

## Distributed robustness in cellular networks: insights from synthetic evolved circuits

Javier Macia and Ricard V Solé

*J. R. Soc. Interface* 2009 **6**, 393-400 first published online 16 September 2008  
doi: 10.1098/rsif.2008.0236

### References

[This article cites 30 articles, 8 of which can be accessed free](#)  
<http://rsif.royalsocietypublishing.org/content/6/33/393.full.html#ref-list-1>

### Subject collections

Articles on similar topics can be found in the following collections

[biocomplexity](#) (6 articles)  
[systems biology](#) (26 articles)  
[systems theory](#) (1 articles)

### Email alerting service

Receive free email alerts when new articles cite this article - sign up in the box at the top right-hand corner of the article or click [here](#)

To subscribe to *J. R. Soc. Interface* go to: <http://rsif.royalsocietypublishing.org/subscriptions>

# Distributed robustness in cellular networks: insights from synthetic evolved circuits

Javier Macia<sup>1</sup> and Ricard V. Solé<sup>1,2,\*</sup>

<sup>1</sup>ICREA-Complex Systems Lab, Universitat Pompeu Fabra,  
Parc de Recerca Biomedica de Barcelona, Dr Aiguader 80, 08003 Barcelona, Spain

<sup>2</sup>Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, NM 87501, USA

Evolved natural systems are known to display some sort of distributed robustness against the loss of individual components. **Such type of robustness is not just the result of redundancy. Instead, it seems to be based on degeneracy, i.e. the ability of elements that are structurally different to perform the same function or yield the same output.** Here, we explore the problem of how relevant is degeneracy in a class of evolved digital systems formed by NAND gates, and what types of network structures underlie the resilience of evolved designs to the removal or loss of a given unit. It is shown that our fault tolerant circuits are obtained only if robustness arises in a distributed manner. No such reliable systems were reached just by means of redundancy, thus suggesting that reliable designs are necessarily tied to degeneracy.

**Keywords:** evolvable hardware; redundancy; degeneracy; robustness; fault tolerance

## 1. INTRODUCTION

One remarkable feature of many biological systems is the presence of a high degree of robustness against perturbations. Such robustness appears at multiple scales (Alon *et al.* 1999; Tononi *et al.* 1999; Edelman & Gally 2001; Gibson 2002; Krakauer & Plotkin 2002; Li *et al.* 2004; Jen 2005; Wagner 2005). Specifically, it is often found that temporal failure or permanent loss of some components has very often little or no impact on overall performance. In this context, such entities as a whole are able to cope with a changing world even under the loss of single units. A standard illustration of such robustness (or fault tolerance) is provided by gene knockouts through directed homologous recombination. In a large number of cases (close to 30%), little or no phenotypic effects are observed (Melton 1994). What is more surprising, it was shown that the mechanisms underlying such reliable behaviour are not based on redundancy. By redundancy, we refer to the presence of multiple copies of a given component: the failure of one of them would be compensated by another identical (isomorphic) copy. Instead, robustness in biology is largely associated with a distributed property, which has been dubbed either *degeneracy* (Tononi *et al.* 1999; Edelman & Gally 2001) or *distributed robustness* (Wagner 2005). By degeneracy, we refer to ‘the ability of elements that are structurally different to perform the same function’ (Tononi *et al.* 1999; Edelman & Gally 2001).

The problem of robustness was a hot topic in the 1950s, in parallel with the design of the first electronic computers. Back then, vacuum tube technology was

rather unreliable and John von Neumann and others (von Neumann 1952; Cowan & Vinograd 1963) explored the problem of how to design reliable computers from unreliable elements. Most of this work was based on digital designs described in terms of logic gates (figure 1*a*). The main conclusion from these studies was that a high degree of redundancy was required in order to achieve such a goal. Since nature seems to deal with faulty behaviour by using non-redundant mechanisms, something different must be at stake. Exploring the problem of degeneracy involves a number of difficulties. Since redundancy does not explain robustness (at least not most of it), we cannot understand the problem in terms of repeated, dissociated pieces. In this paper, we want to address this problem using evolved synthetic circuits performing simple computations.

Digital and switching circuits have been widely used in modelling gene and signalling networks (Kauffman 1962, 1993; Wuensche & Lesser 1992; Mendoza & Alvarez-Buylla 1998; Mendoza *et al.* 1999; Astor & Adami 2000; Kauffman *et al.* 2003; Solé *et al.* 2003; Weiss *et al.* 2003; Sauro & Khodolenko 2004; Simpson *et al.* 2004; Alvarez-Buylla *et al.* 2006; Braunewell & Bornholdt 2007; Willadsen & Wiles 2007), as well as other more general problems concerning the evolution of technology (Arthur & Polak 2006). Evolved circuits provide a good framework where our questions can be explored in a sensible way (Koza 1992; Miller *et al.* 2000). These systems, resulting from artificial evolution, allow one to obtain new designs without direct human intervention, in many cases displaying a higher efficiency (Koza 1992). Our goal here is the generation of digital circuits by evolution under different external conditions, with the purpose of

\*Author and address for correspondence: ICREA-Complex Systems Lab, Universitat Pompeu Fabra, Parc de Recerca Biomedica de Barcelona, Dr Aiguader 80, 08003 Barcelona, Spain (ricard.sole@upf.edu).

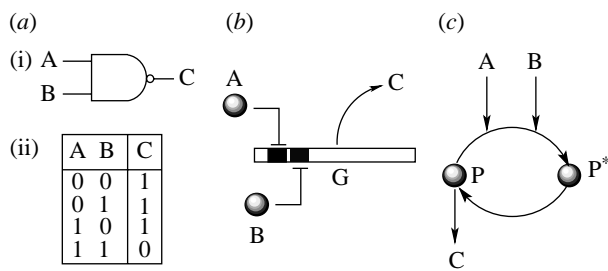


Figure 1. A logic gate, such as the NAND gate shown here, can be easily implemented using a molecular system. In (a) the standard symbol for the (i) NAND gate and (ii) its Boolean table representation are shown. In (b) is shown an example of the molecular implementation based on two proteins (A and B, which can be present or absent) activating a given gene coding for a protein C which defines the output of the gate. Moreover, signalling cascades also allow defining logic blocks. In (c) we give an example of such scenario for our NAND system. Here a protein P can be activated if two inputs A and B (external signals) are present. In that case, it makes a transition  $P \rightarrow P^*$  to an active form. If  $C = [P]$  is the measured output, then a NAND gate is obtained.

exploring the emergence of fault tolerance and how it relates to redundancy and degeneracy. Although the digital metaphor has some limitations, it has been widely used in computational systems biology to address very diverse types of questions. It seems to define an appropriate level of description to the switching behaviour found in many biological systems, from gene regulation to cell signalling (figure 1b,c). The study of the behaviour of these models has been shown to provide deep insight into the origins and importance of robustness (Bornholdt & Sneppen 2000; Klemm & Bornholdt 2005; Ciliberti *et al.* 2007; Fernandez & Solé 2007). In this context, although previous studies have shown that robust structures emerge as a consequence of evolutionary rules, the exact origin of such robustness is often missing. Here, using the formal definitions introduced by Tononi *et al.* (1999) and Edelman & Gally (2001), we show that robust designs are achieved by means of distributed robustness.

## 2. MEASURING DISTRIBUTED ROBUSTNESS

A first step before we present our results on evolved networks is to properly define a set of quantitative measures of network redundancy and degeneracy. To this goal, we will make use of previous measures from Tononi *et al.* (1999) and Edelman & Gally (2001). These information-based measures can be quantified using statistical measures of entropy and mutual information (Ash 1965; Adami 1998). We build these measures on a network  $X$  of  $Z$  interacting units where some computation is being performed. This is the case of logic circuits, such as the one shown in figure 2a. Here, each unit is a NAND gate. The reason for choosing this particular gate is that it allows one to build any possible digital circuit. Specifically, together with the so-called NOR gate, it is one of the two sole sufficient operators that can be used to express all of the Boolean functions of propositional logic (<http://en.wikipedia.org/wiki/Shefferstroke>).

Several measures can be defined on  $X$  based on information theory. An appropriate combination of them allows one to properly define robustness and measure it. The basic measure of information theory is the entropy, defined as (Ash 1965; Adami 1998)

$$H(X) = - \sum_{x \in X} p(x) \log p(x), \quad (2.1)$$

for one single variable  $x$ , and similarly we have

$$H(X, Y) = - \sum_{x \in X} \sum_{y \in Y} p(x, y) \log p(x, y), \quad (2.2)$$

for two variables  $x$  and  $y$ . Here,  $p(x)$  is the probability distribution for the possible values of  $x$  and  $p(x, y)$  is the joint probability distribution associated with  $(x, y)$ -pairs. From these basic measures, it is possible to define the mutual information  $I(X, Y)$  in terms of the entropies as

$$I(X, Y) = H(X) + H(Y) - H(X, Y). \quad (2.3)$$

Mutual information is the key quantity in theoretical approaches to the study of communication systems (Ash 1965; Adami 1998). It measures the information transmitted through the underlying channel. It has also been used as a measure of complexity, since it also describes the presence of (non-trivial) correlations (Solé & Goodwin 2001).

In order to define robustness, we need to provide a quantitative measure where interdependencies among not only elements but also subsets are properly weighted. Such measure must be able to quantify to what extent information transfer among different parts overlaps. Since the success of a given computation is tied to the generation of a right output, the definition of degeneracy (distributed robustness) must somehow incorporate the relation between different subsets and the set of output units (Tononi *et al.* 1998).

In their analysis of robustness, Tononi *et al.* (1999) define the degeneracy  $D_Z$  of the system as

$$D_Z(X) = \sum_{k=1}^Z \left[ \langle I(X_i^k, O) \rangle - \frac{k}{Z} I(X, O) \right], \quad (2.4)$$

which we adopt here as our definition of robustness (and in the following we will use both terms as equivalent). Here,  $X_i^k$  represents the  $i$ th subset of  $k$  elements, which is possible to build from the  $Z$  elements of the network and  $O$  is the subset of elements which form the output layer. The average  $\langle \rangle$  is computed over all the possible subgroups  $X_i^k$  with size  $k$  in which we can divide the system. As defined, it compares the average information transfer between every subset of size  $k$  and the expected value of the whole information between system and output, weighted by  $k/Z$ . A different (and more convenient) way of writing this function reads

$$D_Z(X) = \frac{1}{2} \sum_{k=1}^Z \left[ \langle I(X_i^k, O) + I(\hat{X}_i^k, O) - I(X, O) \rangle \right]. \quad (2.5)$$

In this expression, the balance of mutual information takes into account the possible overlapping of the information being processed by a subset  $X_i^k \subset X$

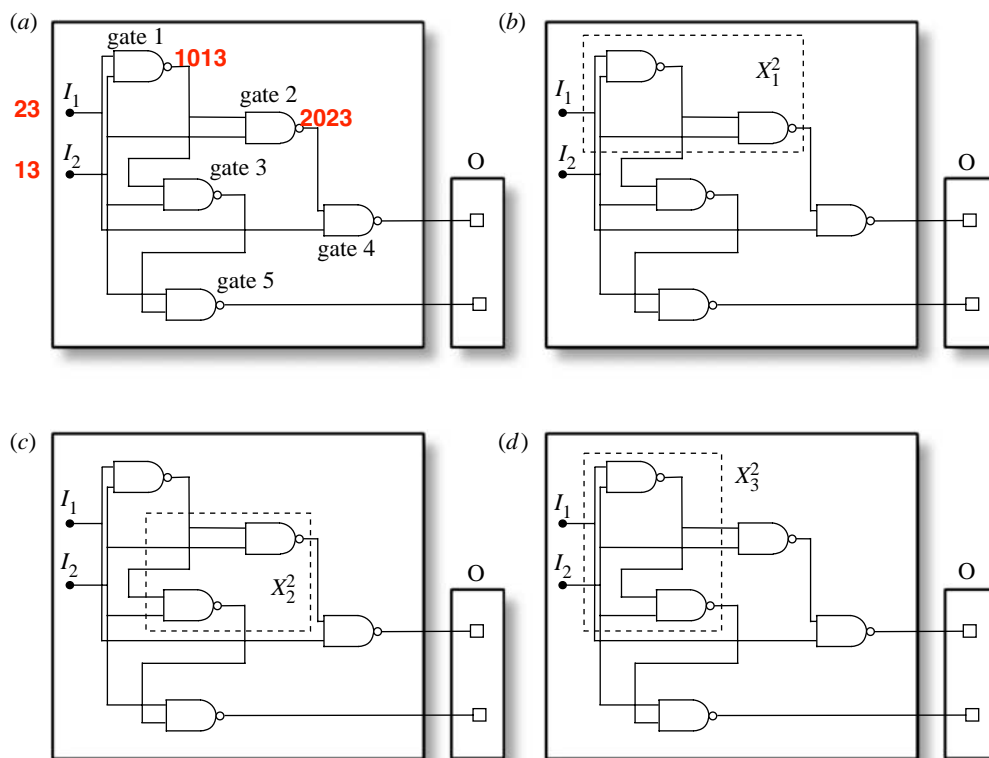


Figure 2. (a) An example of a small digital circuit. Gates 1, 2 and 3 are the elements of the set  $X$ . Gates 4 and 5 define the output layer  $O$ . (b–d) Three possible subsets of size two that can be constructed (see text).

formed by  $k$  elements and the rest of the network, i.e.  $\hat{X}_i^k = X - X_i^k$ . In the case of two independent subgroups, the balance is simply  $I(X_i^k, O) + I(\hat{X}_i^k, O) - I(X, O) = 0$ . This case corresponds to the lower bound of degeneracy. In any other case, it measures the overlap between subgroups. The upper bound corresponds to the extreme case of full degeneracy. In this case, the mutual information between one subgroup of the network and the output layer must be similar to the mutual information between the rest of the network and the output layer and similar to the mutual information between the total network and the output layer:

$$I(X_i^k, O) \approx I(\hat{X}_i^k, O) \approx I(X, O). \quad (2.6)$$

Building a network with such extreme degeneracy is likely to be impossible, but considering this situation it is possible to define an upper bound for degeneracy as

$$D_Z(X) \leq \frac{Z}{2} I(X, O) = D_Z^*, \quad (2.7)$$

which is obtained from (2.4) using the condition given in (2.5).

These measures allow one to unambiguously determine the presence and impact of degeneracy, but they are also involved and computationally costly. For the example chosen in figure 2, the mutual information for the subset  $X_1^2$  (gates 1 and 2) is  $\langle I \rangle = 0.21$ . This value means that part of the computation performed by the network from the input to the output involves some overlap between the subset  $X_1^2$  and the rest of the network  $X - X_1^2$  (gate 3). In other words, two subsets of the network that are structurally different perform, at least partially, the same computation process.

To compute the degeneracy values with (2.4), we need to calculate the average value of the mutual information for all the possible subsets for a given size (in the example if the size is 2, there are three possible subsets depicted in figure 1b–d) and perform the sum over all possible sizes, from  $k=1$  to  $Z$ . Such a scenario implies that, even for small networks, we require a very large number of subset combinations to be considered. This limits the total size of circuits that can be used in our analysis, which in our study is limited to  $Z=15$ .

Additionally, we can also estimate the redundancy of our system using the following definition (Tononi *et al.* 1999):

$$R_Z(X) = \left[ \sum_{i=1}^Z I(X_i, O) \right] - I(X, O). \quad (2.8)$$

This expression measures the overlap of the information processed by one given element and the rest of the network. If the different elements of the network are independent, then  $R_Z(X) = 0$ . Otherwise, some of the elements of the network are redundant.

Finally, a complexity measure can be also defined as (Tononi *et al.* 1999)

$$C_Z = \frac{1}{2} \sum_{i=1}^Z \langle I(X_i^k, \hat{X}_i^k) \rangle. \quad (2.9)$$

This expression measures the level of coherent integration of the different parts of the system (Tononi *et al.* 1998). It takes into account the average mutual information between each possible subset and the rest of the network. If the different subsets  $X_i^k$  have lower values of mutual information  $I(X_i^k, \hat{X}_i^k)$ , this implies that there is lower overlap between the information



Table 1. Different states for each element (here NAND gates) for the circuit shown in figure 2a.

23.13 input	1013 G <sub>1</sub>	2023 G <sub>2</sub>	2013 G <sub>3</sub>	3023 G <sub>4</sub>	3013 G <sub>5</sub>	output
00	1	1	1	1	1	11
01	1	0	0	1	1	11
10	1	1	1	0	1	01
11	0	1	1	0	0	00

processed by each subset and the rest of the network. In this case, the network has a low integration and does not work as a whole, but as a set of more or less independent parts. This case corresponds to small values of  $C_Z$ .

To illustrate the estimation of degeneracy as defined above, we can use a simple logic network as the one shown in figure 2a. It is formed by five NAND gates.

Table 1 shows all possible states for each gate for each input combination. Gates G<sub>4</sub> and G<sub>5</sub> define the output layer. To compute the degeneracy values, we must build all the subsets of NAND gates. It is possible to define three subsets of one gate each,  $X_i^1$  ( $i=1, 2, 3$ ), three subsets of two gates  $X_i^2$  ( $i=1, 2, 3$ ) as displayed in figures 2b–d and one subset of three gates  $X_1^3$ . Here, we illustrate the calculation method for one of these subsets of size two, i.e.  $X_1^2$ . First, we must calculate the different entropy values using the standard definitions (2.1) and (2.2). Table 2 shows the different states for the subset  $X_1^2$ , shown in figure 2b.

In table 3, we give the probabilities associated with the different possible input combinations and the associated entropy calculated directly from the possible states given in table 2.

Finally, different mutual information values can be computed from the previous measures. The results are summarized in table 4.

### 3. EVOLVING CIRCUITS

Different evolutionary rules have been used for the synthesis of circuits (Iba *et al.* 1997; Yu & Miller 2001; Arthur & Polak 2006; Banzhaf & Leier 2006). Some of these methods have been able to produce systems with high levels of fault tolerance and robustness. In our study, the circuits start from a randomly wired set of NAND gates. These circuits evolve until they are able to implement a certain  $N$ -input,  $M$ -output target binary function  $\phi$ , this function being a member of the set of possible Boolean functions described as a mapping

$$\phi_i: \Sigma^N \rightarrow \Sigma^M, \quad (3.1)$$

with  $\Sigma = \{0, 1\}$ . In our study, we will use  $N=3$  and  $M=2$ . The only topological limitation considered here is that there are no backward connections, in order to avoid temporal dependencies. In other words, only downstream effects are considered. The evolutionary rules follow previous work on circuit evolution (Miller & Hartmann 2001) with additional selection constraints in order to canalise the evolution process in two different scenarios: evolutions using correct computation and a fault tolerance measure as an additional constraint.

Table 2. Different possible states for the subset  $X_1^2$  and the rest of the network  $\hat{X}_1 = X - X_1^2$ .

input	$X_1^2(G_{1,2})$	$\hat{X}_1^2(G_3)$	$X(G_{1,2,3})$	$O(G_{4,5})$
00	11	1	111	11
01	10	0	100	11
10	11	1	111	01
00	01	1	011	00

#### 3.1. Selection for accurate computation

The goal of this selection using as fitness measure the correctness of the Boolean computation being implemented. It is thus defined in terms of the matching between the desired target function with no further constraints. The idea here is to see whether circuits that just correctly perform the desired computation are also robust *for free*. The target functions  $\phi$  are chosen at random: the set of outputs for each input combination are generated using 0 and 1 with equal probability. The steps of the algorithm are the following.

- Create a random generated population formed by  $S$  individuals (candidate circuits)  $X_1, \dots, X_S$ . Each one of these individuals is formed by  $Z$  randomly wired gates, with no backward connections. All circuits start with  $Z=5$  NAND gates. In each one of these nodes, there is a NAND gate of two inputs and one output. This population constitutes the starting generation. Here, given the computational constraints, we fixed the number of candidate solutions to  $S=10$ .
- The behaviour of each individual solution is simulated for the  $2^N$  possible inputs, comparing its outputs with the target function outputs. The fitness for the  $k$ th circuit  $X_k$  is defined as

$$F_k = 1 - \frac{1}{M2^N} \sum_{i=1}^{M2^N} |O_i^k - E_i^k|, \quad (3.2)$$

where  $\{O_i^k\}$  is the set of outputs of the circuit for the different inputs;  $M$  is the number of output bits; and  $\{E_i^k\}$  is the set of (expected) target function outputs.

- The individual with greater fitness is chosen to create a new generation formed by the selected individual and  $S-1$  random mutations of it. The random mutations (always respecting the backward patterning) can be: (a) elimination of an existing connection, with probability  $E_c$ , (b) creation of a new connection with probability  $C_c$ , (c) elimination of a node (gate removal) with probability  $I_n$ , and (d) creation of a new node (gate addition) with probability  $C_n$ . Here, we use  $E_c=0.8$ ,  $C_c=0.8$ ,  $I_n=0.3$  and  $C_n=0.6$ .
- Repeat step (ii) until a fitness  $F=1$  is reached.

#### 3.2. Selection using fault tolerance

This algorithm differs from the previous one in that selection of the most optimal individual of each generation introduces fault tolerance as an additional

Table 3. Entropy values for the different elements necessary for mutual information calculations. (The probability distributions are directly obtained from table 2.)

entropy	probability distribution		entropy value
$H(X_1^2)$	$p(11)=2/4, p(10)=1/4$	$p(01)=1/4$	1.04
$H(\hat{X}_1^2)$	$p(1)=3/4, p(0)=1/4$		0.56
$H(X)$	$p(111)=2/4, p(100)=1/4$	$p(011)=1/4$	1.04
$H(O)$	$p(11)=2/4, p(01)=1/4$	$p(00)=1/4$	1.04
$H(X_1^2, O)$	$p(11, 11)=1/4, p(01, 11)=1/4$	$p(10, 01)=1/4, p(11, 00)=1/4$	1.39
$H(\hat{X}_1^2, O)$	$p(1, 11)=1/4, p(0, 11)=1/4$	$p(1, 01)=1/4, p(1, 00)=1/4$	1.39
$H(X, O)$	$p(111, 11)=1/4, p(100, 11)=1/4$	$p(111, 01)=1/4, p(011, 00)=1/4$	1.39

Table 4. Mutual information calculations from entropy values of table 2.

mutual information ( $I$ )	$I$ value
$I(X_1^2, O)$	0.69
$I(\hat{X}_1^2, O)$	0.21
$I(X, O)$	0.69
$I(X_1^2, O) + I(\hat{X}_1^2, O) - I(X, O)$	0.21

requirement. Fault tolerance  $\rho_k$  is measured as

$$\rho_k = 1 - \frac{1}{ZM^{2N}} \sum_{p=1}^Z \left( \sum_{i=1}^{M^{2N}} |O_i - O_i^{kp}| \right), \quad (3.3)$$

where  $\{O_i^k\}$  is the output set of the circuit for the different inputs and  $\{O_i^{kp}\}$  is the output set of the  $k$ th circuit under a perturbation of the  $p$ th node. For each input, all the gates are in a binary state 0 or 1. **The perturbation consists of the inversion of the logical level of the gate located at the  $p$ th node. This perturbation is applied for each node  $p$ .**

In this case, the individual chosen to create a new generation must satisfy two conditions, namely greater fitness and higher fault tolerance. This additional condition of higher fault tolerance on the evolutionary process is prevailing (i.e. we give priority to  $F_k$  on top of  $\rho_k$  when  $F_k < 1$ ). The selection process, in this case, does not stop when  $F=1$ . Once  $F=1$ , the individuals can continue evolving until  $\rho$  reaches a stable value. In this approach, we want to compare the final designs with those resulting from the first evolution approach. Looking at the final circuits generated, we can measure redundancy and degeneracy and see how they contribute to the network robustness.<sup>1</sup>

#### 4. RESULTS

In figure 3, we show two examples of the synthetic circuits obtained from the two previous algorithms. Figure 3a shows a circuit obtained from fault tolerance selection, with a final value  $\rho=0.944$ . For the same Boolean function, figure 3b shows the corresponding

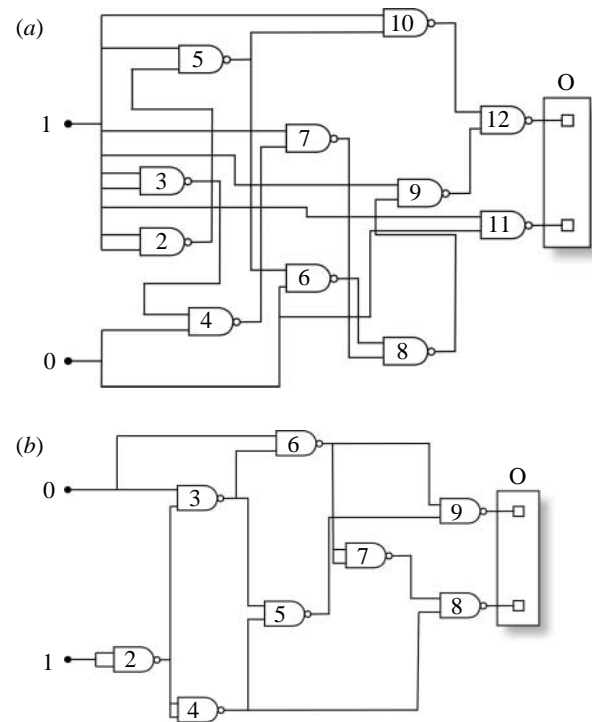


Figure 3. Examples of evolved circuits resulting from a process of evolutionary optimization. In (a) a circuit obtained by conditional evolution is shown. All gates are identical (NAND gates). The first two nodes are the input units. The circuit outputs are located after nodes 11 and 12. This circuit has a very high fault tolerance value ( $\rho=0.944$ ). In (b) we show a circuit obtained using a BFM. This circuit implements the same logic function as the circuit in (a) but involves a smaller fault tolerance of  $\rho=0.54$ .

outcome of the first evolution strategy with a much lower fault tolerance of  $\rho=0.54$ . As a general trend, we have found that evolved circuits from selection using just correct computation are typically smaller than the corresponding circuits selected using a fault tolerance evolutionary dynamics. This is an expected result, since it seems clear that robustness against gate failure must require some kind of internal capacity of reorganization. In order to see what emerges in terms of robustness, we measured redundancy and degeneracy in a set of evolved circuits under both types of selection pressures.

In figure 4a, we show the relationship between redundancy and degeneracy (our measure of robustness) for our evolved circuits under both modes of evolution. The figure clearly shows that circuits evolved

<sup>1</sup>In our numerical experiments, the specific set of parameters used was chosen in such a way that evolution was fast enough to be efficient. Other combinations provided similar results but displayed slower convergence.

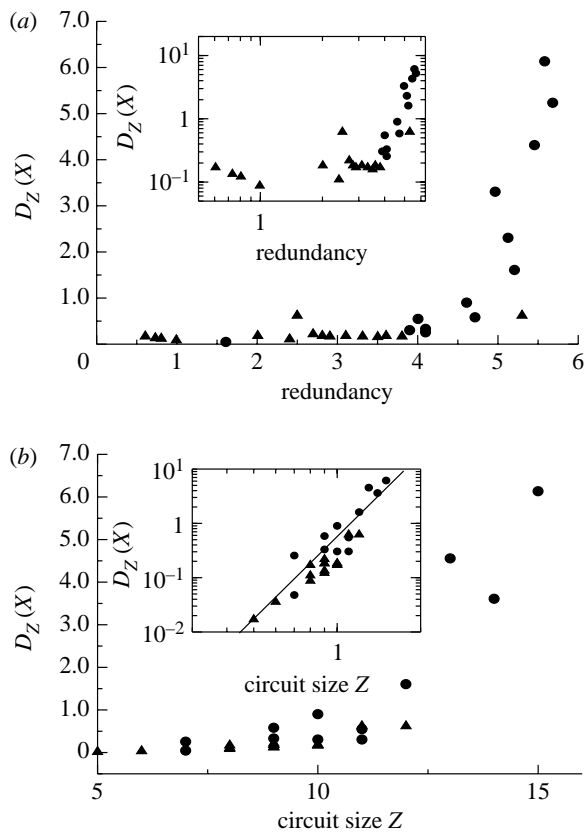


Figure 4. (a) Robustness, as measured in terms of degeneracy  $D_Z$ , increases with redundancy  $R_Z$  in a nonlinear fashion. Here, the first and second algorithms are indicated by means of triangles and circles, respectively. When the first is used, low levels of robustness are achieved, whereas evolution using fault tolerance leads to high robustness, provided that enough redundancy is at play. The inset shows the same results in log-log scale. In (b) we show the correlation between system size  $Z$  and fault tolerance. Here, we can clearly see that high levels of reliability are achieved by increasing the system's size. The inset shows the same plot in log-log scale, where we can appreciate a scaling behaviour.

under properly computing the Boolean function  $\phi$  have diverse levels of redundancy but very small degeneracy. Two relevant implications of this result can be obtained. The first is that an active selection for robustness may be required in order to achieve *highly* reliable designs. The second is that circuits obtained under selection for fault tolerance can achieve large levels of robustness and that such robustness is distributed, but requires some amount of internal redundancy (as shown by the rapid increase of  $D_Z$  at high  $R_Z$  values). **Let us note that the circuit with greater fault tolerance has a degeneracy value of  $D=6.28$ , closer to the upper bound value of  $D_Z^*=7.79$  obtained from (2.6).**

These results support the view that biological designs, which are expected to experience different sources of noise and perturbation, make use of degeneracy, instead of redundancy, in order to properly function. As far as we know, it is actually the first accurate demonstration of such relationship based on a well-defined measure of robustness. Moreover, as shown in figure 4b, there is a growing trend relating fault

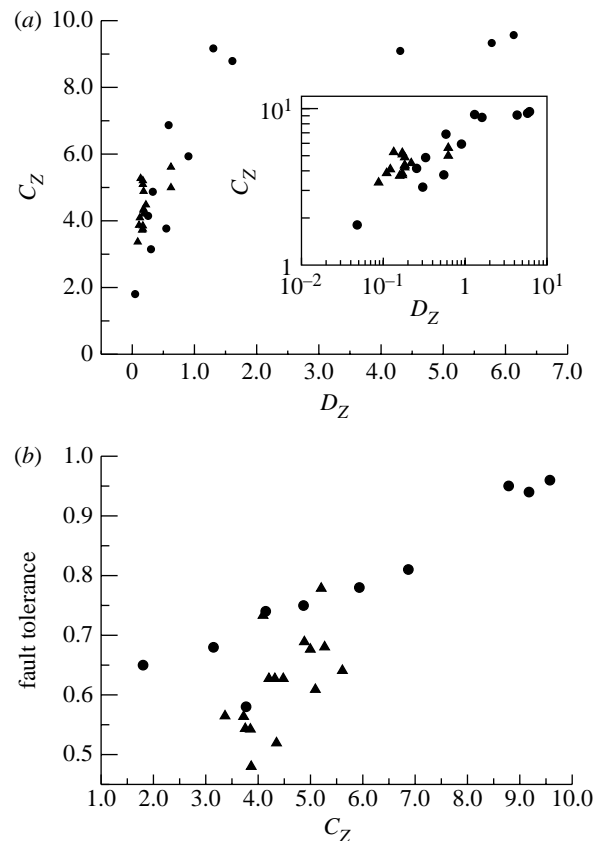


Figure 5. Correlations between circuit complexity and robustness (a) indicate that the system's integration rapidly increases with  $D_Z$ , reaching a plateau. On the other hand, fault tolerance seems well correlated with complexity, as shown in (b) thus indicating that an appropriate internal integration allows the system to be more reliable under failure of single units.

tolerance and the system size achieved through the evolutionary dynamics. Specifically, a power law increase, i.e.  $\rho \sim Z^\gamma$  with  $\gamma \approx 4$ . This trend is highly nonlinear, with a rapid increase in reliability as  $Z$  grows for conditional selection. This implies that high increases in robustness can be achieved by properly adding a single new element to the system.

In figure 5a,b, we also compare our measures of robustness and fault tolerance with the internal organization of the circuits as measured in terms of complexity. In our circuits, we see that large levels of complexity and circuit integration closely follow high degeneracy levels. This is consistent with previous findings using neural networks (Tononi *et al.* 1998). Such high complexity levels indicate that circuits evolve towards structures in which the different parts act with greater levels of coherent integration.

As figure 5a shows, an increase in degeneracy implies an increase in the complexity of the circuits, although a saturation is observed beyond some point (probably due to the small circuit sizes, which does not allow further increases). Such a relation does not happen with the redundancy (not shown): high redundancy levels are not consistent with more complex circuits. Similarly, complexity and fault tolerance are related, although the tendency towards higher fault tolerance with network integration is more obvious in circuits obtained through fault tolerant evolution (figure 5b).

## 5. DISCUSSION

The evolution of complex life forms seems to be inextricably tied to robustness. The problem of how complexity, modularity, adaptation and reliability are connected is an old one (Conrad 1983) but far from being closed (Hogeweg 2002; Lenski *et al.* 2003; Wagner *et al.* 2007). Capturing the details of such relations is a difficult task, and theoretical approaches require strong simplifications that are typically based on a computational picture of the system under consideration. In this paper, we have explored the possible origins of robustness in networks performing computations by using evolved artificial circuits. Although the approach taken is not a realistic biological implementation, it captures some of the logic of cellular networks at least at the level of computation. Nevertheless, our main goal was to determine the possible forms of reaching reliable systems under different selection pressures and understanding how robust designs can be obtained. Given the two potential origins of robust responses, namely either redundancy or degeneracy, we measured the resulting structures in order to determine what are the contributions of each to the observed levels of fault tolerance.

As found in biological systems, we can see that the origins of robustness against the failure of a given element are largely associated with a distributed mechanism of network organization. Both degeneracy and complexity (a measure of network integration and coherence) have been shown to reach high levels provided that redundancy is high enough. In this context, the use of an explicit measure of network reliability based on information exchanges between different subparts allows us to reach well-defined conclusions. Redundancy, on the other hand, is shown to be less relevant to the fault tolerant behaviour of our circuits. However, high redundancy might be needed in order to properly build degeneracy into the system, although what is likely to happen here is that an inevitable, positive correlation is at work.

An important point to be made here concerns the levels of fault tolerance achievable under the first selection criterion. Although maximal levels of fault tolerance have been found in connection to large degeneracy, high levels can also be achieved without explicitly introducing  $\rho$  in the selection process. This is shown for example in figure 5b, where we can see (triangles) that some networks achieving the correct computations are also fault tolerant. Although maximal levels might be interesting for designed systems, the implications for evolved networks are clear: fault tolerance might emerge ‘for free’ as a consequence of the evolutionary rules. This seems consistent with our previous work on evolving networks suggesting that the generative rules of network complexity might pervade many of their desirable functional traits (Solé & Valverde 2006, 2007).

We thank Alfred Borden and the members of the Complex Systems Lab for useful discussions. This work has been supported by the European Union within the 6th Framework Program under contracts FP6-001907 (Dynamically Evolving Large-scale Information Systems), FP6-002035 (Programmable Artificial Cell Evolution), by the James S. McDonnell Foundation and by the Santa Fe Institute.

## REFERENCES

- Adami, C. 1998 *Introduction to artificial life*. New York, NY: Springer.
- Alon, U., Surette, M. G., Barkai, N. & Leibler, S. 1999 Robustness in bacterial chemotaxis. *Nature* **397**, 168–171. (doi:10.1038/16483)
- Alvarez-Buylla, E., Benítez, M., Balleza-Dávila, E., Chaos, Á., Espinosa-Soto, C. & Padilla-Longoria, P. 2006 Gene regulatory network models for plant development. *Curr. Opin. Plant Biol.* **10**, 83–91. (doi:10.1016/j.pbi.2006.11.008)
- Arthur, W. B. & Polak, W. 2006 The evolution of technology within a simple computer model. *Complexity* **11**, 23–31. (doi:10.1002/cplx.20130)
- Ash, R. B. 1965 *Information theory*. New York, NY: Dover.
- Astor, J. C. & Adami, C. 2000 A developmental model for the evolution of artificial neural networks. *Artif. Life* **6**, 189–218. (doi:10.1162/106454600568834)
- Banzhaf, W. & Leier, A. 2006 Evolution on neutral networks in genetic programming. In *Genetic programming—theory and applications III* (eds T. Yu, R. Riolo & B. Worzel), pp. 207–221. Boston, MA: Kluwer Academic.
- Bornholdt, S. & Sneppen, K. 2000 Robustness as an evolutionary principle. *Proc. R. Soc. B* **267**, 2281–2286. (doi:10.1098/rspb.2000.1280)
- Braunewell, S. & Bornholdt, S. 2007 Superstability of the yeast cell-cycle dynamics: ensuring causality in the presence of biochemical stochasticity. *J. Theor. Biol.* **245**, 638–643. (doi:10.1016/j.jtbi.2006.11.012)
- Ciliberti, S., Martin, O. C. & Wagner, A. 2007 Robustness can evolve gradually in complex regulatory gene networks with varying topology. *PLoS Comp. Biol.* **3**, e15. (doi:10.1371/journal.pcbi.0030015)
- Conrad, M. 1983 *Adaptability*. New York, NY: Plenum Press.
- Cowan, J. D. & Vinograd, J. 1963 *Reliable computation in the presence of noise*. Cambridge, MA: MIT Press.
- Edelman, G. M. & Gally, J. A. 2001 Degeneracy and complexity in biological systems. *Proc. Natl Acad. Sci. USA* **98**, 13 763–13 768. (doi:10.1073/pnas.231499798)
- Fernandez, P. & Solé, R. V. 2007 Neutral fitness landscapes in signaling networks. *J. R. Soc. Interface* **4**, 41–47. (doi:10.1098/rsif.2006.0152)
- Gibson, G. 2002 Developmental evolution: getting robust about robustness. *Curr. Biol.* **12**, R347–R349. (doi:10.1016/S0960-9822(02)00855-2)
- Hogeweg, P. 2002 Computing an organism: on the interface between informatic and dynamic processes. *Biosystems* **64**, 97–109. (doi:10.1016/S0303-2647(01)00178-2)
- Iba, H., Iwata, M. & Higuchi, T. 1997 Machine learning approach to gate-level evolvable hardware. In *Proc. 1st Int. Conf. on Evolvable Systems: From Biology to Hardware* (eds T. Higuchi, M. Iwata & W. Liu). Lecture Notes on Computer Science, no. 1259, pp. 327–343. London, UK: Springer.
- Jen, E. (ed.) 2005 *Robust design*. New York, NY: Oxford University Press.
- Kauffman, S. A. 1962 Metabolic stability and epigenesis in randomly constructed genetic nets. *J. Theor. Biol.* **22**, 437–467. (doi:10.1016/0022-5193(69)90015-0)
- Kauffman, S. A. 1993 *The origins of order*. New York, NY: Oxford University Press.
- Kauffman, S. A., Peterson, C., Samuelsson, S. & Troein, C. 2003 Random Boolean network models and the yeast transcriptional network. *Proc. Natl Acad. Sci. USA* **100**, 14 796–14 799. (doi:10.1073/pnas.2036429100)
- Klemm, K. & Bornholdt, S. 2005 Topology of biological networks and reliability of information processing. *Proc. Natl Acad. Sci. USA* **102**, 18 414. (doi:10.1073/pnas.0509132102)



- Koza, J. R. 1992 *Genetic programming: on the programming of computers by means of natural selection*. Cambridge, MA: MIT Press.
- Krakauer, D. & Plotkin, J. 2002 Redundancy, antiredundancy and the robustness of genomes. *Proc. Natl Acad. Sci. USA* **99**, 1405–1409. (doi:10.1073/pnas.032668599)
- Lenski, R. E., Ofria, C., Pennock, R. T. & Adami, C. 2003 The evolutionary origin of complex features. *Nature* **423**, 139–144. (doi:10.1038/nature01568)
- Li, F., Long, T., Lu, Y., Ouyang, Q. & Tang, C. 2004 The yeast cell-cycle network is robustly designed. *Proc. Natl Acad. Sci. USA* **101**, 4781–4786. (doi:10.1073/pnas.0305937101)
- Melton, D. W. 1994 Gene targeting in the mouse. *Bioessays* **16**, 633–638. (doi:10.1002/bies.950160907)
- Mendoza, L. & Alvarez-Buylla, E. R. 1998 Dynamics of the genetic regulatory network for *Arabidopsis thaliana* flower morphogenesis. *J. Theor. Biol.* **193**, 307–319. (doi:10.1006/jtbi.1998.0701)
- Mendoza, L., Thieffry, D. & Alvarez-Buylla, E. R. 1999 Genetic control of flower morphogenesis in *Arabidopsis thaliana*: a logical analysis. *Bioinformatics* **15**, 593–606. (doi:10.1093/bioinformatics/15.7.593)
- Miller, J. & Hartmann, M. 2001 Evolving messy gates for fault tolerance: some preliminary findings. In *Proc. 3rd NASA Workshop on Evolvable Hardware*, pp. 116–123.
- Miller, J., Thompson, A., Thompson, P. & Fogarty, T. (eds) 2000 *Proc. 3rd Int. Conf. on Evolvable Systems: From Biology to Hardware*. Lecture Notes on Computer Science, no. 1801. Berlin, Germany: Springer.
- Sauro, H. H. & Khodolenko, B. N. 2004 Quantitative analysis of signaling networks. *Prog. Biophys. Mol. Biol.* **86**, 5–43. (doi:10.1016/j.pbiomolbio.2004.03.002)
- Simpson, M. L., Cox, C. D., Peterson, G. D. & Sayler, G. S. 2004 Engineering in the biological substrate: information processing in genetic circuits. *Proc. IEEE* **92**, 848–863. (doi:10.1109/JPROC.2004.826600)
- Solé, R. V. & Goodwin, B. C. 2001 *Signs of life: how complexity pervades biology*. New York, NY: Basic Books.
- Solé, R. V. & Valverde, S. 2006 Are network motifs the spandrels of cellular complexity? *Trends Ecol. Evol.* **21**, 419–422. (doi:10.1016/j.tree.2006.05.013)
- Solé, R. V. & Valverde, S. 2007 Spontaneous emergence of modularity in cellular networks. *J. R. Soc. Interface* **5**, 129–133. (doi:10.1098/rsif.2007.1108)
- Solé, R. V., Fernandez, P. & Kauffman, S. A. 2003 Adaptive walks in a gene network model of morphogenesis: insights into the Cambrian explosion. *Int. J. Dev. Biol.* **47**, 685–693.
- Tononi, G., Edelman, G. M. & Sporns, O. 1998 Complexity and coherence: integrating information in the brain. *Trends Cogn. Sci.* **2**, 474–484. (doi:10.1016/S1364-6613(98)01259-5)
- Tononi, G., Sporns, O. & Edelman, G. M. 1999 Measures of degeneracy and redundancy in biological networks. *Proc. Natl Acad. Sci. USA* **96**, 3257–3262. (doi:10.1073/pnas.96.6.3257)
- von Neumann, J. 1952 Probabilistic logics and the synthesis of reliable organisms from unreliable components, pp. 43–98. Lecture Notes, California Institute of Technology.
- Wagner, A. 2005 *Robustness and evolvability in living systems*. Princeton, NJ: Princeton University Press.
- Wagner, G., Pavlicev, M. & Cheverud, J. M. 2007 The road to modularity. *Nat. Rev. Genet.* **8**, 921–931. (doi:10.1038/nrg2267)
- Weiss, R., Basu, S., Hooshangi, S., Kalmbach, A., Karig, D., Mehreja, R. & Netravali, I. 2003 Genetic circuit building blocks for cellular computation, communications, and signal processing. *Nat. Comput.* **2**, 47–84. (doi:10.1023/A:1023307812034)
- Willadsen, K. & Wiles, J. 2007 Robustness and state-space structure of Boolean gene regulatory models. *J. Theor. Biol.* **249**, 749–765. (doi:10.1016/j.jtbi.2007.09.004)
- Wuensche, A. & Lesser, M. J. 1992 *The global dynamics of cellular automata*. SFI Studies in the Sciences of Complexity. Reading, MA: Addison-Wesley.
- Yu, T. & Miller, J. F. 2001 *Neutrality and the evolvability of Boolean function landscape*. Lecture Notes in Computer Science, no. 2038, pp. 204–217. New York, NY: Springer.