

Rhythmic sharing: A bioinspired paradigm for zero-shot adaptive learning in neural networks

Hoony Kang^{*} and Wolfgang Losert[†]

Department of Physics, [University of Maryland, College Park](#), 20740 Maryland, USA



(Received 10 September 2025; accepted 20 February 2026; published 10 March 2026)

The brain rapidly adapts to new contexts and learns from limited data, a coveted characteristic that artificial intelligence (AI) algorithms struggle to mimic. Inspired by the mechanical oscillatory rhythms of neural cells, we developed a learning paradigm utilizing link strength oscillations, where learning is associated with the coordination of these oscillations. Link oscillations can rapidly change coordination, allowing the network to sense and adapt to subtle contextual changes without supervision. The network becomes a generalist AI architecture, capable of predicting dynamics of multiple contexts, including unseen ones. These results make our paradigm a powerful starting point for models of cognition. Because our paradigm is agnostic to specifics of the neural network, our study opens doors for introducing rapid adaptive learning into leading AI models.

DOI: [10.1103/fdc2-ljj6](#)

I. INTRODUCTION

Much of the dynamics that govern the natural world experience drifts in internal parameters, leading to nonstationary dynamics with statistics that change in time. Examples are changes in atmospheric chemistry, tectonic plate movement, or the transition of a tumor's state from benign to malignant. The effects of nonstationarity can be dramatic enough to alter its entire probability space, rendering simple tools such as mean centering of the data insufficient. Although the initial impact of system parameter changes is often barely discernible in the data, detecting them may allow us to anticipate more severe long-term consequences such as climate change, earthquakes, or metastases. Therefore, the classification and prediction of dynamics that experience parameter drifts have garnered considerable attention across disciplines.

While machine learning has been proven to be a powerful tool in predicting the dynamics of complex systems in the absence of mathematical models, most learning methods optimize prediction for tasks originating from the *same* context, i.e., stationary probability distribution. Specifically, learning involves finding optimal weights, or link strengths, which are the strengths of connections between neuronal nodes, that are obtained by minimizing the error between the network's outputs and what we know should be the true outputs for given inputs used during training. This is achieved by having the link strengths of the network gradually increase or decrease monotonically during training, until an optimal configuration

is found [1,2]. Furthermore, to predict nonstationary data, one must also classify the different states present in the data, and then feed these contextual tokens into the machine so that it can understand how the context of the training data changes over time. Here, we define “states” as sets of trajectories that belong to the same stationary probability distribution, such as those of an attractor or some other set with invariant long-term statistical properties. Because each state represents a distinct stationary probability distribution, a state can be considered the underlying “context” of the time series. In the absence of these tokens, the machine cannot make sense of the different probability spaces in the data and tends to forget data whose context clashes with another's (catastrophic forgetting [3,4]). Yet, the brains of living organisms excel at rapidly sensing subtle contextual changes in their environment, and do not suffer from catastrophic forgetting.

A. A learning paradigm

Inspired by the potential physical interactions between electric neuronal signals, chemical astrocytic signals, and mechanical actin signals at the neuronal synapse, we conjecture a paradigm of neuroplasticity. While existing studies have highlighted how calcium signals of astrocytes, a type of glia abundantly present in the brain, may influence neuroplasticity between neurons [5,6], recent studies suggest that mechanical forces by the actin cytoskeleton may represent a crucial, yet overlooked, method of synaptic regulation [7], and that these musclelike forces at the synapses may be linked to associative learning, allowing an organism to map specific events to broader environmental contexts [8]. Based on our team's recent findings that demonstrate localized actin concentration at astrocyte boundaries upon mechanochemical stimuli [9], in addition to studies that reveal spontaneous rhythms in actin dynamics [10–13], we postulate that astrocytes may actively coordinate the rhythms of mechanical oscillations at the neuronal synapse to induce learning [13]. By implementing this bioinspired mechanism in an artificial neural network, we

^{*}Contact author: kang.hoony.2@gmail.com

[†]Contact author: wlosert@umd.edu

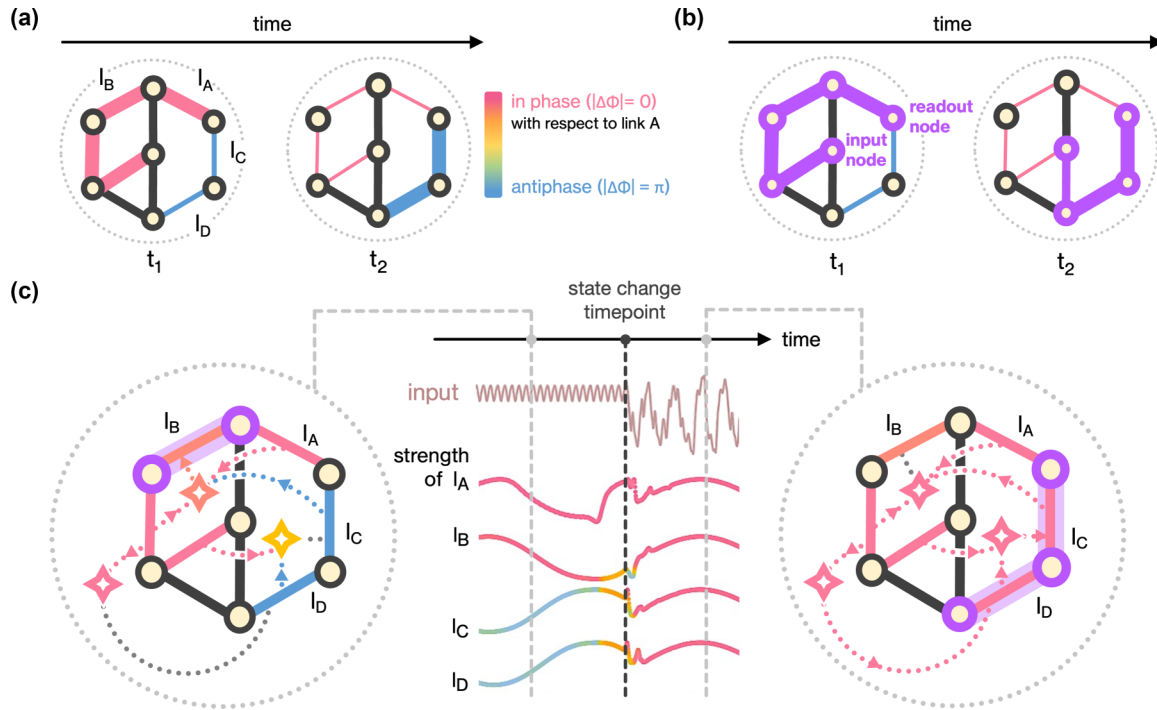


FIG. 1. Schematic of rhythmic sharing. (a) Temporal snapshots of oscillating link strengths, indicated by their widths. In this example, the degree of oscillation coordination, i.e., synchrony, is fixed, meaning the relative phases between links are fixed. (b) The effect of oscillating link strengths on information flow through the network. At each timepoint, the current value of the mean phase alters which links are the strongest, rerouting how information flows through the network. As an example, the most coherent path of information between some input node to some readout node is shown in purple. (c) Changing input alters the subset of active links in the network, altering the coordination among link oscillations. Here, the dotted lines represent the connections between links, and the mean phases between connected links are represented by stars; the degree to which this mean phase is in or out of phase to I_A is represented by a star's color. At earlier times, I_B was activated by the periodic orbit input, encouraging it to synchronize with the mean phase of its connected links (whose corresponding star is mostly still pink). Upon a change of input to a chaotic state, I_C and I_D become active, and they synchronize to their mean fields. This results in links I_C and I_D becoming synchronized to I_A . However, I_B , no longer active, is unaffected by its mean field.

aim to study whether these oscillatory dynamics could endow the network with the capacity for context awareness. If the network is indeed capable of distinguishing contexts in the sense of associative memory, we expect its weights must adapt in some way that is unique to each state.

We draw on these findings to propose a learning model, called rhythmic sharing, that harnesses these multimodal signals in two parts that endow the network with context awareness. First, we propose that learning involves rhythmic variations in link strength, based on spontaneous rhythms exhibited by biomechanics of the actin cytoskeleton. Second, we propose that learning occurs via coordination of the phases of these rhythmic variations. Astrocytes inspire this key element of our algorithm: Astrocytes wrap around thousands of synapses and exhibit rhythmic hotspots of biomechanical activity, which may enable the coordination among synapses [9].

These two concepts work together to generate a powerful learning paradigm: We implement these rhythms as slow oscillations in link strength of some or all links, and allow their phases to be different for each link [Fig. 1(a)]. Therefore, information flowing through the network can utilize multiple subnetworks, depending on the phases of the oscillations at a given time [Fig. 1(b)].

The second part of the paradigm is that the phases of each link can change depending on the information it processes, e.g., different states, as illustrated in Fig. 1(c). We propose that links that are active in processing a given state can adjust their phases rapidly to synchronize with other links. This synchronization is mediated by objects that can integrate information from subsets of links [inspired by the integrative role of astrocytes and denoted as stars in Fig. 1(c)].

Because input of a different state results in active link changes, state changes manifest as collective changes to synchronization of the link phases, as illustrated in Fig. 1(c). Therefore, link phase synchronization also functions as a classification token of each state.

We implement our algorithm on an artificial neural network and demonstrate that it can rapidly sense dynamic state changes in a range of dynamical systems including key model systems of chaos, serving as a digital twin for prediction and state targeting. When we specify the global goal of the network as time-series prediction, we find that our proposed paradigm of phase adaptation becomes a self-supervised mechanism that allows the network to learn how to autonomously adapt this goal for a specific context by coordinating link strengths.

B. Rhythmic sharing

We now expand on the qualitative overview of the learning paradigm with its quantitative formulation. We model the link interactions with the following governing equation:

$$\frac{d\Phi}{dt} = \omega_0 + (\varepsilon_1 + \varepsilon_2 \hat{\mathbf{Q}}^T \mathbf{n}^*) \circ \sin(\Psi - \Phi + \gamma), \quad (1)$$

where $\circ$ denotes the Hadamard product. Φ is a vector of phases $[\phi_1 \cdots \phi_{N_l}]^T$, with N_l being the total number of links. ω_0 is a vector of initially imposed natural frequencies of the links. The vector of *local* mean fields $\Psi(t) \in \mathbb{R}^{N_l}$ that each link is coupled to is defined by

$$\mathbf{r}(t) \circ e^{i\Psi(t)} = \hat{\mathbf{A}}_{\Phi} e^{i\Phi(t)}, \quad (2)$$

where $\mathbf{r} \in \mathbb{R}^{N_l}$ is a vector of *local* order parameters whose elements describe the degree of synchronization across the particular local mean field that each link is coupled to. $e^{i\cdot}$ is taken to operate on $\cdot$ element-wise, e.g., $e^{i\Psi(t)} = (e^{i\psi_1}, e^{i\psi_2}, \dots, e^{i\psi_{N_l}})$. $\hat{\mathbf{A}}_{\Phi_{ij}} \equiv \mathbf{A}_{\Phi_{ij}} / |\mathbf{A}_{\Phi_{i\cdot}}|$, where $\mathbf{A}_{\Phi} \in \mathbb{R}^{N_l \times N_l}$ is a random phase adjacency matrix, binary for simplicity, whose elements equal 1 if the links are connected and 0 otherwise, and the normalization factor $|\mathbf{A}_{\Phi_{i\cdot}}|$ is the L^1 norm of the i th row of $\mathbf{A}_{\Phi}$. Finally, γ is a phase lag, or frustration, vector whose role is akin to that of the Sakaguchi-Kuramoto model in promoting asymmetric communication [14]. The density and weights of the nonzero elements of $\mathbf{A}_{\Phi}$, ω_0 , and γ , are hyperparameters.

The amplitude of the coupling, $\varepsilon_1 + \varepsilon_2 \hat{\mathbf{Q}}^T \mathbf{n}^*$, encodes the causal direction of nodal activity promoting link interactions. Here, the elements of the normalized incidence matrix $(\hat{\mathbf{Q}}^T)_{ij} \equiv (\mathbf{Q}^T)_{ij} / |(\mathbf{Q}^T)_{i\cdot}|$ are of the incidence matrix $\mathbf{Q} \in \mathbb{R}^{N_n \times N_l}$, whose numerator elements equal 1 if a link is incident to a node (self-loops included) and 0 otherwise, and the normalization factor $|(\mathbf{Q}^T)_{i\cdot}|$ is the L^1 norm of $\mathbf{Q}^T$ across the i th row. We rescale the node states $\mathbf{n} \in (-1, 1)$ to $\mathbf{n}^* \in (0, 1)$ by defining $\mathbf{n}^* \equiv (\mathbf{n} + \mathbb{1}_{N_n})/2$ [where $\mathbb{1}_{N_n} = (1, 1, \dots, 1) \in \mathbb{R}^{N_n}$ denotes a vector of 1s of size N_n], because lack of activity of the i th node, $\inf(\mathbf{n}_i) = -1$, should translate to an *absence* of link-to-link interaction, $\inf(\mathbf{n}_i^*) = 0$. The hyperparameter ε_1 can engineer the disparity in phase coherence between different states. However, it is not to be set to a nonzero value if the hyperparameter $\varepsilon_2 = 0$, as this will promote global synchronization or desynchronization across the link network asymptotically, regardless of the state of the input system.

Finally, we discuss the vector hyperparameter of slow natural frequencies ω_0 . While its distribution and values are not constrained *a priori*, we consider the simple binary case where a set percentage of links share the same natural frequency $\omega_0 \neq 0$ and the rest are set to 0. The density of links with nonzero natural frequencies as $\Lambda(\omega_0) \in (0, 1]$ and ω_0 are, then, hyperparameters associated with ω_0 .

The *scalar* global order parameter used to track the coordination of link phases is defined similarly, $R(t) \in [0, 1]$, which is the degree of global synchrony across *all* the link phases:

$$R(t) e^{i\langle \Phi \rangle(t)} = \frac{1}{N_l} \sum_{k=1}^{N_l} e^{i\Phi_k(t)}. \quad (3)$$

Here, $\langle \Phi \rangle(t)$ is the corresponding *scalar* global mean phase. We note that this is equivalent to the definition of

the local mean fields defined previously, except where all elements of $\mathbf{A}_{\Phi}$ are replaced with 1.

Two critical behaviors separate the similar, well-studied Kuramoto [15] model and ours. First, when the forcing strength of a Kuramoto model is driven by a time-varying input, R evolves with the same timescale as the input itself [16]. Consequently, R effectively mirrors the input time series, remaining “blind” to the underlying context because it lacks a distinct stationary signature that distinguishes each input state. It therefore fails to provide information that extracts or represents the underlying context, beyond the raw fluctuations present in the input time series. This contrasts with our model, where we will see that R settles to a stationary value for each stationary state in the time series (Sec. II). Second, when a slowly oscillating link is not connected to any other link, it will quench its own oscillation, unlike the Kuramoto model where the link would simply oscillate at its natural frequency (see Appendix A 1). Our modification is to ensure that an uncoupled link does not introduce “blind” periodic perturbations that are decoupled from the links’ collective consensus, which may act as a source of noncontributory dynamics that would otherwise fail to align with the links’ attempts to coordinate their mean dynamics for context recognition.

Next, the equation describing the evolution of the node states of our recurrent neural network $\mathbf{n} \in \mathbb{R}^{N_n}$ is

$$\mathbf{n}(t + \Delta t) = \alpha \mathbf{n}(t) + (1 - \alpha) \tanh[\mathbf{A} \mathbf{n}(t) + \mathbf{W}_{\text{in}} \mathbf{u}(t) + \boldsymbol{\xi}], \quad (4)$$

where the activation function $\tanh[\cdot]$ is applied to each vector equation component in its argument separately. It is to be noted that when we refer to a node, neuronal, or reservoir state, we mean a point of the vector $\mathbf{n} \in (-1, 1)^{N_n}$; all other uses of the term “state” specifically refer to an invariant set, such as an attractor. α is the leakage hyperparameter that sets the rate of update and memory of $\mathbf{n}$, $\mathbf{A}_{\mathbf{n}} \in \mathbb{R}^{N_n \times N_n}$ is a random adjacency matrix between the nodes scaled to a desired spectral radius, $\mathbf{W}_{\text{in}} \in \mathbb{R}^{N_n \times N_u}$ is the random input matrix whose elements are chosen uniformly from the closed interval $[-\delta, \delta]$, $\mathbf{u}(t) \in \mathbb{R}^{N_u}$ is the input, and $\boldsymbol{\xi} \in \mathbb{R}^{N_n}$ is a neural bias that can be used to shift the activation function of the neuronal nodes. While Eq. (4) subsumes the nonlinearity of activation to each node in order to eventually use the network as a reservoir computer [17,18] in Sec. II B, we note that it is equivalent to other recurrent neural network update equations up to a coordinate transformation [19]. As such, we do not lose generality by using an echo state-type network. The spectral radius and average degree of $\mathbf{A}_{\mathbf{n}}$, the input scaling δ , and the neural bias $\boldsymbol{\xi}$ are hyperparameters.

Finally, to incorporate oscillating link strengths with fixed relative amplitude into the neural network, Eq. (4) is modified by amending $\mathbf{A}_{\mathbf{n}}$ to exhibit a simple oscillation dependence, $\tilde{\mathbf{A}}_{\mathbf{n}}(t)$, where

$$\tilde{\mathbf{A}}_{\mathbf{n}}(t) \equiv \mathbf{A}_{\mathbf{n}} \circ \left[1 - \frac{m}{2} (1 + \sin[\langle \Phi \rangle(t)]) \right]. \quad (5)$$

Here, m is the relative fraction of link strength change during oscillation. The purpose of constructing the modulation this way is so that the strength is always bound between the upper bound that is its original strength $\mathbf{A}_{\mathbf{n}_{ij}}$ and its lower

bound $(1 - m)\mathbf{A}_{n_{ij}}$ to restrict the maximum spectral radius, such that the echo state property does not become inadvertently violated [18]. Finally, the phase matrix $\Phi \in \mathbb{R}^{N_n \times N_n}$ is simply a collection of individual link phases ordered into the same structure as the adjacency matrix:

$$\Phi(t)_{ij} := \begin{cases} \phi_{j \rightarrow i} & : \mathbf{A}_{n_{ij}} \neq 0, \\ 0 & : \mathbf{A}_{n_{ij}} = 0. \end{cases} \quad (6)$$

As such, the vector Φ in Eq. (1) is simply the nonzero elements of Φ collected into a vector of phases $[\phi_1 \cdots \phi_{N_l}]^T$ with N_l being the total number of links; i.e., for some $j, k \in [1, N_n]$, we can represent $\phi_i := \phi_{j \rightarrow k}$ for all $i \in [1, N_l]$.

II. RESULTS

A. Unsupervised state estimation with rhythmic sharing

1. Detecting simple state changes

We choose the 3D Thomas's cyclically symmetric system [20] as our first example input because it hosts numerous states with different values of its internal parameter, λ_{Thomas} . The following equations govern this system:

$$\begin{aligned} \dot{x} &= \sin(y) - \lambda_{\text{Thomas}}x, \\ \dot{y} &= \sin(z) - \lambda_{\text{Thomas}}y, \\ \dot{z} &= \sin(x) - \lambda_{\text{Thomas}}z. \end{aligned} \quad (7)$$

The parameter λ_{Thomas} induces bifurcations in the system. We simulate input data where λ_{Thomas} starts at 0.18, which results in a chaotic trajectory, and then suddenly switches to 0.29, which results in two coexisting periodic orbits (limit cycles). This simulated trajectory $(x(t), y(t), z(t))$ is used as the input $\mathbf{u}(t)$ to our neural network, and is sent frame by frame to the nodes of a neural network whose connection weights and link phases are initially random. The evolution of the link phases as data enter the neural network is then tracked, as shown in Fig. 2(a). The collective dynamics of the link phases are characterized by their order parameter $R \in [0, 1]$ and the mean phase $\langle \Phi \rangle \in [-\pi, \pi)$. We call that $R(t)$ measures the degree of global synchronization of link phases: $R = 0$ indicates uniform dispersion, i.e., complete decoherence, of the phases, while $R = 1$ indicates complete synchronization of the phases.

We emphasize two observations about $R(t)$ in Fig. 2(a). First, R converges to some equilibrium value whenever the input trajectory belongs to a stationary state. For instance, R evolves from its initial condition of 0 and reaches a steady-state value of 1 within roughly two “cycles” of the chaotic state ($\lambda_{\text{Thomas}} = 0.18$) and maintains this value hereafter. This means that the links detected an invariant measure underlying the trajectory, even when the trajectory is chaotic; in other words, the links detected that a stationary probability distribution underlies the time evolution of the trajectory. Second, each state gives rise to a unique equilibrium value of R . Therefore, the evolution of R mimics the evolution of λ_{Thomas} .

The dynamics of the mean phase $\langle \Phi \rangle(t)$ are not as interpretable yet, as they do not respond uniquely to how λ_{Thomas} evolves in time. However, we will see that they will be critical in the ability of the network to recall and predict various states (Sec. II B).

2. Sensing continuous parameter changes on systems with memory

The future evolution of a system may have a nontrivial dependence on its history. These systems are particularly abundant in biology, such as the immune and nervous systems [21,22]. One such example is the one-dimensional (1D) Mackey-Glass system [23]. For this input system, the internal parameter λ_{Mackey} that induces state changes represents the time delay, or memory, of the dynamics. We vary this parameter in a sinusoidal manner when simulating data.

Specifically, the normalized form of the 1D Mackey-Glass system is given by

$$\dot{x}(t) = \beta \frac{x(t - \lambda_{\text{Mackey}})}{1 + x(t - \lambda_{\text{Mackey}})^n} - \gamma x(t). \quad (8)$$

Here, the exact time dependence of the state variable x is provided for clarity. λ_{Mackey} is the time delay parameter responsible for inducing bifurcations of the attractor (parametrized with phase space coordinates $(x(t), x(t - \lambda_{\text{Mackey}}))$). We fix $\beta = 0.2$, $\gamma = 0.1$ and $n = 10$, while we oscillate λ_{Mackey} sinusoidally between $\lambda_{\text{Mackey}} = 20$ and $\lambda_{\text{Mackey}} = 24$ with a slow angular frequency ($= 0.007$) during the simulation of our data.

Figure 2(b) shows the response of the links to this input data. While R is shakier, likely due to the input system's complexity and the continuous nature of the parameter change, it still successfully mimics the evolution of λ_{Mackey} .

3. Detecting warning signs of catastrophic phenomena

Finally, we study a system where changes to its internal parameters are not immediately apparent or measurable with current methods, but whose detection is critical to prevent the impending collapse of a system's dynamics.

A well-known chaotic system that exhibits this behavior is the 3D Lorenz system [24]. The following equations govern this system [24]:

$$\begin{aligned} \dot{x} &= \sigma(y - x), \\ \dot{y} &= x(\lambda_{\text{Lorenz}} - z) - y, \\ \dot{z} &= xy - \beta z. \end{aligned} \quad (9)$$

Here, changes to the parameter λ_{Lorenz} are responsible for bifurcations to the system. We fix the other parameters to values $\sigma = 10$, $\beta = 8/3$ for the duration of the entire simulation, while we switch the value of λ_{Lorenz} abruptly from 24.5 to 23.5 in our simulated data, as in the case for the Thomas system. When $\lambda_{\text{Lorenz}} \gtrsim 24.06$, the Lorenz system exhibits a globally stable chaotic trajectory. When λ_{Lorenz} abruptly drops to a value between $13.93 \lesssim \lambda_{\text{Lorenz}} \lesssim 24.06$, the chaotic trajectory shifts into another similar, but transient chaotic trajectory (saddle) until it suddenly collapses into a dead signal after some probabilistic time [25]. Our choice of $\lambda_{\text{Lorenz}} = 23.5$ is due to its proximity to the bifurcation threshold at $\lambda_{\text{Lorenz}} \approx 24.06$, which minimizes visual changes to the trajectory as the chaotic attractor transitions into a saddle. Indeed, Fig. 2(c) shows the close similarity in the spectral properties of the two chaotic trajectories; to resolve the spectral peak differences (≈ 0.1 – 0.2 Hz), at least 50–100 cycles of the frequency bands containing the peaks must be observed.

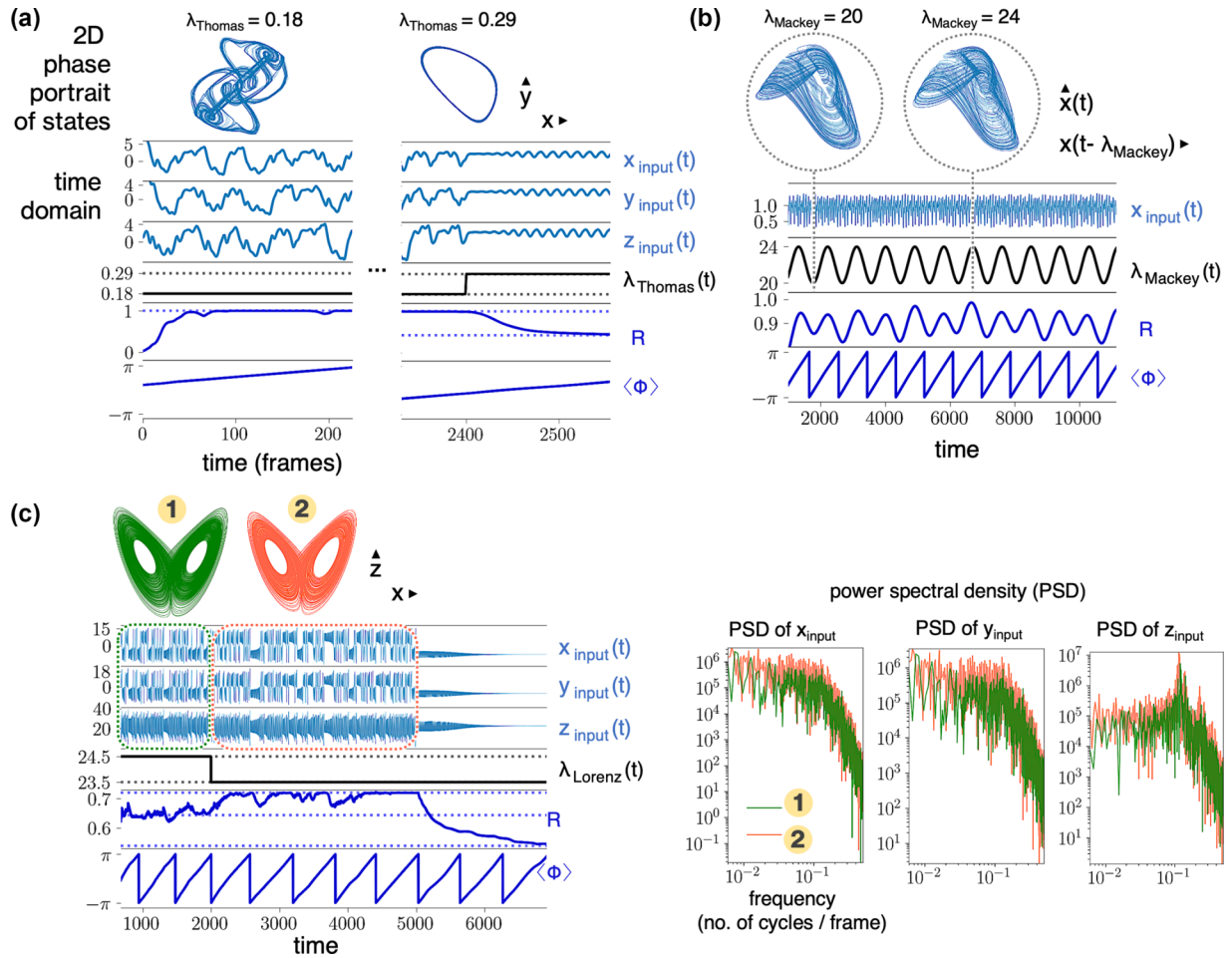


FIG. 2. Rapid sensing of dynamic state changes. (a) A simulated trajectory of the 3D Thomas system (whose state variables are x , y , z) in blue, with the value of its internal parameter λ_{Thomas} that was used to simulate the trajectory shown in black. After an initial transient, R remains near an equilibrium value of 1 for the entire chaotic trajectory. When λ_{Thomas} is abruptly switched to 0.29, the trajectory switches from a chaotic state to a periodic orbit. R senses this switch immediately and settles to a new equilibrium value. (b) The same procedure was conducted for the Mackey-Glass system, whose internal parameter λ_{Mackey} was changed sinusoidally during simulation. (c) (Left) R for the 3D Lorenz system. R , although noisier than the Thomas system, evolves to a different equilibrium value at the exact time λ_{Lorenz} was switched, falling to a different value as the Lorenz system transitions to a dead signal. (Right) The power spectral densities (PSD) of the time series of each state variable, corresponding to when the Lorenz system is at stable chaos (green) and unstable, or transient, chaos (orange). That their spectral differences are also not obvious contributes to the difficulty in sensing this state transition.

The behavior of $R(t)$ of Fig. 2(c) demonstrates that the links sensed the moments when the states were switched: from the first chaotic trajectory to the chaotic transient, and then to the dead signal. The time it took for $R(t)$ to escape its range of values while it was sensing the chaotic trajectory and to enter its range associated with the chaotic transient is around 100 frames, or 11 cycles—approximately 78% fewer observations from that suggested by Fourier analysis.

B. State transformation and recall by rerouting information flow with rhythmic sharing

A powerful feature of a neural network is its ability to store multiple dynamical states. If states were tagged with contextual tokens in the training data, one may recall a specific state by inputting the associated token entry. Without these tokens, recall is very difficult.

While R seems to fulfill the role of the required contextual token, the link phases' mean dynamics are also governed by $\langle \Phi \rangle$, both of which shape the “neural representation” of the input. Therefore, we cannot neglect its specification when attempting to recall a state. Yet, the latter's role is not visually clear from Fig. 2, as it does not seem to respond to state changes. To assess this, we will observe how the neural output changes in response to changes to R and $\langle \Phi \rangle$.

To obtain an output from the network, we train the network as a reservoir computer (RC), a form of a recurrent neural network that has been noted to perform well for chaotic dynamics prediction [17,26–28]. The training of an RC involves finding the optimal combination of nodes such that their combined output best recovers the true value of the input time series at the next time step. More specifically, in order to use our network as a reservoir computer, an additional step is performed that projects the latent space of the nonlinear neuronal states $\mathbb{R}^{N_n}$ to the prediction space via a static linear map to produce

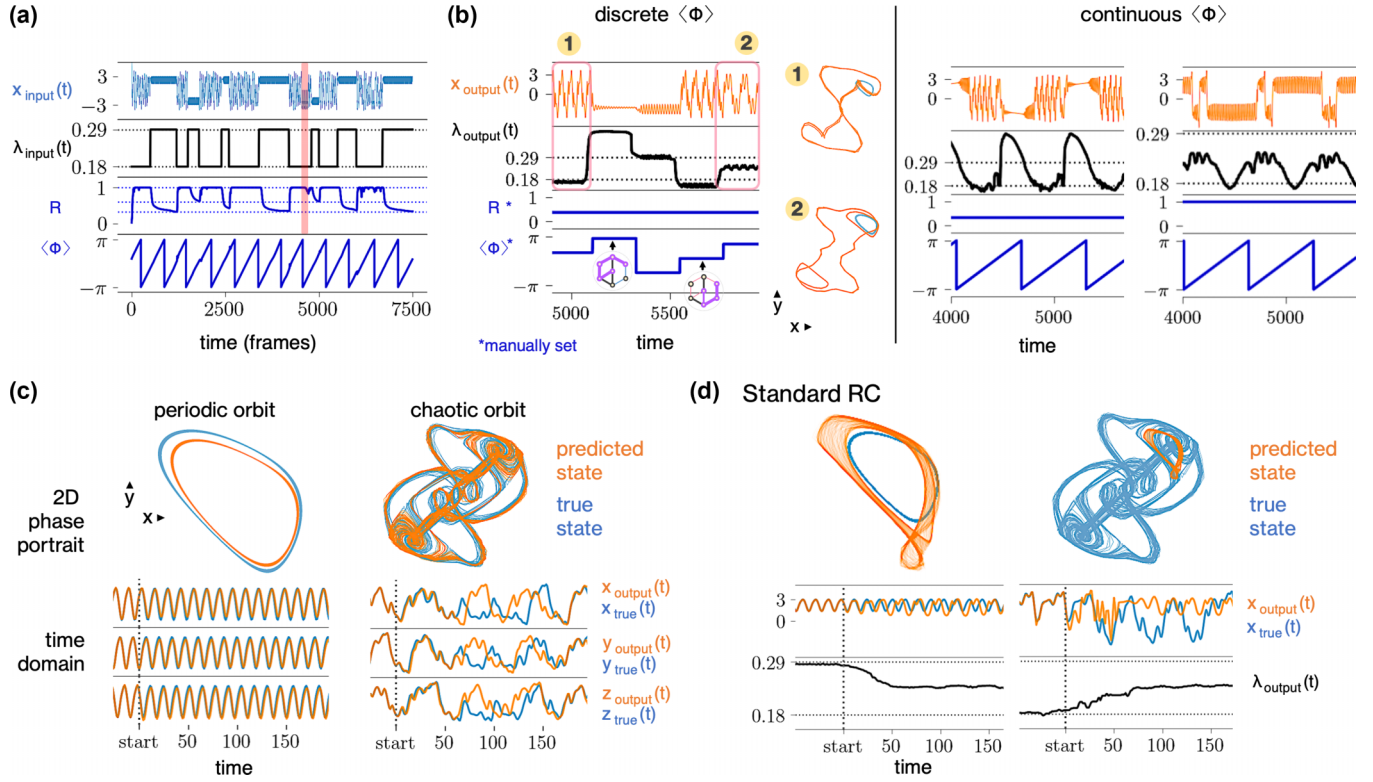


FIG. 3. Generating a digital twin for prediction and state targeting. (a) Training data used for the Thomas system and the response of the links, characterized by R and $\langle \Phi \rangle$. When the chaotic trajectory exhibits limit-cycle-like behavior (inside the red highlighted strip), R briefly falls below its steady-state value, but remains overall robust to these perturbations. (b) Closed-loop predictions of different states by modifying R and $\langle \Phi \rangle$ in real time. Some extrapolated states that the network has never seen before are highlighted with pink boxes in the close-up time series, and their states are, respectively, shown in coordinate space in orange as well. Here, the light blue state represents the initial input state (the periodic orbit), before closed-loop prediction began. Each value of R and $\langle \Phi \rangle$ amounts to sampling different network paths. (c) Closed-loop prediction of individual states present in the training data without being contaminated by other states. The phase portrait representation (top) shows the state of the predicted trajectory that ran for over 5000 frames, which matches closely to the state of the true trajectory. A close-up, time-domain representation of the prediction (bottom) showcases the forecasting accuracy. “Start” represents the beginning of the closed-loop prediction stage. (d) Closed-loop time series predictions of a standard RC, which learned the training data as one state and therefore could not predict each state independently.

the desired outputs, $\eta : \mathbb{R}^{N_n} \rightarrow \mathbb{R}^{N_\eta}$, where

$$\eta(t) = \mathbf{W}_{\text{out}} \mathbf{n}(t). \quad (10)$$

For time-series prediction, $\eta(t)$ corresponds to $\tilde{\mathbf{u}}(t + \Delta t)$, which assumes the prediction of the inputs $\mathbf{u}$ after one time step Δt . The training goal of the reservoir computer, then, is to obtain the matrix $\mathbf{W}_{\text{out}}$ such that $\tilde{\mathbf{u}}(t + \Delta t)$ best approximates $\mathbf{u}(t + \Delta t)$ for chosen hyperparameters. Details on the training and prediction mechanisms are provided in the Appendix A 2.

We reuse the Thomas system, as its large repository of states is well suited for our demonstration of state targeting. We train the RC with simulated data of the Thomas system that alternates between two values of λ_{Thomas} , with our objective being whether the RC can predict the dynamics of each state independently in the training data via rhythmic sharing [Fig. 3(a)]. We will henceforth refer to λ_{Thomas} of the input, or training, data as λ_{input} to clarify that this is the internal parameter of the input trajectory ($x_{\text{input}}(t)$, $y_{\text{input}}(t)$, $z_{\text{input}}(t)$) that is sent to the network. Our objective is motivated by the fact that if the RC can recognize, separate, and learn stationary events differently with the assistance of rhythmic

sharing, it must also be capable of *predicting* each stationary event indefinitely without mixing the other events that it has seen during training by controlling the phases. The alternations between the states in our simulated input data are made aperiodic to ensure that our results are not due to some resonance between the natural frequencies of the link oscillations ω_0 and that of $\lambda_{\text{input}}(t)$. From Fig. 3(a), we see that, as before, $R(t)$ tracks the state changes in the data: $\lambda_{\text{Thomas}} = 0.18$ admits one chaotic state, and $\lambda_{\text{Thomas}} = 0.29$ admits two co-existing periodic orbits, resulting in three equilibrium values of R .

Once the optimal set of output nodes is obtained by finding $\mathbf{W}_{\text{out}}$, we run the network in a closed-loop prediction scheme, where the network output $\tilde{\mathbf{u}}$ (i.e., the prediction of $\mathbf{u}$ of the next frame) serves as the next input $\mathbf{u}$ for the prediction of the subsequent frame. To control the phases, we freeze R once it reaches its equilibrium value for the input state, then commence the closed-loop prediction scheme. Then, as the closed-loop prediction is running, we manually change $\langle \Phi \rangle$ to observe its effects on the output. Further details on how this control mechanism was conducted are provided in Appendix A 2 c.

Figure 3(b) showcases three different examples of how $\langle\Phi\rangle(t)$ was changed during the closed-loop prediction. In the first example, where $\lambda_{\text{input}} = \lambda_{\text{periodic orbit}} \equiv 0.29$ as $\langle\Phi\rangle(t)$ evolves discretely, each discrete value of $\langle\Phi\rangle$ results in neuronal output of a different state of the Thomas system in the neighborhood of $\lambda_{\text{periodic orbit}}$. Most of these states map to values of λ_{Thomas} that the network never encountered during training. Because each $\langle\Phi\rangle$ value qualitatively represents a different static weight configuration of the network, one may say that we are effectively reading out different states embedded in different information pathways associated with each configuration.

In the second example, where $\lambda_{\text{input}} = \lambda_{\text{periodic orbit}}$ but the mean phase $\langle\Phi\rangle$ evolves *continuously* in a linear manner, it is evident that $\lambda_{\text{output}}(t)$ [the internal parameter associated with the output trajectory $(x_{\text{output}}(t), y_{\text{output}}(t), z_{\text{output}}(t))$] is entrained with $\langle\Phi\rangle$. This demonstrates that $\langle\Phi\rangle$ is an effective controller that can reliably steer the neural states to desired trajectories corresponding to various λ_{output} ; in other words, accessing different paths of the network recalls different states. These observations hold for the third example, where setting $\lambda_{\text{input}} = \lambda_{\text{chaos}} \equiv 0.18$ yields a different extrapolation curve in the neighborhood of λ_{chaos} . The results suggest that the mean-field phase parameters can be used to predict how the time series will evolve when the context, or state, of the time series switches in real time—in other words, a digital twin.

With these findings, we are now able to easily achieve our objective: By freezing the proper values of R , $\langle\Phi\rangle$ such that $\lambda_{\text{output}} = \lambda_{\text{input}}$, the network can predict the time series of each state present in the training data separately without contamination of other states, as shown in Fig. 3(c). Although its predictive performance is not as accurate as an RC trained purely on either state alone, a single RC endowed with rhythmic sharing can accurately predict the long-term dynamics of each individual context or state.

In contrast, a standard RC not equipped with contextual tokens cannot learn the different states in the nonstationary training data as distinct events and attempts to treat the data as originating from one *stationary* event. This can be seen in Fig. 3(d), where the RC predicted a trajectory whose λ_{output} is intermediate of the two λ_{input} of the training data. That the reservoir treated the events in the training data as one is evident by the fact that it outputs the same trajectory for both test states.

III. DISCUSSION AND CONCLUSION

Motivated by neuroglial interactions of the brain, we developed a learning paradigm consisting of two key elements: the slow oscillations of the links and the ability to change their coordination. We implemented our model into an artificial neural network to probe the role of these mechanisms in neural representation.

In our model, links rapidly adjust their synchronization to identify the current state dynamics. Thus, the synchrony of links identifies the current state of the dynamics in the data. By self-adjusting link strength rhythms for different states, the network is capable of learning nonstationary dynamics. This is a powerful feature that regular machine learning algorithms

cannot handle without user-provided contextual tokens, despite the ubiquity of nonstationary dynamics in real-world data. Our results demonstrate that the network is capable of rapidly identifying anomalous events in data and signaling warning cues for impending disasters. Furthermore, this is achieved in real time because the network did not require prior training.

Current approaches to identifying state changes in untrained nonstationary data generally rely on spectral methods or density-based anomaly detection. While Fourier analysis serves as a baseline method to identify data requirements in estimating a state change, techniques such as the wavelet and Hilbert-Huang Transforms (HHT) seek to offer higher temporal resolution for resolving state switches from a shorter observation window [29]. However, these methods remain highly sensitive to the specific data window being analyzed, which makes them unsuitable for chaotic time series. For instance, in the case of HHT, the process of extracting the intrinsic frequency modes is purely empirical and window-dependent, such that two snapshots of the same chaotic attractor will often yield nonaligned frequency modes, thereby preventing a stable characterization of the state unless a large number of windows have been analyzed to build a statistical characterization of the modes [30]. Wavelet analysis method relies on convolutions with predefined bases, meaning that two snapshots of the same attractor in some window will also yield different wavelet analysis results, and therefore does not inherently identify the underlying dynamical structure of the state as well. Similarly, anomaly detection algorithms that do not require training, such as distance-based and density-based algorithms, require the system to traverse a significant portion of the strange attractor before a stationary distribution can be defined [31]. Our proposed model addresses these limitations by leveraging a link network that settles into a stationary order parameter for each stationary state, with high resolution and minimal data, that serves as a consistent quantitative signature of the underlying dynamical process regardless of the specific trajectory snippet observed in a window.

Our results demonstrate that slow link strength rhythms in our model enable extrapolation of dynamics of numerous unseen states, and by training the network endowed with rhythmic sharing, we can predict the stationary dynamics of each state. This asset may be used to harness the network as a digital twin, capable of adapting its prediction when the physical twin experiences state changes. As an alternative perspective, we mention that our total neural network system, comprising a standard reservoir computer equipped with rhythmically synchronizing weights, may be thought of as a recurrent neural network where a trained, static output matrix $\mathbf{W}_{\text{out}}$ encodes the fixed global task of time series prediction, while rhythmic sharing adapts this task locally to the specific context by learning the optimal synchronized weight configuration for a given dynamical state. In other words, rhythmic sharing performs a form of self-supervised learning, where they reconfigure the network's internal dynamics to optimize for a specific, self-recognized state, not with external labels or backpropagation but through dynamic rerouting of information flow in a bioinspired learning process.

We restricted the application of our algorithm to input systems whose nonstationarity is induced by a single parameter.

This is because state changes brought on by more than one parameter cannot be causally differentiated from the $R(t)$ plot alone, which will combine all sources of change to the input system. A future direction would be to design a model where multiple clusters can individually track the nonstationary evolution of different internal parameters of the input system, thereby creating small-world networks of links. Such a configuration may enable the network to identify shifts across multiple contextual cues simultaneously and potentially isolate the exact causal mechanism responsible for a global state change. A straightforward mechanism to implement arbitrary orders of $p : q$ synchronization for cluster formation in our model is to modify the argument of the right-hand side of Eq. (1) from $\sin(\Psi - \Phi) \rightarrow \sin(\mathbf{p} \circ \Psi - \mathbf{q} \circ \Phi)$, where $\mathbf{p}$ and $\mathbf{q}$ are vectors of positive integers whose elements $\mathbf{q}_i$, $\mathbf{p}_i$ denote the desired synchronization order between the phase Φ_i and its connected mean field Ψ_i . However, careful engineering of the structure of $\mathbf{A}_\Phi$ is likely to be needed to minimize competition of different synchronization orders between links.

We also take this opportunity to discuss the impact of noise (originating from either the externally provided input or internal neural states) on the link system's stability. Our results provide some insight into the inherent robustness of R against such input disturbances. Specifically, as shown in Fig. 3(a) in the red highlighted strip, when the chaotic trajectory of the Thomas system undergoes transient limit-cycle-like orbits, R exhibits only brief, minor deviations from its steady-state value of 1. Importantly, the change in R during these periods is significantly lower than the rapid decay observed during a true bifurcation. This suggests that the link system effectively distinguishes between transient fluctuations and true state changes, resistant to overreacting to transient fluctuations. This resilience stems from the observation that the mean-field dynamics of the link phases seem to identify the underlying distribution over a specific trajectory. Consequently, when additive stationary noise present in the input enters the phase variables via the forcing strength, $\varepsilon_1 + \varepsilon_2 \mathbf{Q}^T \mathbf{n}(\mathbf{u})$, we expect that the qualitative behavior of R should remain robust. However, the convergence time of R for a specific state may increase if the noise introduces long autocorrelation timescales that require more observations to resolve than the underlying, noiseless input system, potentially delaying context recognition.

An interesting case to consider is multiplicative noise, which is inherent to real neural networks where dynamics are dominated by the vast density of interneuronal connections. In such interaction-heavy systems, multiplicative noise typically yields fat-tailed statistical distributions, e.g., log-normal or Pareto-Lévy distributions, whose resulting power laws are well studied in the dynamics of biological neuronal networks [32–35]. Qualitatively, these power laws emerge as local noisy kicks percolate through the system's metastable states, triggering chains of spatially extended excitations or “avalanches” that upset these states. This process continues until the system settles into clusters of more barely stable states that span multiple length and timescales, resulting in the observed power laws. Since the dynamics of the neuron states $\mathbf{n}$ directly dictate the synchronization forcing strength for their respective incident links, we surmise that a network of clustered neuron states (hereafter “ $\mathbf{n}$ clusters” for brevity)

may inherently promote clustering or chimera states across the link phases. In this regime, distinct groups of links would synchronize locally to the dynamics of a specific $\mathbf{n}$ cluster while remaining phase decoupled from other link clusters; in a sense, this may be seen as a natural manifestation of higher-order $p : q$ synchronization within the network. Given that $\mathbf{n}$ clusters have been effectively utilized in artificial neural networks to store and predict the dynamics of multiple states via external labels that bias each $\mathbf{n}$ -cluster's activity levels [36], it is plausible that a rhythmic sharing network subject to multiplicative noise may be another way of generating link clusters that may be naturally specialized to monitor specific subsets of coupling parameters in the input system.

In our model, one method to stabilize the sensitivity of the order parameter R to input noise is to increase the degree of the link adjacency matrix. When the degree of the link adjacency matrix is increased, the mean phase-field dynamics are smoothed as local link phase perturbations caused by noisy inputs are quenched by the collective consensus of the link phases. While this increases robustness, it also increases the convergence time of R upon an input state change. Another method is to tune the balance between the hyperparameters ε_1 , which is the stimulus-independent strength that intrinsically drives the phases toward a synchronized or desynchronized configuration, and ε_2 , which is the gain on the synchronization forcing from neuronal activation. Together, these hyperparameters control the overall sensitivity of R to input state changes and the separation between steady-state levels of R . In a biological context, we postulate that these parameters may be tuned via neuromodulation. Specifically, the degree of native synchronized activity within the astrocytic and actin networks may modulate ε_1 , while the mechanical coupling strength between the actin network and the neurons at the synapse could serve to modulate ε_2 . Such biological regulation would allow the system to balance its stability to input distraction against its sensitivity in detecting contextual shifts.

Finally, we mention that Eq. (6) of our model assumes a one-to-one correspondence between the phase of link strength modulation and the phases of the links. In other words, the phases of the astrocyte chemical waves are identified with the phases of the actin waves at the synapse. While this was assumed mainly to preserve the simplicity of our model, experimental observations that the chemical oscillations present in astrocytes may be 1:1 synchronized to the actin oscillations that are found in astrocyte hotspots provide potential justification for this simplifying assumption [13]. As a future direction, one may introduce a more accurate map $\Phi = f(\phi)$ that integrates the dynamics of how chemical dynamics of the astrocytes are translated to these mechanical waves, for instance, as an excitable system.

ACKNOWLEDGMENTS

This material is based upon work supported by the Air Force Office of Scientific Research Biophysics Program under Award No. FA9550-21-1-0352 and the Army Research Laboratory Cooperative Agreement No. W911NF2320040. The authors would like to thank Nicholas Mennona, Anna M. Emenheiser, Sylvester J. Gates III, Walter Peregrin, Karima

Jeneh Perry, Noah Chongsiriwatana, and Kate M. O'Neill for discussions.

H.K. and W.L. jointly conceived the learning paradigm. H.K. developed and wrote the algorithm used in the manuscript. H.K. and W.L. have jointly written the manuscript.

DATA AVAILABILITY

The dataset of the Thomas system used in this study is provided on GitHub [37], and the equations and information necessary to generate the other simulated data are provided in the article. The data that support the findings of this article are openly available [37].

APPENDIX

1. Oscillation death for isolated links in rhythmic sharing

In Kuramoto-like models [15], every oscillator is independently coupled to a subset of other oscillators in the network. If an oscillator is not coupled to any other oscillator, its phase will evolve with its unperturbed natural frequency ω_0 .

However, our choice to couple each oscillating link to the mean field of its connections endows the link system with the property that any isolated, active link will eventually quench its oscillations for $|\omega_0| < |\varepsilon|$. To see how oscillation death happens in our model, let us consider a single oscillator, ϕ (where we drop indices for brevity) that evolves without any link-to-link interaction:

$$\dot{\phi} = \omega_0 + \varepsilon \sin(\Psi_0 - \phi), \quad (\text{A1})$$

where Ψ_0 is the constant $\text{atan2}(0./0.)$, which is mathematically undefined yet computationally yields some numerical value due to floating point accuracy. $\varepsilon \equiv \varepsilon_1 + \varepsilon_2(\mathbf{Q}^T \mathbf{n}^*)_\phi$ is the strength of the interaction for this particular oscillator, which is bounded in general between $\varepsilon_1 \pm \varepsilon_2$. Here, ε is frozen as a constant for simplicity and to obtain an analytic solution.

By a translation, $\phi \rightarrow \phi - \Psi_0$, Eq. (A1) reduces to

$$\dot{\phi} = \omega_0 + \varepsilon \sin(-\phi). \quad (\text{A2})$$

By use of a half-tan substitution, the solution can be easily shown to be

$$\phi(t) = 2 \tan^{-1} \left[\frac{\varepsilon}{\omega_0} + \frac{\Delta}{\omega_0} \tan \left(\frac{\Delta}{2} t + \xi_0 \right) \right], \quad (\text{A3})$$

where $\Delta \equiv \sqrt{\omega_0^2 - \varepsilon^2}$ and $\xi_0 \equiv 2 \tan^{-1} \left[-\frac{\varepsilon}{\Delta} + \frac{\omega_0}{\Delta} \tan \frac{\phi_0}{2} \right]$ with ϕ_0 being the initial phase. It suffices to show that if the argument of the $\tan^{-1}$ has a real limit for large time given small ω_0 (compared to the order of ε , which will be on the order of unity for synchronization), then $\tan^{-1}$ (and therefore ϕ) will converge to a real value as $t \rightarrow \infty$.

For $|\omega_0| < |\varepsilon|$, Δ and ξ_0 are purely imaginary, making both the amplitude and the argument of the nested tan in the expression for $\phi(t)$ also purely imaginary. As such, $\phi(t)$ evolves as $2 \tan^{-1}[a + b \tanh(ct + d)]$, where a , b , c , and d are real constants. Since $\tanh(ct + d)$ possesses a real limit for large t , $\phi(t)$ converges to a real limit; i.e., the oscillation has quenched.

On the other hand, consider $|\omega_0| > |\varepsilon|$. Then, Δ and ξ_0 are purely real, and $\phi(t)$ evolves as $2 \tan^{-1}[a + b \tan(ct + d)]$, where a , b , c , and d are reused to denote arbitrary real constants. Because the argument does not have a real limit for large t , the phase $\phi(t)$ also does not. Therefore, $|\omega_0| < |\varepsilon|$ for oscillation death to occur in isolated links.

2. Overview of reservoir computing with echo state networks

This section expands on the training and prediction procedures for an echo state network.

a. Training

First, a reservoir is initiated into a “warm-up phase” for τ_{warm} steps of Δt of the input. As a reservoir computer has leaky memory, this stage is required to provide sufficient time for a reservoir to forget its initial state (usually a blank slate $\mathbf{n} = 0$). Here, the input $\mathbf{u}$ is the training data $\mathbf{u}(t) = \mathbf{u}_{\text{train}}(t)$ so as to entrain $\mathbf{n}(t)$ to the dynamics of the input data without any feedback error. As this stage is used solely to calibrate the nodes’ state space with that of the data, these reservoir states are discarded and not used for training.

Upon completion, the reservoir enters the “training phase” for τ_{train} steps of Δt . The now-calibrated states are once more entrained by $\mathbf{u}(t) = \mathbf{u}_{\text{train}}(t)$, but all the reservoir states during this phase $\mathbf{n}(t_{\text{warm}} + \Delta t)$, $\mathbf{n}(t_{\text{warm}} + 2\Delta t)$, $\dots$, $\mathbf{n}(t_{\text{warm}} + \tau_{\text{train}}\Delta t)$ are recorded. For notational brevity, $t_{\text{warm}} = \tau_{\text{warm}}\Delta t$ marks the end of the warm-up period. Using the concatenated state responses across time, the optimal output matrix $\mathbf{W}_{\text{out}}$ is then calculated via L^2 regression by minimizing

$$\arg \min_{\mathbf{W}_{\text{out}}} \left[\sum_{t=t_{\text{warm}}+\Delta t}^{t_{\text{warm}}+\tau_{\text{train}}\Delta t} \|\mathbf{W}_{\text{out}}\mathbf{n}(t) - \mathbf{u}_{\text{train}}(t)\|^2 + \beta \text{tr}(\mathbf{W}_{\text{out}}\mathbf{W}_{\text{out}}^T) \right], \quad (\text{A4})$$

where β is the regularization hyperparameter.

b. Predicting

Equipped with the trained $\mathbf{W}_{\text{out}}$, we are now ready to commence the prediction phase. A portion of the test data $\mathbf{u}_{\text{test}}$ is sent into the reservoir (which can either start from a blank slate again or from its last state during the training phase) for another warm-up period to forget its previous state and recalibrate it to the data to be predicted. We note in passing that the warm-up period for the test data need not be the same as that of training. The last frame of the test data in the warm-up period $\mathbf{u}_{\text{test}}(t_{\text{warm}})$ is now used as the initial condition to generate the first prediction, $\tilde{\mathbf{u}}(t_{\text{warm}} + \Delta t)$. This output is then fed back as the new input to obtain the reservoir’s next state, forming a *closed-loop* prediction for the next time step. This process is iterated until the reservoir has advanced up to our desired prediction window.

c. State targeting and prediction by steering ($\mathbf{R}$, (Φ))

For our modified architecture, we perform the same procedure as above, but modify the warm-up stage of the prediction slightly if we wish to predict the time series of a single state, as it was done in Fig. 3(c). As the network warms up to the

test data, the links reinitialize their order parameter to the test state. Once the order parameter has reached its steady state (which we denote here as R_0), we freeze R by imposing that the phases evolve in any arbitrary manner according to a prescribed vector-valued forcing $\mathbf{g}$ satisfying

$$\begin{aligned}\dot{\Phi}(t) &= \mathbf{g}(\Phi, t), \quad t \geq t_{R_0}, \\ \mathbf{g}(\Phi, t) &= g(t)\mathbb{1}_{N_l}, \\ \Phi(t_{R_0}) &= \Phi_{R_0},\end{aligned}\tag{A5}$$

where g is some scalar-valued function, $\mathbb{1}_{N_l} = (1, 1, \dots, 1) \in \mathbb{R}^{N_l}$ denotes a vector of 1s of size N_l , and Φ_{R_0} denotes as the vector of phases at the first instance t when $R = R_0$, which we denote as t_{R_0} . This is because the system of equations satisfies the constraint $R(t) = R_0$ for $t \geq t_{R_0}$, where $t_{R_0} < t_{\text{warm}}$, since the forcing does not depend on the phase, and so the Lyapunov exponent of the link system is 0 (the phases do not diverge from each other in time).

We choose $g = \Omega t$, with $\Omega = \omega_0$, such that the states can evolve slowly until the target mean phase $\langle \Phi \rangle = \langle \Phi \rangle_0$ can be obtained. Once our desired $\langle \Phi \rangle_0$ is reached, we replace $g = 0$ to freeze the links permanently. We determine our desired $\langle \Phi \rangle_0$ by finding

$$\begin{aligned}\langle \Phi \rangle_0 &= \arg \min_{\langle \Phi \rangle} [h(\hat{\mathbf{W}}_{\text{out}} \mathbf{n}(\Phi, t), \mathbf{u}_{\text{test}}(t))] \\ &\times \forall t \in [t_{R_0}, t_{R_0} + \tau],\end{aligned}\tag{A6}$$

for $\tau > 2\pi/\Omega$ to ensure the space of static link strength configurations given R_0 is fully sampled during this process, and where h is an error function of choice. The warm-up duration of the prediction is chosen such that all the targeting

of the mean-field variables aforementioned happens before the end of the warm-up period, i.e., $t_{R_0} + \tau \leq \tau_{\text{warm}}\Delta t$. For most input systems, as we have used, a natural candidate is the root mean square error: $h^2 = \|\hat{\mathbf{W}}_{\text{out}} \mathbf{n}(\Phi, t) - \mathbf{u}_{\text{test}}(t)\|^2$, but if the topology of the state does not change considerably with the input system's bifurcation parameter or the system is contaminated with noise, an error function that can more finely resolve differences in states should be formulated and used. In summary,

- (1) Evolve the phase with Eq. (1) with $\mathbf{u}_{\text{test}}$ until $R = R_0$.
- (2) Replace the right side of Eq. (1) with Eq. (A5) for a duration τ such that $R(t) = R_0$, while $\langle \Phi \rangle$ makes at least one full rotation during τ . Identify the mean phase $\langle \Phi \rangle_0$ that embeds the test state $\mathbf{u}_{\text{test}}$ with Eq. (A6).
- (3) Set $g = 0$ upon this instance. Commence the closed-loop prediction as outlined in Appendix A 2 b.

We remind that the method outlined above all takes place during the warm-up phase—that is, the reservoir still runs in an open loop, where the output of the reservoir does not feed back as the input for the next time step. We also note that extrapolation curve during the open-loop stage is different from that during the closed-loop stage. In the former case, an input of one state is continuously sent into the network, which has the effect of an external force that tries to realign the network behavior to output the same state as the input. This prevents the network from extrapolating to a wider range of states in the neighborhood of the input, whereas in the closed-loop case, a small deviation of the output at one time step can amplify during the subsequent loops, thereby providing a large enough force for the network to hop into a basin of attraction of a state λ where $\lambda - \lambda_{\text{input}}$ is much larger.

-
- [1] K. Tyagi, C. Rane, and M. Manry, Chapter 1 - Supervised learning in *Artificial Intelligence and Machine Learning for EDGE Computing*, edited by R. Pandey, S. K. Khatri, N. Kumar Singh, and P. Verma (Academic Press, San Diego, CA, 2022), pp. 3–22.
 - [2] B. Siri, H. Berry, B. Cessac, B. Delord, and M. Quoy, A mathematical analysis of the effects of Hebbian learning rules on the dynamics and structure of discrete-time random recurrent neural networks, *Neural Comput.* **20**, 2937 (2008).
 - [3] R. M. French, Catastrophic forgetting in connectionist networks, *Trends Cognit. Sci.* **3**, 128 (1999).
 - [4] M. McCloskey and N. Cohen, Catastrophic interference in connectionist networks: The sequential learning problem, *Psychol. Learn. Motiv. Adv. Res. Theory* **24**, 109 (1989).
 - [5] Y. Andrade-Talavera, A. Fisahn, and A. Rodríguez-Moreno, Timing to be precise? An overview of spike timing-dependent plasticity, brain rhythmicity, and glial cells interplay within neuronal circuits, *Mol. Psychiatry* **28**, 2177 (2023).
 - [6] S. Valtcheva and L. Venance, Astrocytes gate Hebbian synaptic plasticity in the striatum, *Nat. Commun.* **7**, 13845 (2016).
 - [7] H. Ucar, S. Watanabe, J. Noguchi, Y. Morimoto, Y. Iino, S. Yagishita, N. Takahashi, and H. Kasai, Mechanical actions of dendritic-spine enlargement on presynaptic exocytosis, *Nature (London)* **600**, 686 (2021).
 - [8] H. Kasai, H. Ucar, Y. Morimoto, F. Eto, and H. Okazaki, Mechanical transmission at spine synapses: Short-term potentiation and working memory, *Curr. Opin. Neurobiol.* **80**, 102706 (2023).
 - [9] K. M. O'Neill, E. Saracino, B. Barile, N. J. Mennona, M. G. Mola, S. Pathak, T. Posati, R. Zamboni, G. P. Nicchia, V. Benfenati, and W. Losert, Decoding natural astrocyte rhythms: Dynamic actin waves result from environmental sensing by primary rodent astrocytes, *Adv. Biol.* **7**, e2200269 (2023).
 - [10] Q. Yang, Y. Miao, L. J. Campanello, M. J. Hourwitz, B. Abubaker-Sharif, A. L. Bull, P. N. Devreotes, J. T. Fourkas, and W. Losert, Cortical waves mediate the cellular response to electric fields, *eLife* **11**, e73198 (2022).
 - [11] Q. Yang, Y. Miao, P. Banerjee, M. J. Hourwitz, M. Hu, Q. Qing, P. A. Iglesias, J. T. Fourkas, W. Losert, and P. N. Devreotes, Nanotopography modulates intracellular excitable systems through cytoskeleton actuation, *Proc. Natl. Acad. Sci. USA* **120**, e2218906120 (2023).
 - [12] A. L. Bull, L. Campanello, M. J. Hourwitz, Q. Yang, M. Zhao, J. T. Fourkas, and W. Losert, Actin dynamics as a multiscale integrator of cellular guidance cues, *Front. Cell Dev. Biol.* **10**, 873567 (2022).
 - [13] M. Wu, X. Wu, and P. D. Camilli, Calcium oscillations-coupled conversion of actin travelling waves to standing oscillations, *Proc. Natl. Acad. Sci. USA* **110**, 1339 (2013).

- [14] H. Sakaguchi, S. Shinomoto, and Y. Kuramoto, Mutual entrainment in oscillator lattices with nonvariational type interaction, *Prog. Theor. Phys.* **79**, 1069 (1988).
- [15] Y. Kuramoto, *Chemical Oscillations, Waves, and Turbulence*, *Springer Series in Synergetics*, edited by H. Haken (Springer, Berlin, Heidelberg, 1984), Vol. 19.
- [16] S. Petkoski and A. Stefanovska, Kuramoto model with time-varying parameters, *Phys. Rev. E* **86**, 046212 (2012).
- [17] H. Jaeger and H. Haas, Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication, *Science* **304**, 78 (2004).
- [18] H. Jaeger, The “echo state” approach to analysing and training recurrent neural networks, GMD Report, No. 148, 2001.
- [19] K. D. Miller and F. Fumarola, Mathematical equivalence of two common forms of firing rate models of neural networks, *Neural Comput.* **24**, 25 (2012).
- [20] R. Thomas, Deterministic chaos seen in terms of feedback circuits: Analysis, synthesis, “labyrinth chaos”, *Int. J. Bifurcation Chaos* **09**, 1889 (1999).
- [21] F. Rihan, D. Abdel Rahman, S. Lakshmanan, and A. Alkhajeh, A time delay model of tumour-immune system interactions: Global dynamics, parameter estimation, sensitivity analysis, *Appl. Math. Comput.* **232**, 606 (2014).
- [22] Q. Lu, Q. Wang, and X. Shi, Effects of Time Delay on Synchronization and Firing Patterns in Coupled Neuronal Systems, in *Delay Differential Equations: Recent Advances and New Directions* (Springer US, Boston, MA, 2009), pp. 1–31.
- [23] M. C. Mackey and L. Glass, Oscillation and chaos in physiological control systems, *Science* **197**, 287 (1977).
- [24] E. N. Lorenz, Deterministic nonperiodic flow, *J. Atmos. Sci.* **20**, 130 (1963).
- [25] J. A. Yorke and E. D. Yorke, Metastable chaos: The transition to sustained chaotic behavior in the Lorenz model, *J. Stat. Phys.* **21**, 263 (1979).
- [26] W. Maass, T. Natschläger, and H. Markram, Real-time computing without stable states, *Neural Comput.* **14**, 2531 (2002).
- [27] Z. Lu, J. Pathak, B. Hunt, M. Girvan, R. Brockett, and E. Ott, Reservoir observers: Model-free inference of unmeasured variables in chaotic systems, *Chaos* **27**, 041102 (2017).
- [28] D. Sussillo and L. Abbott, Generating coherent patterns of activity from chaotic neural networks, *Neuron* **63**, 544 (2009).
- [29] Y.-C. Lai and N. Ye, Recent developments in chaotic time series analysis, *Int. J. Bifurcation Chaos* **13**, 1383 (2003),.
- [30] Z. Wu and N. E. Huang, Ensemble empirical mode decomposition: A noise-assisted data analysis method, *Adv. Adapt. Data Anal.* **01**, 1 (2009).
- [31] S. Schmidl, P. Wenig, and T. Papenbrock, Anomaly detection in time series: A comprehensive evaluation, *Proc. VLDB Endow.* **15**, 1779 (2022).
- [32] P. Bak, C. Tang, and K. Wiesenfeld, Self-organized criticality: An explanation of the $1/f$ noise, *Phys. Rev. Lett.* **59**, 381 (1987).
- [33] K. Mizuseki and G. Buzsáki, Preconfigured, skewed distribution of firing rates in the hippocampus and entorhinal cortex, *Cell Rep.* **4**, 1010 (2013).
- [34] G. Hahn, T. Petermann, M. N. Havenith, S. Yu, W. Singer, D. Plenz, and D. Nikolić, Neuronal avalanches in spontaneous activity *in vivo*, *J. Neurophysiol.* **104**, 3312 (2010).
- [35] J. M. Palva, A. Zhigalov, J. Hirvonen, O. Korhonen, K. Linkenkaer-Hansen, and S. Palva, Neuronal long-range temporal correlations and avalanche dynamics are correlated with behavioral scaling laws, *Proc. Natl. Acad. Sci. USA* **110**, 3585 (2013).
- [36] L.-W. Kong, H.-W. Fan, C. Grebogi, and Y.-C. Lai, Machine learning prediction of critical transition and system collapse, *Phys. Rev. Res.* **3**, 013090 (2021).
- [37] H. Kang, Rhythmic_sharing, GitHub, 2025, https://github.com/kangh1/rhythmic_sharing.