Noise-constrained switching times for *heteroclinic computing*

Fabio Schittler Neves,[1,a)] Maximilian Voit,[1,a)] and Marc Timme[1,2,3,4,5]
[1]*Network Dynamics, Max Planck Institute for Dynamics and Self-Organization, 37077 Göttingen, Germany*
[2]*Technical University of Dresden, Institute for Theoretical Physics, 01062 Dresden, Germany*
[3]*Institute for Nonlinear Dynamics, Faculty of Physics, University of Göttingen, 37077 Göttingen, Germany*
[4]*Bernstein Center for Computational Neuroscience Göttingen, 37077 Göttingen, Germany*
[5]*Department of Physics, Technical University of Darmstadt, 64289 Darmstadt, Germany*

(Received 1 October 2016; accepted 15 February 2017; published online 6 March 2017)

Heteroclinic computing offers a novel paradigm for universal computation by collective system dynamics. In such a paradigm, input signals are encoded as complex periodic orbits approaching specific sequences of saddle states. Without inputs, the relevant states together with the heteroclinic connections between them form a network of states—the heteroclinic network. Systems of pulse-coupled oscillators or spiking neurons naturally exhibit such heteroclinic networks of saddles, thereby providing a substrate for general analog computations. Several challenges need to be resolved before it becomes possible to effectively realize heteroclinic computing in hardware. The time scales on which computations are performed crucially depend on the switching times between saddles, which in turn are jointly controlled by the system's intrinsic dynamics and the level of external and measurement noise. The nonlinear dynamics of pulse-coupled systems often strongly deviate from that of time-continuously coupled (e.g., phase-coupled) systems. The factors impacting switching times in pulse-coupled systems are still not well understood. Here we systematically investigate switching times in dependence of the levels of noise and intrinsic dissipation in the system. We specifically reveal how local responses to pulses coact with external noise. Our findings confirm that, like in time-continuous phase-coupled systems, piecewise-continuous pulse-coupled systems exhibit switching times that transiently increase exponentially with the number of switches up to some order of magnitude set by the noise level. Complementarily, we show that switching times may constitute a good predictor for the computation reliability, indicating how often an input signal must be reiterated. By characterizing switching times between two saddles in conjunction with the reliability of a computation, our results provide a first step beyond the coding of input signal identities toward a complementary coding for the intensity of those signals. The results offer insights on how future heteroclinic computing systems may operate under natural, and thus noisy, conditions. *Published by AIP Publishing.* [http://dx.doi.org/10.1063/1.4977552]

Standard paradigms of digital computation use collections of zero–one (or yes–no) bits to represent inputs, intermediate states of a computation as well as outputs of a computational system. Whereas such a system of N digital bits can represent an exponential number of 2^N states, it is not robust to any errors: each single bit flip moves the system to another state that does not reflect the desired one. Alternative concepts developed for "natural" dynamical systems, often of time-continuously coupled units with continuous state variables, typically do exhibit some robustness but often are limited to represent only a linear fraction αN of states. Recent progress showed that *heteroclinic computing*—a novel type of computation by bio-inspired dynamical systems—exhibits both, a certain degree of robustness to inexact inputs and an almost exponential ($N^{-\gamma}\exp(\beta N)$) scaling of the number of represented states. *Heteroclinic computation* has been demonstrated to be capable of performing universal computation in the sense that all digital functions can be represented in principle. However, the demonstration was focused on an idealized system setting where the reception of exchanged signals between units experiences no dissipation and there was no noise perpetually adding errors in time. In this article, we systematically study the joint influence of such processing dissipation and noise in systems of pulse-coupled oscillators, akin to coupled neurons. We show that computational times are bounded by noise levels (as was known for smoothly coupled systems) and that the noise ultimately overrides dissipation, setting a natural time scale for typical computations.

I. INTRODUCTION

Networks of coupled oscillators can exhibit a large variety of dynamical regimes, e.g., synchronous and chaotic regimes. Due to this variety, such systems have been used to model and design computation in natural[1–7] and artificial[8–11] systems. In particular, recent works on heteroclinic networks have shown interesting mathematical[12–19] and computational properties, including universal computation and an encoding capacity that increases exponentially with system size.[9,11]

The fundamental structures underlying *heteroclinic computing* are *saddle* states in state space with heteroclinic connections between them: Saddles are invariant sets, in the simplest example fixed points, exhibiting both stable and

[a)]F. S. Neves and M. Voit contributed equally to this work.

unstable manifolds, i.e., "directions" in state space along which small perturbations from a saddle grow or shrink, respectively. If an unstable manifold of one saddle A is contained in the stable manifold of a second saddle B, these two saddles are said to be connected by a *heteroclinic connection*, here denoted $A \to B$. Sufficiently small perturbations away from A may then induce a switch of the system towards B. A closed sequence of such saddles linked by heteroclinic connections is called a *heteroclinic cycle*, e.g., $A \to B \to C \to A$. Systems of globally coupled oscillators may exhibit many saddles and many more heteroclinic connections forming *a heteroclinic network*, a complex network of saddle states that typically offers two or more options to switch from every saddle, i.e., for instance $A \to B$ or $A \to B'$ with $B \neq B'$. In the paradigm of *heteroclinic computing*, information is encoded either as switching events between saddles[8,15] or as cyclic sequences of such events,[11] i.e., complex periodic orbits in state space.

Symmetrical systems composed of coupled oscillators naturally exhibit heteroclinic networks. The saddles composing such networks are, by themselves, periodic orbits exhibiting poly-synchrony where subsets (clusters) of oscillators synchronize to exhibit (nearly) identical dynamics. The cyclic dynamics traced out by trajectories in state space under a sequence of suitable periodic perturbations[10] or due to temporarily fixed, small symmetry-breaking input signals[11] are thus complex periodic orbits created through two periodicities: that of the periodic saddle orbits and the periodic sequence of switchings between different saddle orbits.

Due to permutation symmetry, typically all saddle states forming the heteroclinic network are isomorphic and exhibit the same local and global stability properties, in particular, the same number of oscillators per cluster, the same stability eigenvalues as well as the same type and number of heteroclinic connections towards and away from them. What differentiates each individual saddle state is exclusively the identity of the oscillators composing each cluster. Under such symmetry conditions, switching transitions between saddles preserve the state symmetry class.

To better understand how computations are actually performed, consider a simple example of a small network of $N = 3$ oscillators with three interconnected saddle orbits (see Figure 1). Each exhibits permutation symmetry $S_2 \times S_1$, i.e., all combinations of one cluster of two synchronized oscillators and one singleton not synchronized to the other two (More details of model settings are explained below; however, these details are not relevant for the characteristic heteroclinic dynamics illustrated now). Let those saddle states be denoted $A = (a, a, b)$, $B = (a, b, a)$, and $C = (b, a, a)$. Here, the position of the label in the vector is the oscillator index, a is the label for the cluster, and b the label for the singleton. In this system, the cluster is unstable to desynchronizing its two oscillators and the resulting heteroclinic network is connected all-to-all. Furthermore, saddle switches follow a simple rule: without loss of generality, starting from $A = (a, a, b)$ a general perturbation $(\Delta_1, \Delta_2, \Delta_3)$ either yields the transition

$$A = (a, a, b) \to (b, a, a) = B, \tag{1}$$

for $\Delta_1 > \Delta_2$, independent of Δ_3, or

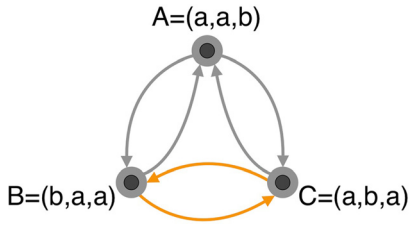

FIG. 1. Heteroclinic computing in a small network of states. In this example, a system composed of three oscillators exhibit three saddles connected via six heteroclinic connections. Dark circles represent saddles; large gray circles represent the saddle vicinities; and arrows represent heteroclinic connections. In real systems that are noisy and exhibit heterogeneities and imperfections, switching processes spontaneously arise between neighbourhoods of saddles, cf. Section III. We denote these saddles as $A = (a, a, b)$, $B = (a, b, a)$, and $C = (b, a, a)$, where the position of the label in the vector is the oscillator index, a is the label for a two-oscillator cluster and b the label for a singleton. Saddle switches follow rules (1) and (2) (and their permutations). The orange set of arrows highlight the "complex" periodic orbit $B = (b, a, a) \to C = (a, b, a) \to B = (b, a, a)$ triggered by a persistent $\Delta_1 > \Delta_2 > \Delta_3$ perturbation, where the sub index indicate the oscillator being perturbed.

$$A = (a, a, b) \to (a, b, a) = C, \tag{2}$$

for $\Delta_1 < \Delta_2$, also independent of Δ_3. By permuting the oscillator indices we find the other four possible switching options. Thus, observing any one such switching process between two saddles tells us which of the two inputs to the two oscillators in the clusters was the larger one. Such a system thereby computes the rank order of the perturbation vector projected onto the (unstable) cluster. If the perturbation signals comes in form of a continuing input (or consists of small, sufficiently many perturbations with identical rank order), for $\Delta_1 > \Delta_2 > \Delta_3$, the orbit

$$A \xrightarrow{\Delta_1 > \Delta_2} B \xrightarrow{\Delta_2 > \Delta_3} C \xrightarrow{\Delta_1 > \Delta_3} B \tag{3}$$

is established. After transiently leaving the state A, the dynamics continuously switches between the B and C states (Figure 1). This two-states orbit is complex in that it is an orbit alternating between B and C, where both B and C are already periodic. Observing the specific cycle (complex periodic orbit) $B \to C \to B\ldots$ thus tells a partial rank order of the input vector: it tells that Δ_1 and Δ_2 are both larger than Δ_3. The collective dynamics of the system, as steered by the input perturbations, thus constitutes a k-winners-take-all computation (with $N = 3$ and $k = 2$). We recently demonstrated theoretically[11] that this principle of *heteroclinic computing* is universal and in principle scalable. For instance, a system of $N = 100$ coupled oscillators exhibits a k-winners-takes-all computation with $k = 20$, with of the order of 10^{65} saddles and 10^{20} distinguishable cycles of saddles. The system can thus process 10^{20} qualitatively different inputs (at identical parameters) and thus, in principle, perform as many different individual computations.

As the switching dynamics are decisive of the overall heteroclinic dynamics and thus essential for any type of heteroclinic computing, switching times play an important role in the overall encoding and decoding processes. Switching times after instantaneous perturbations[20] have been studied for time-continuously coupled systems. Pulse-coupled

oscillator systems enable analytic access to periodic and even non-periodic dynamics and thereby improved qualitative and quantitative insights useful for understanding options and properties of heteroclinic computing. Yet, most results, including those for switching times do not trivially transfer to pulse-coupled and other hybrid systems from systems coupled continuously in time. Indeed, pulse-coupled systems are known to exhibit a number of non-standard dynamical properties, including unstable Milnor attractors and local stability operators that are non-linear,[15,16,21,22] a breakdown of order preservation in symmetric systems[23] and non-standard bifurcations.[24,25] How noise sources impact the switching dynamics and in particular, the switching times is not well understood for pulse-coupled systems.

In this article, we show how providing independent noise to all oscillators in a system exhibiting heteroclinic networks provides an upper bound for the switching times between saddles. This work is divided in four main sections: we first introduce (in Section II) the model of coupled oscillators and briefly review specific previous results; in Section III, we present working definitions of switchings and switching times; in Section IV, we characterize the switching dynamics induced by noise and the combined effects of partial reset and noise. Finally, we briefly discuss the results and their impact in neural computing (Section V).

II. PULSE-COUPLED OSCILLATORS AND PARTIAL RESETS

Consider all-to-all connected networks of pulse-coupled oscillators (no self-connections). The unit dynamics is a modified version of the model introduced in Ref. 1, now incorporating partial resets[25,26] and delays.[27,28] More specifically, consider an Integrate-and-fire neuronal model, where a voltage-like state variable x follows the dynamical rule:

$$\frac{dx_i}{dt} = f(x_i) + \sum_{j=1}^{N} \sum_{t_{j,l}\in P_j} \varepsilon\delta(t - t_{j,l} - \tau), \quad 0 \le x_i < x^\theta, \quad i \in \{1, \ldots, N\}, \tag{4}$$

where i is the neuron index, N is the number of neurons, $x_i = x^\theta$ is the firing threshold, and $f(x_i)$ is a positive function, i.e., $f(x_i) > 0$. The summation in Equation (4) is the contribution of all pulses arriving from the other $N-1$ neurons to neuron i, where τ represents the connection delays, ε the connection weight, and P_j is the set of all times of pulses sent by neuron j. Whenever the upper threshold at x^θ is crossed, either by reaching the threshold or receiving incoming pulses, the neuron is reset to a phase

$$x_i(t^+) = c \cdot (x_i(t) + m\varepsilon - x^\theta), \tag{5}$$

and a pulse is sent to all other neurons. Here, t^+ denotes the time immediately after t, m is the number of arriving pulses at time t, and $c \in [0, 1]$ is a partial reset constant that defines the fraction of supra-threshold voltage kept after reset. The partial reset models memory effects of signals received before the most recent reset.[25] Mathematically, $c \neq 0$ implies an invertible flow, see Ref. 26.

Because the voltage-like dynamics presented above is monotonic between events, it can be mapped to a simple phase variable $\phi_i(t) \in [0, 1]$ such that

$$\frac{d\phi_i}{dt} = 1 \quad \text{with} \quad \phi_i(t^+) \to 0 \quad \text{if} \quad \phi_i(t) = 1. \tag{6}$$

Here the time of one free oscillation is rescaled to one, $\phi_i(t) \to 1$ as $x_i(t) \to x^\theta$ and $\phi_i(t) \to 0$ as $x_i(t) \to 0$. In this phase representation, to compute the contribution of incoming pulses, the phase is first transformed back to the voltage variable, the effect of all pulses are computed, and, after that, transformed again into its phase representation. Let the map be denoted $x_i(t) = U(\phi_i(t))$. The resulting phase immediately after the arrival of incoming pulses is either

$$\phi_i(t^+) = U^{-1}(U(\phi_i(t)) + m\varepsilon), \tag{7}$$

if the resulting phase is sub-threshold, or

$$\phi_i(t^+) = U^{-1}(c \cdot (U(\phi_i(t)) + m\varepsilon - x^\theta)), \tag{8}$$

if the resulting phase is supra-threshold. Furthermore, this system of equations is integrable between pulse events and, thereby, yields an analytical event based description of its time evolution. Throughout this work, we define a unit of time as one oscillation of an uncoupled oscillator.

To be concrete, we choose a well studied oscillator model[1] with

$$U(\phi_i) = \frac{1}{b}\ln\left(1 + (e^b - 1) \cdot \phi_i\right), \tag{9}$$

where b is a constant controlling the concavity of x_i. For the sake of simplicity and clarity, we here study a network composed of four oscillators exhibiting two interconnected saddles. Notice that in this work the size of the network of states is irrelevant, because we are interested in the timings during switching dynamics and not the saddles identities. We here consider a simple and small example system of $N = 4$ oscillators and illustrate that the results are qualitatively the same for a system of $N = 100$ oscillators in Appendix. For parameters $b = 4.2$, $\tau = 0.02$, $\varepsilon = 0.23$, $c = 0$, and threshold $x^\theta = 1$, this four-oscillator system exhibits two unstable attractors linked by two heteroclinic connections, see Figure 2(a). More specifically, two clusters of two oscillators perfectly synchronize. One cluster is unstable, whereas the other is stable.[25] For $c = 0$, upon a small perturbation to an initial saddle state, the unstable cluster transiently desynchronizes and resynchronizes along the heteroclinic orbit. After this transition, the elements in each cluster are still the same, but their stability is interchanged, marking the arrival at the other saddle state. The final state is identical to the initial one after a proper permutation of the units' indices.

It was shown in Ref. 24 that increasing c to values above 0 yields local invertibility because the supra-threshold fraction in reception events is not erased, see Equation (6). Thus, we have a bifurcation at $c = 0$, in which the saddle periodic orbits cannot be perfectly reached anymore in finite time because perfect resynchronization is prevented. Instead, the system exhibits spiral orbits approaching the heteroclinic

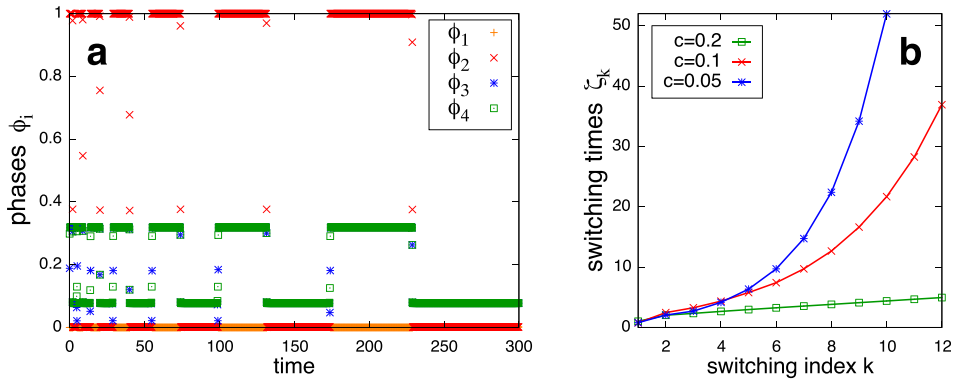

FIG. 2. Switchings induced by instantaneous perturbations with amplitudes 10^{-5} and their switching times. (a) For $c = 0.1$, all phases are plotted each time oscillator $i = 1$ is reset. Notice the ever increasing plateaus after the fast part of switchings; (b) the switching times after the perturbation for three values of c. The switching times increase exponentially with the number of switches.[24] Small differences in the switching times ranking order during the first three switches are caused by small differences in the times in which the perturbations were applied in each case.

cycle, yielding an ongoing switching process. Indeed, the switching times between saddles (roughly the time spent in each plateau in Figure 2(a)) after a one-time perturbation grow exponentially with the number of switchings, see Figure 2(b) and Refs. 15, 16, and 25 for more details.

III. SWITCHING TIMES BETWEEN SADDLE STATES

In this work, a precise working definition of "switching" is required, even though what is meant by "switching" seems to be intuitively clear. Naively, one may define a switching event as the trajectory along a heteroclinic connection from one saddle to another. Thus, a switch would be a transient process with seemingly well-defined start and end points. This definition does not apply for systems with partial resets or under noise because these conditions prevent the system from precisely reaching the saddles (cf. Figure 1). To characterize the switching times under those conditions, we propose a less restrictive definition, in which the saddles do not need to be precisely reached, but only a certain vicinity of the saddle orbits. More specifically, we consider that a saddle has been "reached" at a time t_k if the (so far) stable cluster loses stability and begins to consistently desynchronize. To compute the t_k values we actually observe the system's event sequence (sending and receiving pulses). As mentioned in Section II after a perturbation the oscillators will not perfectly resynchronize in finite time for $c > 0$. Nevertheless, the overall stability properties close to the saddles for $0 < c < 1$ are still similar to the case $c = 0$. That is, one quasi-synchronized cluster tends to resynchronize after perturbations while the other tends to desynchronize, until a new saddle is "reached." Figure 2(a) shows the two-clusters dynamics, where the elements in each cluster are virtually synchronized except for the short period between plateaus. At these times, a large difference in the phases of the oscillators in the unstable cluster appears transiently. We here choose two consecutive changes in cluster stability, from stable to unstable, as our reference points to calculate the switching times, i.e., the times in which the phase difference in a cluster starts to consistently increase.

Having defined the $\{t_k\}$ as the change of stability times, we here denote by ζ_k the switching time between arriving in the vicinity of a saddle, leaving it again, and arriving in the next saddle's vicinity. Therefore, given two consecutive switchings we have

$$\zeta_k := t_{k+1} - t_k, \tag{10}$$

where t_k is the time of the k-th departure event. We are also interested in the average switching time

$$\bar{\zeta} := \frac{1}{L}\sum_{k=1}^{L} t_{k+1} - t_k, \tag{11}$$

where L is the number of switches, given different noise levels and fractions of partial resets.

IV. NETWORK DYNAMICS SUBJECT TO NOISE

This section deals with the aspects of noise in networks of pulse-coupled oscillators. We substantiate the importance of noise in systems with discontinuous dynamics and define the type of noise used throughout this work. Based on these fundamentals, we characterize how noise promotes sustained switchings between two saddles in a $N = 4$ network and later the combined effect of noise and partial resets.

A. Noise induced switching

In Section II we showed that the dynamics of delta-pulse-coupled oscillators have a discrete event sequence description, providing a precise and fast alternative to numerical integration. To keep these advantages, we approximate a continuous Gaussian noise source as the sum of two high frequency Poisson distributed delta-pulse-trains with very small, fixed amplitudes, one with positive pulses and one with negative pulses. In this way, noise with zero mean contribution can be added to each neuron as the sum of two independent pulse trains on top of the pulses exchanged between oscillators (network pulses) and, thus, their contribution is simply computed via Equations (7) and (8). In the unlikely event of a negative pulse, arriving just after a reset event, producing a slightly negative phase, the phase is instead set to zero. Furthermore, for each oscillator i, the

noise generated in an interval Δ, denoted as $\eta_i(t,\, t+\Delta)$, is given by a simple summation

$$\eta_i(t, t+\Delta) = \sum_{\{j:t<\iota_{ij}<t+\Delta\}} s_{ij} n_{ij}, \tag{12}$$

where ι_{ij} is the time of the j-th "noise pulse" arriving at the i-th oscillator (Poissonian pulse train), n_{ij} is a fixed pulse amplitude ($n_{ij} \ll \epsilon$), and s_{ij} is a random variable that assumes either 1 or -1 with the same probability, to keep the mean of the generated noise at zero. The introduction of the sign s_{ij} means that, in fact, we have two Poisson processes with the same frequency (half of the original) and same variance, but different signs. Altogether, the final distribution of times between spikes approximates a Gaussian centered at zero with the same variance as the originals, because the sum of the variances is compensated by the drop in half in the individual variances due to the decrease in individual frequencies. In practice, algorithmically, $\{\iota_{ij}\}$ is a large number of new pulse events with low amplitude, with their time differences drawn from

$$\iota_{i(j+1)} - \iota_{ij} = -\frac{\log(1-u)}{f}, \tag{13}$$

where u is a random number drawn from a uniform distribution between 0 and 1 and f is a chosen frequency for the Poisson process. Therefore, our noise level parameter is neither the pulse strength nor the frequency, but how they combine to provide the noise standard deviation, given by

$$\sigma_i = |n_{ij}|\sqrt{f_i}. \tag{14}$$

In this work, without loss of generality, we fixed f at 100 pulses per a neuron free oscillation and we vary n_{ij} in order to vary σ_i.

To show how even a small noise level can strongly influence the dynamics for small c (typical case), we now describe how a small phase shift affects the overall dynamics. For simplicity, we consider the limiting case $c=0$. Starting from fully synchronized clusters, pulses from a cluster are sent at the same instant and, thus, arrive at the same time. If the pulses cause a supra-threshold event, the surplus of phase is lost, due to the reset rule (for small c it is almost erased). On the other hand, if a small perturbation is present, one pulse may be able to cause a reset, and the second pulse would be computed in full. As a result, the difference in phases is rather different in the two cases, after the pulses arrived. Even an arbitrarily small perturbation can thus cause a large change in dynamics.

Similar to the switchings sustained by partial reset, noise also leads to a persistent switching process, because the saddles are unstable, see Figure 3. The following results concern the switching times for a network with $N=4$ and $c=0$, to study the switching times independently of partial resets effects. We find that the smaller the noise strength, the larger are the mean switching times and their variances (Figure 4). This happens because, at small noise values, the system spends a larger time at the slow dynamics close to the saddle point, and thus, there is a larger probability that a switching will occur due to low probability events with a larger magnitude. As the noise strength increases, the system departs shortly after approaching the vicinity of a saddle, and thus its variance is small. Furthermore, we find that the dependence of $\bar{\zeta}(\sigma)$ on σ, as known for smooth dynamical systems,[29] is logarithmical

$$\bar{\zeta}(\sigma) \propto \ln(\sigma), \tag{15}$$

as shown in Figure 4(b). Even though this scaling was already shown for networks of phase coupled oscillators,[8,20,30,31] its emergence in the context of pulse-coupled systems was not established.

B. Combined effects of noise and partial reset ($N=4$)

We found that the resulting dynamics from the combined effects of noise and partial reset is roughly a combination of the separate effects. Starting from a relatively large perturbation (10^{-5}), we first observe a continuous increase in switching times, consistent with the partial reset dynamics. As the switching times approach the times predicted by the noise induced switchings, it stagnates, because the amplitude of the noise contribution has the same order of magnitude as the residual phase differences given by the partial reset events. In Figure 5 we provide an example of the time evolution in those conditions and the switching times for the same initial perturbation but for different noise levels. Small variation at the upper boundary is caused by random fluctuations in the generated noise.

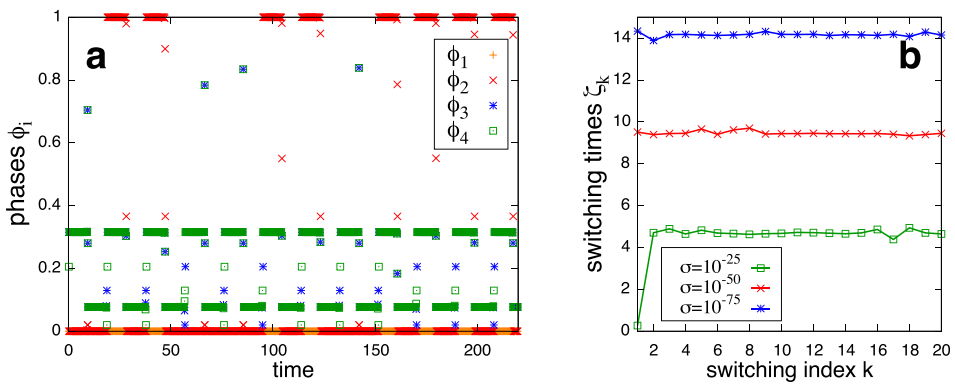

FIG. 3. Switchings induced by noise and their switching times. (a) For $\sigma = 10^{-50}$, all phases are plotted each time oscillator $i=1$ is reset. The plateaus exhibit approximately the same length; (b) for three values of σ, the switching times for the first 20 switches are shown. Switching times decrease with the strength of the noise.

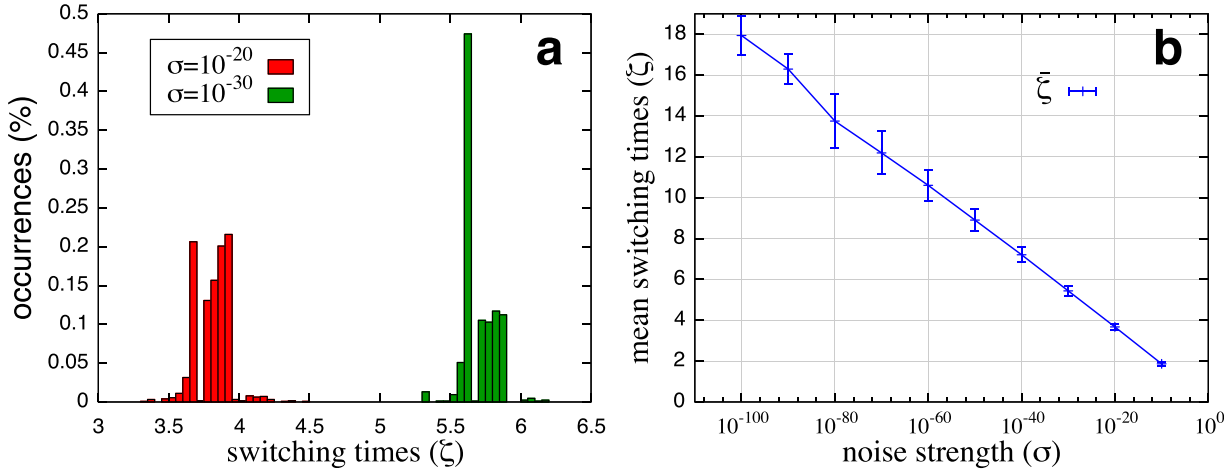

FIG. 4. Dependence of switching times ζ on the noise strength σ. (a) Example of distribution of switching times for $\sigma = 10^{-20}$ and $\sigma = 10^{-30}$ after, respectively, 1270 and 846 switches. Both exhibit a relatively small variance and are roughly distributed in a one unit range; (b) dependence of the mean switching times and the switching time variance on σ. Notice that the variance increases as the noise strength decreases.

C. Combined effects of noise and partial reset ($N = 5$)

How may noise affect computations by altering the switching times as well as the switching directions after one instantaneous perturbation signal? Because the system studies so far exhibit only one switching direction per saddle, to the only other saddle available in a network of states composed of two saddles, we here study a slightly larger system composed of $N = 5$ Integrate-and-Fire oscillators. Specifically, we choose, for Equation (4), $f(x_i) = (I - x_i)$, $I = 1.04$, $\tau = 0.49$ (already rescaled for ϕ), and $\epsilon = 0.025$. For these parameters, this system exhibits thirty interconnected saddles with two incoming directions and two outgoing directions each (see Refs. 10 and 11). Furthermore, switching times for this new system increase much more rapidly than for the system of $N = 4$ units studied before. To compensate in part for this increase of partial reset events per switch, we here set $c = 0.7$, preserving the signal for a while longer, and focus only in the first five switching events.

The results shown in Figure 6(a) conform with the ones presented in Sections III and IV B, see in particular Figures 2(b), 3(b) and 5(b) i.e., after an instantaneous perturbation signal, the switching times between saddles increase exponentially up to some level set by the noise. The main difference is that in this example there are two exit directions per saddle, i.e., the trajectory may approach one out of two possible saddles right after leaving the neighborhood of one saddle. In this particular example, if the system is driven by zero-centered Gaussian noise only, the chance of following any one of these paths is 50%.[10] If input signals are applied in addition to the noise, these signals may generate switches with deterministic directions, cf. Ref. 11. Yet due to dissipation (set by c) and noise, the influence of one instantaneous perturbation decreases with each supra-threshold input pulse, triggering a reset. Simultaneously, the switching times will exponentially increase. Thus, to evaluate the computational reliability in the presence of noise, we numerically determine the fraction of switching events that do not go into the direction predicted by the same signal in the absence of noise. We define this fraction to be the *error probability* $P_{err}(\sigma)$. Figure 6(b) illustrates estimations of the error probability for a sequence of five switches for three different noise levels after the same instantaneous perturbation. Qualitatively, the error probability stays essentially zero as long as the switching

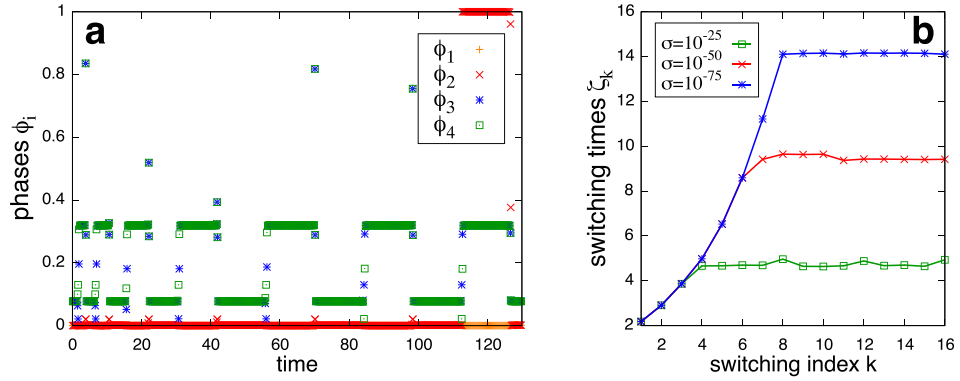

FIG. 5. Noise constrains switching times. Switchings in a $N = 4$ system subject to partial reset and concurrent noise ($c = 0.1$ and single signal amplitude of 10^{-5} in both panels). (a) The phases of all oscillators are plotted whenever oscillator $i = 1$ is reset. The switchings are promoted by partial resets after an initial perturbation; the switching time interval increases until a value close to the mean $\bar{\zeta}(\sigma)$ determined by the noise strength, here set to $\sigma = 10^{-75}$. (b) Switching times depending on the number of switchings for different noise strengths. The three curves are qualitatively the same, with the approximate maximum ζ_k value depending on σ. Early deviations from the noiseless case are caused, with low probability, by random fluctuations of the realized noise value.

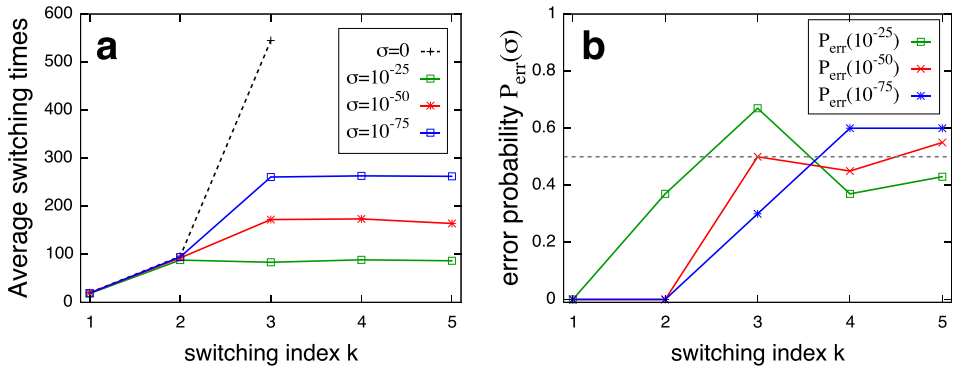

FIG. 6. Noise constrains switching times and switching error probability. In all simulations, the initial perturbation has amplitude of 10^{-4}. Switchings in a $N=5$ system subject to partial reset and concurrent noise ($c=0.7$ in both panels). (a) Switching times depending on the number of switchings for different noise strengths. The three solid curves are qualitatively the same, with the approximate maximum ζ_k value depending on σ. The dashed line shows the switching times in the absence of noise growing exponentially. (b) The switching error probability (turn to the wrong saddle) is calculated for the three solid curves in panel (a); average switching times and error probabilities are calculated as averages over 30 simulations.

times are below those set by the noise, whereas it becomes substantially large (30% and more in the sample estimates shown in Figure 6(b)) once the switching times are of the order set by the noise. These results shows how often a signal must be reiterated to be properly computed and thus indicate how low the frequency of an spike train signal (signal repetition) can be to still enable reliable computations at a given noise level. We note that in the limit of high frequency inputs with sufficiently small amplitudes, the input signal sequence resembles a constant current, which do not induce exponentially growing switching times, but a fixed value[11] instead.

V. SUMMARY AND CONCLUSIONS

How does noise impact the dynamics along networks of states and thus *heteroclinic computing*[11]? In systems of pulse-coupled oscillators with partial resets, we systematically studied how noise changes the switching times. Whereas initially the switching times increase exponentially due to the local instability of the saddles, in the long term, they saturate as they are driven by the noise.

For simplicity and numerical tractability in a reasonable time, we here considered systems of $N=4$ pulse-coupled oscillators. As argued above, the results do not qualitatively depend on system size, because we are interested in the saddle dynamics (How does a trajectory pass one saddle?) and not the saddle identity (Which sequence of saddles does the trajectory pass?). This small system exhibits a network of two saddles and already captures the necessary switching aspects (switching times).

We first briefly reviewed results on a bifurcation occurring due to partial resets,[24] which leads to sustained switching between the neighborhoods of saddle periodic orbits. As specific new results, we showed that switching times grow exponentially, because deviations of the trajectory from the saddles decrease at every cycle by some factor. Thus, the trajectory more and more closely approaches the heteroclinic cycle and, thereby, slows the dynamics close to the saddle orbits. Second, we showed that similar switchings occur due to noise without partial resets. More specifically, we showed that the mean switching times scale logarithmically with the noise variance (mean zero), consistent with numerical observations[8,20,31] and analytic estimates[29] for continuous-time systems. Particularly, our results do not trivially follow from those for continuous-time systems, because the dynamics of systems of pulse-coupled oscillator are not continuous but of (discrete-continuous) hybrid type. Third, we investigated the dynamics simultaneously under the influence of noise and partial resets. We demonstrated how the switching times induced by noise may override the ones induced by partial resets and thus constrain them. Furthermore, we studied a system of $N=5$ oscillators (exhibiting a heteroclinic network of 30 saddles[10,11] and at each saddle two options for subsequent saddles) and show that computational reliability switches from faithful (close to 0% error probability) to unfaithful (error probability close to chance level). This result provides a lower bound on how frequently an input signal must be presented again to maintain a reliable computation or, equivalently, how low the frequency of a spike train signal can be to still be properly computed. A simple but important fact is that each instantaneous perturbation signal applied yields some head time during which noise can act on the system without being capable of blurring the computation in any substantial way.

In summary, our results suggest that systems of pulse-coupled oscillators exhibit characteristic switching times available for the dynamics near networks of saddle states and thus should, in principle, enable *heteroclinic computing*, cf. Ref. 11 in noisy pulse-coupled systems with dissipation. In particular, these results indicate that noise may constrain a computation to a maximum time scale available for switchings. Mathematically, it remains to be shown that the mechanisms for this constraint are the same for both continuous and pulse-coupled systems. If so, any engineered *heteroclinic computing* device may come, on the one hand with a lower bound on the time interval to present a given instantaneous signal again, and on the other hand with an upper bound guarantee for the times of completing a computation. As heteroclinic computing constitutes a novel type of analog computing by exploiting collective dynamics emerging in a broad class of systems, our results provide hints about how such systems operate under noisy conditions. The results thus do not only offer theoretical insights about scaling

properties of systems' dynamics, but may also open up ways for hardware implementations of heteroclinic computing interacting with the environment, including robotic applications, decision making in general, and short-term memory.

APPENDIX: HETEROCLINIC DYNAMICS: 100 OSCILLATORS' NETWORK

We here present an example of a system composed of $N = 100$ oscillators exhibiting heteroclinic dynamics. The model is the same as introduced in the main text, Equation (9), with $U = 3$, $\tau = 0.15$, and $\epsilon = 0.2$. Consistent with the example in the main text, this system exhibits decreasing switching times with increasing noise strengths. As shown in Figure 7, the saddles are periodic orbits exhibiting five clusters. Moreover, this saddle states have $S_{21} \times S_{21} \times S_{21} \times S_{21} \times S_{16}$ cluster permutation symmetry and are unstables only to perturbations to one of the clusters. We here call such cluster the "unstable cluster," and the remaining clusters are simply called "stable clusters." The system also exhibits symmetry preserving saddle switches. As shown in Figure 7, after approaching a saddle state, a perturbation (noise) triggers a saddle switch. In this process, the five oscillators in the unstable cluster subject to the stronger perturbations depart from the cluster and join the cluster originally composed of 16 oscillators, forming a new stable 21 cluster; after a transient, the remaining 16 oscillators from the originally unstable cluster form a new stable cluster; and another 21 cluster losses stability after the relative phase differences between cluster are dynamically rearranged. As show in Ref. 11, a persistent detuning perturbation to such systems provides a 20-winners-take-all out of a 100 detuning currents.

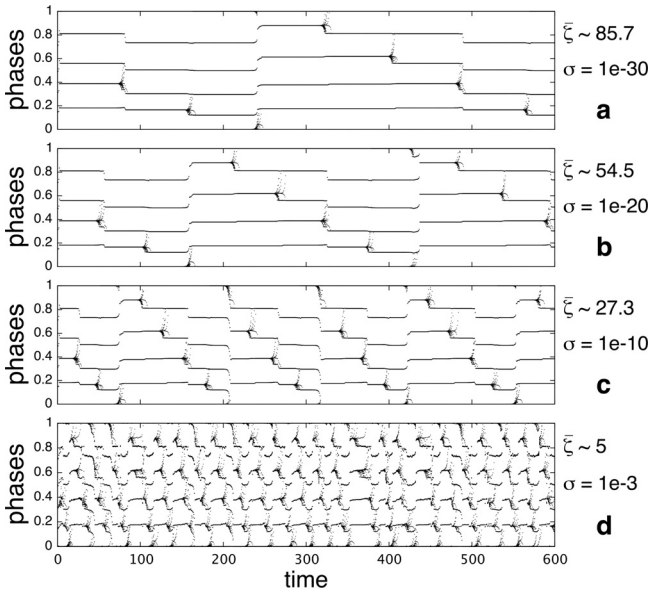

FIG. 7. Average switching times for different noise levels, $N = 100$ oscillators, $U = 3$, $\tau = 0.15$, and $\epsilon = 0.2$. (a)–(d) Each time oscillator $i = 1$ is reset, the phases of all oscillators are plotted as single dots. The average switching times $\bar{\zeta}$ and noise strength σ are indicated at the right side of the corresponding graph.

[1] R. E. Mirollo and S. H. Strogatz, "Synchronization of pulse-coupled biological oscillators," SIAM J. Appl. Math. **50**(6), 1645–1662 (1990).
[2] M. Rabinovich, A. Volkovski, 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).
[3] M. Rabinovich, R. Huerta, and G. Laurent, "Transient dynamics for neural processing," Science **321**, 48–50 (2008).
[4] P. Ashwin and M. Timme, "When instability makes sense," Nature **436**, 36–37 (2005).
[5] C. Bick and M. Rabinovich, "Dynamical origin of the effective storage capacity in the brain's working memory," Phys. Rev. Lett. **103**, 218101 (2009).
[6] M. I. Rabinovich, P. Varona, I. Tristan, and V. S. Afraimovich, "Chunking dynamics: Heteroclinics in mind," Front. Comput. Neurosci. **8**, 22 (2014).
[7] P. Chosat and M. Krupa, "Heteroclinic cycles in hopfield networks," J. Nonlinear Sci. **26**, 315–344 (2016).
[8] P. Ashwin and J. Borresen, "Discrete computation using a perturbed heteroclinic network," Phys. Lett. A **347**(4–6), 208–214 (2005).
[9] J. Wordsworth and P. Ashwin, "Spatiotemporal coding of inputs for a system of globally coupled phase oscillators," Phys. Rev. E **78**, 066203 (2008).
[10] F. S. Neves and M. Timme, "Controlled perturbation-induced switching in pulse-coupled oscillator networks," J. Phys. A **42**(34), 345103 (2009).
[11] F. S. Neves and M. Timme, "Computation by switching in complex networks of states," Phys. Rev. Lett. **109**(1), 018701 (2012).
[12] T. Chawanya, "Infinitely many attractors in game dynamics system," Prog. Theor. Phys. **95**(3), 679–684 (1996).
[13] T. Chawanya, "Coexistence of infinitely many attractors in a simple flow," Physica D **109**, 201–241 (1997).
[14] M. Krupa, "Robust heteroclinic cycles," J. Nonlinear Sci. **7**, 129–176 (1997).
[15] M. Timme, F. Wolf, and T. Geisel, "Prevalence of unstable attractors in networks of pulse-coupled oscillators," Phys. Rev. Lett. **89**(15), 154105 (2002).
[16] M. Timme, F. Wolf, and T. Geisel, "Unstable attractors induce perpetual synchronization and desynchronization," Chaos **13**(1), 377–387 (2003).
[17] P. Ashwin and M. Timme, "Unstable attractors: existence and robustness in networks of oscillators with delayed pulse coupling," Nonlinearity **18**, 2035–2060 (2005).
[18] P. Ashwin, G. Orosz, and J. Borresen, "Heteroclinic switching in coupled oscillator networks: Dynamics on odd graphs," in *Nonlinear Dynamics and Chaos: Advances and Perspectives* (Springer, Berlin/Heidelberg, 2010), pp. 31–50. ISBN 978-3-642-04629-2.
[19] S. Castro and A. Lohse, "Switching in heteroclinic networks," SIAM J. Appl. Dyn. Syst. **15**(2), 1085–1103 (2016).
[20] D. Hansel, G. Mato, and C. Meunier, "Clustering and slow switching in globally coupled phase oscillators," Phys. Rev. E **48**(5), 3470–3477 (1993).
[21] H. Broer, K. Efstathiou, and E. Subramanian, "Robustness of unstable attractors in arbitrarily sized pulse-coupled networks with delay," Nonlinearity **21**(1), 13 (2008).
[22] H. Broer, K. Efstathiou, and E. Subramanian, "Heteroclinic cycles between unstable attractors," Nonlinearity **21**(6), 1385 (2008).
[23] H. Kielblock, C. Kirst, and M. Timme, "Breakdown of order preservation in symmetric oscillator networks with pulse-coupling," Chaos **21**, 025113 (2011).
[24] C. Kirst and M. Timme, "From networks of unstable attractors to heteroclinic switching," Phys. Rev. E **78**, 065201(R) (2008).
[25] C. Kirst, T. Geisel, and M. Timme, "Sequential desynchronization in networks of spikin neurons with partial resets," Phys. Rev. Lett. **102**, 068101 (2009).
[26] C. Kirst and M. Timme, "Partial reset in pulse-coupled oscillators," SIAM J. Appl. Math. **70**(7), 2119–2149 (2010).
[27] U. Ernst, K. Pawelzik, and T. Geisel, "Synchronization induced by temporal delays in pulse-coupled oscillators," Phys. Rev. Lett. **74**, 1570 (1995).
[28] U. Ernst, K. Pawelzik, and T. Geisel, "Delay-induced multistable synchronization of biological oscillators," Phys. Rev. E **57**, 2150 (1998).
[29] P. Ashwin and C. Postlethwaite, "Quantifying noisy attractors: from heteroclinic to excitable networks," SIAM J. Appl. Dyn. Syst. **15**(4), 1989–2016 (2016).
[30] E. Stone and P. Holmes, "Random perturbations of heteroclinic attractors," SIAM J. Appl. Math. **50**(3), 726–743 (1990).
[31] D. Armbruster, E. Stone, and V. Kirk, "Noisy heteroclinic networks," Chaos **13**(1), 71–79 (2003).