

date 15 September 2021

## Decoding Complex State Space Trajectories for Neural Computing

Fabio Schittler Neves<sup>1, a)</sup> and Marc Timme<sup>1, 2, 3, b)</sup>

<sup>1)</sup>*Chair for Network Dynamics, Center for Advancing Electronics Dresden (cfaed) and Institute for Theoretical Physics, TU Dresden, 01062 Dresden, Germany*

<sup>2)</sup>*Cluster of Excellence Physics of Life, TU Dresden, 01062 Dresden, Germany*

<sup>3)</sup>*Lakeside Labs, 9020 Klagenfurt am Wörthersee, Austria*

In biological neural circuits as well as in bio-inspired information processing systems, trajectories in high-dimensional state-space encode the solutions to computational tasks performed by complex dynamical systems. Due to the high state-space dimensionality and the number of possible encoding trajectories rapidly growing with input signal dimension, decoding these trajectories constitutes a major challenge on its own, in particular as exponentially growing (space or time) requirements for decoding would render the original computational paradigm inefficient. Here we suggest an approach to overcome this problem. We propose an efficient decoding scheme for trajectories emerging in spiking neural circuits that exhibits linear scaling with input signal dimensionality. We focus on the dynamics near a sequence of unstable saddle states that naturally emerge in a range of physical systems and provide a novel paradigm for analog computing, for instance in the form of heteroclinic computing. Identifying simple measures of coordinated activity (synchrony) that are commonly applicable to all trajectories representing the same percept, we design robust readouts whose sizes and time requirements increase only linearly with the system size. These results move the conceptual boundary so far hindering the implementation of heteroclinic computing in hardware and may also catalyze efficient decoding strategies in spiking neural networks in general.

**Keywords:** neural networks, nonlinear dynamics, network dynamics, synchrony

---

<sup>a)</sup>Electronic mail: fabio.neves@tu-dresden.de

<sup>b)</sup>Electronic mail: marc.timme@tu-dresden.de

## Decoding Complex State Space Trajectories

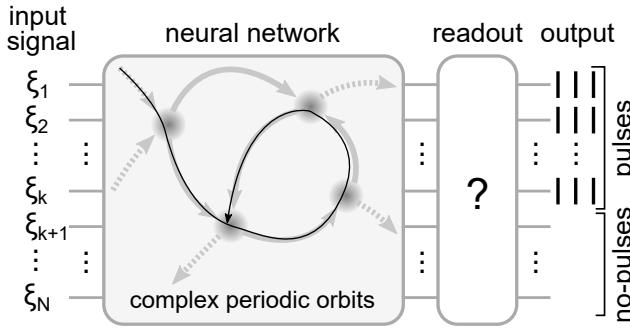
**Heteroclinic networks naturally emerge in systems of pulse-coupled neuronal units and provide the basis for bio-inspired analog computing:** switching dynamics in networks of unstable states. External signals induce complex trajectories resembling sequences of saddle states, thus yielding computation via a re-encoding of input signals. Remarkably, for a large class of systems exhibiting heteroclinic dynamics, the number of distinguishable inputs (percepts) the system is capable of encoding grows exponentially with the number of units composing the system. Nevertheless, inputs may be not uniquely represented by a single orbit, but, depending on initial conditions, as anyone out of a set of orbits that also grows exponentially for each percept. This potentially large number of orbits representing the same information could, in principle, undermine the system's main computational features, i.e. its large encoding capacity, by also requiring a readout with exponentially growing size or complexity. Here we show how to overcome this problem. We identify simple synchrony measures common to all orbits representing the same percept in heteroclinic networks and exploit the system symmetry to design robust readouts whose sizes increase linearly with the system size. Our results thus provide a vital step towards a viable implementation of a **Heteroclinic Computer.**

---

### I. INTRODUCTION

In neuronal or neuro-inspired computations, state-space trajectories of high-dimensional networked systems typically encode sensory signals or the results of computational tasks. Decoding such trajectories, however, constitutes a formidable task, specifically because of the high dimensionality and the often large number of possible trajectories to be decoded. One particularly transparent example of neural information processing is offered by Heteroclinic Computing<sup>1–8</sup>. Heteroclinic dynamics<sup>9–11</sup> naturally emerges in a range of physical systems and provides a novel paradigm for analog computing. In the basic paradigm, excitatory pulses and delayed connections among symmetrically interacting neural units robustly enable the emergence of closed networks of unstable saddle states (orbits). Computations are performed when external signals, serving as inputs, break the system symmetry<sup>1,7,12,13</sup> and induce specific periodic orbits resembling cyclic sequences of saddle states (subset of the original network), thus re-encoding the inputs<sup>7,8</sup>. By observing only the sequence of visited saddles, one can determine that a set of  $k$  neurons received

## Decoding Complex State Space Trajectories



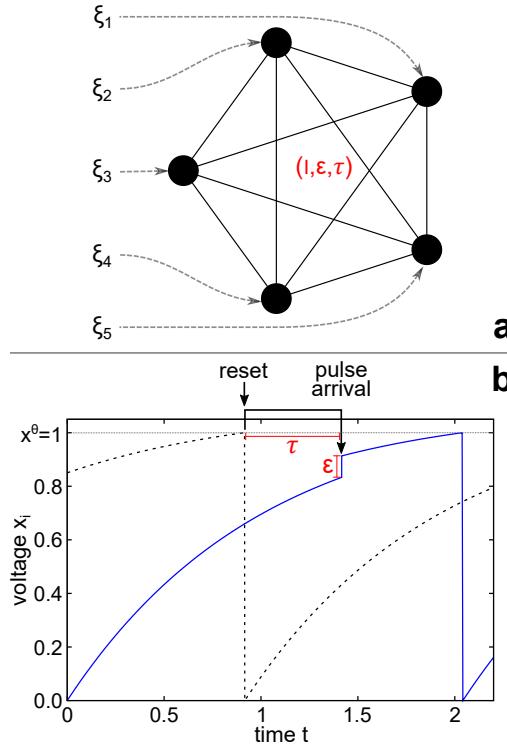
**FIG. 1. Idealized diagram of a Heteroclinic Computing System.** Input signals  $\xi_i$  induce complex periodic orbits (black line) resembling cyclic sequences of saddle states (circles). The resulting orbit is decoded by a readout into a simple spike train representation where the  $k$  output units corresponding the  $k$  strongest inputs exhibit spike activity while the  $N - k$  output units corresponding the  $N - k$  weakest inputs do not exhibit spikes.  $k$  is a system feature. In this example  $\{\xi_1, \dots, \xi_k\} > \{\xi_{k+1}, \dots, \xi_N\}$ .

a larger input than the other  $N - k$  neurons, thus computing a  $k$ -winners-take-all function. Figure 1 depicts an idealized diagram of a Heteroclinic Computing System.

Heteroclinic Computing presents two major aspects of interest: first, the number of internal representations grows exponential with the number of neurons in the system, in stark contrast to the linear capacity of stable attractor systems; the second aspect is their tractability and transparency of how computations are performed in such spiking neural circuits. Due to their symmetry properties, the characterization of a heteroclinic network, i.e. of all saddle-to-saddle connections, is reduced to the characterization of a single heteroclinic orbit, independent of the number of orbits, because all other orbits and their stability are determined by simply permuting the neurons in a single state in all possible manners<sup>13</sup>.

In this work we investigate and solve a potentially major problem for heteroclinic computers. Even though the number of discernible inputs (percepts) encoded in networks of states grows exponentially with the system size, other system properties may grow even faster and hinder a heteroclinic computer viability. Notably, the number of periodic orbits representing the same input also grows exponentially and could, in principle, represent a simple shift of computing load from an encoding to a decoding step, with no real gain. The fundamental problem is not related to limitations on traditional approaches to decoding, which may rely on spike times correlation<sup>14–16</sup> or learning algorithms<sup>17–21</sup>, but rather on the lack of understanding the dynamics guided by the symmetry of the heteroclinic network in state-space and its implications for decoding options. It has been unknown so far how to decode combinatorial numbers of multi-dimensional dynamic spiking patterns effectively and robustly. Below, we present a novel design for a readout layer

## Decoding Complex State Space Trajectories



**FIG. 2. Symmetrical neural network and pulse interaction.** (a) Example of a fully symmetrical network of  $N = 5$  identical neurons defined by three global parameters,  $I, \varepsilon$  and  $\tau$ . Circles represent neurons, solid lines represent internal connections and dashed arrows represent external connections,  $\xi$  represent the external signal. (b) The first pulse interaction between two neurons in the network is shown. After reaching the threshold  $x^\theta$ , neuron 1 sends a pulse that takes a time  $\tau$  to arrive. The pulse causes an instantaneous voltage jump equal to  $\varepsilon$  on all other neurons (only one is shown). Parameters,  $I_i = 1.1$ ,  $\tau = 0.5$  and  $\varepsilon = 0.08$ ,  $\gamma = 1$  and  $\xi_i(t) = 0$ , for all  $i$ .

whose size (number of units) scale only linearly with the size of the core heteroclinic processing network. To achieve this design, we first identify features common to all orbits representing the same percept (the average degree of synchrony between sets of neurons) and thereby we were able to avoid the multiplicity of decoding many orbits individually for each percept. Second, we demonstrate that the network symmetry may be exploited to tackle the exponential growth in the number of percepts themselves. Combining these results, we present readouts based on standard spiking layered networks whose size scale linearly with the number of neurons in the heteroclinic system and moreover requires only a few spikes per readout unit. To illustrate a specific example, we present a complete system consisting of computing network and readout, all composed of only one type of unit, the Leaky-Integrate-and-Fire neuron and tested its performance at different signal-to-noise ratios.

## II. HETEROCLINIC COMPUTING AND SCALING OF SYSTEM PROPERTIES

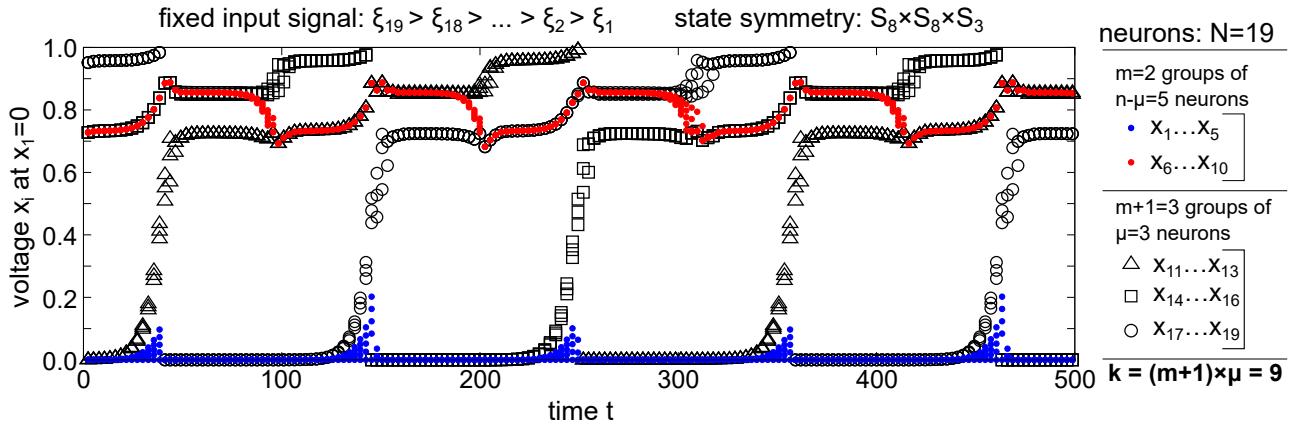
Even though the concepts studied here depends only on the existence of a robust heteroclinic network, for illustration we consider networks of oscillatory Leaky-Integrate-and-Fire neurons with all-to-all connections. Each neural unit  $i \in \{1, \dots, N\}$  exhibits a voltage-like state variable  $x_i$  satisfying the differential equations

$$\frac{dx_i(t)}{dt} = I - \gamma x_i(t) + \xi_i(t) + \sum_{j=1}^N \sum_{t_{j,\rho} \in P_j} \varepsilon \delta(t - t_{j,\rho} - \tau), \quad (1)$$

for  $x_i \in [0, x^\theta]$  where  $x^\theta$  is a spiking threshold. The parameter  $I$  represents a constant current,  $\gamma$  is a dissipation parameter, and  $\xi_i$  an external signal serving as input. Whenever a threshold is reached  $x_i(t^-) \geq x^\theta$ , the state variable is reset to  $x_i(t) = 0$  and a pulse (spike) is sent to all other neurons, mathematically reflected in the time  $t = t_{j,\rho}$  of the  $\rho$ th threshold crossing ( $\rho \in \mathbb{Z}$ ), by the neuron  $j$ . Without loss of generality we fix  $x^\theta = 1$ . The sum in (1) is the contribution of all spikes arriving from the other  $N - 1$  neurons  $j$  to  $i$  at time  $t$ , where  $P_j$  is the set of all times  $t_{j,\rho}$  of spikes sent by neuron  $j$ , where  $\varepsilon > 0$  is the coupling strength and  $\tau$  is the coupling delay. Figure 2 shown and example of time evolution and the delayed pulse-interaction for a simple network of  $N = 2$  neurons.

It has been shown that the combination of excitatory couplings and delays may promote the emergence of partial synchronization in networks of pulse-coupled oscillators<sup>5,22,23</sup>, also called polysynchrony<sup>24</sup>, where subsets of neurons synchronize in the absence of global synchrony. In this work we refer to these subsets simply as “clusters”. How the neurons are distributed among clusters is characterized by its permutation symmetry. For example, a state with two clusters, one with 3 synchronized neurons and one with 2 synchronized neurons, has a permutation symmetry  $S_3 \times S_2$ . That is, given five identical neurons, we can distribute them in  $\binom{5}{3} = 10$  saddle states with the same spike-time differences between clusters. The underlying mechanism behind such dynamics are simultaneous resets induced by incoming spikes. When one neuron is reset, it sends pulses to all other neurons, which arrive simultaneously after a fixed amount of time  $\tau$ . If the pulses are strong enough to cause the simultaneous threshold crossings for a set of neurons, they are simultaneously reset to zero, and thus synchronized. If the frequencies of the neurons are similar enough, these resets via supra-threshold incoming pulses may become cyclic, leading to a periodic orbit (in the absence of noise). If all resets during one orbit’s period are triggered by incoming

## Decoding Complex State Space Trajectories



**FIG. 3. Complex periodic orbit computing a  $(k = 9)$ -winners-take-all function over 19 inputs.** The voltage of all neurons are plotted each time  $x_1(t) = 0$ . The ten neurons receiving the smallest inputs form two groups of five neurons (in blue and red). The neurons receiving the strongest 9 inputs form three groups of three neurons each (in black). Non of those three groups of three neither the two groups of five ever synchronize among themselves, only across the group types. Notably, only the three groups of three neurons (in black) sequentially form the smaller cluster, i.e. form a cluster of size  $\mu$ . The states' permutation symmetry for this example is  $S_8 \times S_8 \times S_3$ . Thus,  $n = 8$ ,  $\mu = 3$  and  $m = 2$ .

pulses, the orbit is called stable, because any small variation in the voltages of synchronized neurons is reset within one cycle; if one or more resets are not triggered by incoming spikes, small desynchronizations will induce ever larger desynchronization after each cycle, because incoming pulses will have different contribution to each neuron due to the nonlinear (concave down) voltage curves<sup>13</sup>, and the orbit is called unstable. In particular, if an orbit exhibits stable and unstable manifolds, we called it a “saddle periodic orbit”.

Interestingly, because the neural network is fully symmetrical, if one saddle periodic orbit exists, any permutation of neuron will result in another saddle orbit with similar stability properties and the same symmetry. Furthermore, for a broad range of parameters, those symmetry-related saddles form closed networks of unstable states, i.e. their stable and unstable manifolds are interconnected<sup>22</sup>. Such switchings occur between saddles exhibiting the same symmetry, a feature we will make use for the decoding process.

Small but persistent input signals breaks the system symmetry and induce complex periodic orbits visiting the vicinity of a fixed number of saddles from the original network of states<sup>7,8</sup>. But, what is encoded in these resulting orbits? We analyze a frequently occurring type of saddle periodic orbits near networks of states with each saddle exhibiting  $m$  clusters of  $n$  neurons and a single smaller cluster of size  $\mu < n$ . Known examples include states, for instance, with permutation

## Decoding Complex State Space Trajectories

symmetries  $S_2 \times S_1$ ,  $S_8 \times S_8 \times S_3$ ,  $S_{21} \times S_{21} \times S_{21} \times S_{21} \times S_{16}$ , etc. Furthermore, their stability is such that a general perturbation to all neurons desynchronizes only one large cluster with size  $n$ , here denoted as “the unstable cluster”, all other (stable) clusters are resynchronized by incoming spikes shortly after a perturbation. Near each saddle the unstable cluster splits deterministically. There are only two possible ways that after breaking a single (unstable) cluster the state symmetry is recovered after a transient (symmetry preserving switch), either the  $\mu$  neurons receiving the strongest inputs in this cluster or the  $\mu$  neurons receiving the weakest inputs in this cluster must resynchronize to form a new small cluster. Meanwhile, the other  $n - \mu$  neurons must join the originally small cluster to form a new large cluster of size  $n$ . Whether the strongest or the weakest will form the small cluster is a system property, i.e. depends on neural model, the coupling strength and delays. Finally, an overall shift in relative voltages renders another cluster unstable, thus, marking the arrival at a new saddle with the same symmetry and stability as the original. As previously shown<sup>7</sup> and also illustrated in Figure 3, if the input signals persist for long enough, the dynamics converge to a periodic orbit resembling a cyclic sequence of visited saddles. During this orbit, from the  $N = mn + \mu$  neurons,  $m + 1$  groups of  $\mu$  neurons are recurrently compared to  $m$  groups of  $n - \mu$  neurons but not among themselves, thus altogether dynamically categorize inputs into two groups, one of size  $(m + 1) \times \mu$ , that visits the smaller cluster, and one of size  $m \times (n - \mu)$ , that does not, see Figure 3 for a specific example. Therefore, the resulting computation is a  $k$ -winners-take-all function where either

$$k = (m + 1) \times \mu, \quad (2)$$

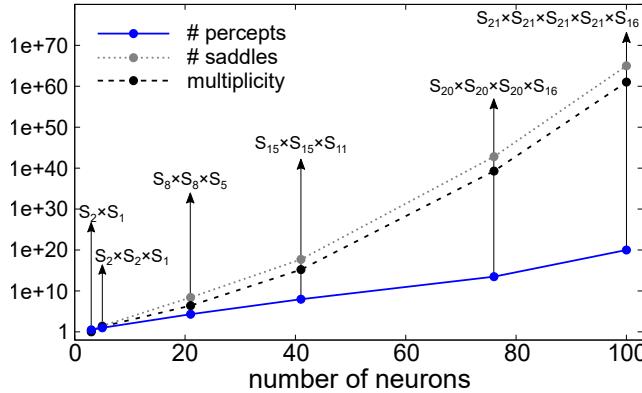
if the neurons receiving the strongest inputs sequentially form the smaller cluster or,

$$k = m \times (n - \mu), \quad (3)$$

if not.

With Equations 2 and 3 and the saddle permutation symmetry we can now estimate how the relevant system properties scale with network size. There are two quantities that are particularly interesting, namely the discernible inputs (percepts) and the number of orbits representing the same input (percept multiplicity). We remark that for different initial conditions (states), even though the computation is the same (same percept), may lead to different resulting orbits. The

## Decoding Complex State Space Trajectories



**FIG. 4. Number of orbits, percepts and orbits representing the same percepts (multiplicity) grow exponentially with the number of neurons.** Lines represent the exponential interpolation between the data point represented by circles. The first two points and the last point are real examples, while the others are predicted states. Arrows together with permutation symmetry labels indicate the symmetry of the state consider for each chosen  $N$ . An increase of two orders of magnitude in the number of neurons yields an increase of 20 orders of magnitude in the number of percepts. Nevertheless, the number of orbits representing the same percept grows even faster, with an increase of 60 orders of magnitude.

number of percepts for a network with  $N$  neurons computing a  $k$ -winners-take-all function is given by

$$P_{N,k} = \frac{N!}{k!(N-k)!}. \quad (4)$$

That is, the number of ways we can divide the input into one group with  $k$  inputs and a second group of  $N - k$  inputs. The number of orbits  $M_{m,n,\mu}$  representing the same input depends on the number of clusters  $m + 1$  and the size of each cluster,  $n$  and  $\mu$ . As discussed above, we have two distinct sets of neurons: one of size  $m(n - \mu)$  that is divided into  $m$  subsets of the same size  $(n - \mu)$  and another of size  $(m + 1)\mu$  divided into  $m + 1$  subsets of size  $\mu$ . Within each of the two main sets, the subsets order's exhibit a circular permutation, see Figure 3. Thus, the multiplicity of a percept is analogous to finding all the ways we can distribute  $m(n - \mu)$  distinguishable elements into  $m$  sets of size  $(n - \mu)$  over a circle combined with all the ways we can distribute  $(m + 1)\mu$  distinguishable elements in  $(m + 1)$  sets of size  $\mu$  over a circle. That is,

$$M_{m,n,\mu} = \frac{((m)(n - \mu))!}{((n - \mu)!)^m} \frac{1}{m} \times \frac{((m + 1)\mu)!}{(\mu!)^{m+1}} \frac{1}{(m + 1)}. \quad (5)$$

As shown in Figure 4, Equations 4 and 5 describe system properties that scale exponentially with the number of neurons composing the system. In particular, the number of percepts indeed scales exponentially with the number of neurons, providing a way to encode a large number of

## Decoding Complex State Space Trajectories

items in relatively small networks. Nevertheless, other properties such as the number of saddles and the number of orbits representing the same percept increase even faster. For example, for  $N = 100$ , a network of saddles with cluster permutation symmetry  $S_{21} \times S_{21} \times S_{21} \times S_{21} \times S_{16}$  can already encode  $P_{100,20} \approx 5.3 \times 10^{20}$  percepts<sup>7</sup>, but it also exhibits a network of about  $6.5 \times 10^{65}$  saddles and a multiplicity of  $M_{4,21,16} \approx 1.04 \times 10^{61}$  different orbits for each percept.

This large number of orbits representing the same input could in principle undermine the large encoding capacity of the system by also requiring a readout which size or complexity also grows exponentially.

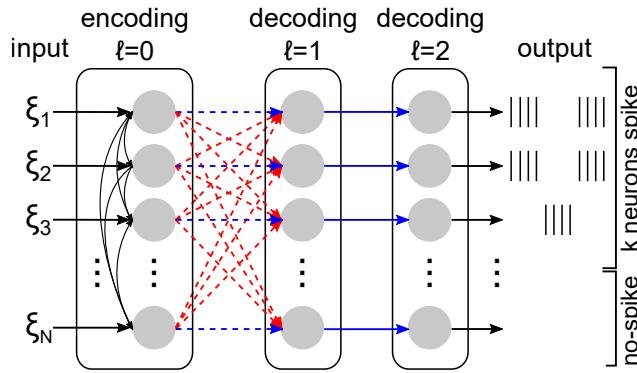
### III. DECODING COMPLEX PERIODIC ORBITS

We now show that the difficulties imposed by the multiplicity of a percept can be overcome by identifying a simple synchrony measure that is the same for all orbits representing the same percept and show that a readout which size increases linearly with the system size is possible by exploiting the system's complete permutation symmetry.

In networks of spiking neurons, each orbit representing/encoding a percept may be represented by  $N$  concurrent spike trains, one per neuron (periodic). Thus, from the readout perspective, there are two simple candidate quantities available for detecting an orbit, (a) differences in firing rates among the neurons and (b) the synchrony/timing between spikes elicited by different neurons. Our numerical experiments have shown that the increase in firing rates for the neurons receiving the strongest inputs is small ( $\approx 0.1\%$  in some cases), thus approach (a) may not provide a good indicator to which neurons received the strongest inputs. On the other hand, approach (b) is promising. The groups receiving the  $k$  strongest inputs and the  $N - k$  weakest inputs exhibit quite different synchrony properties during a complex periodic orbit. A characteristic of all networks-of-states discussed here is the symmetry preserving saddle-to-saddle switches. That is, after the transient time that follows a perturbation, the final periodic orbit is a simple permutation of the original, thus not only preserving its symmetry but also its stability properties. Therefore, analyzing one such switch should already yield the synchrony measure for detecting a percept (orbits resembling sequences of switches).

One simple characteristic of long periodic orbits representing percepts (as discussed above) is that only a set of  $k$  neurons divided in  $m + 1$  groups of size  $\mu$  may sequentially form a small cluster. Either the  $k$  neurons receiving the strongest inputs or the  $k$  weakest. Thus, detecting which neurons

## Decoding Complex State Space Trajectories



**FIG. 5. Connectivity between layers in a Heteroclinic Computing unit.** Signals  $\xi_i$  are first encoded by a fully connected neural network with only excitatory connections (solid arrows); each neuron from the decoder is connected to all neurons in the first encoder layer  $\ell = 1$ . Neurons with same index have excitatory connections and inhibitory connections (dashed arrows) with all other neurons (symmetrical). There is no in layer connections in the decoder. Connections between layer  $\ell = 1$  and  $\ell = 2$  are excitatory and exist only between nodes with the same index. The  $\ell = 2$  function is to detect neurons with the  $k$  highest input spike frequencies from  $\ell = 1$ . The overall output are spike activity from the neurons with same index as the  $k$  stronger inputs.

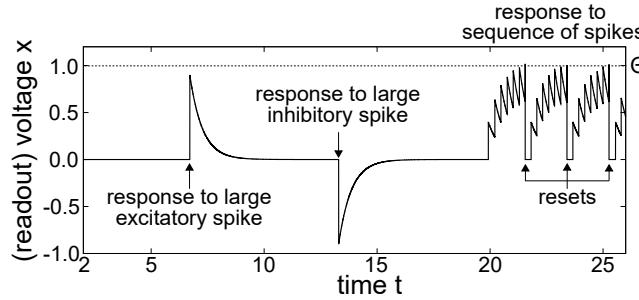
cyclically assume a position in the smaller cluster, exactly decode the result of a computation.

This approach seems reasonable because not only the smaller cluster is only composed of the neurons of one type ( $k$  strongest or  $k$  weakest), but it is also synchronized by incoming spikes, thus small variations in parameters or noise should not be an obstacle for detection. This stability by incoming pulses also produces a clear pulse pattern of  $\mu$  simultaneous spikes even during the saddle-to-saddle transient times, because the small cluster must first merge with other neurons to form a large cluster before becoming an unstable cluster.

We argued that identifying the neurons recurrently composing the smaller cluster during a complex periodic orbit is a good indicator of which are the weakest, and which are the strongest input signals applied to a heteroclinic system. But what scalable readout (system) can detect this marker? We propose a two layered network (Figure 5). The first layer computes whether a neuron in the encoding layer spikes with at most other  $\mu - 1$  neurons. This includes all neurons in the smaller cluster and the neurons that are desynchronized during transient. The second layer computes exactly the neurons that frequently spike with only  $\mu - 1$  other neurons, thus decoding the percept.

The connectivity between the encoding network and the first layer of the decoder ( $\ell = 1$ ) is as follows: it has a single excitatory connection between neurons with the same index which are strong enough to generate a spike from the neurons in the readout. That is, they always

## Decoding Complex State Space Trajectories



**FIG. 6. Readout neuron's response to incoming pulses.** Excitatory incoming pulses produce a positive spike in voltage; Inhibitory incoming pulses produce a negative spike in voltage. After a spike in voltage, its value exponentially relax to its resting potential at  $x_i = 0$ . Low amplitude incoming spike trains can also lead to a reset, if their frequency is high enough.

generate spikes in the absence of concurrent inhibitory incoming pulses. It also exhibits inhibitory connections between neurons with different indices, such that if more than  $\mu - 1$  inhibitory spikes arrive at a neuron in the readout almost or at the same time as an excitatory pulse, a new spike is prevented. Moreover, inhibitory connections exhibit slightly shorter delays than excitatory ones, such that the inhibitory pulses from synchronized or almost synchronized neurons does not arrive shortly after an excitatory pulse. The connectivity between the first and the second layer ( $\ell = 2$ ) is very simple. It has a single excitatory connection between neurons with the same index, which can only induce a spike in layer  $\ell = 2$  if pulses arrive with at least a minimum frequency. It takes advantage of the stability of the smaller cluster, that promotes reliably  $\mu$  concurrent pulses in the vicinity of a saddle and during transient times, because it stay stable during the whole process. It losses stability only after forming a larger cluster.

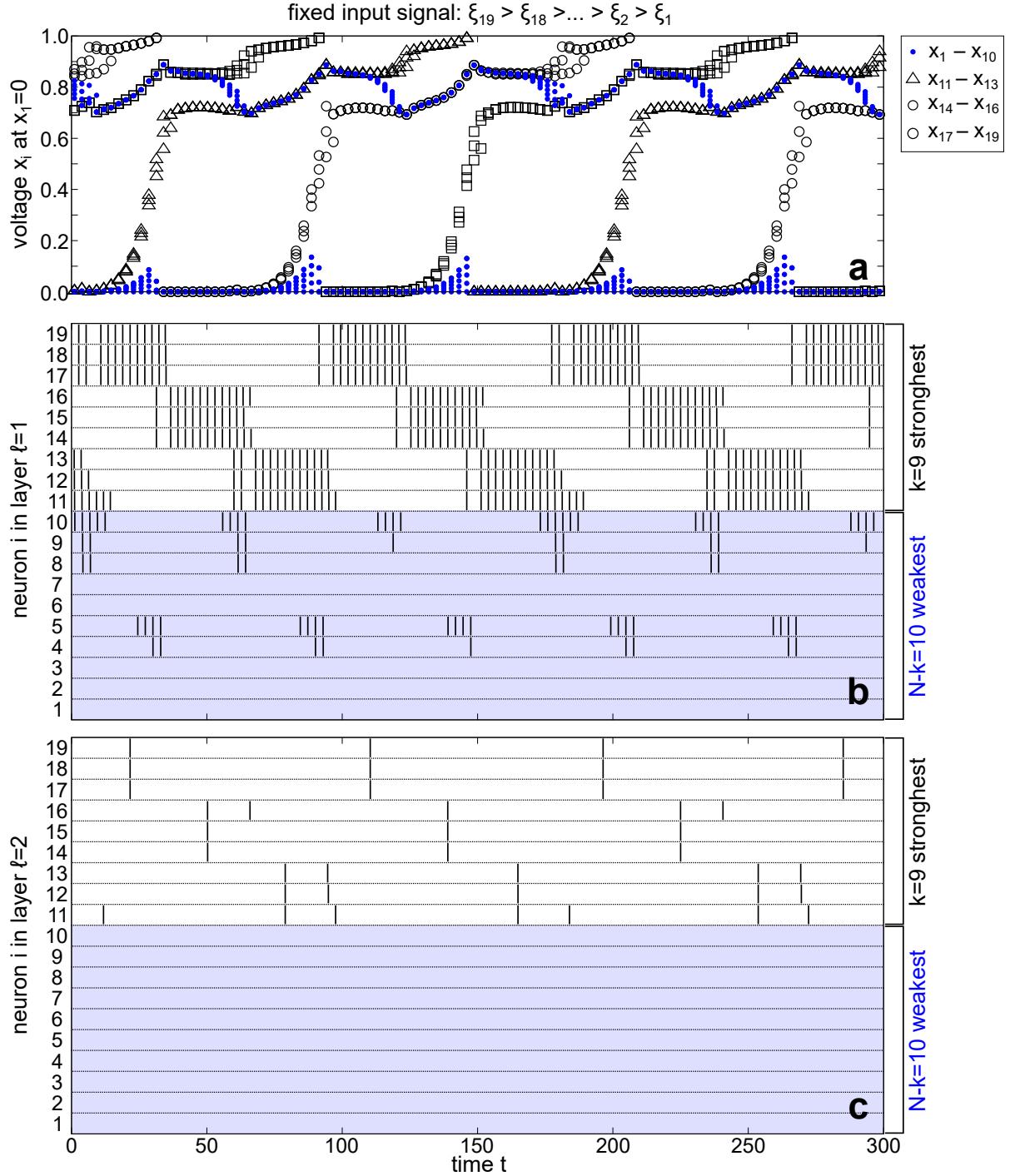
Specifically, our readout's nodes consist of Leaky-Integrate-and-Fire neurons which are non-oscillatory ( $I = 0$ ),

$$\frac{dx_i^\ell(t)}{dt} = -\gamma x_i^\ell(t) + \eta_i^\ell(t) + \sum_{j \in N} \sum_{t_{j,\rho} \in P_j^\ell} w_{ij}^\ell \delta(t - t_{j,\rho} - \tau_j), \quad (6)$$

where  $\gamma$  is a leak constant,  $\ell$  indicates the network layer ( $\ell = 0$  refers the heteroclinic system,  $\ell > 0$  the readout),  $w_{ij}^\ell$  are the connection weights between neurons  $i$  from layer  $\ell$  and  $j$  from the previous layer and  $\eta_i(t)$  is a white Gaussian noise signal. The positive finite leak grants and limits the memory of the neuron to a short window. Without it, the system would either integrate all pulses (very small  $\gamma$ ), thus not decode, or do not detect synchrony due to infinitesimal or very small voltage differences induced by noise (very large  $\gamma$ ), see Figure 6 for a depiction of voltage's

## Decoding Complex State Space Trajectories

This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
 PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429



**FIG. 7. Decoding a 9-winners-take-all computation in a  $N = 19$  neural network.** (a) The voltage of all neurons are plotted each time  $V_1(t) = 0$ . The ten neurons receiving the smallest inputs form two groups of fine neurons (in blue). The neurons receiving the strongest 9 inputs form three groups of three neurons each (in black). (b-c) Each dash represents a time at which a neuron spiked. (b) In layer  $\ell = 1$ , the neurons corresponding to the strongest inputs (white background) spike more often than the others (blue background). (c) In layer  $\ell = 2$ , only the neurons corresponding to the  $k$  strongest inputs (white background) spike. Parameters  $w_{ij}^1 = 1.35$  for  $i = j$ ,  $w_{ij}^1 = -0.1$  for  $i \neq j$  and  $w_{ii}^2 = 0.1$

## Decoding Complex State Space Trajectories

response to different inputs.

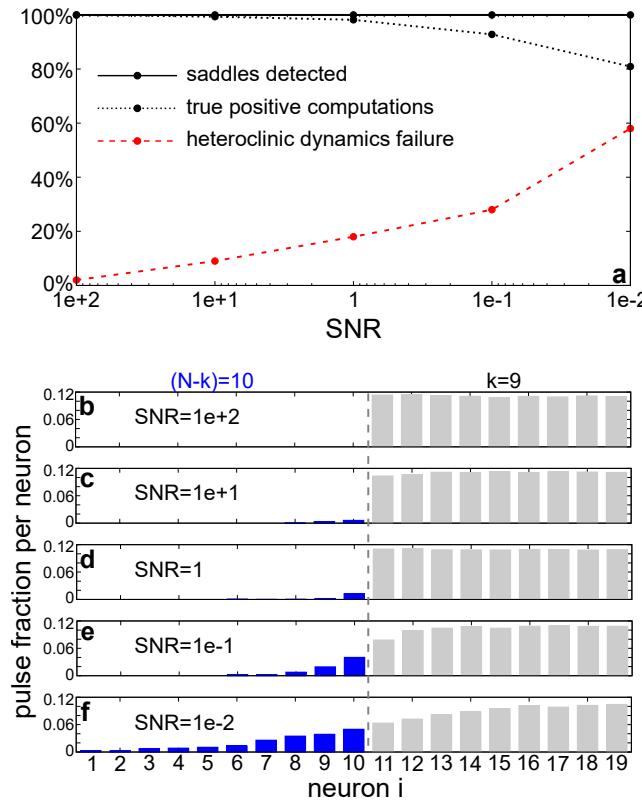
For illustration, we present a readout for the  $N = 19$  system introduced above. This is a good example because it already exhibits a large percept multiplicity while it is still possible to depict it in a concise way. In this system three groups of three neurons switch between two groups of five neurons, thus  $\mu = 3$  and  $n = 8$ . Figure 7 shows that the first layer detects whether a neuron spikes (almost) concurrently with less than  $\mu = 3$  other neurons. As expected, output neurons with the same index as the ones in the smaller cluster spikes during transient and when close to a saddle orbit, thus, more often than other neurons. The second layer takes advantage of this feature and compute the inputs with higher frequency, thereby, decoding the original result of the computation, i.e. ( $k = 9$ )-winners-take-all.

As a proposed computing paradigm, it is important to discuss how does Heteroclinic Computing performs in non-ideal condition. Previous works have shown how noise affects switching times<sup>1,25</sup> and switching probabilities<sup>26,27</sup> in heteroclinic networks. But, how does noise affect our overall encoding-decoding process? As the computing system is only composed by neurons of the same type, that is, separate just in functionality, we will consider the same type of noise acting over all network nodes. We, thus, also added the noise term  $\eta_i(t)$  to all nodes in the encoding layer. Noise can affect the computation in different ways. In what concerns the heteroclinic dynamics, noise can either induce undesired switches, thus transiently computing the wrong rank order, or completely break the dynamics causing a total system failure. From the readout perspective, noise can either promote extra spikes by increasing the neural voltage at the receiving pulses times in  $\ell = 1$  or preventing them by reducing their voltage at those times.  $\ell = 2$  is mostly unaffected directly by noise because it only integrates the incoming excitatory pulse-trains.

To evaluate the heteroclinic encoder-decoder system under noise, we compute the fraction of false positives (always accompanied by a false negative) and the fraction of saddles detected as functions of time for different signal-to-noise ratios (SNR), averaged over 50 independent trials per SNR. To avoid irrelevant spike timing variations across trials due to different initial conditions and noise, we measure time in number of output pulses ( $\ell = 2$ ), thus aligning all trials for a relevant average measure. Input signals are fixed and set to

$$\xi_i = i \times \Delta\xi, \quad i \in \{1, \dots, N\}, \quad (7)$$

## Decoding Complex State Space Trajectories



**FIG. 8. Decoder and system performance at different SNR for a network with  $N = 19$  neurons and  $k = 9$ .** (a) The fractions of detected saddles (solid line), the fraction of true positive detections (dotted line) and the fraction of system failures (dashed line) as functions of the SNR. Average over 50 trials per SNR and measured over 250 pulses from the output layer  $\ell = 2$ . For the consider SNR interval all visited saddles have been detected; the quality of detection increases with SNR; (b-f) the average fraction of pulses elicited by each neuron over all pulses for different SNRs. In blue, false positive pulses; in light-gray, true positive pulses.

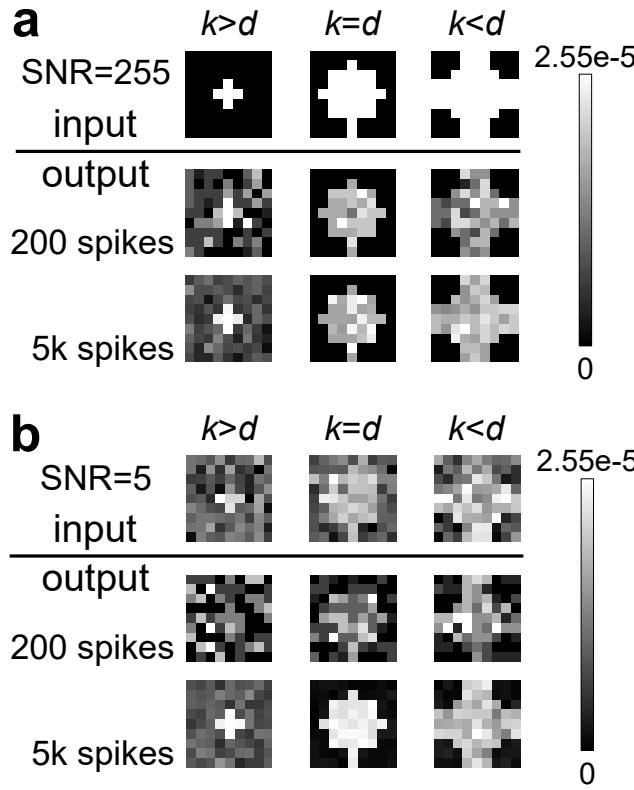
where  $\Delta\xi$  is a fixed constant completely determining the input signal. We define a SNR ration as

$$\text{SNR} = \frac{\Delta\xi^2}{\sigma^2}, \quad (8)$$

where  $\sigma$  is the standard deviation of the white Gaussian noise signal  $\eta(t)$ . Furthermore, we define as a true positive any pulse from the layer  $\ell = 2$  indicating a correct result and as a false positive any pulse from  $\ell = 2$  indicating the wrong result. Their fraction is simply given by dividing their count by the total number of  $\ell = 2$  pulses.

As an example, we apply the approach described above to a network of  $N = 19$  neurons, details in Figure 8. Our results shows first and foremost that the readout is robust, because it detected the vicinity of all approached saddle (correctly computed or not) for the broad range of SNRs

## Decoding Complex State Space Trajectories



**FIG. 9. Image encoding via output spike-rates** Inputs and outputs are represented as  $9 \times 9$  pixels images. Input image's pixel show the combined noise an external signals for a neuron, averaged of approximately a time period  $[0, 3.26)$  between two subsequent spikes of one neuron. Input are depicted in a grey gradient between zero (black) and  $2.55 \times 10^{-5}$  (white). A network of 90 neurons (81 depicted) computing a ( $k = 30$ )-wta function encodes the input.  $d$  is the number of signals' non-zero inputs,  $d = 5$  (left),  $d = 30$  (center)  $d = 49$  (right). Output images show the average spike-rate of each neurons (one per pixel), normalized to the interval  $[0, 2.55 \times 10^{-5}]$  after 200 and five thousand output pulses. **(a-b)** For  $k > d$ , the input is overrepresented, i.e. more neurons are activated than necessary to represent it. False positives occur repeatedly and generate a grey background, representing neurons that receive no inputs and spike stochastically. For  $k = d$ , only a subset is visited with high probability, so there is a fast convergence to an (almost) homogeneous spike rate. For  $k < d$ , the input is underrepresented by a single periodic orbit but, for long enough simulations, all neurons receiving input signals eventually generate spikes, because noise induces changes to the set of winners. **(a)** the noise contribution ( $\sigma = 10^{-7}$ ) to the total input is not visible as it is smaller than the color gradient precision, nevertheless it supports switching. **(b)** The noise contribution is large ( $\sigma = 5.1 \times 10^{-6}$ ), but the dynamics still computes properly after five thousand output pulses. Parameters  $\tau = 1.61$ ,  $\varepsilon = 10^{-3}$ ,  $w_{ij}^1 = 1.32$  for  $i = j$ ,  $w_{ij}^1 = -0.015$  for  $i \neq j$  and  $w_{ii}^2 = 0.12$ .

considered. The quality of the computation, measured as the fraction of true positives, decrease monotonically but not linearly with decreasing SNRs. Furthermore, the number of system failures within the first 250 spikes form layer  $\ell = 2$ , that is, in which the dynamics leave the heteroclinic network, increases inversely proportional to the SNR.

As a second example of computation, we apply a heteroclinic computing unit to represent a set

## Decoding Complex State Space Trajectories

of images serving as inputs, see Figures 9 and 10. Specifically, a network composed of 90 neurons computing a ( $k = 30$ )-winners-take-all function is used to represent images with  $9 \times 9$  pixels. From the 90 neurons, 81 receive direct connections from the image, one per pixel. The other nine receive no input. A black pixel represents an input equal zero; a white pixel represents a signal with amplitude equal  $2.55 \times 10^{-5}$ . Independent Gaussian noise of variance  $\sigma = 10^{-7}$  is added to all nodes, promoting switching when a set of inputs is in part identical. The output is also expressed in an image of  $9 \times 9$  pixels. The color of each output pixel reflects its firing rate in relation to the other neurons' firing rates. The largest firing rate is depicted as white. If a neuron does not spike, the firing rate is zero, represented by the color black. Intermediary values are scaled accordingly and depicted in a linear grey scale. For generic inputs, three cases may emerge (see Figures 9): first, the number of non-zero inputs  $d$  is smaller than the  $k$  being computed. In this case,  $d$  neurons fire the most and repeatedly, while the other neurons fire with a lower probability and stochastically. For  $k = d$  the input is precisely represented, with neurons receiving no input not producing spikes (for small enough noise). For  $k < d$ , the image is underrepresented at each cycle, but the whole image can be represented if the computation is long enough. Again, neurons receiving no input do not generate any spikes. To show how this system responds to a more natural set of inputs, we also compute its response to low resolution representations of the numerals  $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$  and depict the results in Figure 10. The results are similar to the ones depicted in Figure 9. All numeral representations are overrepresented by the network, because  $d < k$  for all numerals. Notice that, differently from our previous examples, the numerical experiments shown in Figures 9 and 10 exhibit cases in which there are multiple neurons receiving the same input, so there is no ranking order difference between many of them. Momentary differences are induced by noise, thereby the outputs are stochastic and their firing rates reflect their probability of being in the set of “winners”.

We argue that our results are general because the underlying features exploited by our readout is independent of the precise neural model or system size: first, simultaneous pulses generated by the heteroclinic system are robust and precise, because suprathreshold inputs erase small enough voltage differences; and second, the readout does not operate on unstable states as the heteroclinic system, but is rather driven from its single and stable ground state at  $V_i = 0$  for all  $i$  by recurrent incoming spikes.

## Decoding Complex State Space Trajectories



FIG. 10. **Numerals' representation via output spike-rates.** Inputs and outputs are represented as  $9 \times 9$  pixels images. Input image's pixel show the combined noise an external signals for a neuron, averaged of approximately a time period  $[0, 3.26]$  between two subsequent spikes of one neuron. Input are depicted in a grey gradient between zero (black) and  $2.55 \times 10^{-5}$  (white). A network of 90 neurons (81 depicted) computing a ( $k = 30$ )-wta function encodes the input.  $d$  is the number of non-zero inputs. Output images represent the average spike-rate of the neurons (one per pixel), normalized to the interval  $[0, 2.55 \times 10^{-5}]$  of shades of grey after 200 and after five thousand output pulses. Each numeral exhibits a different  $d$  value, all overrepresented ( $d < k$ ). Parameters  $\tau = 1.61$ ,  $\varepsilon = 10^{-3}$ ,  $\sigma = 10^{-7}$  (noise variance),  $w_{ij}^1 = 1.32$  for  $i = j$ ,  $w_{ij}^1 = -0.015$  for  $i \neq j$  and  $w_{ii}^2 = 0.12$ .

## IV. DISCUSSION

In this work we identified and solved a potential major problem on computing with state-space trajectories in bio-inspired dynamical systems. After a brief review about heteroclinic computing, we have shown that not only the number of percepts, but also the number of periodic orbits representing (encoding) the same percept grows exponentially with the number ( $N$ ) of neurons in the system. The exponential variety brings the benefit of a large number of outputs that are computable but also poses the question whether a readout is required that is exponentially large or needs exponentially long time for decoding the state space trajectory that reflects the result of the computation. We have shown that this is not the case by first defining a readout strategy that exploits features that are common to exactly orbits that represent the same input type. A simple synchrony measure, i.e. concurrent pulses from an (always present) small cluster serves to identify the percepts independent of the actual orbit. Second, we exploit the system symmetry to mediate the exponentially growing number of percepts with the system size, yielding readouts that grows linearly with the system size. To present a concrete example, we implemented a heteroclinic computing unit, including a readout, using only leaky Integrate-and-Fire neurons. Interestingly, because it is based mostly on symmetry, the readout approach presented here is independent of network size or saddle permutation symmetry, as long as only a single smaller cluster is present

## Decoding Complex State Space Trajectories

(the predominant case). Furthermore, the approach is robust to noise because, if the dynamics is not disrupted, noise is first filtered in the heteroclinic system, due to simultaneous supra-threshold events, and second the readout performs a short average over its inputs, with memory that scale inversely proportional to their voltage leak constant (exponential decay).

We emphasize that we were able to design the general readout presented above only by following a conceptual theoretical approach, exploiting symmetry features that are not constrained to any specific system. The readout is thus applicable across system settings (units, parameters and all sizes of systems that exhibit the type of heteroclinic dynamics we discuss) and is moreover independent of any details of synchronization patterns generated by the (near-) heteroclinic dynamics. For instance, to obtain similar results using any purely computational approach, through machine learning<sup>17–21</sup>, would be feasible only for any fixed system at given parameters, system size and heteroclinic dynamics. In addition, the system size would be strongly limited due to the combinatorial growth of the number of encoded features. To generalize any such result to arbitrary settings is viable only with some additional information about the mechanisms underlying the dynamics, and thus would require further theoretical insight of the type we provided in this article. Moreover, machine learning approaches uninformed about mechanisms underlying the dynamics would likely be hugely computationally demanding as all readout parameters (a priori an exponential number of them) would need to be determined. Our results are largely independent of the neural model used in the readout. In fact, any readout subsystem capable of discerning between two sets of almost synchronized incoming pulses, one smaller than the other, should be able to decode the signal. Our specific readout architecture simply stresses that symmetry informs us on how to build readouts of sub-exponential (and even linear) size, independent of the grouping of the neurons in the (near-) heteroclinic dynamics and the resulting symmetry of saddle states. So, the readout design presented is unique in its generality. We still mention that we do not claim any optimality of the readout, in terms of, e.g. the number of spikes, units, or processing steps required. The readout presented thus just constitute the first (to date) example to function, with many questions about optimality to be addressed.

The proposed framework for readouts has some interesting implications. By providing a spatial correlation between inputs and outputs, our readout brings the applicability of the Heteroclinic Computing paradigm on par with other approaches of computing partial rank order with spiking systems<sup>28–32</sup>. Furthermore, building readouts using the same substrate as the main system, neuron-like units, may help to model and explain how complex trajectories are decoded in neural system

## Decoding Complex State Space Trajectories

models based in heteroclinic dynamics, for example models of the olfactory system<sup>3,6,33</sup>. Overall, we present the first general and scalable readout approach to a heteroclinic computing systems, making them theoretically feasible and bring them one step closer to an actual implementation, also providing a clear way to couple such system with other systems and, possibly, machine learning approaches.

## ACKNOWLEDGEMENTS

Partially supported by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under project number 419424741 and under Germany's Excellence Strategy – EXC-2068 – 390729961 – Cluster of Excellence Physics of Life at TU Dresden, and the Saxonian State Ministry for Science, Culture and Tourism under grant number 100400118.

## DATA AVAILABILITY

The data that supports the findings of this study are available within the article.

## REFERENCES

- <sup>1</sup>P Ashwin and J Borresen. Discrete computation using a perturbed heteroclinic network. *Phys. Lett. A*, 374(4–6):208–214, 2005.
- <sup>2</sup>U Ernst, K Pawelzik, and T Geisel. Synchronization induced by temporal delays in pulse-coupled oscillators. *Phys. Rev. Lett.*, 74:1570, 1995.
- <sup>3</sup>M Rabinovich, A Volkovskii, P Lecanda, R Huerta, H D I Abarbanel, and G Laurent. Dynamical encoding by networks of competing neuron groups: winnerless competition. *Phys. Rev. Lett.*, 87(6):068102, 2001.
- <sup>4</sup>M Timme, F Wolf, and T Geisel. Prevalence of unstable attractors in networks of pulse-coupled oscillators. *Phys. Rev. Lett.*, 89:154105, 2002.
- <sup>5</sup>U Ernst and K Pawelzik. Delay-induced multistable synchronization of biological oscillators. *Phys. Rev. E*, 57:2150, 1998.
- <sup>6</sup>M Rabinovich, R Huerta, and G Laurent. Transient dynamics for neural processing. *Science*, 321:48–50, 2008.

## Decoding Complex State Space Trajectories

This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429

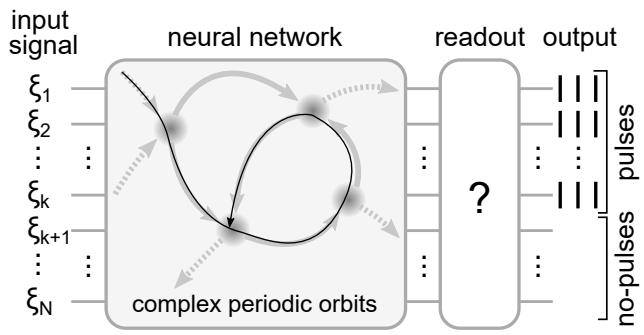
- <sup>7</sup>F S Neves and M Timme. Computation by switching in complex networks of states. *Phys. Rev. Lett.*, 109(1):018701, 2012.
- <sup>8</sup>J Wordsworth and P Ashwin. Spatiotemporal coding of inputs for a system of globally coupled phase oscillators. *Phys. Rev. E*, 78:066203, 2008.
- <sup>9</sup>M Krupa. Robust heteroclinic cycles. *J. Nonlinear Sci.*, 7:129–176, 1997.
- <sup>10</sup>J Guckenheimer and P Holmes. Structurally stable heteroclinic cycles. *Proc. Camb. Phil. Soc.*, 103:189–192, 1988.
- <sup>11</sup>P. Ashwin and J. Borresen. Dynamics on networks of cluster states for globally coupled phase oscillators. *SIAM J. Appl. Dyn. Syst.*, 6:728–758, 2007.
- <sup>12</sup>P Ashwin and M Timme. When instability makes sense. *Nature*, 436:36–37, 2005.
- <sup>13</sup>F S Neves and M Timme. Controlled perturbation-induced switching in pulse-coupled oscillator networks. *J. Phys. A*, 42(34):345103, 2009.
- <sup>14</sup>S Panzeri, S R Schultz, A Treves, and E T Rolls. Correlations and the encoding of information in the nervous system. *Proc. R. Soc. Lond. B Biol. Sci.*, 266:1001–1012, 1999.
- <sup>15</sup>S Nirenberg and P E Latham. Decoding neuronal spike trains: How important are correlations? *Proceedings of the National Academy of Sciences*, 100(12):7348–7353, 2003.
- <sup>16</sup>A Dettner, S Münzberg, and T Tchumatchenko. Temporal pairwise spike correlations fully capture single-neuron information. *Nat. Commun.*, 7:13805, 2016.
- <sup>17</sup>R Gütig and H Sompolinsky. The tempotron: A neuron that learns spike timing-based decision. *Nature Neuroscience*, 9:420 – 428, 2006.
- <sup>18</sup>C M Bishop. *Pattern recognition and machine learning*. Springer, 2006.
- <sup>19</sup>R Urbanczik and S Walter. A gradient learning rule for the tempotron. *Neural Computation*, 21:340 – 352, 2009.
- <sup>20</sup>J Schmidhuber. Deep learning in neural networks: An overview. *Neural Networks*, 61:85 – 117, 2015.
- <sup>21</sup>J I Glaser, A S Benjamin, R H Chowdhury, M G Perich, L E Miller, and K P Kording. Machine learning for neural decoding. *eNeuro*, 31, 2020.
- <sup>22</sup>M Timme, F Wolf, and T Geisel. Unstable attractors induce perpetual synchronization and desynchronization. *Chaos*, 13(1):377–387, 2003.
- <sup>23</sup>P Ashwin and M Timme. Unstable attractors: existence and robustness in networks of oscillators with delayed pulse coupling. *Nonlinearity*, 18:2035–2060, 2005.

## Decoding Complex State Space Trajectories

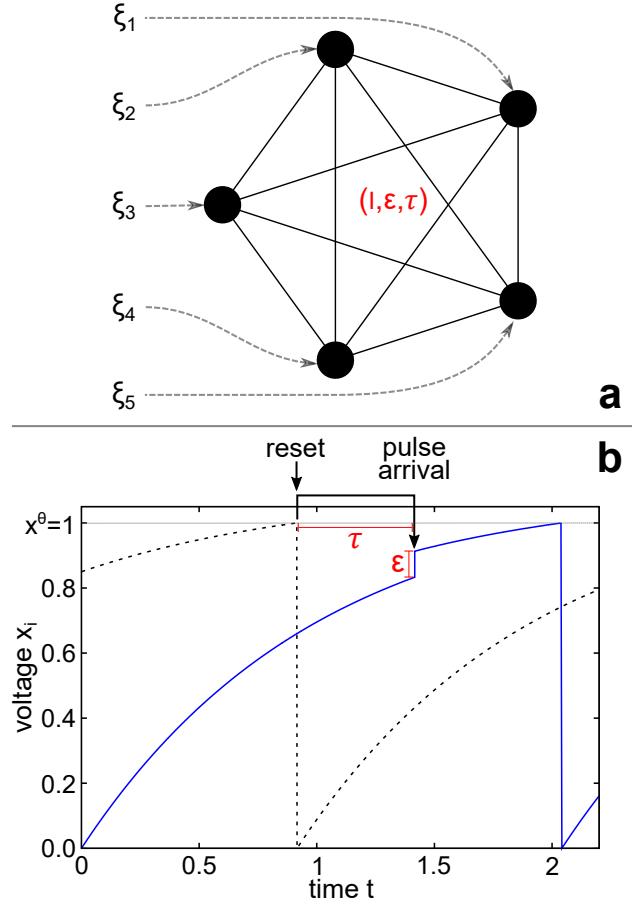
This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429

- <sup>24</sup>E M Izhikevich. Polychronization: Computation with spikes. *Neural Comput.*, 18(2):245–282, 2006.
- <sup>25</sup>F S Neves, M Voit, and M Timme. Noise-constrained switching times for heteroclinic computing. *Chaos*, 27(3):033107, 2017.
- <sup>26</sup>D Armbruster, E Stone, and V Kirk. Noisy heteroclinic networks. *Chaos*, 13:71–79, 2003.
- <sup>27</sup>P Ashwin and C Postlethwaite. Quantifying noisy attractors: From heteroclinic to excitable networks. *SIADS*, 15(4):1989–2016, 2016.
- <sup>28</sup>Y Sandamirskaya. Dynamic neural fields as a step toward cognitive neuromorphic architectures. *Front. Neurosci.*, 22, 2014.
- <sup>29</sup>C Strub, G Schöner, F Wörgötter, and Y Sandamirskaya. Dynamic neural fields with intrinsic plasticity. *Front. Comp. Neurosci.*, 31, 2017.
- <sup>30</sup>Y Chen. Mechanisms of winner-take-all and group selection in neuronal spiking networks. *Front. in Comput. Neurosci.*, 11:20, 2017.
- <sup>31</sup>J J Wang, Q Yu, S G Hu, Y Liu, R Guo, T P Chen, Y Yin, and Y Liu. Winner-takes-all mechanism realized by memristive neural network. *Appl. Phys. Lett.*, 115(24):243701, 2019.
- <sup>32</sup>F S Neves and M Timme. Reconfigurable computation in spiking neural networks. *IEEE Access*, 8:179648–179655, 2020.
- <sup>33</sup>G Laurent, M Stopfer, R W Friedrich, M I Rabinovich, A Volkovskii, and H D I Abarbanel. Odor encoding as an active, dynamical process: experiments, computation, and theory. *Annual Review of Neuroscience*, 24(1):263–297, 2001.

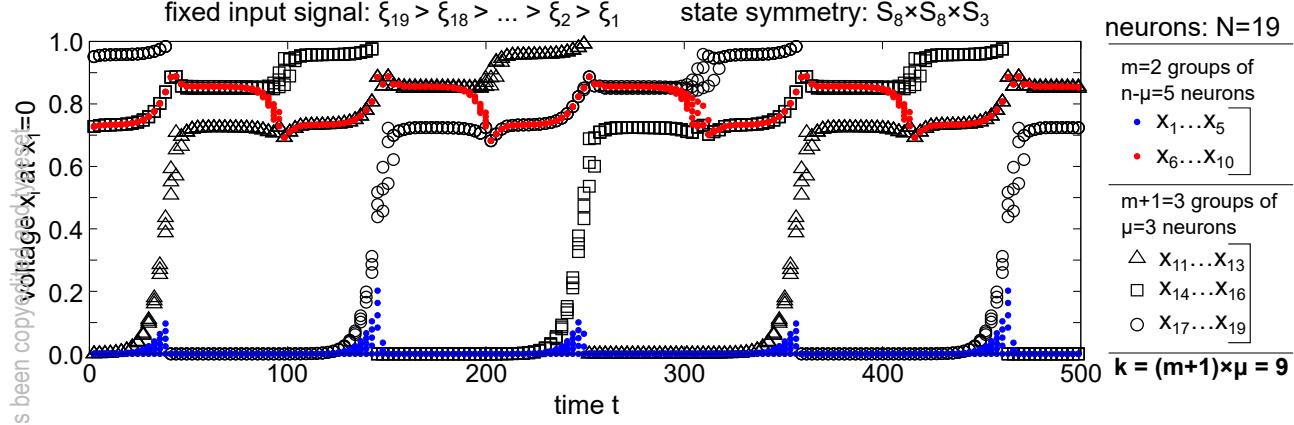
This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
PLEASE CITE THIS ARTICLE AS DOI: [10.1063/5.0053429](https://doi.org/10.1063/5.0053429)



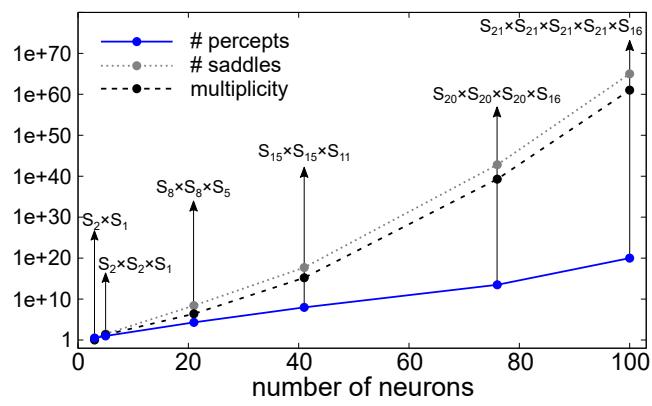
This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429



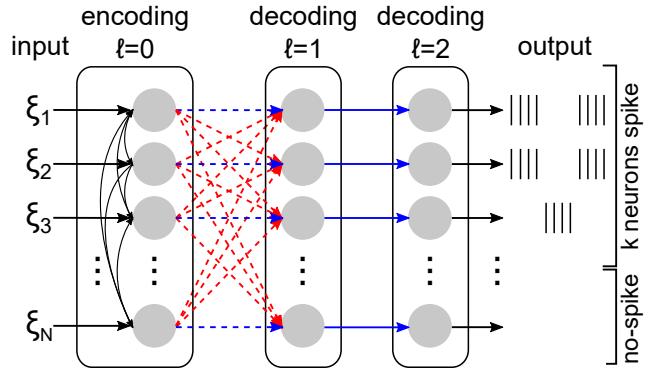
This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copy/revised.  
PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429



This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
 PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429

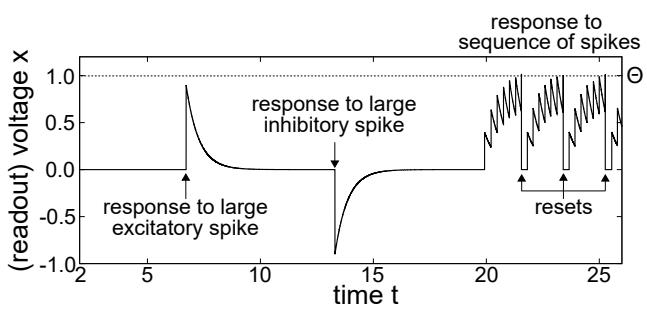


This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429

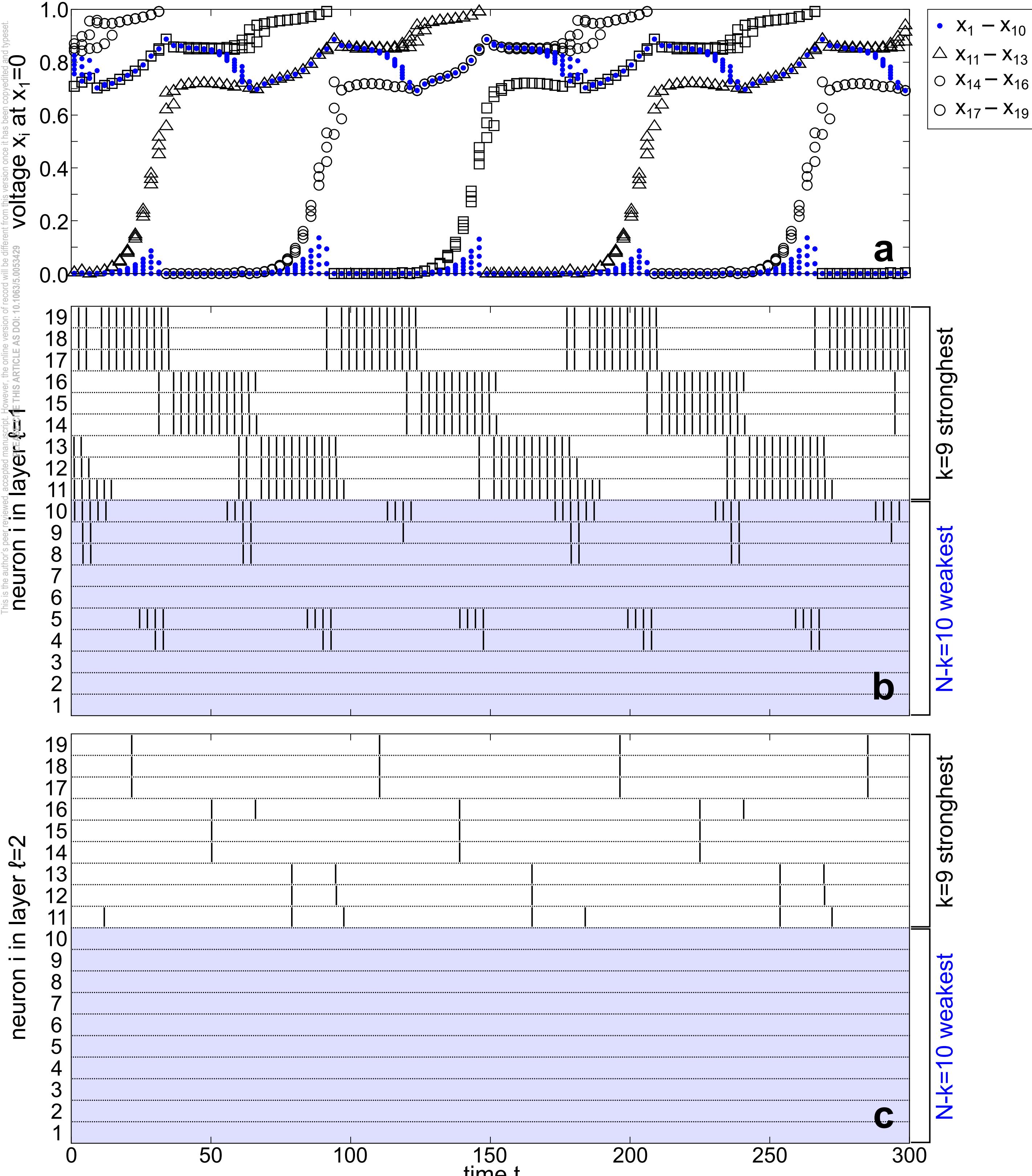


This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.

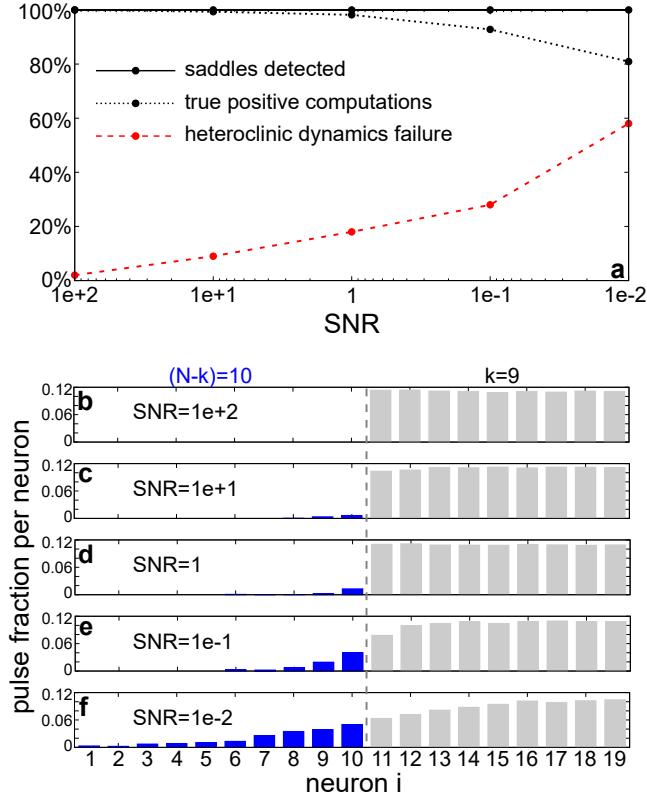
PLEASE CITE THIS ARTICLE AS DOI: [10.1063/5.0053429](https://doi.org/10.1063/5.0053429)



This is the author's peer-reviewed accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
E-1 THIS ARTICLE AS DOI: 10.1063/5.0053429

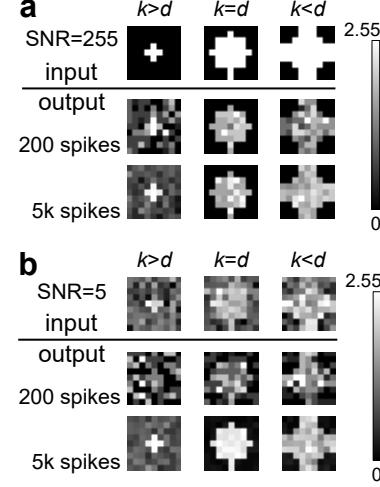
fixed input signal:  $\xi_{19} > \xi_{18} > \dots > \xi_2 > \xi_1$ 

This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.  
PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429



This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.

PLEASE CITE THIS ARTICLE AS DOI: [10.1063/5.0053429](https://doi.org/10.1063/5.0053429)



This is the author's peer reviewed, accepted manuscript. However, the online version of record will be different from this version once it has been copyedited and typeset.

PLEASE CITE THIS ARTICLE AS DOI: 10.1063/5.0053429

