PHYSICAL REVIEW E **77**, 060902(R) (2008)

Reliability of genetic networks is evolvable

Stefan Braunewell and Stefan Bornholdt
Institute for Theoretical Physics, University of Bremen, D-28359 Bremen, Germany

(Received 26 July 2007; published 20 June 2008)

Control of the living cell functions with remarkable reliability despite the stochastic nature of the underlying molecular networks—a property presumably optimized by biological evolution. We ask here to what extent the ability of a stochastic dynamical network to produce reliable dynamics is an evolvable trait. Using an evolutionary algorithm based on a deterministic selection criterion for the reliability of dynamical attractors, we evolve networks of noisy discrete threshold nodes. We find that, starting from any random network, reliability of the attractor landscape can often be achieved with only a few small changes to the network structure. Further, the evolvability of networks toward reliable dynamics while retaining their function is investigated and a high success rate is found.

DOI: 10.1103/PhysRevE.77.060902 PACS number(s): 87.16.Yc, 87.17.Aa, 89.75.Hc

Processes in living cells are largely controlled by complex networks of molecular interactions. A remarkable feature of these networks is their reliable functioning, despite the stochastic nature of their molecular components [1,2]. There are several indications that properties of the circuitry influence reliability of noisy systems [3–6].

Taking an evolutionary perspective, noise resistance of biological networks must have emerged from the interplay of mutation and selection. We study here the question of how accessible noise-resistant dynamical networks are to evolution and what the costs are in terms of topological rearrangements in order to achieve a reliable dynamical network. We study this question in the framework of numerical experiments, evolving discrete dynamical networks in the computer.

Different notions of robustness have been considered in this framework. It was found that networks can evolve toward realizations that are robust against mutations of the network structure [7]. Further, robustness of dynamical (expression) patterns has been studied as a guiding principle of network evolution [8]. Recent numerical experiments show that networks can simultaneously be robust against structural changes (attractors are left unchanged) as well as evolvable (new attractors emerge) [9]. Finally, network evolution toward robustness against gene state errors in the initial configuration turns out to be an easy and rapid process [10,11].

In this Rapid Communication, we extend these viewpoints by studying the evolution of networks toward robustness against small timing fluctuations or "reliability" (to avoid confusion with existing definitions of the term "robustness"). While gene switching errors (a type of "perturbation" commonly used by many authors) are not at all small perturbations and not typical for real cells, small perturbations in timing and activity levels are ubiquitous in biological systems. Such small noise levels have recently proven to destroy most attractors in Boolean networks that are observed under parallel update [12,13]. Obviously, only those attractors that are stable against such small noise (i.e., "reliable") can be relevant in the biological context. Indeed, in the biological example of the yeast cell cycle network, this type of stability against timing perturbations is observed [14].

Here, we investigate whether such reliability of a dynamical network can readily result from an evolutionary procedure, only involving rewiring of the network structure. Defining biologically motivated mutation-selection processes, we will evolve random networks toward realizations that exhibit reliable dynamics. We investigate both the emergence of entirely stable attractor landscapes as well as the ability of networks to evolve in such a way that a given attractor is stabilized.

On a different route, the question of evolving networks toward performing a specified target function has been investigated in various works since the first numerical experiments of Kauffman and Smith in 1986 [15]. Boolean networks can be evolved to perform specific tasks and several authors have discussed how such problems are affected by noise from external sources [16], modular goals [17], and specific degree distributions [18]. In contrast to these models, we do not evolve toward a specific target function but instead select networks solely for reliable functioning of their (existing) dynamics. This follows along the lines of an earlier network evolution model that selects for robust reproduction of individual network dynamics [8,19]. As in [8], we choose here a computationally simple subset of all possible Boolean networks, the so-called threshold networks with a majority rule in the inputs of each node. Networks of threshold nodes are best known as the prototypical model for neural networks introduced by McCulloch and Pitts [20]. They later made a second career as models for genetic networks [7,8,21,22]. The main dynamical features of Boolean networks are represented in threshold networks as well [23,24] and a close correspondence between the two has been found in the earlier evolution studies [8,19]. We choose threshold networks in this study, and expect that using networks with the full set of Boolean functions would at least match or even exceed the results reported below (as increasing the number of possible Boolean functions at each node would increase—not decrease—the number of options in each evolutionary step).

We model genes as nodes in a network where every node i assumes a state of either $\sigma_i(t)=+1$ (active) or -1 (inactive). Interactions between the nodes are directed links A_{ij} with an assigned weight of either $A_{ij}=+1$ or -1, corresponding to an activating or inhibiting interaction, respectively, or $A_{ij}=0$ where interaction is absent. The update rule is given by

$$\sigma_i(t+1) = \begin{cases} +1 & \text{if } \sum_{i=1}^{n} A_{ij}\sigma_j(t) \geq 0, \\ -1 & \text{otherwise}, \end{cases} \tag{1}$$

where nodes are updated in parallel. As the dynamics is deterministic in a finite state space, the system always ends up in an attractor, which can be either a fixed point or a limit cycle (as, in general, our weight matrix is asymmetric, $A_{ij} \neq A_{ji}$). The "basin size" of an attractor is defined as the number of all states that lead to this attractor. We assess the stability of a network against fluctuations of the signal transmission times using the reliability criterion of [13], which provides a deterministic measure for a network under investigation. It requires two principal assumptions: First, the nodes implement a low-pass filter that removes the effect of short-term fluctuations of the activity states. This is justified by the buildup and decay processes of protein concentrations [25]. Second, the signal time fluctuations are small compared to the time scales of the processes and that of the filter. Thus, a single signal fluctuation does not dramatically perturb the system, but only the addition of many similar perturbations over time can drive the system away from an initially synchronous behavior.

To determine the stability of an attractor, first the synchronous state sequence is recorded. Starting from one state of the limit cycle, we determine all switching events that occur in the next synchronous step and call the set of switching nodes M. For every subset $S \subset M$ we change the switching times from $t=0$ to $t=\epsilon$, i.e., we retard the switching times for these nodes by a small number. Thus, a new intermediate state from time $t=0$ to $t=\epsilon$ is created, where only some nodes have already switched. We then follow the dynamics, with two times for every synchronous time step:

(i) Determine the states at times $t=i$, $i=1,2,\ldots$ and $t'=i+\epsilon$ from the states at $t=i-1$ and $t'=i-1+\epsilon$, respectively.

(ii) Apply the filter rule: if a node switches both at integer and perturbed time, flip the state at integer time, thus effectively removing both switching events. As the activity state has persisted only for a time span of ϵ, we assume it does not affect the system further.

(iii) If all nodes switch at either integer or perturbed time, the system has regained synchrony and the attractor is stable for this particular subset of perturbed nodes. If, however, the system reaches a new attractor in the combined state space of both times, the system is unstable as the perturbation can in general persist in the system and might diverge, thus leading to a different attractor or to a "chaotic" regime of incessant switchings.

We call an attractor "stable" if it is stable against all subset perturbations, otherwise we call it "unstable." Fixed points are trivially stable by this definition. We use two different evolutionary selection criteria. In the first part, the stability of the full attractor landscape is investigated and the fitness score is given by the sum of the basin sizes of all stable attractors. In the second part, the fitness score is determined by the stability and basin size of one "functional attractor" only.

In every step of the evolution, the network is copied and mutated and its stability assessment is compared with the mother network. If the fitness of the mutant is higher than that of the mother network, the mutant is kept and replaces the original, otherwise a new mutant is tested. This is repeated until the requested criterion is maximally fulfilled.

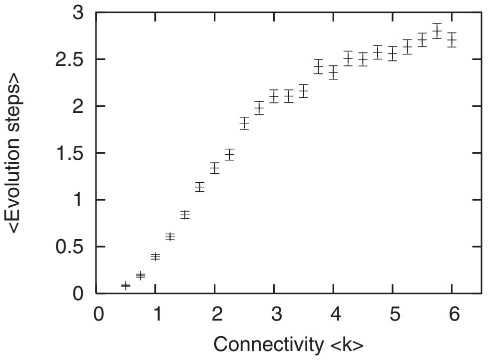

FIG. 1. Network evolution rapidly leads to stable attractor landscapes: average number of evolution steps vs connectivity of the networks. The networks consist of 16 nodes; each data point corresponds to the average over 1000 runs.

Mutation is performed through a single link rewiring, i.e., one connection between two nodes is removed and another new connection between two nodes is added. This procedure keeps the average connectivity of the network during evolution. As our method requires full enumeration of the space of 2^N states, where N is the number of nodes, we can only perform this analysis for small networks. We show the results for $N=16$ nodes, but have checked that the conclusions also hold for networks with $N=12$ and 20 nodes.

In the first part, let us evolve networks toward stability regarding the complete attractor landscape. Given a network, we accept a mutation of it if the mutant has a higher number of initial states leading to a stable attractor. If so, the network is replaced by the mutant and the next evolution step is taken, otherwise a new mutation is tested. This procedure stops as soon as all initial states lead to stable attractors. In Fig. 1, we show the average number of evolution steps necessary to reach full stability of the attractor landscape, plotted against the average connectivity, defined by the total number of edges divided by the number of nodes. Networks consist of 16 nodes and 1000 repetitions were run for every data point. One can see that for all connectivities a very small number of mutations already suffices to find a completely stable network. Using a more restrictive method of selection, such as, for example, choosing the fittest out of several tested mutant networks, further reduces the average evolution steps significantly (data not shown). To ensure that we do not simply observe the effects of networks evolving toward fixed points (which are always stable), we have checked all results also with the rule that a fixed point is counted as an unstable attractor. We do not show the results here, but the conclusions drawn above hold also in this case. The only qualitative difference is that the pronounced drop at low connectivities of the average number of evolution steps (Fig. 1) is not present if fixed points are regarded as unstable attractors. Thus, the two different slopes for $\langle k \rangle < 3$ and $\langle k \rangle > 3$ are due to the abundance of fixed points in networks with low connectivities.

TABLE I. Characteristics of random and evolved networks for $N=16$, $\langle k\rangle=3$. Averages over 20 000 runs.

	Random networks	Evolved networks
Number of attractors	3.98 ± 0.02	2.12 ± 0.01
Largest basin size	$47\,800 \pm 100$	$57\,100 \pm 100$
Fitness score	$40\,300 \pm 200$	65 536
No. of evolution steps		2.07 ± 0.02

Next, it is interesting to look at dynamical properties and how they change during the course of the evolution process. In Table I, we compare random networks with networks that have undergone the evolution process for an average connectivity of $\langle k\rangle=3$ (results are typical for any value of $\langle k\rangle$). One can see that the average number of attractors has decreased and that the size of the largest basin has increased at the same time. Again, these significant effects take place within very few evolution steps. Thus, we find that the dynamical landscape of a threshold network can be significantly altered by only a few mutations of the network topology. Stability of the attractor landscape can be achieved without significant changes of the overall network structure.

So far we have not constrained the dynamics in any way, so the evolved networks might show different dynamical behavior from the original networks. If we think of attractors as a function performed by a genetic network, we should restrict evolution to networks that are able to reproduce the original attractor dynamics.

This leads to a modified selection criterion with the following target: We choose the largest attractor of the original network as the "functional attractor" and require stabilization of this attractor. If it is a fixed point or a stable limit cycle, there is trivially nothing to do in the evolution, so we just discard these networks and create a new one until we find a network with an unstable largest attractor. During evolution, every mutant has to reproduce this attractor. This means that, starting at one step of the attractor cycle, the dynamics of the original network and of the mutant have to be exactly the same. If the mutant does not reproduce the attractor, it is immediately discarded. We do not request that the networks reproduce the transient states as this constraint is too strict and disallows practically every mutation.

The fitness score is given by the basin size of the functional attractor or is 0 if the functional attractor is unstable. We have employed two different selection criteria: strict or neutral selection. In the strict selection scheme, a network is only accepted if it increases the fitness score, whereas in the neutral selection a larger or equal fitness score suffices. This means that in the strict scheme, the stabilization has to occur within a single rewiring, whereas the neutral criterion allows for a random walk through the space of networks that exhibit the functional attractor. The evolution process is complete as soon as the functional attractor is stable with a basin size of half the total state space, which makes the functional attractor the dominant dynamical expression pattern. In Fig. 2, we show the results of the evolution processes using the functional attractor criterion for a network size of $N=16$ and 1000 attempted evolution runs. The ratio of networks that can be stabilized in both selection schemes is plotted against the average connectivity of the networks. For each evolution step, we have attempted 20 000 mutations before marking a network as not evolvable toward stability (this simulation parameter does not influence the results as long as it is sufficiently high). In the neutral selection, a stable network has to be found within 10^6 mutation attempts during the full evolution run.

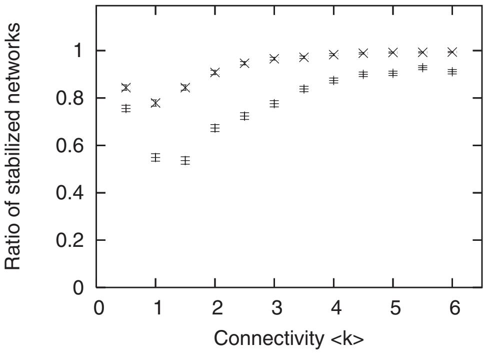

FIG. 2. Ratio of networks stabilized in the evolution against average connectivity. $N=16$, every data point averaged over 1000 evolution runs for neutral mutations (×) and single link rewirings (+).

First, one can see that even in the single-step evolution (points marked by +), more than half of all networks can be stabilized. For very low connectivities as well as connectivities above 3, more than $\frac{3}{4}$ of all networks fulfill the criterion. In the case of neutral selection (points marked by ×), this ratio is even higher. Especially for networks of connectivities around 1.5, the probability of evolving toward a stable realization is significantly increased. For connectivities above 2, practically every network can be stabilized using this evolution process.

There is a dip at connectivities around 1 in the single-step evolution and around 1.5 in the neutral selection. The reason for this is that at small connectivities, attractor cycles are

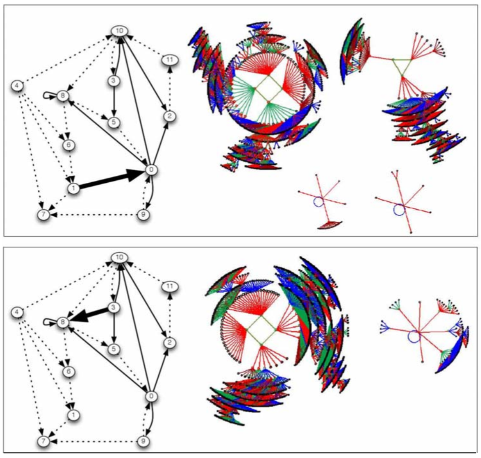

FIG. 3. (Color online) Attractor stabilization can be achieved by small topological changes. See text for details.

caused by short loops in the network topology—at very low connectivities unstable dynamics caused by two nodes influencing each other can often be stabilized by the formation of a self-loop of one of these nodes. At higher connectivities, larger dynamical cores emerge, which are, however, still dependent on a small number of links and cannot easily be stabilized. At even larger connectivities, these dynamical cores are often formed by dense connections of many links, some of which are dispensable and can be rewired without destroying the attractor.

In Fig. 3, we show an example of an evolutionary process for a network with 12 nodes and $\langle k \rangle = 2$. The network structure as well as the full synchronous attractor landscape are shown, before (top) and after (bottom) a single mutation. In the attractor landscape figure, each network state is represented by a dot that is connected to the concurrent state by a line. The central shape denotes the attractor. All four attractors of the original network are unstable. After mutation (depicted by thick lines), only two attractors remain. The functional attractor with a cycle length of 4 is now stable. One can see how the single mutation dramatically affects the attractor landscape of the network.

Three implications of our results are at hand. First, we find that the topological features of a network do not strictly dictate the stability of the resulting network dynamics. Small changes in the rewiring can have dramatic effects on the attractor landscape, including complete stabilization. Second, the (synchronous) state sequence of an attractor does not determine the stability. Even within small topological changes, it is often possible to find networks that exhibit the same attractor, but perform it in a reliable way. And third, neutral mutations ease network evolution and increase the fraction of networks that can be stabilized.

This work was supported by Deutsche Forschungsgemeinschaft Grants No. BO1242/5-1 and No. BO1242/5-2.

[1] C. V. Rao, D. M. Wolf, and A. P. Arkin, Nature **420**, 231 (2002).

[2] M. Thattai and A. van Oudenaarden, Proc. Natl. Acad. Sci. U.S.A. **98**, 8614 (2001).

[3] H. H. McAdams and A. Arkin, Trends Genet. **15**, 65 (1999).

[4] S. S. Shen-Orr, R. Milo, S. Mangan, and U. Alon, Nat. Genet. **31**, 64 (2002).

[5] M. Aldana and P. Cluzel, Proc. Natl. Acad. Sci. U.S.A. **100**, 8710 (2003).

[6] K. Klemm and S. Bornholdt, Proc. Natl. Acad. Sci. U.S.A. **102**, 18414 (2005).

[7] A. Wagner, Evolution (Lawrence, Kans.) **50**, 1008 (1996).

[8] S. Bornholdt and K. Sneppen, Proc. R. Soc. London, Ser. B **267**, 2281 (2000).

[9] M. Aldana, E. Balleza, S. Kauffman, and O. Resendiz, J. Theor. Biol. **245**, 433 (2007).

[10] A. Szejka and B. Drossel, Eur. Phys. J. B **56**, 373 (2007).

[11] S. Ciliberti, O. C. Martin, and A. Wagner, PLOS Comput. Biol. **3**, e15 (2007).

[12] F. Greil and B. Drossel, Phys. Rev. Lett. **95**, 048701 (2005).

[13] K. Klemm and S. Bornholdt, Phys. Rev. E **72**, 055101(R) (2005).

[14] S. Braunewell and S. Bornholdt, J. Theor. Biol. **245**, 638 (2007).

[15] S. A. Kauffman and R. G. Smith, Physica D **22**, 68 (1986).

[16] M. D. Stern, Proc. Natl. Acad. Sci. U.S.A. **96**, 10746 (1999).

[17] N. Kashtan and U. Alon, Proc. Natl. Acad. Sci. U.S.A. **102**, 13773 (2005).

[18] P. Oikonomou and P. Cluzel, Nat. Phys. **2**, 532 (2006).

[19] S. Bornholdt and K. Sneppen, Phys. Rev. Lett. **81**, 236 (1998).

[20] W. McCulloch and W. Pitts, Bull. Math. Biophys. **5**, 115 (1943).

[21] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang, Proc. Natl. Acad. Sci. U.S.A. **101**, 4781 (2004).

[22] M. I. Davidich and S. Bornholdt, e-print arXiv:0704.2200.

[23] K. E. Kürten, J. Phys. A **21**, L615 (1988).

[24] K. E. Kürten, Phys. Lett. A **129**, 157 (1988).

[25] H. Hirata, S. Yoshiura, T. Ohtsuka, Y. Bessho, T. Harada, K. Yoshikawa, and R. Kageyama, Science **298**, 840 (2002).