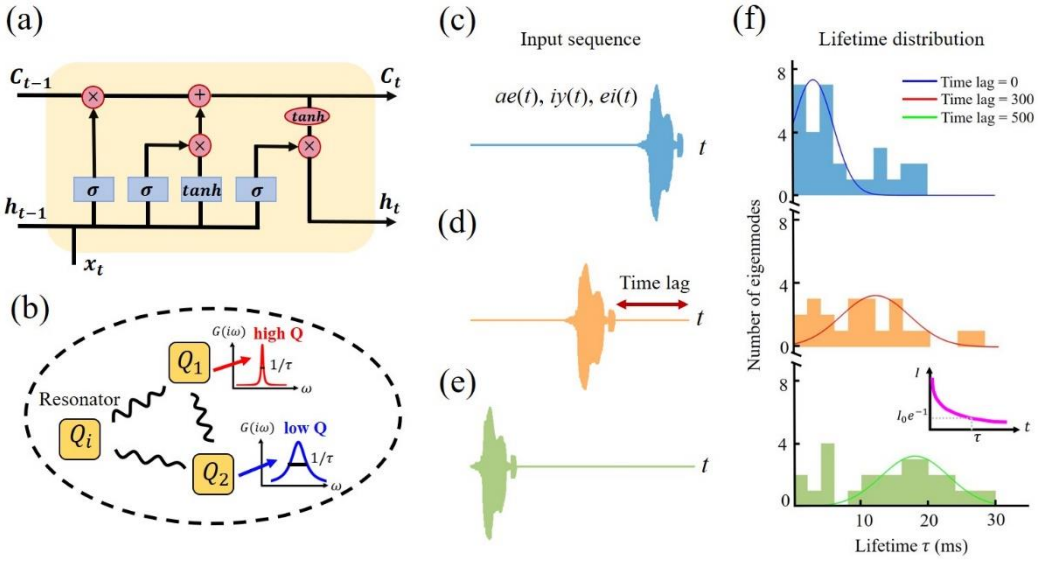


Fig. 4. Task of long-term and short-term memory. (a) Diagram of LSTM architecture in a digital RNN. (b) Diagram of coupled resonators with different quality factors Q and lifetimes τ. (c-e) Schematics showing that the timing of vowel sound varies for three different training sets. The vowel signal contains three classes: *ae*(*t*), *iy*(*t*) and *ei*(*t*). (f) The lifetime distribution of resonators after training for three cases. Long-term and short-term memory arise spontaneously in response to different memory requirement in three different training settings.

We now address a more challenging task by making the previous task of vowel classification more practical. Previously, the input signal consists of pure vowel without any quiet lapse of time preceding or following the vowel. This would be an unusual situation. In practice, we often take a temporal window of recorded sound and ask if there is a vowel in this window. The temporal window is often much longer than the duration of vowel. Now we need relatively short-term memory to recognize the inner structure of the vowel sound, and at the same time, we also need long-term memory because the duration of temporal window can be much longer than the length of vowel sound.

The specific case study considered here uses a temporal window of 600 time steps. A vowel sound only consists of 100 time steps. We position the vowel at different temporal locations as shown in Fig. 4c-e. Here we use voices from three classes: *ae*(*t*), *iy*(*t*) and *ei*(*t*). In the first training case, we place a vowel sound at the end of the temporal window [Fig. 4(c)]. In the second case, we place it in the middle, starting at time step number 300 [Fig. 4(d)]. And lastly, we place it towards the beginning of the temporal window [Fig. 4(e)]. The neural network makes its decision of the vowel class at the end of temporal window. For the first case, the vowel sound is located at the end of the temporal window. There is no time lag between the vowel sound and the output so that the network requires mostly short-term memory. For the last case, however, there is a long time lag, and therefore long-term memory is needed. This setup allows us to see how short and long-term memories spontaneously arise from training. More details about the input sequence, the output, the loss function and the physical configuration are included in Supplementary Materials.

The three trained computing systems all function well with accuracies above 70%. It is interesting to examine the lifetimes of resonators in the three computing systems. Because of the coupling between resonators, it is more informative to look at the lifetimes of eigenmodes after modal hybridization. Eigenvalues of the matrix $\mathcal{H} = i\Omega - \Gamma_e - \mathcal{H}_{int}$ are calculated, where the imaginary and real parts are the resonant frequency ω and decay rate $1/\tau$ of each eigenmode, respectively. The lifetime τ of each eigenmode can then be calculated. The histogram of eigenmodes shows distinct lifetime distributions for the three cases [Fig. 4(f)]. For the first case, the computing system consists of resonators mostly with short lifetimes. For the second case, the histogram shows a middle-range lifetime distribution. For the third case with the longest time lag, the resonance system evolves to possess the capability of long-term memory by including a significant portion of resonators with long lifetimes; in fact, the computing system develops both short-term and long- term memory simultaneously, in a similar way as the memory cells in LSTM. Thus, we observe that the distribution of lifetime

semi-quantitatively reflects the length of the memory required in each computing situation.

We note that the resonance systems have many intrinsic advantages over digital RNNs. These include advantages associated with analog wave computing, such as fast speed, ultra-low energy consumption, and intrinsic parallelism. Moreover, the proposed system can process continuous temporal signals whereas digital RNNs have the risk of insufficient temporal resolution. Lastly, an important but subtle advantage is that it is much easier to train a resonance system than to train a digital RNN. Training a digital RNN has an inherent issue of instability known as the problem of exploding gradient [44], which is due to the great depth of RNNs. One has to take great care to address exploding gradient problems by using techniques such as gradient clipping and weight regularization. It has been shown that in a physical system, recurrent computing is fundamentally stable [45]. We can quantify the advantage of training stability by treating the computing device as a dynamic system described by $\dot{a}(t) = \mathcal{H} \cdot a(t)$. This dynamic system is stable if all the eigenvalues of the matrix $\mathcal{H}$ have negative real parts. As shown in Fig. 5, all eigenvalues of the trained matrix H in the problem shown in Fig. 3 are in the left half the space; that is, the training of this computing system is fundamentally stable without the issue of exploding gradients.

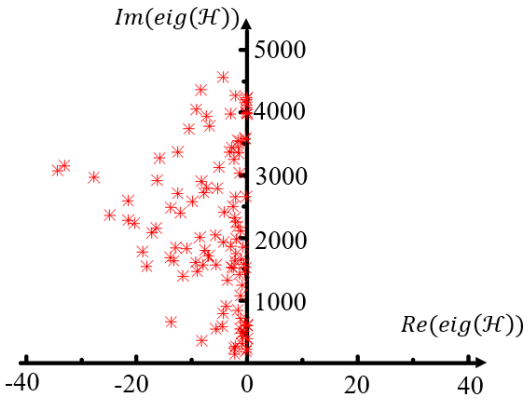


Fig. 5. Real and imaginary parts of the eigenvalues of matrix $\mathcal{H}$. The training case we studied here is the same as the one in Figure 3.

Finally, we comment on the computing time and energy cost in comparison to the optical RNN that is based on electronic feedback [24]. Unlike [24], here the feedback is provided in the optical domain. Thus, there is no need to perform opto-electronic conversion for each feedback step. Without opto-electronic conversion, resonance RNN significantly speed up the computing. Moreover, the energy consumption is reduced because of the absence of opto-electronic conversion. The training of the resonance RNN is performed in digital computer with a similar energy and time scaling as those digital RNN. The inference time scales linearly with the number of resonators. More detailed discussions are included in Supplementary Materials.

In conclusion, we have shown that resonance can be used to construct stable and scalable recurrent neural networks. Resonance provides an explicit form of memory. Short-term or long-term memories can be directly constructed and interpreted by the lifetime of resonators. This extends the analog computing capability into the complex neural network architecture such as LSTM. While we use acoustic as an example, the strategy can be broadly applied to other physical domains. The computing speed and footprint analyses are included in Supplementary Materials. Photonic analog computing has been used in a number of applications such as reservoir computing [46], photonic Ising machines [47, 48], self-learning Machines [49], image edge detection [50-52] and analog signal processing [53, 54].

Acknowledgement: Y. Q, N. Y, Z. Y acknowledge the financial support by DARPA Award No. FA8650-20-1-7028. M. Z. and Z.Y acknowledge the financial support by DARPA NLM program Award HR00111820046.

Author contributions: Y. Q and Z. Y conceived the idea of the paper. Y. Q, M. Z and E. K. designed the neural network architectures. Y. Q, M. Z, E. K., N. Y and Z. Y analyzed the results. N. Y and Z. Y directed this project. Y. Q, N. Y and Z. Y wrote the draft of the paper and revised the paper. All authors reviewed and commented on the paper.

Competing interests: The authors declare no competing interests.

Reference


[1] OpenAI. AI and Compute [Online]. 2018. https://openai.com/blog/ai-and-compute/.

[2] Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks, In Advances in Neural Information Processing Systems (NeurIPS, Tahoe, Nevada, 2012), pp. 1097–1105.

[3] T. Young, D. Hazarika, S. Poria, and E. Cambria. Recent Trends in Deep Learning Based Natural Language Processing, IEEE Comput. Intell. Mag. 13, 55 (2018)

[4] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel et al. A General Reinforcement Learning Algorithm that Masters Chess, Shogi, and Go Through Self-Play, Science 362, 1140 (2018).

[5] J. Trisno, H. Wang, H. T. Wang, R. J. Ng, S. Daqiqeh Rezaei, and J. K. Yang. Applying Machine Learning to the Optics of Dielectric Nanoblobs, Advanced Photonics Research, 1, 2000068 (2020).

[6] E. Gawehn, J. A. Hiss, and G. Schneider. Deep learning in drug discovery, Molecular informatics 35, 3 (2016).

[7] D. Wang, A. Khosla, R. Gargeya, H. Irshad, and A. H. Beck. Deep Learning for Identifying Metastatic Breast Cancer, arXiv:1606.05718.

[8] S. S. Schoenholz, E. D. Cubuk, D. M. Sussman, E. Kaxiras, and A. J. Liu. A structural approach to relaxation in glassy liquids, Nat. Phys. 12, 469 (2016).

[9] L. Wang, Discovering phase transitions with unsupervised learning. Phys. Rev. B 94, 195105 (2016).

[10] D. Liu, Y. Tan, E. Khoram, and Z. Yu. Training deep neural networks for the inverse design of nanophotonic structures, ACS Photonics 5, 1365 (2018).

[11] J. Peurifoy, Y. Shen, L. Jing, Y. Yang, F. Cano-Renteria, B. G. DeLacy, J. D. Joannopoulos, M. Tegmark, and M. Soljacič. Nanophotonic particle simulation and inverse design using artificial neural networks, Sci. Adv. 4, No. eaar4206 (2018).

[12] Y. Qu, L. Jing, Y. Shen, M. Qiu, and M. Soljacic. Migrating knowledge between physical scenarios based on artificial neural networks, ACS Photonics 6, 1168 (2019).

[13] W. Ma, F. Cheng, and Y. Liu. Deep-Learning Enabled OnDemand Design of Chiral Metamaterials, ACS Nano 12, 6326 (2018).

[14] P. Baldi, P. Sadowski, and D. Whiteson. Searching for exotic particles in high-energy physics with deep learning, Nat. Commun. 5, 4308 (2014).

[15] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd. Quantum machine learning, Nature 549, 195 (2017).

[16] G. Carleo, and M. Troyer. Solving the quantum many-body problem with artificial neural networks, Science 355, 602 (2017).

[17] J. Chang, V. Sitzmann, X. Dun, W. Heidrich, and G. Wetzstein. Hybrid optical-electronic convolutional neural networks with optimized diffractive optics for image classification, Sci. Rep. 8, 1 (2018).

[18] J. Bueno, S. Maktoobi, L. Froehly, I. Fischer, M. Jacquot, L. Larger, and D. Brunner. Reinforcement learning in a large-scale photonic recurrent neural network, Optica 5, 756 (2018).

[19] Y. Qu, H. Zhu, Y. Shen, J. Zhang, C. Tao, P. Ghosh, and M. Qiu. Inverse design of an integrated-nanophotonics optical neural network, Sci. Bull. 65, 1117 (2020).

[20] X. Lin, Y. Rivenson, N. T. Yardimci, M. Veli, Y. Luo, M. Jarrahi, and A. Ozcan. All-optical machine learning using diffractive deep neural networks, Science 361, 1004 (2018).

[21] E. Khoram, A. Chen, D. Liu, L. Ying, Q. Wang, M. Yuan, and Z. Yu. Nanophotonic media for artificial neural inference, Photonics Res. 7, 823 (2019).

[22] Z. Wu, M. Zhou, E. Khoram, B. Liu, and Z. Yu. Neuromorphic metasurface, Photonics Res. 8, 46 (2020).

[23] A. Silva1, F. Monticone, G. Castaldi, V. Galdi, A. Alù, N. Engheta. Performing mathematical operations with metamaterials, Science 343, 160 (2014).

[24] 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. Deep learning with coherent nanophotonic circuits, Nat. Photonics, 11, 441 (2017).

[25] A. N. Tait, T. F. De Lima, E. Zhou, A. X. Wu, M. A. Nahmias, B. J. Shastri, and P. R. Prucnal. Neuromorphic photonic networks using silicon photonic weight banks, Sci. Rep. 7, 1 (2017).

[26] T. W. Hughes, I. A. Williamson, M. Minkov, and S. Fan. Wave physics as an analog recurrent

neural network, Sci. Adv. 5, eaay6946 (2019).

[27] A. Graves, and J. Schmidhuber. Framewise phoneme classification with bidirectional LSTM and other neural network architectures, Neural Netw. 18, 602 (2005).

[28] G. K. Anumanchipalli, J. Chartier, and E. F. Chang. Speech synthesis from neural decoding of spoken sentences, Nature 568, 493 (2019).

[29] I. Sutskever, O. Vinyals, and Q. V. Le. Sequence to sequence learning with neural networks, Advances in neural information processing systems. (NeurIPS, Montréal, Canada, 2014).

[30] J. L. Elman. Finding structure in time, Cogn. Sci. 14, 179 (1990).

[31] F. Denis-Le Coarer, M. Sciamanna, A. Katumba, M. Freiberger, J. Dambre, P. Bienstman, and D. Rontani. All-optical reservoir computing on a photonic chip using silicon-based ring resonators, IEEE J. Sel. Top Quantum Electron. 24, 1 (2018).

[32] S. Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions, Int J Uncertain Fuzz, 6, 107 (1998).

[33] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult, IEEE Trans Neural Netw, 5, 157 (1994).

[34] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput, 9, 1735 (1997).

[35] W. Suh, Z. Wang, and S. Fan. Temporal coupled-mode theory and the presence of non-orthogonal modes in lossless multimode cavities, IEEE Journal of Quantum Electronics 40, 10 (2004).

[36] V. G. Ataloglou, T. Christopoulos, and E. E. Kriezis. Nonlinear coupled-mode-theory framework for graphene-induced saturable absorption in nanophotonic resonant structures, Phys. Rev. A 97, 063836 (2018).

[37] M. H. Teimourpour, L. Ge, D. N. Christodoulides, and R. El-Ganainy. Non-Hermitian engineering of single mode two dimensional laser arrays, Sci. Rep. 6, 1 (2016).

[38] R. W. Boyd, Nonlinear Optics (Academic Press, 2008).

[39] T. Rossing, Springer Handbook of Acoustics (Springer Science & Business Media, 2007).

[40] G. Bahl, X. Fan, and T. Carmon. Acoustic whispering-gallery modes in optomechanical shells, New J. Phys. 14, 115026 (2012).

[41] R. T. Chen, Y. Rubanova, J. Bettencourt, and D. K. Duvenaud. Neural ordinary differential equations, In Advances in Neural Information Processing Systems (NeurIPS, Montréal,

Canada, 2018, pp. 6571-6583).

[42] F. Li, M. Xuan, Y. Wu, and F. Bastien. Acoustic whispering gallery mode coupling with Lamb waves in liquid, Sens. Actuator A Phys. 189, 335 (2013).

[43] C. He, Z. Li, X. Ni, X. C. Sun, S. Y. Yu, M. H. Lu, X. Liu, and Y. F. Chen. Topological phononic states of underwater sound based on coupled ring resonators, Appl. Phys. Lett. 108, 031904 (2016).

[44] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks, International conference on machine learning. 2013. (ICML, Atlanta, Georgia, USA, 2013, pp. 1310-1318).

[45] L. Jing, Y. Shen, T. Dubcek, J. Peurifoy, S. Skirlo, Y. LeCun, M. Tegmark, and M. Soljačić. Tunable efficient unitary neural networks (eunn) and their application to rnns." In International Conference on Machine Learning, pp. 1733-1741. PMLR, 2017.

[46] G. Marcucci, D. Pierangeli, and C. Conti. Theory of neuromorphic computing by waves: machine learning by rogue waves, dispersive shocks, and solitons. Phys. Rev. Lett. 125, 9 (2020).

[47] C. Roques-Carmes, Y. Shen, C. Zanoci, M. Prabhu, F. Atieh, L. Jing, T. Dubček, C. Mao, M.R. Johnson, V. Čeperić, J. D. Joannopoulos, D. Englund, and M. Soljačić. Heuristic recurrent algorithms for photonic Ising machines. Nat. Commun. 11, 1 (2020).

[48] 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. Englund, and M. Soljačić. Accelerating recurrent Ising machines in photonic integrated circuits. Optica, 7, 5 (2020).

[49] V. Lopez-Pastor, F. Marquardt. Self-learning Machines based on Hamiltonian Echo Backpropagation. arXiv, 2103.04992 (2021).

[50] T. Zhu, Y. Zhou, Y. Lou, H. Ye, M. Qiu, Z. Ruan and S. Fan. Plasmonic computing of spatial differentiation. Nat. Commun. 8, 1 (2017).

[51] T. Zhu, Y. Lou, Y. Zhou, J. Zhang, J. Huang, Y. Li, H. Luo, S. Wen, S. Zhu, Q. Gong, M. Qiu, and Z. Ruan. Generalized spatial differentiation from the spin hall effect of light and its application in image processing of edge detection. Phys. Rev. Appl. 11, 3, (2019).

[52] Y. Zhou, H. Zheng, I. I. Kravchenko and J. Valentine, Flat optics for image differentiation. Nat. Photonics 14, 5 (2020).

[53] F. Zangeneh-Nejad, and R. Fleury. Topological analog signal processing. Nat. Commun. 10, 1, (2019).

[54] F. Zangeneh-Nejad, D. L. Sounas, A. Alù and R. Fleury. Analogue computing with metamaterials, Nature Reviews Materials 6, 207 (2020).

# Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- Supportinginformation.docx