Evolving complex dynamics in electronic models of genetic networks

Jonathan Mason
Center for BioDynamics and Department of Biomedical Engineering, Boston University, Boston, Massachusetts 02215

Paul S. Linsay
Superior Methods, Inc., Brookline, New Hampshire 02459

J. J. Collins
Center for BioDynamics and Department of Biomedical Engineering, Boston University, Boston, Massachusetts 02215

Leon Glass[a)]
Department of Physiology, McGill University, Montreal, Quebec H3G 1Y6, Canada

Ordinary differential equations are often used to model the dynamics and interactions in genetic networks. In one particularly simple class of models, the model genes control the production rates of products of other genes by a logical function, resulting in piecewise linear differential equations. In this article, we construct and analyze an electronic circuit that models this class of piecewise linear equations. This circuit combines CMOS logic and RC circuits to model the logical control of the increase and decay of protein concentrations in genetic networks. We use these electronic networks to study the evolution of limit cycle dynamics. By mutating the truth tables giving the logical functions for these networks, we evolve the networks to obtain limit cycle oscillations of desired period. We also investigate the fitness landscapes of our networks to determine the optimal mutation rate for evolution. © *2004 American Institute of Physics.* [DOI: 10.1063/1.1786683]

Networks giving rise to complex dynamics exist in a wide range of physical, biological, and engineered systems. Recent studies have focused on the structure of such networks, and examined how the structure is linked to functional properties such as robustness and error tolerance. In general, however, a theory to predict the dynamics based on network structure is lacking, and consequently, it is often unclear what structural architecture is needed to produce desired dynamics. Here we show that networks with desired complex dynamics can be obtained by evolving their structure rather than by designing it from the outset. We construct and experimentally analyze an electronic circuit that is based on a class of ordinary differential equations that model genetic networks. Networks in this system can display a variety of dynamics, including steady states, limit cycles, and chaos. Here we focus on limit cycles and show that it is possible to evolve networks that display stable oscillations of a specified cycle length. By analyzing the fitness landscape, we demonstrate that there is an optimal evolution rate for obtaining such dynamics. This work shows how mutations in model gene networks can lead to the evolution of dynamic behaviors.

I. INTRODUCTION

Research carried out by Jacob and Monod in the early 1960s provided early insights into the regulation of the activities of genes. They discovered that specialized protein molecules, called transcription factors, could bind directly to DNA thereby regulating the activity of regions of DNA proximate to the binding site of the protein. Since the DNA carries the code for the structure of proteins, products from one DNA site could affect the activity at another DNA site, thereby leading to a network of genes interacting through protein intermediaries. In an early paper, Jacob and Monod outlined simple genetic control circuits that they imagined could underlie biological processes associated with multistability and oscillation.[1] Shortly after this seminal work, mathematical models of genetic control networks were developed in which the ''on–off'' dynamics of genes could be modeled by networks of Boolean logical devices that updated at discrete times.[2–6]

The notion that the regulation of gene activity can be modeled using logical functions has persisted to the present.[7–11] Further developing these ideas, recent work has demonstrated that different logical functions can be combinatorially synthesized in bacteria,[12] directed evolution can be used to generate a genetic circuit that acts as an inverter,[13] and that the binding of transcription factors to DNA is ideally suited to generate modular and evolvable transcriptional control.[14] Finally, following up on Jacob and Monod's early proposals, genetic circuits in bacteria have now been designed and synthesized that show simple dynamic behaviors including bistability[15–17] and oscillations.[18]

The above papers provide a rationale for studying networks of genes that display switchlike behavior. However, since there is no evidence of clocking devices that update states of networks at discrete times in genetic networks, we

---

[a)]Electronic mail: glass@cnd.mcgill.ca

believe it is more appropriate to study dynamics in differential equations representing networks in which time is continuous but the activities of the elements of the networks are regulated using logical functions.[19–25] In this formulation, the concentrations of the protein transcription factors increase or decrease exponentially, but the target genes that are under control switch between two states. In the ''on'' state, a gene is activated leading to the synthesis of the protein for which it codes, and in the ''off'' state, the gene is inactive and the protein for which it codes decays towards zero concentration. A given gene is turned on or off, depending on the logical function that controls it, and whether the transcription factors that regulate it are above or below threshold levels.

The ways in which real genetic control networks evolve is not understood. The current work shows how the modifications in the rules controlling gene expression can be used to seek a desired dynamics, even though there is no theory to predict what structural architecture is needed to produce the desired dynamics. Many recent papers have analyzed evolution in a variety of systems including electronic networks,[26,27] mathematical models of genes,[28] computer models of genes,[29–31] computer models of gene networks,[32] complex networks,[33,34] and genetic circuits in bacteria.[13] Our work is complementary to these studies in that we use a localized random search method to find and explore novel dynamical behaviors, and to study how the properties of these behaviors change as a network becomes progressively modified.

In this work, rather than analyze dynamics in a theoretical model, we have chosen to study dynamics in a hybrid digital–analog system that models the differential equations. We have two main reasons for doing this. First, in the electronic system there are necessarily small amounts of noise, and consequently all observed dynamics will be robust to small perturbations in the circuit. Second, we believe that the class of circuits we consider has intrinsic interest, and at some stage might lead to novel ways to build oscillators displaying robust dynamically different nonlinear oscillations. Our emphasis on the design of real circuits, also places the current work in the area of evolutionary electronics.[26,27] However, most of the work in evolutionary electronics is concerned with the design of circuits that compute functions of input data, rather than the evolution of circuits that have novel dynamic behaviors.

The paper is organized as follows: In Sec. II, we introduce differential equations that have been used to model genetic networks. Section III describes the design of an electronic circuit that models the differential equations, and shows the dynamics for circuits of three and five elements, respectively. Section IV describes the evolution algorithm we employ to search for novel dynamic behaviors. In Sec. V, we provide theoretical insight into the optimal mutation rate by analyzing the fitness landscape of the model system. Finally, in Sec. VI, we discuss the significance and implications of the presented work.

II. A DIFFERENTIAL EQUATION MODEL FOR GENE NETWORKS

The class of piecewise linear differential equations that underlie the design of the circuit has been proposed as a highly simplified model of genetic networks,[20–25]

$$\frac{dx_i}{dt} = -\gamma_i x_i + B_i(X_{i_1}(t), X_{i_2}(t), \ldots, X_{i_K}(t)),$$

$$i = 1, \ldots, N, \tag{1}$$

where x_i is a continuous variable, X_i is a discrete binary variable, $X_i = 1$ if $x_i \geq \theta_i$ and $X_i = 0$ if $x_i < \theta_i$, where θ_i is a threshold, γ_i is a decay constant, and $B_i(X_{i_1}(t), X_{i_2}(t), \ldots, X_{i_K}(t))$ is a function that depends only on the logical values of its K inputs, $X_{i_1}(t), X_{i_2}(t), \ldots, X_{i_K}(t)$. We assume that there is no self-input, so that the inputs to B_i do not include variable i. In some situations, and in particular in the current paper, we may assume that B_i is a Boolean variable that only assumes two values (which through rescaling can be set to be 0 and 1). In the biological context, we can think of x_i as a class of proteins, called transcription factors, that regulate the production of other transcription factors, e.g., see (8). Given that the circuit elements act as simple integrators, the dynamics of any particular network are governed completely by the truth table and initial conditions.

Letting $\{t_1, t_2, \ldots, t_k\}$ denote the switch times when any element of the network crosses its threshold, we can obtain the solution of Eq. (1) for each variable x_i for $t_j < t < t_{j+1}$:

$$x_i(t) = x_i(t_j) \, e^{-(t-t_j)} + B_i(X_{i1}(t), X_{i2}(t), \ldots, X_{iK}(t))$$

$$\times (1 - e^{-(t-t_j)}). \tag{2}$$

Thus, by piecing together the trajectories, it is possible to determine the dynamics for future times. As we show below, this differential equation can be implemented by a hybrid digital–analog circuit in which different elements are changing state at different times. Consequently, it differs significantly from synchronous Boolean switching networks, such as those proposed by Kauffman,[3,4] in which the logical states of all network elements are updated simultaneously.

Because of their simple structure, these equations are amenable to theoretical analysis. The equations can display fixed points, stable limit oscillations, and chaotic dynamics. Further, as the number of variables in the networks increases, there is a combinatorial explosion in the number of possible networks. We are interested in constructing an electronic network with a comparatively small number of elements that can have rich dynamic behavior. We choose to construct a system of five variables in which each receives four inputs. Since there are 2^4 logical states of four variables, there are $2^{2^4} = 2^{16}$ different logic functions of four variables. Consequently, the total number of networks is 2^{80}. Using group theoretic arguments based on the symmetries of the truth tables, the number of distinct networks is $\approx 3.14 \times 10^{20}$.[22] These different networks are generated by designating the 80 entries in the five truth tables of the five elements of the network. Edwards gives a comprehensive review of the properties of these equations.[23]
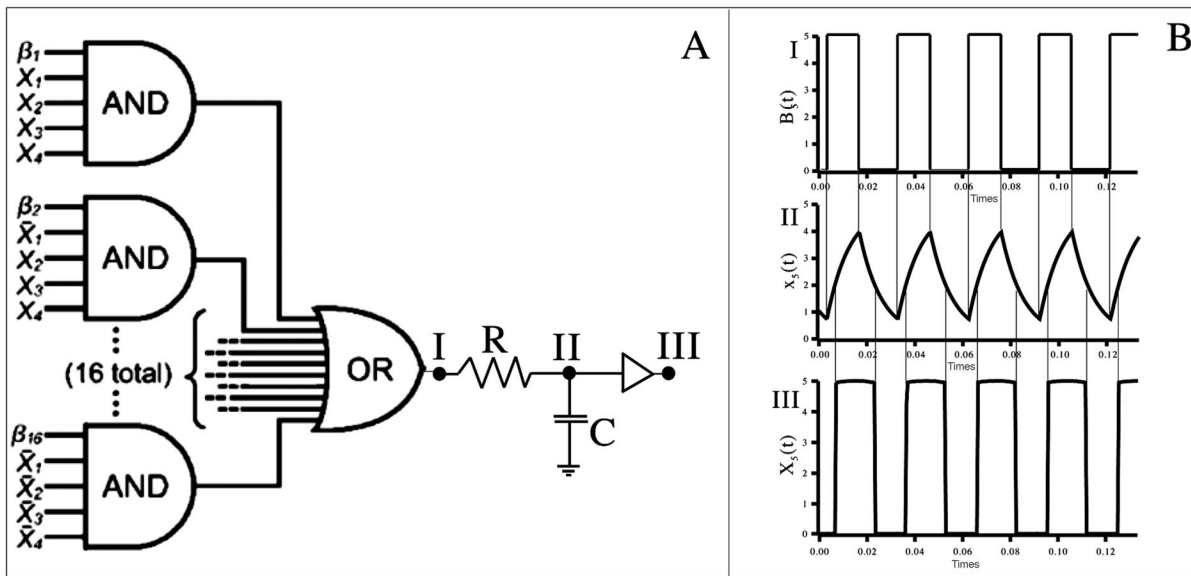
FIG. 1. Schematic diagram of element 5 in the electronic circuit (A) and the output at three points in this circuit (B). The output of the circuit is controlled by inputs from X_1, X_2, X_3, X_4 and the truth table $\beta_1, \beta_2, \ldots, \beta_{16}$. The AND functions and OR functions are realized using CMOS chips. The waveforms at points I, II, and III in the circuit are indicated in panel B. The output of the truth table at I is converted to an exponentially increasing or decreasing sawtooth at II by passing the voltage at I into an RC circuit. The sawtooth is then converted to a step function using two inverter chips in series. An inverter chip converts an input voltage that is less than a threshold of approximately 2.1 V, to a "high" output, and an input voltage that is greater than the threshold to a "low" output. The output at III from this circuit is fed into the inputs of the other elements. The circuits for the other elements are constructed in an analogous fashion. The bar over a variable indicates negation, i.e., $\bar{0} = 1$ and $\bar{1} = 0$.

III. A HYBRID DIGITAL–ANALOG CIRCUIT

A. Design of the circuit

We constructed a hybrid digital–analog circuit that models networks of five genes described by the above equations. Since this system is subjected to intrinsic noise and time delays associated with its operation, and since its parameters depend on the actual values of capacitances and resistors, any observed oscillations must be robust.

The construction of this circuit relies on RC circuits and combinatorial switching circuits. In an RC circuit, a voltage E is provided to a circuit with a resistance R and capacitance C. In our system, the voltage E arises from the logical controller as it switches back and forth between 5 V and 0 V, as determined by the dynamics of the network described below [Fig. 1(A)]. Following a change in the value of E at $t=0$, we find that the voltage across the capacitor is

$$V(t) = E + (V(0) - E)e^{-(t/RC)}. \qquad (3)$$

Thus, the voltage across the capacitor is an exponential function that approaches E with a time constant equal to RC. In this sense, the voltage $V(t)$ is analogous to the concentration of a protein transcription factor $x_i(t)$. Moreover, we can pass the voltage through a threshold element to provide a logical variable analogous to the logical variable $X_i(t)$.

Figure 1(A) shows a schematic diagram of the circuit for element 5. To model the regulation of genes, we apply methods of combinatorial switching circuit design[35] using CMOS logic. This allows us to model in a programmable way, the logical functions that control the regulation of the "on–off" states the genes. Any of the 2^{16} logic functions of four variables can be synthesized by an appropriate combination of the control lines $\beta_1, \beta_2, \ldots, \beta_{16}$. In order to generate any

logical function, the β_i are combined in an AND function with all possible logical states of the four model genes that are inputs to gene 5. For example, setting $\beta_1 = 1$ and $\beta_i = 0$ for $i = 2, \ldots, 16$, only produces a value "true" or "1" if $X_1 = X_2 = X_3 = X_4 = 1$ at the same time.

The operation of the network can be appreciated by considering the voltage relative to ground that would be recorded at three different places in the circuit [see Fig. 1(B)]. The voltage at I corresponds to the Boolean function B_5 in Eq. (2). This voltage would be 5 V if the logical switch at that time was "true," otherwise it would be 0 V. The voltage at point II corresponds to the value of x_5 in Eq. (2). As follows from Eq. (3), if the voltage at I is 5 V, then at point II in the circuit, there will be an exponentially increasing function; if the voltage at I is 0 V, then at point II in the circuit, there will be an exponentially decreasing function. Finally, the voltage at III corresponds to the value of X_5 in Eq. (2). By passing the signal at II through two inverters, we find a voltage of 5 V at point III if the voltage is above the threshold, or we find a voltage of 0 V if the voltage is less than the threshold. The feedback is provided by feeding back X_5, and its complement $\bar{X}_5$, into the combinatorial logic functions for the other elements. By selecting $R = 100$ kΩ, $C = 0.1$ μF, we set the time constant to be 10 ms.

The circuit was modified and data were analyzed under Labview (National Instruments, Austin, TX, USA) with a digital I/O card to initialize the control functions and an analog data acquisition card to collect the output. All modifications involved only making changes in the set of β_i that define the truth tables for the network. We analyzed the resulting dynamics for stable periodic oscillations. The analyses were carried out for time series of 3.5 s length sampled at

2 kHz. Each time a circuit element switched from high to low, or low to high, we recorded the element that switched. The resulting sequence of integers was then analyzed for the shortest repeating sequence, of at least three integers, in which each integer appears an even number of times. Periodic switching sequences in the solutions of Eq. (2) are associated with stable limit cycle oscillations.[22] If the identified sequence repeated at least 15 times, we determined the period of the resulting cycle. This analysis procedure was checked in a large number of cases, and in all cases, it identified stable large-amplitude periodic solutions. Since these networks can also display switching at a rapid rate set by the time delays of the circuit that are associated with a stable focus in the differential equations, and since we are interested in stable limit cycle oscillations with long periods, we only consider cycles whose period was greater than 10 ms.

B. The repressilator: Oscillations in a three-gene network

In order to illustrate the operation of the circuit, we show the dynamics for a simple circuit with three genes that has been designed to oscillate. One mechanism for generating oscillations involves feedback circuits composed of a ring of elements, each of which either inhibits or activates the next element in the ring. Situations in which there are an odd number of inhibitory interactions often display stable limit cycle oscillations. Elowitz and Leibler[18] implemented this type of circuit in bacteria by constructing an inhibitory ring of three genes. Each gene coded for a transcription factor that in turn inhibited the synthesis of the next gene in the ring, Fig. 2(A). The network was constructed using plasmids in *E. coli*, and the dynamics were monitored using green fluorescent protein (GFP) that was under control of the transcriptional circuit. The resulting network, called the repressilator, exhibited oscillations in the expression of GFP. A differential equation of the form in Eq. (2) shows stable limit cycle oscillations with period 2.887... when $\gamma_i = 1$.[22]

As a first test for our circuit, we implemented a network that had the same logical structure as the repressilator.[22] The truth table for this corresponding network is shown in Fig. 2(B). The network has a stable oscillation, shown in Fig. 2(C). Taking into account the time constant for our circuit, we compute that the period of the limit cycle oscillation in the circuit would be 28.87 ms, provided all the time constants are equal, and the thresholds of all switches are exactly 2.5 V. In the current case, measurement of the time constants show that they differ from the rated 10 ms by up to 6%, and that the thresholds are approximately 2.1 V. These small differences lead to a period of the digital–analog repressilator circuit of approximately 29.5 ms. The pattern of oscillation in the circuit is similar to that observed in the repressilator and in differential equation models of the repressilator.[18,22]

C. More complex oscillations in a five-gene network

A five-gene network is capable of an extremely rich variety of oscillatory behaviors. Examples of two truth tables and their corresponding dynamics are shown in Fig. 3. For compactness, the truth tables are written in the following
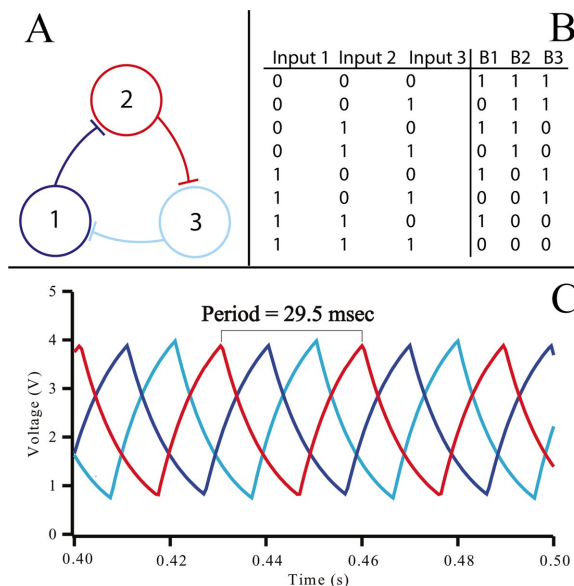

FIG. 2. (Color) (A) Schematic diagram of the repressilator. (B) Truth table for the repressilator. (C) Dynamics of the digital–analog repressilator circuit. There is a stable oscillation with period 29.5 ms.

way: all combinations of four inputs for any individual network element are given on the left, and the logic functions represented by the five elements are given in columns $B_1 - B_5$ on the right. Recalling that each element receives input from the other four elements with no self-input, the four inputs on the right correspond to inputs from elements 1, 2, 3, 4, and 5, minus the element in question. In other words, for element 1, inputs 1–4 come from elements 2–5, respectively. For element 2, inputs 1–4 come from elements 1, 3, 4, and 5, respectively, and so on. As seen in Fig. 3, it would be difficult to predict these dynamics based on the logic of the network.

IV. SEARCHING FOR COMPLEX OSCILLATIONS IN THE ELECTRONIC CIRCUIT

We set a search task of finding networks that display complex oscillations which occur infrequently by chance. To do this, we first carried out a survey of the distribution of the periods of randomly generated networks. Figure 4 shows a histogram displaying the periods found in the circuit for 300 randomly generated networks. There are a comparatively small number of networks with periods greater than 60 ms. Based on this observation, we selected a target period, denoted T^*, of 80 ms (± 5 ms) as the target period for our search procedure.

Evolution was implemented in the following manner. Random networks were generated until a network displayed periodic dynamics. Then a random local search was initiated. For each element in the truth table, a random number was generated. If this random number was less than the mutation rate, called ρ, then a random 1 or 0 replaced that element in the truth table. This led to truth table mutations at an average rate of $\rho/2$. In each generation, there was only one "progeny." If the progeny had a limit cycle oscillation whose period was the same as the period of the parent or closer to the

Input 1	Input 2	Input 3	Input 4	B1	B2	B3	B4	B5
0	0	0	0	1	1	1	0	1
0	0	0	1	1	0	1	0	0
0	0	1	0	0	0	0	1	1
0	0	1	1	1	1	0	1	1
0	1	0	0	1	1	1	1	0
0	1	0	1	1	0	0	1	0
0	1	1	0	1	1	1	1	0
0	1	1	1	0	1	0	1	1
1	0	0	0	1	1	0	0	1
1	0	0	1	0	0	1	0	1
1	0	1	0	1	0	1	0	0
1	0	1	1	0	0	0	0	0
1	1	0	0	0	1	1	1	0
1	1	0	1	0	0	0	1	1
1	1	1	0	0	0	1	0	0
1	1	1	1	1	1	0	1	1

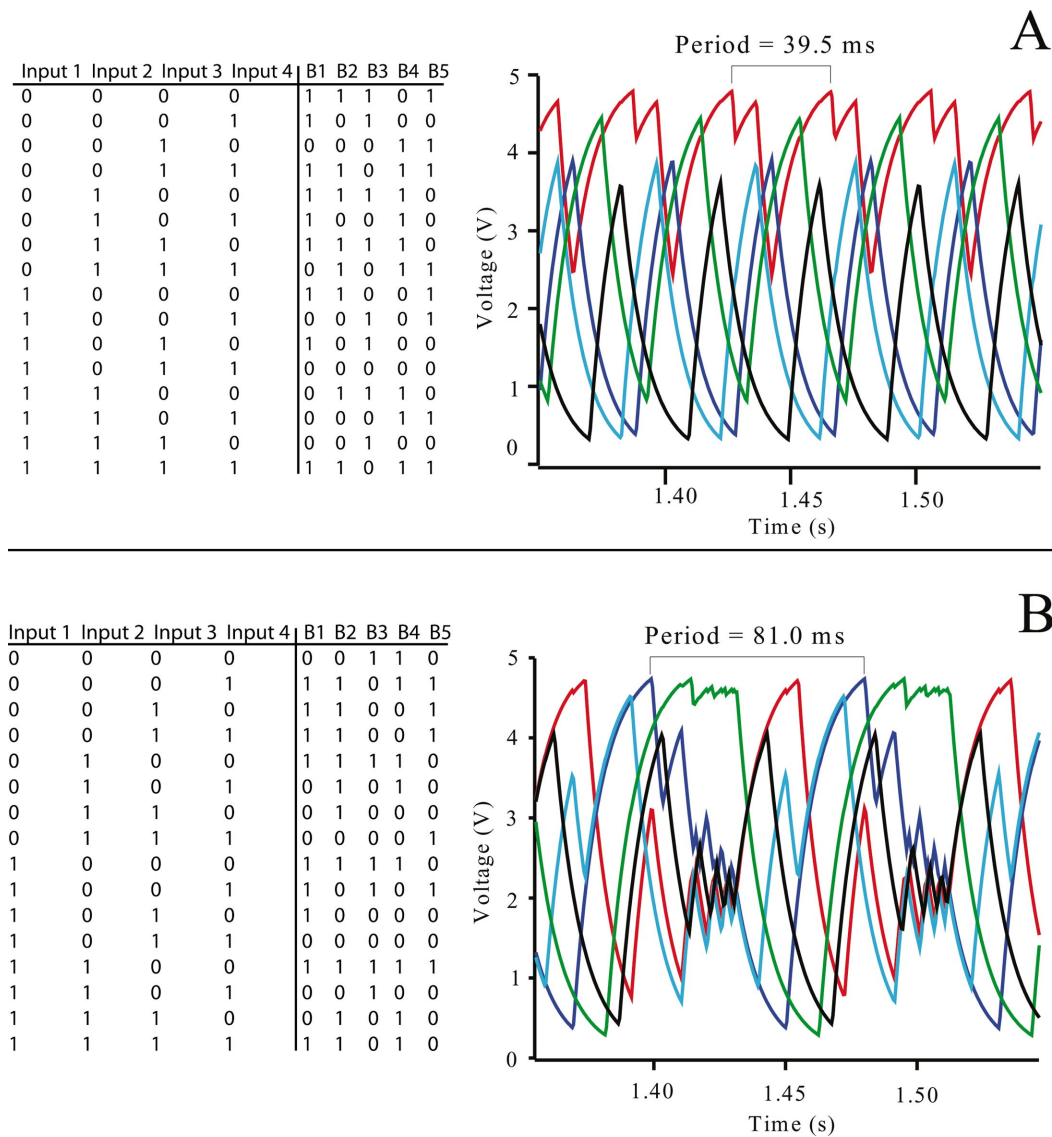| Input 1	Input 2	Input 3	Input 4	B1	B2	B3	B4	B5
0	0	0	0	0	0	1	1	0
0	0	0	1	1	1	0	1	1
0	0	1	0	1	1	0	0	1
0	0	1	1	1	1	0	0	1
0	1	0	0	1	1	1	1	0
0	1	0	1	0	1	0	1	0
0	1	1	0	0	1	0	0	0
0	1	1	1	0	0	0	0	1
1	0	0	0	1	1	1	1	0
1	0	0	1	1	0	1	0	1
1	0	1	0	1	0	0	0	0
1	0	1	1	0	0	0	0	0
1	1	0	0	1	1	1	1	1
1	1	0	1	0	0	1	0	0
1	1	1	0	0	1	0	1	0
1	1	1	1	1	1	0	1	0

FIG. 3. (Color) The truth tables and data for two different networks. To read the truth tables, recall that each element receives four inputs (one from each other element, with no self input). For each element, then, inputs 1–4 represent inputs from the five elements minus itself. For example, for element 3, inputs 1–5 come from elements 1, 2, 4, and 5, respectively. The logic function defined for each element in the circuit, 1–5, is given by B_1 through B_5 on the right side of the truth table.
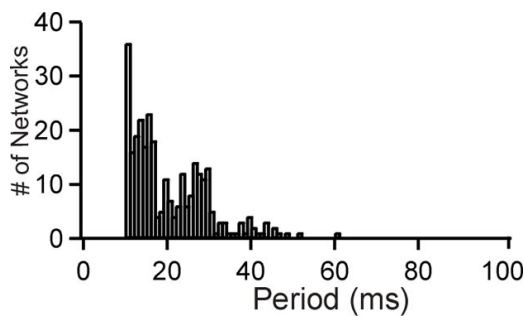

FIG. 4. Histogram showing the periods of stable limit cycle oscillations in the electronic circuit based on an analysis of 300 randomly generated networks which display limit cycle oscillations. A period of oscillation for a network in the differential equation literature must be multiplied by approximately 10 to find the comparable period in ms for the electronic circuit. The percentage of randomly generated networks showing stable oscillations is approximately 2.8%.

target than the period of the parent, then future evolution was carried out from its truth table. Otherwise, evolution was carried out using the truth table of the parent.

Figure 5 shows the results of a typical evolution trial. The initial period of the circuit was 27.0 ms. The period increases through a series of plateaus of different heights and durations, and at the end of 540 trials achieves a period of 75.5 ms. The observed oscillation of the final network differs considerably from that of the initial circuit. The course of evolution in each run was different, and the final circuits obtained are also typically different. Even though a small percentage (about 2%–3%) of the possible networks show stable oscillations, this is still a very large number of circuits displaying stable limit cycles.

Figure 6 compares the average results of 25 trials for several different rates of evolution. Each trial was carried out
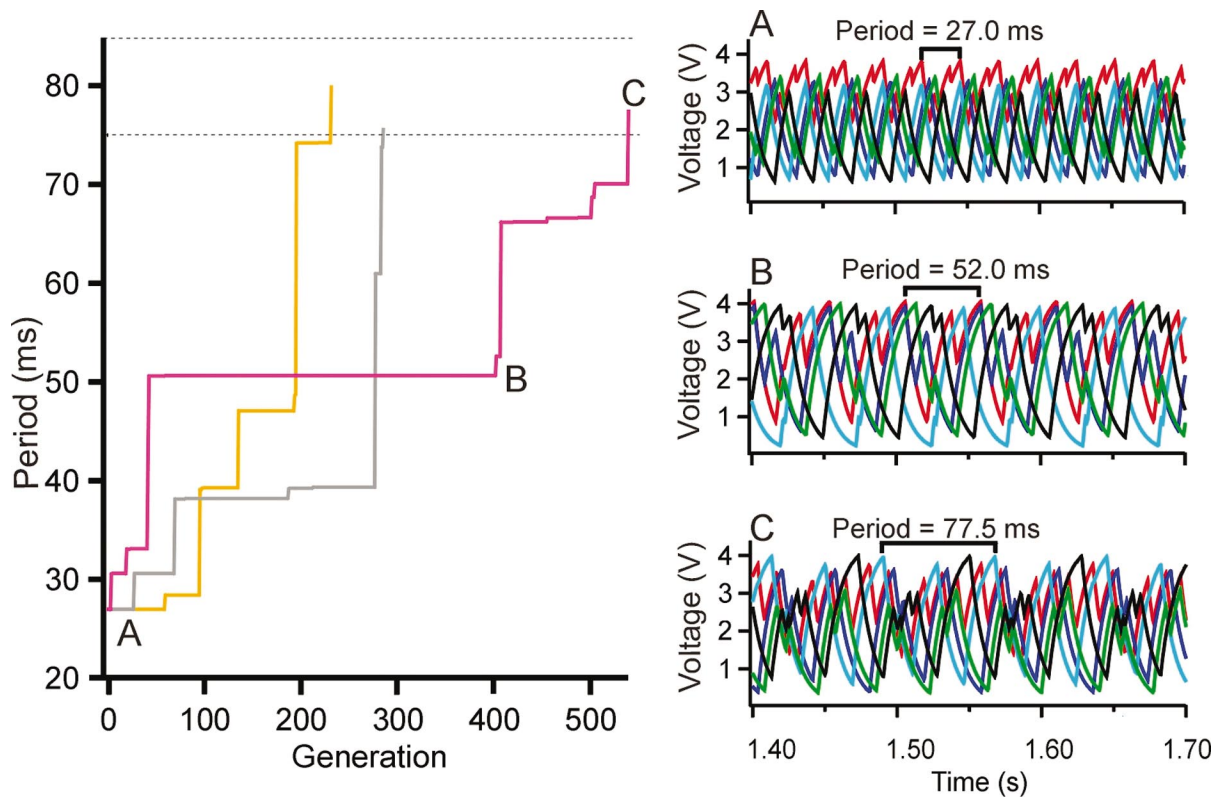
FIG. 5. (Color) Typical example of evolution of the limit cycle oscillation in the electronic circuit. The three traces in the left panel show the increases in period of the network in three different runs as the network evolves towards the goal. Each run begins with the same initial network, whose dynamics are displayed in panel A. Panels B and C show the dynamics of an intermediate network and the ending network, respectively, for one of these runs. The colored traces in panels A, B, and C represent the output of the five network elements. Both the complexity of the oscillation and the period increase during evolution, and the dynamics of the final network differs dramatically from that of the initial network.

for 250 generations. These results demonstrate that the approach to the target is maximally fast for a mutation rate of $\rho = 5\% - 10\%$.

V. DETERMINATION OF THE FITNESS LANDSCAPE

One of the main theoretical means for understanding evolution in biological systems and model systems is to construct the fitness landscape, a graph in which the vertices represent the different genetic makeup of organisms. A fitness is associated with each vertex, where the fitness reflects
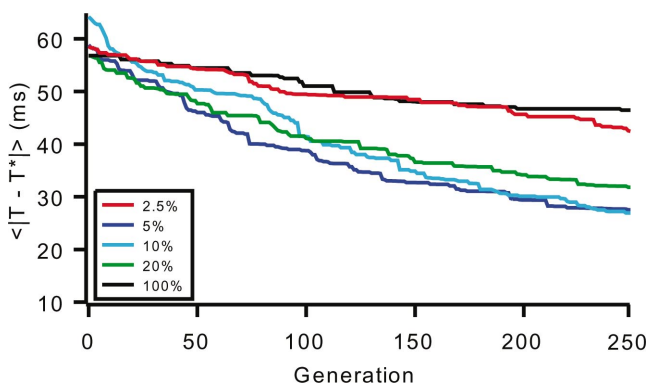

FIG. 6. (Color) Evolution of the electronic circuit (schematically represented in Fig. 1) for different mutation rates ρ. The average deviation of the period from the target value $T^* = 80$ ms is plotted as a function of generation number.

the relative ability of the organism to compete with other organisms. Higher fitnesses are associated with organisms that tend to outcompete other organisms of the same species. Various models of the fitness landscape have been proposed and features such as the number of local maxima and the mean path length from any state to its nearest local maxima have been computed; for example, see Refs. 30, 36–38, and references therein.

For our circuit, the fitness landscape is an 80-dimensional Boolean hypercube where each of the 2^{80} vertices represents a different truth table for the network. The Hamming distance between two Boolean vectors of the same length represents the number of loci in which the two vectors differ. Consequently, in the Boolean hypercube, vertices that share a common edge represent states with a Hamming distance of one, vertices separated by two edges represent states with a Hamming distance of two, and so forth. Networks that do not give rise to periodic dynamics have fitness 0. The fitness of a given network with period T and target period T^* is inversely proportional to $|T - T^*|$. We sampled the fitness landscape in the neighborhood of many different periodic networks. Random networks were generated until a network displayed periodic dynamics.

Once a network which displayed periodic dynamics was randomly generated, the effects of flipping a fixed number of truth table entries were determined. All 80 truth tables with a Hamming distance of one from the parent network were sampled. One thousand randomly selected networks that lie

FIG. 8. The expected improvement per generation, $\Delta(\rho)$, as a function of the mutation rate ρ.
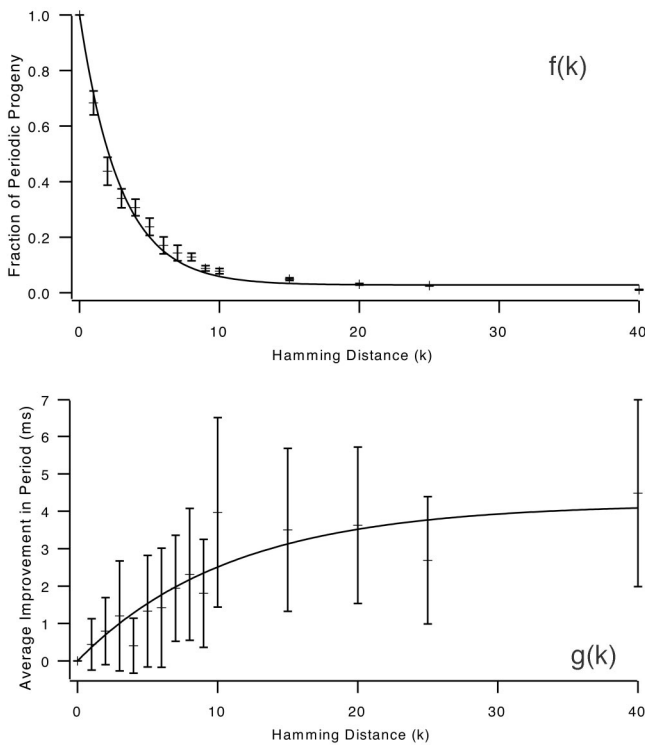
FIG. 7. The fraction of periodic networks, $f(k)$, and the average improvement in period of a periodic network, $g(k)$, as a function of the Hamming distance k of a network from its parent. The error bars show the standard error and the solid curves represent fits to exponential functions.

Hamming distances $k=2-10$, 15, 20, and 25, from the parent network were also sampled. We repeated this process for 10 different periodic networks at each condition, and determined the mean fitness and fraction of periodic networks for each condition.

To study properties of the random local search in these networks, we developed an approximate characterization of the fitness landscape. Let k represent the Hamming distance of a network from another network displaying a stable periodic cycle. The fitness landscape is characterized by two functions: $f(k)$, which is the fraction of networks displaying periodicity, and $g(k)$, the average improvement in the period of a progeny network displaying a stable limit cycle towards the target period. As k increases, $f(k)$ decreases and $g(k)$ increases. These data are shown with standard errors in Fig. 7 and were fit to exponential functions (solid lines).

The probability that any given element of the truth table will change given the mutation rate ρ is $\rho/2$. Using the binomial theorem in a truth table of N entries, the fraction of truth tables that are a Hamming distance k from the initial network, $\Phi(N,k,\rho)$ is

$$\Phi(N,k,\rho)=\binom{N}{k}\left(\frac{\rho}{2}\right)^{k}\left(1-\frac{\rho}{2}\right)^{N-k}. \tag{4}$$

Using this result, we can compute the expected mean improvement per generation $\Delta(\rho)$
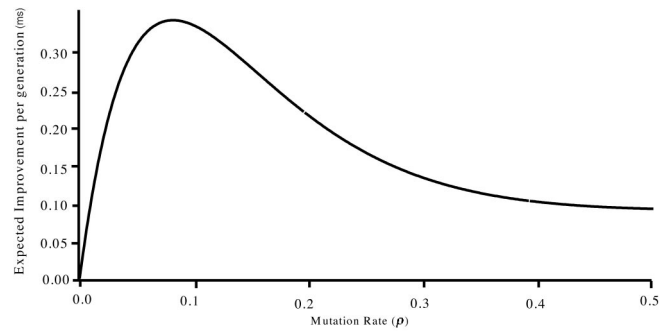
$$\Delta(\rho)=\sum_{k}f(k)g(k)\Phi(N,k,\rho). \tag{5}$$

The results are shown in Fig. 8. Based on the above computation, the search towards the target period is predicted to be most rapid for $\rho\approx0.08$. This is in agreement with the experimental data in Fig. 6.

Figure 7 shows that while there is a small amount of variation of each data point for $f(k)$, there is considerably more variation in the values of the data points for $g(k)$. This occurs since we only sample a small portion of the networks a Hamming distance k away from a given network, and as k increases, the standard error also increases. Consequently, many different functional forms could have been used to fit $g(k)$ in Fig. 7. However, choosing other functional forms such as a second-order polynomial or Gaussian for $g(k)$ also yield optimal mutation rates approximately the same as the one found using the exponential fit.

VI. DISCUSSION

In biological systems, stable oscillations are commonly exhibited, but it is unclear how these dynamics can arise and evolve. The present work shows that robust physically realizable oscillations can arise quite easily in model genetic networks, and that such networks can be modified to produce oscillations of different periods. Further, by analyzing the fitness landscape in the neighborhood of periodic networks, we demonstrate that the search procedure is optimized for an intermediate mutation rate (Fig. 8).

In the evolution runs carried out in this work, the circuit starts each trial with with each element at a low voltage near zero in the low logic state. Therefore, during all evolutionary trials, the period of a particular network was the period starting from this initial condition. As we demonstrate, we obtain an approach to the target period using this scheme. However, these networks can also display more than one attractor starting from different initial conditions. Consequently, it would be of interest to analyze the evolution rates, if at any trial we selected the best period possible from some limited subset of initial conditions.

We think that this system, in which there is a precisely defined fitness landscape for a real physical system, poses an interesting model for further theoretical analysis. In this system, there are a large number of good solutions that are sparsely scattered through the space of all possible networks. As a consequence, in contrast to recent theoretical results in which there is an evolution through adjacent states to a local

maximum after a small number of steps,[37] the evolution in the current model occurs over many intermediates with long plateaus, to one of a large number of possible end states (Fig. 5). The discontinuous changes in dynamic behavior, are similar to the discontinuous evolution that has been observed in many different biological systems, e.g., see Ref. 39, and references therein. In the current case, the discontinuous changes in the period reflect the combined effects of neutral mutations that do not lead to a change in the period and other mutations that lead to large changes in the period.

The data in Fig. 7 indicate that the fitness landscapes are correlated, so that in the neighborhood of a network displaying periodic dynamics, there tend to be a high density of periodic networks with a similar period. For correlated fitness landscapes, optimal mutation schemes would incorporate a mutation rate inversely proportional to the fitness, similar to observed mutation rates in other biological and model systems.[38] It would also be interesting to study evolutionary schemes based on genetic algorithms,[40] and schemes that allow the possibility of detrimental mutations.

The current work is also relevant to nonlinear dynamics. In addition to the periodic dynamics that we have described in the current paper, we anticipate that the circuit would support chaotic dynamics, since Eq. (2) does.[21,24] Thus, it would be interesting to study the dynamics obtained when the truth tables or time constants of networks displaying chaos are varied. Just as there are families of networks displaying closely related types of periodic oscillations, there might be families of networks that display closely related chaotic dynamics,[22,25] and it might be possible to evolve networks displaying chaotic dynamics with specified dynamical properties.

This work shows that it is possible to evolve electronic networks with desired dynamics without needing to design the circuit architecture from the outset. This has implications for engineering since it shows that it should be possible to develop autonomous electronic circuits that evolve their dynamics, based upon environmental demands, by changing the logical structure of the network without reconfiguring its hardware. Moreover, because of the intimate connection between our circuits and genetic networks, it may be feasible to adapt these methods to evolve genetic oscillators. Indeed, we think that this work represents an important middle ground between more theoretical work[21,22] and more biological work.[12,15,18] Further, since the building blocks of our networks depend on functions that can be implemented using the binding of transcription factors to DNA, the current work also shows how complex biological functions may have evolved dynamic behaviors even though the combinatorial complexity of possible network states is astronomical.

[1] J. Monod and F. Jacob, "General conclusions: Teleonomic mechanisms in cellular metabolism, growth, and differentiation," Cold Spring Harb Symp. Quant Biol. **XXV**, 389–401 (1961).

[2] M. Sugita, "Functional analysis of chemical systems in vivo using a logical circuit equivalent," J. Theor. Biol. **4**, 179–192 (1963).

[3] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," J. Theor. Biol. **22**, 437–467 (1969).

[4] S. A. Kauffman, *Origins of Order: Self-Organization and Selection in Evolution* (Oxford University Press, Oxford, 1993).

[5] R. Thomas, "Boolean formalization of genetic control circuits," J. Theor. Biol. **42**, 563–585 (1973).

[6] R. Thomas and R. D'Ari, *Biological Feedback* (CRC Press, Boca Raton, 1990).

[7] H. H. McAdams and L. Shapiro, "Circuit simulation of genetic networks," Science **269**, 650–656 (1995).

[8] C.-H. Yuh, H. Bolouri, and E. H. Davidson, "Genomic cis-regulatory logic: Experimental and computational analysis of a sea urchin gene," Science **279**, 1896–1902 (1998).

[9] S. Bornholdt and K. Sneppen, "Neutral mutations and punctuated equilibrium in evolving genetic networks," Phys. Rev. Lett. **81**, 236–239 (1998).

[10] G. von Dassow, E. Meir, E. Munro, and G. M. Odell, "The segment polarity network is a robust developmental module," Nature (London) **406**, 188–192 (2000).

[11] H. Bolouri and E. H. Davidson, "Modeling transcriptional regulatory networks," BioEssays **24**, 1118–1129 (2002).

[12] C. C. Guet, M. B. Elowitz, W. Hsing, and S. Leibler, "Combinatorial synthesis of genetic networks," Science **296**, 1466–1470 (2002).

[13] Y. Yokobayashi, R. Weiss, and F. H. Arnold, "Directed evolution of a genetic circuit," Proc. Natl. Acad. Sci. U.S.A. **99**, 16587–16591 (2002).

[14] N. E. Buchler, U. Gerland, and T. Hwa, "On schemes of combinatorial transcription logic," Proc. Natl. Acad. Sci. U.S.A. **100**, 5136–5141 (2003).

[15] T. S. Gardner, C. R. Cantor, and J. J. Collins, "Construction of genetic toggle switch in *Escherichia coli*," Nature (London) **403**, 339–342 (2000).

[16] F. J. Isaacs, J. Hasty, C. R. Cantor, and J. J. Collins, "Prediction and measurement of an autoregulatory genetic model," Proc. Natl. Acad. Sci. U.S.A. **100**, 7714–7719 (2003).

[17] H. Kobayashi, M. Kaern, K. Chung, T. S. Gardner, C. R. Cantor, and J. J. Collins, "Programmable cells: Interfacing natural and engineered gene networks," Proc. Natl. Acad. Sci. U.S.A. **101**, 8414–8419 (2004).

[18] M. B. Elowitz and S. Leibler, "A synthetic oscillatory network of transcriptional regulators," Nature (London) **403**, 335–338 (2000).

[19] L. Glass and S. A. Kauffman, "Cooperative components, spatial localization, and oscillatory cellular dynamics," J. Theor. Biol. **34**, 219–237 (1972).

[20] L. Glass, "Combinatorial and topological methods in nonlinear chemical kinetics," J. Chem. Phys. **63**, 1325–1335 (1975).

[21] T. Mestl, R. J. Bagley, and L. Glass, "Common chaos in arbitrarily complex feedback networks," Phys. Rev. Lett. **79**, 653–656 (1997).

[22] R. Edwards and L. Glass, "Combinatorial explosion in model gene networks," Chaos **10**, 691–704 (2000).

[23] R. Edwards, "Analysis of continuous-time switching networks," Physica D **146**, 165–199 (2000).

[24] R. Edwards, "Chaos in neural and gene networks with hard switching," Diff. Eq. **9**, 187–220 (2001).

[25] R. Edwards, H. T. Siegelmann, K. Aziza, and L. Glass, "Symbolic dynamics and computation in model gene networks," Chaos **11**, 160–169 (2001).

[26] *Evolvable Systems: From Biology to Hardware*, edited by J. Miller, A. Thompson, P. Thompson, and T. C. Fogarty (Springer-Verlag, Berlin Heidelberg, 2000).

[27] R. S. Zebulum, M. A. C. Pacheco, and M. M. B. R. Vellasco, *Evolutionary Electronics. Automatic Design of Electronic Circuits and Systems by Genetic Algorithms* (CRC Press, Boca Raton, 2002).

[28] J. Masel and A. Bergman, "The Evolution of the Evolvability Properties of the Yeast Prion [PSI+]," Evolution (Lawrence, Kans.) **57**, 1498–1512 (2003).

[29] S. Bornholdt and T. Rohlf, ''Topological evolution of dynamical networks: Global criticality from local dynamics,'' Phys. Rev. Lett. **84**, 6114–6117 (2000).

[30] P. R. A. Campos, C. Adami, and C. O. Wilke, ''Optimal adaptive performance and delocalization of NK fitness landscapes,'' Physica A **304**, 495–506 (2002).

[31] S. A. Kauffman and R. G. Smith, ''Adaptive automata based on Darwinian selection,'' Physica D **22**, 68–82 (1986).

[32] P. François and V. Hakim, ''Design of genetic networks with specified functions by evolution *in silico*,'' Proc. Natl. Acad. Sci. U.S.A. **101**, 580–585 (2004).

[33] A.-L. Barabási and R. Albert, ''Emergence of scaling in random networks,'' Science **286**, 509–512 (1999).

[34] S. H. Strogatz, ''Exploring complex networks,'' Nature (London) **410**, 268–276 (2001).

[35] M. B. Muroga, *Logic Design and Switching Theory* (Wiley, New York, 1979).

[36] S. A. Kauffman, *The Origins of Order* (Oxford University Press, New York, 1993).

[37] H. A. Orr, ''A Minimum on the mean number of steps taken in adaptive walks,'' J. Theor. Biol. **220**, 241–247 (2003).

[38] L. Hadany and T. Beker, ''Fitness-associated recombination on rugged adaptive landscapes,'' J. Evol. Biol. **16**, 862–870 (2003).

[39] W. Fontana and P. Schuster, ''Continuity in Evolution: On the Nature of Transitions,'' Science **280**, 1451–1455 (1998).

[40] J. H. Holland, *Adaptation in Natural and Artificial Systems* (University of Michigan, Ann Arbor, 1975).