

Genetic networks with canalizing Boolean rules are always stable

Stuart Kauffman*[†], Carsten Peterson*^{‡§}, Björn Samuelsson*[‡], and Carl Troein*[‡]

*Department of Cell Biology and Physiology, Health Sciences Center, University of New Mexico, Albuquerque, NM 87131; and [‡]Complex Systems Division, Department of Theoretical Physics, Lund University, Sölvegatan 14A, S-223 62 Lund, Sweden

Communicated by Philip W. Anderson, Princeton University, Princeton, NJ, October 22, 2004 (received for review May 10, 2004)

We determine stability and attractor properties of random Boolean genetic network models with canalizing rules for a variety of architectures. For all power law, exponential, and flat in-degree distributions, we find that the networks are dynamically stable. Furthermore, for architectures with few inputs per node, the dynamics of the networks is close to critical. In addition, the fraction of genes that are active decreases with the number of inputs per node. These results are based upon investigating ensembles of networks using analytical methods. Also, for different in-degree distributions, the numbers of fixed points and cycles are calculated, with results intuitively consistent with stability analysis; fewer inputs per node implies more cycles, and vice versa. There are hints that genetic networks acquire broader degree distributions with evolution, and hence our results indicate that for single cells, the dynamics should become more stable with evolution. However, such an effect is very likely compensated for by multicellular dynamics, because one expects less stability when interactions among cells are included. We verify this by simulations of a simple model for interactions among cells.

With the advent of high-throughput genomic measurement methods, it will soon be within reach to apply reverse engineering techniques and map out genetic networks inside cells. These networks should perform a task, and, very importantly, be stable from a dynamical point of view. It is therefore of utmost interest to investigate the generic properties of networks models, such as architecture, dynamic stability, and degree of activation as functions of system size. Random Boolean networks have for several decades received much attention in these contexts. These networks consist of nodes, representing genes and proteins, connected by directed edges, representing gene regulation. The number of inputs to and outputs from each node, the in- and out-degrees, are drawn from some distribution.

It has been shown that with a fixed number, K , of inputs per node, such network models exhibit some interesting properties (1). Specifically, for $K = 2$, the dynamics is critical, i.e., right between stable and chaotic. Furthermore, one might interpret the solutions, i.e., fixed points and cycles, as different cell types. Being critical is considered advantageous, because it should promote evolution. These results were obtained with no constraints on the architectures and assume a flat distribution of the Boolean rules.

It appears natural to revisit the study of Boolean network ensembles, given recent experimental hints on network architectures and rule distributions. For transcriptional networks, there are indications from extracted gene–gene networks that, for both *Escherichia coli* (2) and yeast (3), the out-degree distribution is of power law nature. The in-degree distribution appears to be exponential in *E. coli*, whereas it may equally well be a power law in yeast. In ref. 4, stability properties of random Boolean networks were probed with power law in-degree distributions, and regions of robustness were identified.

The distribution of rules is likely to be structured and not random. Indeed, in a recent compilation (5) (see also ref. 6), all rules are canalizing (1); a canalizing Boolean function has at least one input, such that for at least one input value, the output value is fixed. It is not straightforward to generate biologically relevant canalizing

functions. In ref. 6, the notion of nested canalizing functions was introduced, which facilitates generation of rule distributions.

We find that networks with nested canalizing rules are stable for all power law and exponential in-degree distributions. Furthermore, as the degree distribution gets flatter, one moves further away from criticality. Also, the average number of active genes (fraction of genes that take the value “true”) is predicted for different powers.

There are experimental hints that in-degree distributions get flatter with genome size. This could be understood intuitively, because higher organisms in general have acquired more complexity in terms of redundancy in signal integration. In such a picture, our robustness analysis indicates that, with multicellular species, one would move away from critical dynamics, thereby making evolution less accessible. However, the picture is complicated by the presence of extracellular interactions. We model these with a simple scheme allowing for 5–10% extracellular traffic, and, not unexpectedly, the system, though still robust, moves toward criticality.

Methods and Models

Degree Distributions. Our results turn out to be qualitatively equivalent for power law, exponential, and flat in-degree distributions. (By flat, we mean a uniform distribution for up to K_{\max} inputs.) In what follows, we choose to illustrate the results with a power law (often denoted scale-free) distribution, with a cutoff in the number of inputs, K ,

$$p_K^N \equiv P(\#inputs = K) \propto \begin{cases} \frac{1}{K^\gamma} & \text{if } 1 \leq K \leq N \\ 0 & \text{otherwise,} \end{cases} \quad [1]$$

where N is the number of nodes. In yeast protein–protein networks (7) and also in other applications, e.g., the Internet and social networks, γ appears to lie in the range 2–3. In our calculations, we explore the region 0–5, varying N from 20 to infinity. The connectivity of gene–gene networks extracted from yeast (3) appears to behave like in Eq. 1 for the in-degree distributions, with an exponent γ in the range 1.5–2. For *E. coli* (2) data, an exponential fits somewhat better than a power law, but the data are statistically inconclusive. For mammalian cell cycle genes, slightly lower γ has been extracted (8). The average number of inputs varies with N and γ and grows with decreasing γ . For very high γ , it is essentially 1. In Fig. 1, typical network realizations for $N = 20$ are shown for $\gamma = 1, 2$, and 3, respectively.

Canalizing Boolean Rule Distributions. In most studies of Boolean models of genetic networks, all Boolean rules have been used (1). In a previous paper (6), we introduced nested canalizing rules and showed that it is possible to generate a distribution of such rules that fits well with rule data from the literature.

Freely available online through the PNAS open access option.

[†]S.K., C.P., B.S., and C.T. contributed equally to this work.

[§]To whom correspondence should be addressed. E-mail: carsten@thep.lu.se.

© 2004 by The National Academy of Sciences of the USA

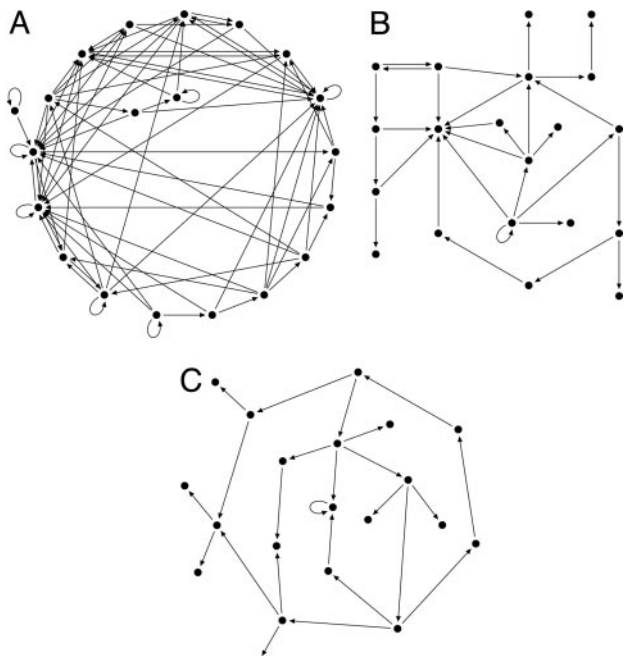


Fig. 1. Examples of $N = 20$ networks with $\gamma = 1$ (A), $\gamma = 2$ (B), and $\gamma = 3$ (C).

A canalyzing rule is a Boolean rule with the property that one of its inputs alone can determine the output value, for either “true” or “false” input. This input value is referred to as the canalyzing value, the output value is the canalyzed value, and we refer to the rule as being canalyzing on this particular input.

Nested canalyzing rules are a very natural subset of canalyzing rules, stemming from the question of what happens in the non-canalyzing case, that is, when the rule does not get the canalyzing value as input but instead has to consult its other inputs. If the rule then is canalyzing on one of the remaining inputs and again for that input’s noncanalyzing value, and so on for all inputs, we call the rule nested canalyzing. All but 6 of the ≈ 150 observed rules were nested canalyzing (6).

With I_m and O_m denoting the canalyzing and canalyzed values, respectively, and suitable renumbering of the inputs, i_1, \dots, i_K , the output, o , of a nested canalyzing rule can be expressed on the form

$$o = \begin{cases} O_1 & \text{if } i_1 = I_1 \\ O_2 & \text{if } i_1 \neq I_1 \text{ and } i_2 = I_2 \\ O_3 & \text{if } i_1 \neq I_1 \text{ and } i_2 \neq I_2 \text{ and } i_3 = I_3 \\ \vdots & \\ O_K & \text{if } i_1 \neq I_1 \text{ and } \dots \text{ and } i_{K-1} \neq I_{K-1} \text{ and } i_K = I_K \\ O_{\text{default}} & \text{if } i_1 \neq I_1 \text{ and } \dots \text{ and } i_K \neq I_K. \end{cases} \quad [2]$$

For each value of K , we generate a distribution of K -input rules, with the inputs to each rule taken from distinct nodes. All rules should depend on every input, and this dependency requirement is fulfilled if and only if $O_{\text{default}} \neq O_K$. Then it remains to choose values for I_1, \dots, I_K and O_1, \dots, O_K . These values are independently and randomly drawn with the probabilities

$$P(I_m = \text{true}) = P(O_m = \text{true}) = \frac{\exp(-2^{-m}\alpha)}{1 + \exp(-2^{-m}\alpha)} \quad [3]$$

for $m = 1, \dots, K$, where α is a constant. Eq. 3 can be seen as a way to put a penalty both on outputting true and on letting a true input determine the output. More precisely: Let f be the fraction of true outputs in the truth table, and let g be the

fraction of input states such that the first input that has its canalyzing value is true. Then, the probability to retrieve a specific rule is proportional to $\exp[-\alpha(f + g)/2]$.

Our rule distribution fits observed data well (6), given that $\alpha = 7$. For all generated distributions, we keep $\alpha = 7$. A high value of α means a high penalty on active genes, whereas $\alpha = 0$ means equal probabilities for activity and inactivity.

Robustness Calculations. We wish to address the question of robustness in network models. In a stable system, small initial perturbations should not grow in time. In ref. 6, this was probed by monitoring how the Hamming distance H between random initial states evolved in a “Derrida plot” (9). Specifically, the slope in the low- H region shows the fate of a small perturbation after a single time step. This implicitly assumes that true and false are equally probable in the initial states.

Our chosen distribution of Boolean rules will favor false node values. Preferably, a robustness measure should reflect the network properties in the vicinity of the equilibrium distribution, where the in- and out-degree distributions of true and false are identical (see Appendix A). We therefore define the robustness r_N for an ensemble of N -node networks as the average effect, after a single time step, of a small perturbation at this equilibrium distribution.

To compute r_N , we introduce the total sensitivity, $S(R)$, of a given K -input rule R . $S(R)$ is the sum of the probabilities that a single flipped input will alter the output of R . Thus,

$$S(R) = \sum_{j=1}^K P[R(i_1, \dots, i_{j-1}, 0, i_{j+1}, \dots, i_K) \neq R(i_1, \dots, i_{j-1}, 1, i_{j+1}, \dots, i_K)], \quad [4]$$

where the probability is calculated over the equilibrium distribution of input values, i_1, \dots, i_K . Then, r is given by $r = \langle S(R) \rangle$, where the average is taken over all rules in a given network (see Appendix A). This also allows us to calculate r when the rules are drawn from a distribution. Note that r is calculated without any assumption on how the inputs to the rules are chosen. This means that r stays the same for any output connectivity and for any way to build a network containing a certain set of rules. In other words, r is a strictly local stability measure that is independent of whether the network is divided into some kind of clusters or not.

Let S_K denote the average of $S(R)$ over a distribution of K -input rules. The average robustness of a randomly chosen network with N nodes is then given by

$$r_N = \sum_{K=1}^N p_K^N S_K. \quad [5]$$

See Appendix A on calculation of S_K for nested canalyzing rules. With the nested canalyzing rule distribution defined by Eq. 3, $S_K < 1$ for $K = 2, 3, \dots$, provided that $\alpha \neq 0$. S_1 is always 1, because every rule has to depend on all of its inputs. ($\alpha = 0$ yields $S_K = 1$ for all K .) This means that every network ensemble with the canalyzing rule distribution, and $\alpha \neq 0$, that not solely consists of one-input nodes, is stable in the sense that $r < 1$.

Number of Attractors. Attractors in the Boolean model can be seen as distinct cell types (1). It is, however, not straightforward to tell which attractors are biologically relevant. First, one can ask what cycle lengths are relevant. Second, the attractor basin sizes vary in a very broad range, and attractors with small attractor basins may be biologically irrelevant.

The broad distribution of attractor basin sizes also means that the number of attractors found by random sampling strongly depends on the number of samples (10).

We choose to monitor the number of attractors, $\langle C_L \rangle_N$, of different periods, L , using exact methods (for the limit $N \rightarrow \infty$) and full enumerations of the state space (for small networks).

Given that $r < 1$, which means that the network is subcritical, the average number of attractors of a certain length, L , will approach a constant, $\langle C_L \rangle_\infty$, as the system size, N , approaches infinity. We find an analytic expression for $\langle C_L \rangle_\infty$ and find that it is qualitatively consistent with results from a full state space search for $N = 20$.

To investigate the limit $N \rightarrow \infty$, we split up the robustness measure, r , into r^C and r^I , where r is the average number of outputs that, after one time step, will be affected by one flipped node. We define r^C and r^I as the numbers of nodes that, respectively, copy and invert the state of the flipped node. This means, e.g., that $r = r^C + r^I$. For convenience, we define $\Delta r = r^C - r^I$.

$\langle C_L \rangle_\infty$ can be expressed as a function of r_∞ and Δr_∞ for each L . For $L = 1, 2, 3$ we get

$$\langle C_1 \rangle_\infty = \frac{1}{1 - \Delta r_\infty}, \quad [6]$$

$$\langle C_2 \rangle_\infty = \frac{r_\infty - \Delta r_\infty + r_\infty \Delta r_\infty}{2(1 - r_\infty)(1 - [\Delta r_\infty]^2)}, \quad \text{and} \quad [7]$$

$$\langle C_3 \rangle_\infty = \frac{r_\infty^3 + (\Delta r_\infty)^3 - r_\infty^3 (\Delta r_\infty)^3}{3(1 - r_\infty^3)(1 - \Delta r_\infty)(1 - [\Delta r_\infty]^3)}. \quad [8]$$

See *Appendix B*, where the derivations needed to calculate $\langle C_L \rangle_\infty$ are presented.

The canalyzing rule distribution satisfies $r^C > r^I$, meaning that $\Delta r > 0$. This condition yields an increased number of fixed points and attractors of odd length, compared with the symmetric case $\Delta r = 0$.

It is interesting to note that the limit of the total number of attractors,

$$\langle C \rangle_\infty = \sum_{L=1}^{\infty} \langle C_L \rangle_\infty, \quad [9]$$

is convergent for $r < 1/2$ and divergent for $r > 1/2$. This transition occurs at $\gamma = 1.376$, with convergence below this value. See *Supporting Text*, which is published as supporting information on the PNAS web site, for details.

Tissue Simulations. In multicellular organisms, we expect communication among cells to influence the network dynamics. In the real world, there exist several different types of intercellular signaling. Here we make an initial exploration using a simple model. Nevertheless, we think the results reflect some core properties.

In our model, each cell communicates with its four nearest neighbors on a square lattice with periodic boundary conditions. All cells have the same genotype and hence identical internal network architecture and rules. Each connection in the network represents how a gene product influences the transcription of some gene. Some molecules or signals can cross cell membranes and possibly diffuse far, but we consider only the case of local cell-to-cell signaling at the level of specific gene–gene interactions. Specifically, a fraction, κ , of the connections are flagged as being intercellular, and for such connections, the value is true if any of the four neighbors has true.

For the overall robustness of such tissue networks, it is not sufficient to measure the robustness r , because r depends only on interactions during a single time step, during which a perturbation can propagate only to the nearest neighbor cells. Hence, we desire a multistep robustness measure, which requires simulations, be-

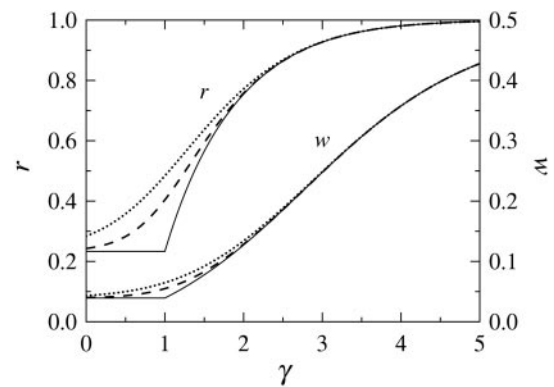


Fig. 2. Robustness, r , and the probability of a node being true, w , at the equilibrium distribution of Boolean values, as functions of the exponent γ in the in-degree distribution p_K (Eq. 1) for $N = 20$ (dotted), 100 (dashed), and ∞ (solid).

cause it is outside the scope of our analysis. Rather than following trajectories from random initial states, we have chosen to identify fixed points, perturb these randomly by Hamming distance $H(0) = 1$, and then track the mean of $H(t)$ for 20 time steps. In our simulations, we generate ensembles of networks using 5×5 lattices of cells, where each cell contains an identical network of $N = 50$ genes.

Results and Discussion

Three major findings emerge from Fig. 2, where the average robustness r and the fraction of active genes are shown as functions of γ in Eq. 1.

- (i) The dynamics of the networks is always stable, regardless of the power in the in-degree distribution.
- (ii) The stability of the networks increases with the average number of inputs to the nodes. For flatter distributions, r approaches the critical value, $r = 1$.
- (iii) The average number of active genes decreases with increasing in-degree.

Not unexpectedly, the number of attractors increases as the networks approach criticality (see Fig. 3). This increase is particularly rapid for long cycles. These results were obtained with analytical calculation and exhaustive enumeration of state space. Given the undersampling problems when simulating Boolean networks (10),

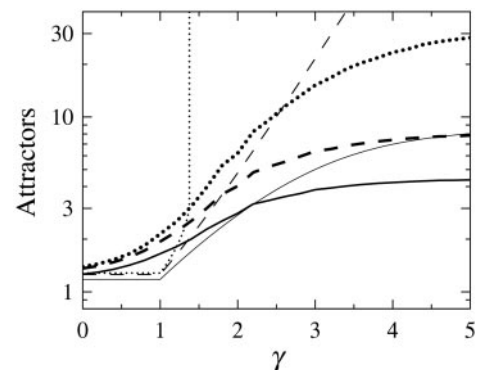


Fig. 3. The number of attractors as a function of the exponent γ in the in-degree distribution p_K (Eq. 1) for $N = 20$ (thick lines) and $N \rightarrow \infty$ (thin lines). The curves show the cumulative number of attractors of length L for $L = 1$ (solid), $L = 2$ (dashed), and $L = \infty$ (dotted). The values for $N \rightarrow \infty$ were calculated analytically, whereas the values for $N = 20$ are taken from full enumerations of the state space for at least 5,000 networks, with more networks at higher γ .

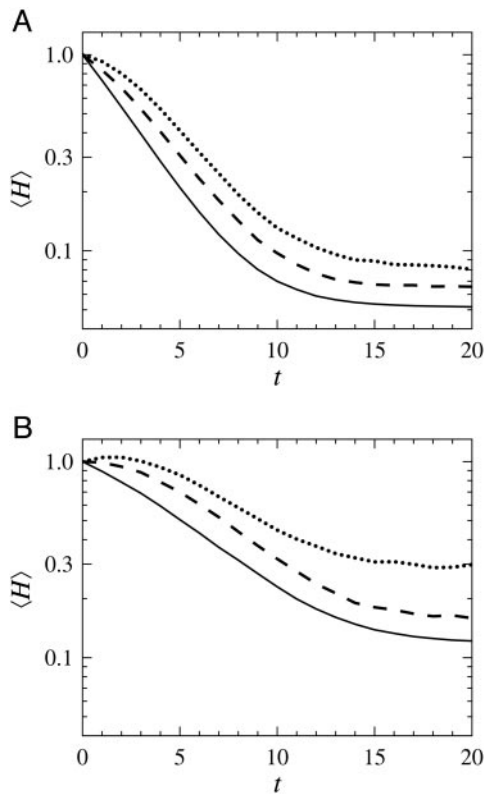


Fig. 4. The average time evolution of perturbed fixed points in 5×5 cell tissues with periodic boundary conditions and $N = 50$ nodes (over many network realizations). Simulations from random initial states in generated networks were used to locate fixed points, which were perturbed by toggling the value of a single node. The mean distances to the unperturbed fixed point, $\langle H(t) \rangle$, as given by 20 subsequent simulation steps, are shown for $\gamma = 2$ (A) and $\gamma = 3$ (B), for three different degrees of cell connectivity: $\kappa = 0$ (solid), 0.05 (dashed), and 0.1 (dotted).

the feasibility of analytical calculations is crucial for drawing the firm conclusions above. However, we did not attempt to extract the distribution of attractor basin sizes. In future research, it could be interesting to compare this distribution to the exact results for one-input networks in ref. 11.

The results in Figs. 2 and 3 are shown for power law in-degree distributions. However, they are quite general. The corresponding curves for other distributions exhibit very similar behavior, when the x axis (γ) has been transformed to appropriate parameters for other distributions. In all tested degree distributions, constant nodes ($K = 0$) are excluded. Recall that we also exclude rules that have redundant inputs. Thus, for low values of the average K , most of the rules will be copy or invert operators, which puts the network close to criticality. The strong stability for wide in-degree distributions, however, is a result of the canalizing property of the rules, which makes forcing structures (1) prevalent.

From analysis of network data from yeast (3) and the mammal cell cycle (8), there are indications that γ decreases with the number of genes. Within the framework of our results, this means that, as the genome size increases, the networks get more stable. However, with increased number of genes, multicellularity becomes common. Including interactions among cells should make the overall system less stable. Indeed, when investigating this issue by simulations of a simplified tissue model, the stability decreases with interconnectivity between the cells, κ , as can be seen from Fig. 4.

We predict how the average number of active genes increases with γ . This may not be easy to verify, given that such a number will depend upon experimental conditions. It should be pointed out that

the order of magnitude of active genes is set by the rule generator in Eq. 3, which is derived from fitting to the observed rules in ref. 5, many of which originate from *Drosophila melanogaster*. The fitted parameter α sets the scale of the fraction of active genes, with high α corresponding to low gene activity and vice versa. The qualitative behavior, however, is rather insensitive to the value of α .

Conclusion

We have designed and analyzed a class of Boolean genetic network models consistent with observed interaction rules. The emerging ensemble properties exhibit not only remarkable stability for the dynamics, but also several generic properties that make predictions, such as how stability varies with genome size, and how the number of active genes depends on the in-degree distribution. Because single-cell calculations are performed analytically, the results are transparent in terms of understanding the underlying dynamics.

Appendix A: Robustness

The stability measure r expresses the average number of perturbed nodes one time step after perturbing one node, given that the network has reached equilibrium in a mean field sense. Both the mean field equilibrium distribution (of true and false) and r are closely related to attractors in the true non-mean field dynamics (see Appendix B).

Let $W(w)$ denote the probability that a randomly chosen rule will output true, given that the input values are randomly and independently chosen with probability w to be true. Let $W_K(w)$ denote $W(w)$ when the selected rule has K inputs. Then, the equilibrium probability for an N -node network, w_{eq}^N , satisfies

$$w_{\text{eq}}^N = W(w_{\text{eq}}^N) = \sum_{K=1}^N P_K^N W_K(w_{\text{eq}}^N). \quad [10]$$

Eq. 10 can be solved numerically for nested canalizing rules, taking advantage of the fast (exponential) convergence of $W_K(w)$ as $K \rightarrow \infty$ and using standard (integration-based) methods to calculate the sum in the case that $N \rightarrow \infty$. Note that w_{eq}^N is referring only to the mean field equilibrium distribution, which is essentially the same as, but not identical to, the distribution of true and false after many time steps in a simulation.

For nested canalizing rules, $W_K(w)$ is given by

$$W_K(w) = \sum_{k=1}^K P_1^{\text{nc}} \cdots P_{k-1}^{\text{nc}} P_k^{\text{can}} P(O_k = \text{true}) + P_1^{\text{nc}} \cdots P_K^{\text{nc}} P(O_{\text{default}} = \text{true}), \quad [11]$$

where $P_k^{\text{can}} = P(i_k = I_k)$ and $P_k^{\text{nc}} = P(i_k \neq I_k)$. The input values, i_1, \dots, i_K , are true with probability w , whereas the corresponding probabilities for I_1, \dots, I_K and O_1, \dots, O_K are given by Eq. 3.

Let $r(R)$ denote the probability that the rule R depends on a randomly picked input, given that the other inputs are independently set to true with probability w_{eq}^N . We can express r for a specific N -node network with rules R_1, \dots, R_N as

$$r = \sum_{i=1}^N r(R_i) \frac{K(R_i)}{N} = \frac{1}{N} \sum_{i=1}^N S(R_i), \quad [12]$$

where $K(R_i)$ is the number of inputs to R_i . We have defined $S(R) = K(R)r(R)$, so that r is the average of $S(R)$ over all rules in the network. This is also valid for a distribution of networks, meaning that

$$r_N = \sum_{K=1}^N p_K^N S_K \quad [13]$$

for N -node networks, where S_K is the average of $S(R)$ when random K input rules are chosen.

Eq. 13 is exact, given that the state of the network is randomly picked from the mean field equilibrium distribution of true and false. Because the derivations are completely independent of the specific network architecture, this result holds for any procedure to generate architectures, so long as the average fraction of nodes with K inputs are given by p_K^N (over many network realizations).

For nested canalizing rules, we can calculate S_K as a sum of probabilities. If the rule is canalizing on input k , and changing i_k makes the rule canalize on input j , there is some probability that the output value changes. The case that the rule falls back to O_{default} corresponds to the last term in the square brackets.

$$S_K = \sum_{k=1}^{K-1} P_1^{\text{nc}} \cdots P_{k-1}^{\text{nc}} \left[\sum_{j=k+1}^K P_{k+1}^{\text{nc}} \cdots P_{j-1}^{\text{nc}} P_j^{\text{can}} P(O_k \neq O_j) + P_{k+1}^{\text{nc}} \cdots P_K^{\text{nc}} P(O_k \neq O_{\text{default}}) \right], \quad [14]$$

where $P_k^{\text{can}} = P(i_k = I_k)$ and $P_k^{\text{nc}} = P(i_k \neq I_k)$.

Let v and $V(v, w)$ denote the fraction of input and output values, respectively, that are toggled (from false to true or vice versa) when going one step forward in time, given that the fraction true input values is w both before and after the input is toggled. Then, $V(0, w) = 0$, because constant input renders constant output. A small change in v will change the output with r times the change v . This means that $\partial_v V(v, w_{\text{eq}}) \leq r$, where the inequality comes from the possibility that new changes undo previous ones as v is increasing. Combining these relations yields

$$V(v, w_{\text{eq}}) \leq rv, \quad [15]$$

which means that, for $r < 1$, $V(v, w_{\text{eq}}) \leq v$ with equality if and only if $v = 0$. Hence, frozen states, where the fraction of true nodes is w_{eq} , are the only solutions to the mean field dynamics, given that $r < 1$, which is true for the nested canalizing rule distribution.

Note that v can be seen as the distance, i.e., fraction of differing states, between two separate time series. Then, the mapping $v \mapsto V(v, w_{\text{eq}})$ gives the evolution of the distance during one time step. Similar calculations have been carried out in e.g., refs. 12 and 13, yielding results consistent with Eq. 15.

Appendix B: Attractors

To calculate the average number of attractors, we use the same approach as in ref. 10. This approach means that we first transform the problem of finding an L cycle to a fixed point problem, and then find a mathematical expression for the average number of solutions to that problem.

Assume that a Boolean network performs an L cycle. Then, each node performs one of 2^L series of output values. We call these L cycle series. Consider what a rule does when it is subjected to such L cycle series on the inputs. It performs some Boolean operation, but it also delays the output, giving a one-step difference in phase for the output L cycle series. If we view each L cycle series as a state, an L cycle turns into a fixed point (in this enlarged state space). $L\langle C_L \rangle_N$ is then the average number of input states (choices of L cycle series) for the whole network, such that the output is the same as the input.

Let \mathbf{Q} denote a specific choice of L cycle series for the network, and let Q_L^N be the set of all \mathbf{Q} , such that \mathbf{Q} is a proper

L cycle. A proper L cycle has no period shorter than L . The average number of L cycles is then given by

$$\langle C_L \rangle_N = \frac{1}{L} \sum_{\mathbf{Q} \in Q_L^N} P_L^N(\mathbf{Q}), \quad [16]$$

where $P_L^N(\mathbf{Q})$ denotes the probability that the output of the network is the same as the input \mathbf{Q} .

Because the inputs to each K input rule are chosen from a flat distribution over all nodes, $P_L^N(\mathbf{Q})$ is invariant with respect to permutations of the nodes. Let $\mathbf{n} = (n_0, \dots, n_{m-1})$ denote the number of nodes expressing each of the $m = 2^L$ series. For n_i , we refer to i as the index of the considered L cycle series. For convenience, let the constant series of false and true have indices 0 and 1, respectively. Then,

$$\langle C_L \rangle_N = \frac{1}{L} \sum_{\mathbf{n} \in \mathcal{P}_L^N} \binom{N}{\mathbf{n}} P_L^N(\mathbf{Q}), \quad [17]$$

where $\binom{N}{\mathbf{n}}$ denotes the multinomial $N!/(n_0! \cdots n_{m-1}!)$, and \mathcal{P}_L^N is the set of partitions \mathbf{n} of N such that $\mathbf{Q} \in Q_L^N$. That is, \mathbf{n} represents a proper L cycle.

Let $A_L^i(\mathbf{x})$ denote the probability that a randomly selected rule will output L cycle series i , given that the input series are selected from the distribution $\mathbf{x} = (x_0, \dots, x_{m-1})$. Because a node may not be used more than once as an input to a specific rule, $A_L^i(\mathbf{x})$ should also depend on N . However, the difference between allowing or not allowing coinciding inputs vanishes as N goes to infinity, because the output is effectively determined by relatively few inputs for nested canalizing rules.

Because these calculations aim to reveal the asymptotic behavior as $N \rightarrow \infty$, we allow for coinciding inputs in the following. Then, we get

$$P_L^N(\mathbf{Q}) = \prod_{\substack{0 \leq i < m \\ n_i \neq 0}} [A_L^i(\mathbf{n}/N)]^{n_i}. \quad [18]$$

By combining Eqs. 17 and 18 and applying Stirling's formula to $\binom{N}{\mathbf{n}}$, we get

$$\langle C_L \rangle_N \approx \frac{1}{L} \sum_{\mathbf{n} \in \mathcal{P}_L^N} \frac{\sqrt{2\pi N}}{\prod_{\substack{0 \leq i < m \\ n_i \neq 0}} \sqrt{2\pi n_i}} e^{N f_L(\mathbf{n}/N)}, \quad [19]$$

where

$$f_L(\mathbf{x}) = \sum_{\substack{0 \leq i < m \\ x_i \neq 0}} x_i \ln \left[\frac{1}{x_i} A_L^i(\mathbf{x}) \right]. \quad [20]$$

See Supporting Text regarding the use of Stirling's formula.

Eq. 20 can be seen as an average $\langle \ln[(1/x_i) A_L^i(\mathbf{x})] \rangle_i$ with weights x_0, \dots, x_{m-1} . Hence, the concavity of $x \mapsto \ln x$ yields

$$\begin{aligned} f_L(\mathbf{x}) &= \left\langle \ln \left[\frac{1}{x_i} A_L^i(\mathbf{x}) \right] \right\rangle_i \\ &\leq \ln \left\langle \frac{1}{x_i} A_L^i(\mathbf{x}) \right\rangle_i = \ln \sum_{\substack{0 \leq i < m \\ x_i \neq 0}} A_L^i(\mathbf{x}) \leq 0 \end{aligned} \quad [21]$$

with equality if and only if

$$\mathbf{x} = \mathbf{A}_L(\mathbf{x}), \quad [22]$$

where $\mathbf{A}_L(\mathbf{x}) = (A_L^0(\mathbf{x}), \dots, A_L^{m-1}(\mathbf{x}))$. Eq. 22 can be seen as a criterion for mean field equilibrium in the distribution of L cycle

