Experiments on autonomous Boolean networks

David P. Rosin,[1,2] Damien Rontani,[1] Daniel J. Gauthier,[1] and Eckehard Schöll[2]
[1]*Duke University, Department of Physics, Science Drive, Durham, North Carolina 27708, USA*
[2]*Institut für Theoretische Physik, Technische Universität Berlin, Hardenbergstr. 36, D-10623 Berlin, Germany*

(Received 22 January 2013; accepted 23 April 2013; published online 4 June 2013)

We realize autonomous Boolean networks by using logic gates in their autonomous mode of operation on a field-programmable gate array. This allows us to implement time-continuous systems with complex dynamical behaviors that can be conveniently interconnected into large-scale networks with flexible topologies that consist of time-delay links and a large number of nodes. We demonstrate how we realize networks with periodic, chaotic, and excitable dynamics and study their properties. Field-programmable gate arrays define a new experimental paradigm that holds great potential to test a large body of theoretical results on the dynamics of complex networks, which has been beyond reach of traditional experimental approaches. © *2013 AIP Publishing LLC*. [http://dx.doi.org/10.1063/1.4807481]

Dynamical networks have attracted considerable attention because of their ubiquitous presence in numerous fields,[1,2] such as biology (cellular and metabolic networks, food webs, neural networks)[3–6] and social sciences (mobile communication networks, scientific collaboration networks).[7,8] Insight into the dynamics of networks comes predominately from studies of mathematical models and observations of real-world networks as a test bed for theoretical results. There is also need to realize networks in the laboratory to test theoretical predictions in a controlled environment. But hitherto, the difficulty to connect a large number of dynamical nodes has restricted experiments to coupling topologies with at most 20 nodes.[9,10] As a solution, computer algorithms have been used to manage the coupling between experimental dynamical systems.[11,12] Here, we present an approach without computer-assisted coupling for the experimental realization of networks of potentially large sizes using a field-programmable gate array (FPGA)—an integrated circuit with millions of reconfigurable logic gates. Using its autonomous mode of operation, we implement continuous-time dynamical systems with periodic, chaotic, and excitable dynamics that can be coupled to arbitrary topologies and display collective phenomena such as synchronization.

I. INTRODUCTION

Logic gates on FPGAs can be assigned to arbitrary Boolean functions[13] and interconnected to form Boolean networks, which are used typically to model diverse biological processes, such as gene and metabolism regulation,[14–16] cell-cycle dynamics,[17] neural interactions,[18] social networks.[19]

The Boolean states of the nodes in a Boolean network evolve in time according to logic functions.[20] Typically, the dynamical state of the network is updated either synchronously or asynchronously, where the Boolean states of the nodes are updated according to their logic functions simultaneously or successively with randomly chosen updating order, respectively.[21] These updating strategies for the network dynamics simplify the mathematical analysis[22] and allow for exact numerical simulations but are, in some regard, unrealistic.

For example, gene and metabolism regulation networks are not updated by a discrete global clock in nature and should, therefore, be modeled continuously in time with dynamical updating of the logic functions.[23] To account for continuous temporal evolution, Boolean delay equations (BDEs)[24,25] and ordinary differential equations[26] have been introduced and the networks are then referred to as autonomous Boolean networks (ABN). A node in an ABN updates its Boolean state whenever Boolean transitions are present at its inputs. Because of the finite response time of the node, intermediate output states have been numerically and experimentally observed. The consequence of the autonomous operation and the resulting non-ideal behavior of the gate is the existence of rich and complex dynamics, such as chaos[27,28] and quasi-periodicity.[29]

FPGAs allow one to realize experimentally large ABNs to test theoretical predictions[23,26,29] of models of Boolean networks. The experimental approach reveals dynamics that is not predicted in theoretical studies, which usually neglect non-ideal behaviors that may appear both in the biological and electrical systems, such as the sigmoidal activation functions of the gates, intrinsic parameter heterogeneity, noise. Glass *et al*. have already identified differences between the dynamics of an idealized model and the electronic implementation of a simple Boolean network.[30]

Realizing large ABNs on an FPGA also has the advantage of fast dynamics, where the network nodes evolve with fast rise and fall times on the order of 300 ps. In contrast, a numerical simulation of an ABN requires multiple calculations for the continuous rises and falls for each logic gate in the network, which usually takes time on the order of milliseconds with current computer technology. With our approach, the network dynamics are, therefore, much faster generated than with the corresponding simulation.

In this article, we demonstrate the potential of FPGAs to realize experimentally large-scale complex networks with controllable node dynamics and arbitrary topology. The networks are meta-networks consisting of interconnected

autonomous logic circuits—an electronic realization of ABNs—that represent dynamical nodes with various types of dynamics. We first introduce the FPGA and its autonomous mode of operation. Then, we introduce a Boolean phase oscillator and couple it to networks that display phase synchronization. We continue by introducing an ABN that displays chaotic dynamics. Finally, we realize ABNs with excitable dynamics that can be coupled to large networks and synchronize with zero-time lag in the presence of coupling time delays.

II. EXPERIMENTAL NETWORKS ON A CHIP

In this section, we introduce the main working principles of FPGAs. We detail how to implement time-continuous dynamical nodes (ABNs) and connect them with time delay links into meta-networks.

A. Programmable logic gates

An FPGA has up to 2×10^6 re-assignable logic gates that generate high and low voltages $V_{H,L}$ corresponding to the Boolean states 0 and 1. They can execute any of the 2^n possible logic functions, where n is the number of inputs whose value is typically four or six depending on the FPGA technology. For our experimental platform (Altera Cyclone IV EP4CE115F29C7N), $n=4$. Logic operations with more inputs can be realized by combining multiple logic gates.

The logic circuit—the logic gates and their interconnection—is specified using a hardware description language, such as Verilog or VHDL.[13] A compiler optimizes and converts the logic design, so that it can be loaded on the FPGA, thereby specifying the Boolean operation of each logic element and the manner in which they are connected. These operations take as little as a few seconds depending on the complexity of the design. The flexibility, the speed, and the large number of available logic gates render the FPGA a promising platform for the realization of network experiments with large network sizes and complex topologies.

B. Autonomous mode of operation

The mode of operation of the logic gates has important implications for the dynamics of the logic circuit on the FPGA. In most applications of FPGAs, they are used in the synchronous operation with a clock period slow enough so that all logic gates can settle to their Boolean states between two consecutive clock cycles.[13] Then, the logic gates behave in a digital fashion consistent with the Boolean algebra. In the autonomous mode of operation, in contrast, the logic circuit displays an analog dynamical evolution governed by the logic gates' propagation delays, gate activation function, and low-pass filtering characteristics.[27] These properties vary between the logic gates on a chip because of manufacturing imperfections. Consequently, two autonomous logic circuits of identical layout that are realized on different regions on the FPGA can display somewhat non-identical dynamics.

C. Design of time delay links and dynamical nodes

To build dynamical networks, we identify a circuit design for the dynamical nodes and the network topology using the built-in switching fabric of the FPGA. Direct links can be realized with on-chip wires that have a delay of a few tens of picoseconds, which can often be neglected compared to the propagation delay of logic gates $\tau_{\mathrm{LG}} = (280 \pm 10)$ ps (numeric values for the FPGA used in our experiments). Links with substantially longer time delay can also be realized by exploiting the finite propagation time of logic gates. Specifically, time delay links are built by cascading an even number of n_k inverter gates to achieve a time delay of $\tau_{n_k} = n_k\tau_{\mathrm{LG}}$. A delay line built from an even number of consecutive inverter (NOT) gates transmits the logic state of the input to the output. This construction can, however, alter the signal due to degradation effects.[28] In principle, cascaded buffer logic gates (executing the identity operation) can also be used to build a delay line, but they introduce larger degradation.[31]

Dynamical nodes are built by tailoring the autonomous logic circuit. For example, a unidirectional ring of an odd number of autonomous inverter gates—a ring oscillator[32]—generates periodic square-wave oscillations useful for clock generation[33,34] and for physical random number generation by exploiting the inherent jitter.[32,35] It is also possible to assemble logic gates executing other Boolean operations to achieve more complex dynamics such as chaos[27] and type-II excitability.[36]

D. Hardware description language for an autonomous Boolean network

In the following, we describe typical Verilog code to demonstrate the flexibility in realizing physical design with logic gates on an FPGA and creating networks. An ABN with periodic dynamics is introduced in Sec. III; its Verilog code reads

```
module my_osc(s_in,s_out);
 wire [20:0] delay /*synthesis keep*/;
 assign delay[0] = delay[20] | s_in;
 assign delay[1] = ~ delay[0];
   :      :      =     :     :
 assign delay[20] = ~ delay[19];
 assign s_out = delay[20];
endmodule
```

This module, called `my_osc`, describes an ABN with a closed unidirectional chain of inverter (NOT) gates and one OR gate. The NOT and OR logic operations are generated with the `~` and the `|` operators. The ABN has an input and output `s_in` and `s_out`, respectively. The directive `/*synthesis keep*/` (for Altera FPGAs) guarantees that all logic gates involved with the name delay are implemented by the compiler. Some logic gates would be redundant in synchronous operation and would, therefore, be removed by the compiler. To realize an experimental meta network with two of these ABNs, we consider a module main in which we call two instantiations of the periodic oscillators `osc1` and `osc2` as described by the following hardware description:

```
module main(out);
 assign out = net1;
 my_osc osc1(net2, net1);
 my_osc osc2(net1, net2);
endmodule
```

The output net1 of the oscillator called osc1 is input to oscillator `osc2`. The output `net2` of `osc2` is coupled back, realizing a network of two mutually coupled (here without time delays) dynamical nodes. The output port of the FPGA, called out, is connected to `net1`; hence, it will output the dynamics of the first network node. By extending the Verilog code for the main module by a few lines, networks of many nodes can be implemented easily. Note that the definitions of the variables are required at the beginning of the code and are omitted here. By compiling this high-level logical hardware description and loading it on the FPGA, we obtain a true physical (not emulated) network.

III. PERIODIC DYNAMICS IN AUTONOMOUS BOOLEAN NETWORKS

In this section, we show that periodic oscillators can be realized and coupled to form networks, thereby achieving phase synchronization. We adapt an existing ring oscillator to ensure unidirectional and bidirectional coupling and observe in-phase and anti-phase synchronization of the oscillators depending on the coupling time delay.

A. Periodic autonomous Boolean network: Ring oscillators

A schematic representation of a ring-oscillator design is shown in Fig. 1(a). It comprises one inverter gate subject to time-delayed feedback realized with n_k (even number) inverter logic gates. Not shown are output buffer gates through which the signals pass before they are recorded by an oscilloscope. The design of the ring oscillator prevents the existence of a Boolean fixed point that satisfies simultaneously all inverter logic gates. It displays periodic square-wave oscillations that correspond to a Boolean transition between V_L and V_H

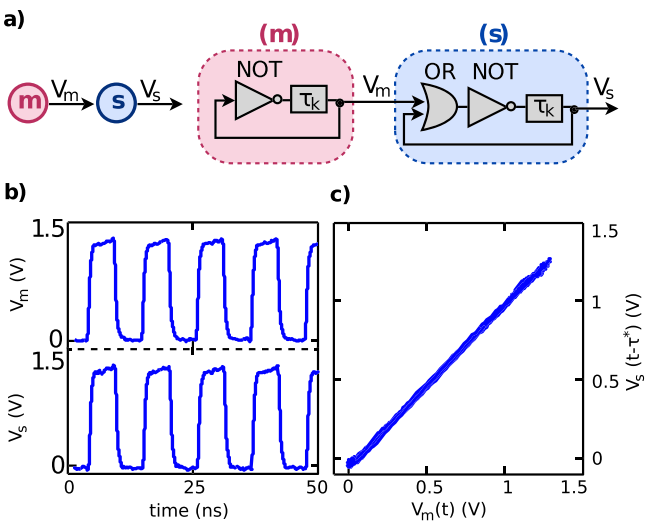

FIG. 1. Experimental demonstration of unidirectional synchronization of Boolean phase oscillators. (a) Illustration of the setup with master (m) and slave (s) oscillators. (b) Temporal evolutions of the oscillators (m) and (s) showing in-phase square-wave oscillations with period $T_{m,s} = 10.9 \pm 0.4$ ns. (c) Evolution in phase plane $(V_m(t), V_s(t-\tau^*))$. The time series are acquired with a high-speed oscilloscope (DSO80804A) with 8 GHz bandwidth and 40 GSa/s sampling rate.

propagating through the ring twice per period.[37] Its fundamental oscillation frequency is $f_k = 1/[2(n_k+1)\tau_{LG}]$.

Usually, ring oscillators are not designed to be coupled. However, a simple modification by the addition of an OR gate allows external Boolean transitions to be injected into the feedback loop.

Using this modified design, we couple two ring oscillators uni-directionally as shown in Fig. 1(a). They are both realized with an identical number of inverter logic gates $n_m + 1 = n_s + 1 = 21$; with frequencies $f_m = (92.1 \pm 0.9)$ MHz and $f_s = (87.5 \pm 1.2)$ MHz, respectively. Their frequencies differ because of the additional OR gate in (s) and heterogeneity in the propagation delay of the logic gates. As a result, the two oscillators are not frequency-locked without coupling. However, when the master oscillator (m) injects its output waveform into the slave oscillator (s), phase- and frequency-locking is achieved with frequency $f_m = f_s = (92.2 \pm 0.1)$ MHz, as illustrated in Fig. 1(b). Further confirmation of phase synchronization is given in Fig. 1, where the phase portrait $(V_m(t), V_s(t-\tau^*))$ shows a straight line with slope of one approximately. We measure the quality of synchronization by computing the cross-correlation coefficient between V_m and V_s, which is $\rho_{V_m V_s} \approx 0.995$. A skew time $\tau^* \approx 225$ ps is used to compensate for the additional propagation time of the OR gate, the difference in propagation time of the two signals to the output port of the FPGA to the oscilloscope, and for a small propagation delay in the coupling.

The stable phase-locked dynamics corresponds to one Boolean transition propagating in each oscillator with constant relative phase shift. The OR gate used in (s) leads to the creation of a Boolean transition in (s) whenever (m) generates a Boolean transition ($V_{H,L} \rightarrow V_{L,H}$) and (s) is in the V_L state. This implies that multiple transitions can potentially propagate in (s) if (m) and (s) are not phase locked. However, the most stable evolution for (s) has a single transition propagating; this results in $V_s(t)$ adjusting to $V_m(t-\tau^*)$. Using an OR gate for the coupling of ring oscillators prevents an accumulation of Boolean transitions in (s): if one of the two inputs is in V_H, then a Boolean transition ($V_{H,L} \rightarrow V_{L,H}$) in the other input has no influence on the output of the OR logic gate.

Interestingly, such a master-slave architecture realizes a very efficient, yet simple, phase-locked loop (PLL) architecture that does not require a voltage-controlled oscillator, a phase detector, or a complex digital design.[38]

B. Mutual phase synchronization of ring oscillators

With our modified ring architecture, we can also couple two ring oscillators bidirectionally, as illustrated in Fig. 2(a), with a flexible choice of the coupling time delays τ_{12} and τ_{21}.

When the coupling time delays are negligible $\tau_{12} \approx \tau_{21} \approx 0$ ns, the two oscillators are synchronized in phase, as shown in Fig. 2(b) with the frequency of each oscillator being slightly pulled from their respective free-running frequencies $f_1 = (81.9 \pm 0.7)$ MHz and $f_2 = (87.54 \pm 0.7)$ MHz to a common frequency $f = (87.7 \pm 0.7)$ MHz.

The synchronization patterns change when time delays along the links are included. The two ring oscillators display

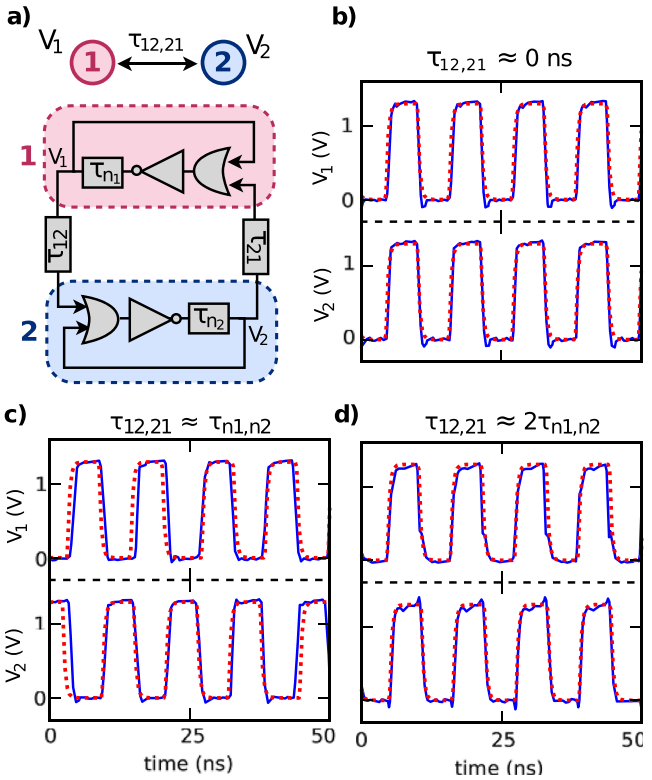

FIG. 2. Demonstration of bidirectional synchronization of two Boolean phase oscillators. (a) Boolean implementation of the two oscillators labeled (1) and (2) built with $n_1 = n_2 = 21$ inverter gates and coupled by two links with time delays $\tau_{12} \approx \tau_{21}$. (b) Temporal evolution of each Boolean oscillator showing in-phase square-wave oscillations with identical period $T_1 = T_2 = 10.7 \pm 0.4$ ns for $\tau_{ij} = 0$ ns. The time delay is due only to the on-chip wires connecting the two oscillators and can be neglected ($\tau_{12} \approx \tau_{21} \approx 0$). ((c)-(d)) Temporal evolution for the oscillators with $\tau_{12} \approx \tau_{21} \approx \tau_{n_1} \approx \tau_{n_2}$ ($\tau_{12} = 6.2$ ns, $\tau_{21} = 6.5$ ns) and $\tau_{12} \approx \tau_{21} \approx 2\tau_{n_1} \approx 2\tau_{n_2}$ ($\tau_{12} = 11.7$ ns, $\tau_{21} = 11.05$ ns), respectively. The blue solid lines show the experimental time series. The red dotted lines show the dynamics of x_{buf1} and x_{buf2} from numerical simulation of Eqs. (2)–(4) with $\tau_1 = \tau_2 = 5.4$ ns and mutual time delays τ_{12}, τ_{21} as stated above. The dimensionless quantities x_{buf1} and x_{buf2} are scaled in amplitude and time ($V_{1,2} \to x_{buf1,2} V_H$ and $t \to t T_{\text{rise}}/\ln(2)$, with $V_H = 1.3$ V and $T_{\text{rise}} = 0.26$ ns).

either in-phase or anti-phase synchronization depending on the coupling time delays τ_{12} and τ_{21} with respect to the period of the oscillators $T_1 \approx 2\tau_{n_1}$ and $T_2 \approx 2\tau_{n_2}$. After a series of experiments, we identified that, when $\tau_{12} \approx \tau_{21} \approx p\tau_{n_1} \approx p\tau_{n_2}$ with $p \in \mathbb{N}$ even (odd), the two oscillators are in-(anti-)phase synchronized. To illustrate this, the temporal evolution of each oscillator is shown for $\tau_{12} \approx \tau_{21} \approx \tau_{n_1} \approx \tau_{n_2}$ and $\tau_{12} \approx \tau_{21} \approx 2\tau_{n_1} \approx 2\tau_{n_2}$ in Figs. 2(c) and 2(d), respectively.

Interestingly, our experimental result on mutual synchronization is reminiscent of phase synchronization states predicted theoretically for two coupled Kuramoto oscillators with time-delay feedback loops and links.[39] In their study, however, the periodic oscillator can oscillate without the presence of time-delayed feedback, which is not the case for our Boolean phase oscillator—without the time-delayed feedback, our Boolean oscillator reduces to an OR and a NOT gate with a fixed Boolean state. Similar behavior has been observed numerically for two delay-coupled FitzHugh-Nagumo systems, each of which is in the excitable regime, i.e., does not exhibit self-sustained oscillations in the uncoupled case.[40,41]

In our experiments, the coupling mechanism is the same in both the unidirectional and bidirectional case but differs from typical Kuramoto oscillator, as it can be understood in terms of the exchange of Boolean transitions. The OR gate in each oscillator generates Boolean transitions when the signal of an external Boolean transition is received. Simultaneously, each oscillator maintains a single transition because of the stability associated with a single transition propagating in each ring and the low sensitivity of the OR gate.

C. Model for ring oscillators

The dynamics of the electronic logic gates are usually modeled from first principles using SPICE models.[42] Here, we describe a piecewise-linear switching model of our ABN adapted from previous models of genetic networks[29] and compare the theoretically predicted and experimentally observed dynamics.

The dynamics of the ABN are modeled by considering the continuous output variables of the nodes $x_i(t)$ and the associated Boolean states $X_i(t)$,

$$X_i(t) = 0 \text{ if } x_i(t) < x_{th}; \text{ otherwise } X_i(t) = 1, \tag{1}$$

with the low and high Boolean values 0 and 1 and threshold $x_{th} = 0.5$. Specifically, the network in Fig. 2, which consists of two OR gates with consecutive inverter gates and two delay lines composed of consecutive inverter gates, can be described as two inverted OR (NOR) Boolean functions with delayed feedback. We model the dynamics of this setup, following the formalism introduced by Glass *et al.*,[29] and extending it by time delays

$$\frac{dx_1}{dt} = -x_1 + \text{NOR}[X_1(t-\tau_1), X_2(t-\tau_{2,1}-\tau_2)], \tag{2}$$

$$\frac{dx_2}{dt} = -x_2 + \text{NOR}[X_2(t-\tau_2), X_1(t-\tau_{1,2}-\tau_1)], \tag{3}$$

$$\frac{dx_{buf1,2}}{dt} = -x_{buf1,2} + X_{1,2}(t), \tag{4}$$

where NOR: $\{0,1\} \times \{0,1\} \to \{0,1\}$ denotes the inverted OR operation on the Boolean states. The time delays originate from chains of consecutive inverter gates in the setup (τ_1, τ_2, τ_{12}, and τ_{21}). The third equation describes the temporal evolution of two buffer logic gates x_{buf1} and x_{buf2} that perform the Boolean identity operation on $X_1(t)$ and $X_2(t)$; the buffer gates correspond to output gates on the FPGA.

In Figs. 2(b)–2(d), the dotted red line denotes the solutions obtained from the model for x_{buf1} and x_{buf2} by evolving the analytical solution of the piecewise linear differential equations between the switching of the NOR Boolean function, similar to Ref. 29. Apart from a low level of amplitude noise in the experiment, the dynamics generated by the model agrees well with the experiment. Both display waveforms with an exponential approach to the Boolean states, similar rise times, and similar periodicity of the oscillations.

The discrepancy between model and experiment can be quantified via differences in timing of transitions, which is a common measure in autonomous Boolean systems,[27] and amounts to average values of 0.20 ns, 0.94 ns, and 0.49 ns for the waveforms in Figs. 2(b)–2(d), respectively. The error is small in comparison to the oscillation period of $T = 10.7 \pm 0.4$ ns.

D. Discussion

Above in this section, we show that FPGAs are well suited to realize coupled dynamical systems with periodic dynamics. We assemble the periodic oscillators in simple network motifs and observe phase synchronization. Our experiments display interesting features that are similar to general theoretical predictions of coupled phase oscillators.[43]

Our approach is scalable to larger network sizes and nodes of higher in-degree. For example, before injecting the input signal into the oscillator, another logic gate with multiple inputs can be used to combine and pre-process multiple input signals from the neighboring nodes.

A limitation of our current approach, however, is the lack of control of the coupling strength. In our design, the coupling is either on or off. In ongoing research, we are developing an autonomous logic circuit to allow for an adjustable coupling strength so that we can test the various theoretical predictions involving a variation in the coupling strength, such as chimera states[44–46] and waves on networks.[47]

IV. CHAOTIC DYNAMICS IN AUTONOMOUS BOOLEAN NETWORKS

In addition to periodic oscillations, ABNs can display chaos for topologies with multiple loops and multiple-input logic functions.[24,27,29,48] For example, Cavalcante *et al.*[28] showed that chaos emerges in an ABN of two XOR and one XNOR logic gates with links of incommensurate time delays.

In this section, we show that an ABN with a simple topology composed of an XNOR logic gate with three delayed feedback lines also displays chaotic dynamics depending on the time delay of the feedback lines.

A. Realization of a small ABN with complex dynamics

Our design of a chaotic dynamical system is motivated by a study of Ghil *et al.*,[24] who demonstrated that complex dynamics can emerge in a feedback system comprising one XOR logic gate with two delayed feedback lines of incommensurate time delays τ_{n_k,n_l} as shown in Fig. 3(a). Theoretical analysis predicts a power-law increase in time of the number of Boolean transition in this circuit. Boolean transitions in the output of the XOR gate are fed back to its input via the two incommensurate delay lines and they trigger new Boolean transitions. In fact, any single change of the input of an XOR logic function leads to a change of the output value, which is called maximum Boolean sensitivity.[49]

This situation, however, cannot occur in our experimental ABN because of the finite bandwidth of logic gates that limits the rate of Boolean transitions.[27,28] The low-pass filter effect erases transitions that are too close to each other, a phenomenon called *short-pulse rejection*. Instead, Boolean transitions appear at unpredictable time: A dynamical state referred to as *Boolean chaos*.[27]

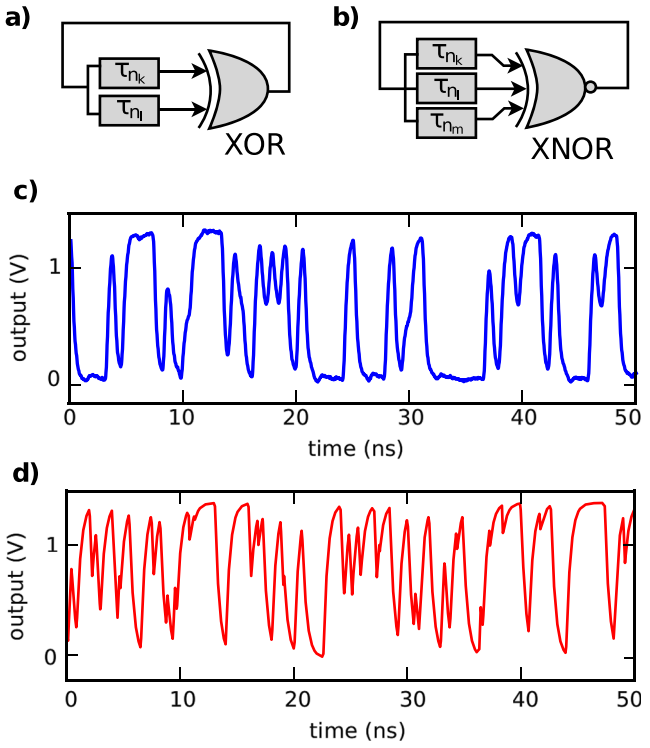

FIG. 3. Experimental demonstration of chaotic dynamics. (a) ABN made of one two-input XOR gate with two time delays τ_{n_k,n_l}, as proposed by Ghil *et al.*[24] (b) ABN made of one three-input XNOR gate with three delayed feedback lines. This ABN does not have a Boolean fixed point because $\mathrm{XNOR}(1,1,1) = 0$ and $\mathrm{XNOR}(0,0,0) = 1$. Not shown are output buffer gates that the signal is routed through prior to the acquisition. (c) Chaotic dynamics of the circuit shown in (b) for $\tau_{n_k} = (2.8 \pm 0.1)$ ns, $\tau_{n_l} = (1.7 \pm 0.1)$ ns, and $\tau_{n_m} = (0.56 \pm 0.02)$ ns ($n_k = 10$, $n_l = 6$, and $n_m = 2$). (d) Numerical simulation of Eqs. (5) and (6) with parameters $\tau_1 = 3.11$ ns, $\tau_2 = 1.73$ ns, and $\tau_3 = 0.597$ ns. The dimensionless quantity x_{buf} is scaled in amplitude and time ($V \to xV_H$ and $t \to tT_{\mathrm{rise}}/\ln(2)$, with $V_H = 1.3$ V and $T_{\mathrm{rise}} = 0.26$ ns).

When realized with electronic logic gates, Ghil's network relaxes to a Boolean fixed state that satisfies the XOR input-output relationship $V_{\mathrm{in}} = V_{\mathrm{out}} = V_{\mathrm{L}}$. This fixed point is always reached after a transient time regardless of the initial conditions and combinations of time delays that we have tested. To observe other dynamics, we need to design a similar network without a Boolean fixed point. For this, we use an XNOR gate (instead of an XOR) and three delayed feedback links, as shown in Fig. 3. The generalization of the XNOR logic operation to more than two inputs corresponds to the inverted parity operation on the Boolean input states.

When implemented on the FPGA, this ABN can display Boolean chaos for a range of values of time delays for each of the three feedback links. Chaotic dynamics is shown in Fig. 3(c) for feedback links with delays of $\tau_{n_k} = (2.8 \pm 0.1)$ ns, $\tau_{n_l} = (1.7 \pm 0.1)$ ns, and $\tau_{n_m} = (0.56 \pm 0.02)$ ns (corresponding to $n_k = 10$, $n_l = 6$, and $n_m = 2$ inverter gates, respectively). There is strong evidence of the chaotic nature of the waveforms because the mechanism of the generation

of Boolean transitions is similar to that of an ABN proven to be chaotic by Zhang *et al.*[27] Other evidence is given by the measured fast-decaying and quasi-unstructured autocorrelation function. However, for a rigorous proof of the chaotic nature, the waveforms should be analyzed more carefully.

B. Model for the chaotic oscillator

Similar to our considerations in Sec. III C, we model the dynamics of the ABN with delay differential equations that switch between two piecewise linear right hand sides

$$\frac{dx}{dt} = -x + \mathrm{XNOR}[X(t-\tau_1), X(t-\tau_2), X(t-\tau_3)], \quad (5)$$

$$\frac{dx_{buf}}{dt} = -x_{buf} + X(t), \quad (6)$$

where $x(t)$ and $X(t)$ denote the continuous and Boolean state of the XNOR logic gate, respectively, XNOR: $\{0,1\} \times \{0,1\} \times \{0,1\} \rightarrow \{0,1\}$ denotes the inverted XOR operation on three Boolean states and the time delays originate from the consecutive inverter gates in the setup. The second equation describes, similar to our consideration in Eq. (4), the temporal evolution of a buffer logic gate x_{buf}.

Figure 3(d) shows the dynamics obtained from the model for $x_{buf}(t)$, using similar time delays as in the experiment and a threshold voltage of $x_{th} = 0.46$ in Eq. (1). The numerical dynamics of $x_{buf}(t)$ can be compared to the experimental dynamics in Fig. 3(c). Similar features in the two waveforms, such as irregular timing of transitions, can be seen. However, the numerical simulation displays a higher rate of transitions in comparison to the experimental waveform, which returns to the Boolean states for a longer time. In addition, the simulation displays chaos only for narrow ranges of the feedback delays and threshold voltage x_{th}, whereas the experiment shows chaotic dynamics consistently for large enough time delays. The differences between experiment and simulations could be due to state-dependency of the feedback delays in the experiment[28] and other non-ideal behaviors not captured by the simplified model.

C. Transition to chaos

The network displays chaos only when the time delays of the feedback are sufficiently large. In this section, we keep the two time delays $\tau_{n_l} = (1.7 \pm 0.1)$ ns and $\tau_{n_m} = (0.56 \pm 0.02)$ ns fixed and only vary the value of τ_{n_k}. For short time delay τ_{n_k}, however, regular dynamics is observed. A feedback link of a direct on-chip wire with only a few tens of picoseconds delay leads to a steady-state dynamics. This is because two transitions propagating through that short feedback link in the network will have a frequency that is higher than the cut-off of the low-pass filter of the logic gates. Therefore, the ABN cannot generate Boolean transitions without falling into the previous scenario, thereby only producing a constant voltage at a value that is in between the two Boolean voltage levels, as shown in Fig. 4(a). With a short value of the time delay, the threshold value V_{th} of the XNOR gate is stabilized.

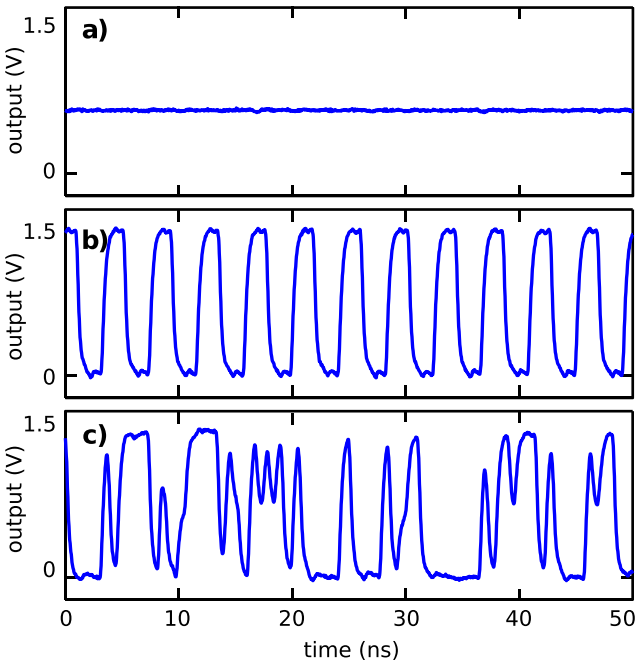

FIG. 4. Dynamics of the oscillator shown in Fig. 3 for feedback lines built from fixed numbers of inverter gates $n_l = 6$, $n_m = 2$, corresponding to $\tau_{n_l} = (1.7 \pm 0.1)$ ns, $\tau_{n_m} = (0.56 \pm 0.02)$ ns, and different numbers n_k. (a) Fixed point with voltage value between the two Boolean voltages $V_{\mathrm{L,H}}$ for $n_k = 0$ corresponding to τ_{n_k} of a few picoseconds. (b) Oscillatory dynamics with period $T = 4.2 \pm 0.3$ ns for $n_k = 2$ [$\tau_{n_k} = (1.7 \pm 0.1)$ ns]. (c) Chaotic dynamics for $n_k = 10$ [$\tau_{n_k} = (2.8 \pm 0.1)$ ns].

A feedback link of $n_k = 2$ inverters, corresponding to a time delay of $\tau_{n_k} = (0.56 \pm 0.02)$ ns, leads to periodic oscillations, as shown in Fig. 4(b). For this value of the time delay, the longest feedback loop is large enough to allow for a single Boolean transition to propagate, while additional transitions cannot be generated by the XNOR gate.

For larger values of n_k, the system can display complex dynamics such as quasi-periodicity or periodic oscillations with multiple harmonics. However, the observation of these dynamics depends heavily on the experimental conditions and their existence or the waveform properties may vary significantly when the system is moved to different locations on the FPGA.

Finally, when the number of inverter gates used to realize the feedback time delay reaches or exceeds a threshold value $n_k = 10$, the ABN displays Boolean chaos. A measurement of the autocorrelation function calculated from the ABN time series reveals a correlation time of 650 ps, and the autocorrelation function decays almost to zero for a lag time greater than 100 ns. The threshold value n_k at which the network displays chaos is sensitive to the specific placement of the logic circuit on the FPGA.

D. Discussion

We demonstrate in this section that ABNs realized on an FPGA can display chaotic dynamics. In agreement with previous studies, our experiments show that the non-ideal behavior of electronic logic gates plays an important role for the dynamics of ABNs.[27,28] Short time delays of the

feedback loops lead to fixed points and stable periodic dynamics. Longer time delays lead to chaos.

As shown in Sec. III, it is possible to couple ABNs with periodic oscillatory dynamics to meta-networks and observe synchronization phenomena. However, realizing similar experiments with chaotic ABNs is difficult: Inhomogeneities and inconsistencies in the autonomous mode of operation of logic gates (propagation delays, low-pass filter characteristics, and electronic noise) result in significant parameter mismatch when implementing multiple copies of chaotic oscillators on the same FPGA. Consequently, chaos synchronization has not yet been achieved in our experiments. Nevertheless, Boolean chaos in ABNs has several applications. For example, it has already been used for ultra-high-speed random number generation[48] and is also promising for chaos-based radar applications.[50,51]

V. EXCITABLE DYNAMICS IN AUTONOMOUS BOOLEAN NETWORKS

In this section, we demonstrate that ABNs can be designed to exhibit excitable dynamics as an artificial neuron and, when connected to a meta-network, they constitute an artificial neural network. We previously used this approach to build small neural-like networks,[36,52] and here we show that we can implement larger networks with random topologies, community structures, and large in-degree of nodes. Building such systems can be useful for understanding large-scale properties of neural systems and for building ultra-fast neuromorphic systems.[53]

A. Realization of an ABN with excitable dynamics

Excitability is a property of dynamical systems that generate large excursions in phase space (spikes) in response to small perturbations above a threshold, the stimulus. Such dynamics is often detected for neurons. Another feature of excitable systems is the refractory phase of duration T_{ref}, where the excitable system cannot respond to stimuli. We implement excitable systems with these two characteristic features using autonomous logic gates on an FPGA.

An excitable system based on an ABN is shown schematically in Fig. 5. The excitable node consists of two pulse generators (PGs), which are autonomous logic circuits that generate pulses of constant width in response to a Boolean or continuous voltage that exceeds the threshold voltage of logic gates. The two PGs generate the output voltage of the excitable system V_{out} and the refractory voltage V_{ref}, which indicates the refractory phase.

We have shown by experiments and numerical simulations[36] of the excitable nodes that these two fundamental properties of neural systems (the pulse generation for above-threshold inputs and the refractory period) leads to basic excitable dynamics that reproduces dynamics of neural networks, such as cluster synchronization, that have been observed previously in complex neuronal models.[52]

The refractory mechanism is implemented with an AND gate that receives inputs from an external input voltage V_{in} and V_{ref}. When the system is (not) in the refractory phase, indicated by $V_{\text{ref}} = V_H$ ($V_{\text{ref}} = V_L$), the AND gate prevents

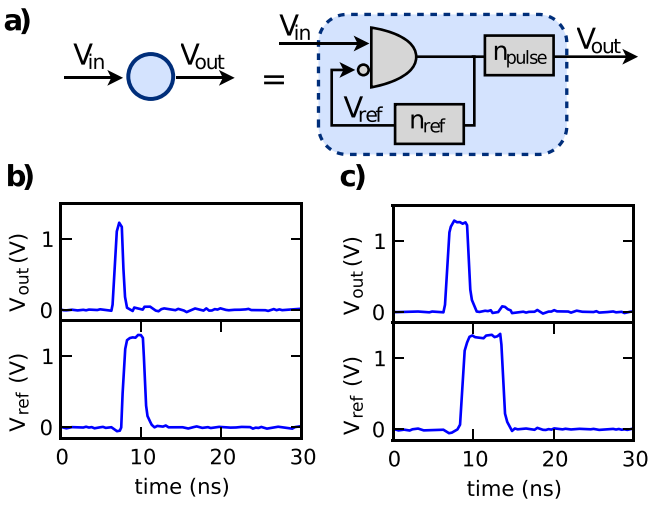

FIG. 5. Excitable node using an ABN. (a) Scheme of a single excitable node composed of two pulse generators labeled by integers $n_{\text{pulse,ref}}$ characterizing the pulse width and refractory period: $T_{\text{pulse,ref}} = n_{\text{pulse,ref}}\tau_{\text{LG}}$. ((b)–(c)) Temporal evolution of each pulse generator after a single stimulus of the excitable node with (b) $n_{\text{pulse}} = 4$ [$T_{\text{pulse}} = (1.12 \pm 0.04)$ ns] and $n_{\text{ref}} = 10$ [$T_{\text{ref}} = (2.8 \pm 0.1)$ ns] and (c) $n_{\text{pulse}} = 10$ [$T_{\text{pulse}} = (2.8 \pm 0.1)$ ns] and $n_{\text{ref}} = 20$ [$T_{\text{ref}} = (5.6 \pm 0.2)$ ns].

(allows for) an external stimulus to activate the PGs to generate output pulses V_{out} and to excite the node. We can adjust the refractory period of the excitable node by changing the width of the pulse in V_{ref}.[36] We implement an excitable Boolean node on the FPGA and we observe its dynamics in response to a single input stimulus. As shown in Figs. 5(b) and 5(c), the excitable node generates an output and a refractory signal with pulse widths $T_{\text{pulse}} = (1.12 \pm 0.06)$ ns, $T_{\text{ref}} = (2.8 \pm 0.1)$ ns and $T_{\text{pulse}} = (2.8 \pm 0.1)$ ns, $T_{\text{ref}} = (5.6 \pm 0.2)$ ns, respectively.

Our excitable node has only a single input V_{in}. However, in biological neural networks, the in-degree of nodes can be much higher than unity. Therefore, we add another autonomous gate that integrates and combines various incoming signals into a single stimulus. In the literature, such a pre-processing unit is referred to as the synapse of the artificial neuron.[53] Here, we use an OR gate—similar to our experiments in Sec. III.

B. Dynamics of neural-like networks of Boolean excitable nodes

The flexibility of the FPGA allows us to, for example, duplicate excitable Boolean nodes and assemble them in a network of four distinct neural populations, as illustrated in Figs. 6(a) and 6(b). Dynamical properties of similar networks of excitable systems with community structure have been investigated theoretically[54,55] because of their relevance in analyzing neural circuits such as the thalamic circuitry embedded in the brain.[56,57]

In our experiment, each population consists of 20 excitable nodes, totaling 80 nodes for the entire network. The links within a population are connected with on-chip wires on the FPGA so that the associated link time delays are small. The nodes within a population are randomly connected with probability $p = 0.3$. Between the populations, the links are

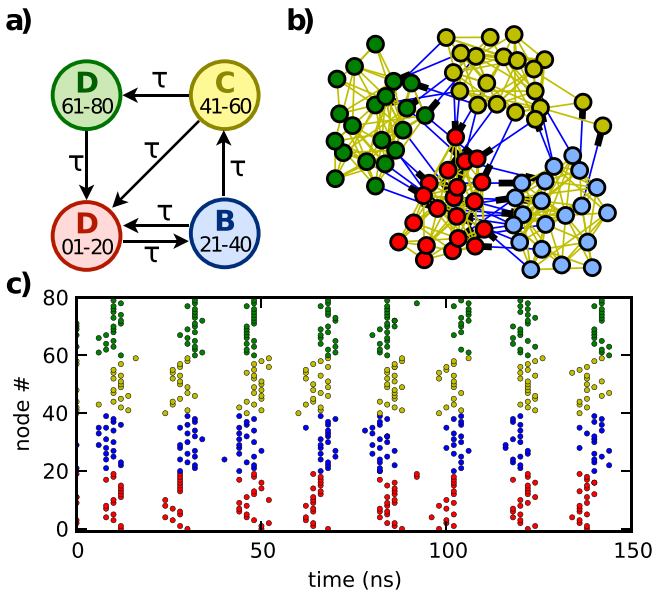

FIG. 6. Experimental demonstration of excitable dynamics. (a) Topology of four coupled populations that involves loops of four, three, and two elements. (b) Implemented topology, where nodes of the same (different) populations are connected with directed links with probability $P_{\text{intra}} = 30\%$ ($P_{\text{inter}} = 1.5\%$). Nodes within a population are strongly coupled with negligible link delay and nodes of different populations are loosely coupled with significant time delay τ. An initial pulse is sent to one node to perturb the network out of its quiescent state. (c) Raster diagram of the network for $n_\tau = 60$ [$\tau = (16.8 \pm 0.6)$ ns], $n_{\text{pulse}} = 4$ [$T_{\text{pulse}} = (1.12 \pm 0.04)$ ns], and (b) $n_{\text{ref}} = 20$ [$T_{\text{ref}} = (5.6 \pm 0.2)$ ns].

realized with delay lines as defined in Sec. II with value $\tau = (16.8 \pm 0.6)$ ns and probability of connection $p = 0.015$.

The dynamics of our artificial neural network is described theoretically by Kanter *et al.*[58,59] According to the theory, the network dynamics is given by the network topology of the community structure by the greatest common divisor (GCD) of the sizes of directed loops. In the network topology in Fig. 6, inspired by Fig. 1(a) in Ref. 58, there are three directed loops of two, three, and four neural populations, respectively. Therefore, the theory predicts a number of zero-lag synchronized clusters of $\text{GCD}(2, 3, 4) = 1$, i.e., all the populations are predicted to be synchronized with zero time lag.

The experimental dynamics of the network is reported in the raster diagram of Fig. 6(c), where each circle corresponds to a pulse generated by a node. The dynamics of the 80 nodes is acquired with an integrated measurement system based on a processing unit on the FPGA with a timing resolution of $\sim$2 ns. When a pulse is generated by an artificial neuron, its time is recorded and stored on the FPGA on-chip memory.

We observe that all the artificial neurons of the four populations generate pulse trains with period $\tau \pm \Delta\tau$ with $\Delta\tau = 5$ ns. The dispersion $\Delta\tau$ in the period of the pulse trains originates from heterogeneities in the values of the link time delays and limited resolution of our integrated measurement system. The experimental network is considered to be in a near-zero-lag single synchronization cluster, which is consistent with the theoretical predictions. This experimental confirmation suggests that our Boolean excitable nodes can be used generically to observe other collective phenomena in artificial neural networks.

C. Discussion

In this section, we measure synchronized neural activity in a neural network of 80 excitable nodes. We show that ABNs can be used to build large meta-networks of neural excitable dynamics. Various synchronization patterns and more general dynamics are expected for high in-degrees of nodes and for a different choice of the synapses than an OR gate. For example, the flexibility of the logic function will allow for implementation of inhibiting connections.

Besides potential insights into neurodynamics, our excitable Boolean node may become invaluable for neuro-inspired computing, such as reservoir computing,[60] especially because the nanosecond time-scale of the dynamics will allow for fast processing rates.

VI. CONCLUSION

We demonstrate that an FPGA is a versatile experimental platform to conduct integrated experiments on the dynamics of complex networks. When assembling logic gates in their autonomous mode of operation, one can create ABNs that display rich and complex dynamics, such as periodic oscillations, chaos, excitable dynamics. The ABNs with these dynamics can be further coupled with time-delay links to form autonomous Boolean meta-networks that are used to conduct experiments on collective phenomena.

We propose Boolean analogies of three paradigmatic configurations arising in nonlinear dynamics: (i) phase synchronization in simple network motifs of oscillatory systems, (ii) chaotic dynamics, and (iii) synchronization phenomena in networks of excitable systems. These three sets of experiments pave the way towards filling the gap between the theory of dynamic networks and desirable experiments, since our approach allows for the realization of large networks with arbitrary topologies on an FPGA.

Nevertheless, our approach still presents many technical and scientific challenges. For example, the experimental extraction of data from each node is only partially solved for networks of excitable systems using the data acquisition capabilities of FPGAs. The greatest challenge in using FPGAs for network experiments is to find the Boolean analogy for the desired dynamical node and the coupling while satisfying technological constraints imposed by the FPGA platform.

ACKNOWLEDGMENTS

D.P.R., D.R., and D.J.G. gratefully acknowledge the financial support of the U.S. Army Research Office Grants W911NF-11-1-0451 and W911NF-12-1-0099. E.S. and D.P.R. acknowledge support by the DFG in the framework of SFB 910. We would like to express our thanks to Leon Glass for insightful discussions on the theoretical model.

[1]S. H. Strogatz, Nature **410**, 268 (2001).
[2]M. E. J. Newman, *Networks: An Introduction* (Oxford University Press, Inc., New York, NY, 2010).

[3]A.-L. Barabási and Z. N. Oltvai, Nat. Rev. Genet. **5**, 101 (2004).
[4]H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási, Nature **407**, 651 (2000).
[5]J. M. Montoya, S. L. Pimm, and R. V. Solé, Nature **442**, 259 (2006).
[6]E. d. Bullmore and O. Sporns, Nat. Rev. Neurosci. **10**, 186 (2009).
[7]J. P. Onnela, J. Saramäki, J. Hyvönen, G. Szabó, D. Lazer, K. Kaski, J. Kertész, and A.-L. Barabási, Proc. Natl. Acad. Sci. U.S.A. **104**, 7332 (2007).
[8]M. E. J. Newman, Proc. Natl. Acad. Sci. U.S.A. **98**, 404 (2001).
[9]M. Nixon, M. Fridman, E. Ronen, A. Friesem, N. Davidson, and I. Kanter, Phys. Rev. Lett. **108**, 214101 (2012).
[10]A. A. Temirbayev, Z. Z. Zhanabaev, S. B. Tarasov, V. I. Ponomarenko, and M. Rosenblum, Phys. Rev. E **85**, 015204 (2012).
[11]A. M. Hagerstrom, T. E. Murphy, R. Roy, P. Hövel, P. I. Omelchenko, and E. Schöll, Nat. Phys. **8**, 658 (2012).
[12]M. R. Tinsley, S. Nkomo, and K. Showalter, Nat. Phys. **8**, 662 (2012).
[13]S. Brown and Z. Vranesic, *Fundamentals of Digital Logic with Verilog Design* (Mc Graw Hill, New York, NY, 2008).
[14]F. Jacob and J. Monod, J. Mol. Biol. **3**, 318 (1961).
[15]S. Kauffman, C. Peterson, B. Samuelsson, and C. Troein, Proc. Natl. Acad. Sci. U.S.A. **100**, 14796 (2003).
[16]D. Sahoo, J. Seita, D. Bhattacharya, M. A. Inlay, I. L. Weissman, S. K. Plevritis, and D. L. Dill, Proc. Natl. Acad. Sci. U.S.A. **107**, 5732 (2010).
[17]F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang, Proc. Natl. Acad. Sci. U.S.A. **101**, 4781 (2004).
[18]S. Bornholdt and K. Sneppen, Phys. Rev. Lett. **84**, 6114 (2000).
[19]M. Paczuski, K. E. Bassler, and A. Corral, Phys. Rev. Lett. **84**, 3185 (2000).
[20]S. A. Kauffman, *The origins of Order: Self Organization and Selection in Evolution* (Oxford University Press, New York, NY, 1993).
[21]F. Greil and B. Drossel, Phys. Rev. Lett. **95**, 048701 (2005).
[22]A. Pomerance, E. Ott, M. Girvan, and W. Losert, Proc. Natl. Acad. Sci. U.S.A. **106**, 8209 (2009).
[23]T. Mestl, R. J. Bagley, and L. Glass, Phys. Rev. Lett. **79**, 653 (1997).
[24]M. Ghil and A. Mullhaupt, J. Stat. Phys. **41**, 125 (1985).
[25]M. Ghil, I. Zaliapin, and B. Coluzzi, Physica D **237**, 2967 (2008).
[26]T. Mestl, C. Lemay, and L. Glass, Physica D **98**, 33 (1996).
[27]R. Zhang, H. L. D. de S. Cavalcante, Z. Gao, D. J. Gauthier, J. E. S. Socolar, M. M. Adams, and D. P. Lathrop, Phys. Rev. E **80**, 045202 (2009).
[28]H. L. D. de S. Cavalcante, D. J. Gauthier, J. E. S. Socolar, and R. Zhang, Philos. Trans. R. Soc. London, Ser. A **368**, 495 (2010).
[29]L. Glass and C. Hill, Europhys. Lett. **41**, 599 (1998).
[30]L. Glass, T. J. Perkins, J. Mason, H. T. Siegelmann, and R. Edwards, J. Stat. Phys. **121**, 969 (2005).
[31]M. Bellido-Diaz, J. Juan-Chico, A. Acosta, M. Valencia, and J. Huertas, in *IEE Proc.-Circuits Devices Syst.* (IET, 2000), Vol. 147, pp. 107–117.
[32]A. Hajimiri and T. H. Lee, IEEE J. Solid-State Circuits **33**, 179 (1998).
[33]J. Alvarez, H. Sanchez, G. Gerosa, and R. Countryman, IEEE J. Solid-State Circuits **30**, 383 (1995).
[34]L. Sun and T. A. Kwasniewski, IEEE J. Solid-State Circuits **36**, 910 (2001).
[35]B. Sunar, W. J. Martin, and D. R. Stinson, IEEE Trans. Comput. **56**, 109 (2007).
[36]D. P. Rosin, D. Rontani, D. J. Gauthier, and E. Schöll, Europhys. Lett. **100**, 30003 (2012).
[37]H. Kato, IEEE Trans. Circuits Syst., I: Fundam. Theory Appl. **45**, 98 (1998).
[38]D. H. Wolaver, *Phase-Locked Loop Design Circuit* (Prentice Hall, Upper Saddle River, NJ, 1991).
[39]O. D'Huys, R. Vicente, T. Erneux, J. Danckaert, and I. Fischer, Chaos **18**, 037116 (2008).
[40]E. Schöll, G. Hiller, P. Hövel, and M. A. Dahlem, Philos. Trans. R. Soc. London, Ser. A **367**, 1079 (2009).
[41]A. Panchuk, D. P. Rosin, P. Hövel, and E. Schöll, Int. J. Bifurcation Chaos (to be published); e-print arXiv:1206.0789.
[42]A. Vladimirescu, *The SPICE Book* (John Wiley & Sons, Inc., Hoboken, NJ, 1994).
[43]A. S. Pikovsky, M. G. Rosenblum, and J. Kurths, *Synchronization, A Universal Concept in Nonlinear Sciences* (Cambridge University Press, Cambridge, UK, 2001).
[44]Y. Kuramoto and D. Battogtokh, Nonlinear Phenom. Complex Syst. **5**, 380 (2002).
[45]D. M. Abrams and S. H. Strogatz, Phys. Rev. Lett. **93**, 174102 (2004).
[46]I. Omelchenko, Y. L. Maistrenko, P. Hövel, and E. Schöll, Phys. Rev. Lett. **106**, 234102 (2011).
[47]S.-i. Shima and Y. Kuramoto, Phys. Rev. E **69**, 036213 (2004).
[48]D. P. Rosin, D. Rontani, and D. J. Gauthier, Phys. Rev. E **87**, 040902(R) (2013).
[49]I. Shmulevich and S. Kauffman, Phys. Rev. Lett. **93**, 048701 (2004).
[50]M. I. Sobhy and A. R. Shehata, in *IEEE MTT-S Int. Microwave Symp. Dig.* (IEEE, 2000), Vol. 3, pp. 1701–1704.
[51]F.-Y. Lin and J.-M. Liu, IEEE J. Quantum Electron. **40**, 815 (2004).
[52]D. P. Rosin, D. Rontani, D. J. Gauthier, and E. Schöll, Phys. Rev. Lett. **110**, 104102 (2013).
[53]G. Indiveri, B. Linares-Barranco, T. J. Hamilton, A. van Schaik, R. Etienne-Cummings, T. Delbruck, S. C. Liu, P. Dudek, P. Häfliger, S. Renaud, J. Schemmel, G. Cauwenberghs, J. Arthur, K. Hynna, F. Folowosele, S. Saighi, T. Serrano-Gotarredona, J. Wijekoon, Y. Wang, and K. Boahen, Front. Neurosci. **5**, 73 (2011).
[54]R. Vicente, L. Gollo, C. R. Mirasso, I. Fischer, and G. Pipa, Proc. Natl. Acad. Sci. U.S.A. **105**, 17157 (2008).
[55]E. Kopelowitz, M. Abeles, D. Cohen, and I. Kanter, Phys. Rev. E **85**, 051902 (2012).
[56]E. G. Jones, Philos. Trans. R. Soc. London, Ser. B **357**, 1659 (2002).
[57]S. Shipp, Philos. Trans. R. Soc. London, Ser. B **358**, 1605 (2003).
[58]I. Kanter, E. Kopelowitz, R. Vardi, M. Zigzag, W. Kinzel, M. Abeles, and D. Cohen, Europhys. Lett. **93**, 66001 (2011).
[59]R. Vardi, A. Wallach, E. Kopelowitz, M. Abeles, S. Marom, and I. Kanter, Europhys. Lett. **97**, 066002 (2012).
[60]H. Jaeger and H. Haas, Science **304**, 78 (2004).