Journal Name

Dynamic Article Links ►

Cite this: DOI: 10.1039/c0xx00000x

www.rsc.org/xxxxxx

ARTICLE TYPE

Supplementary Information - Network function shapes network structure: the case of the *Arabidopsis* flower organ specification genetic network

Adrien Henry, Françoise Monéger, Areejit Samal and Olivier C. Martin

Received (in XXX, XXX) Xth XXXXXXXXX 20XX, Accepted Xth XXXXXXXXX 20XX
DOI: 10.1039/b000000x

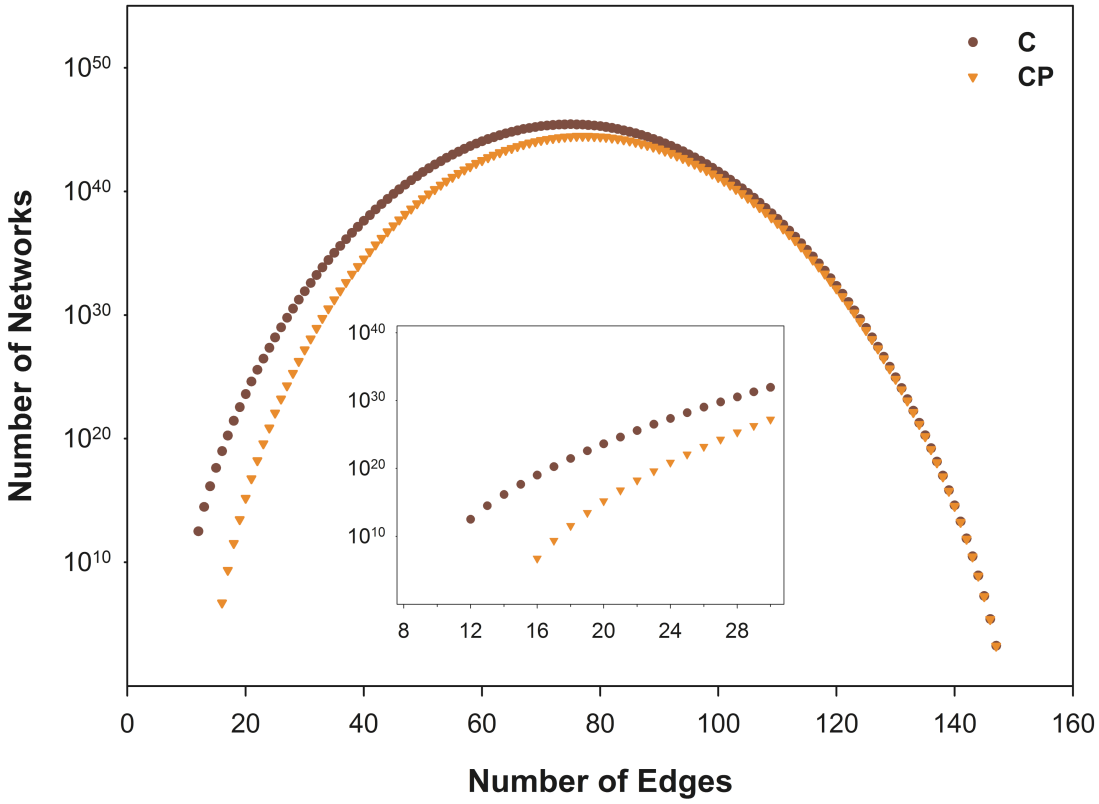

Fig.S1 Size of the ensembles *C* and *CP* as a function of number of edges in the network. The horizontal axis shows the number of edges in the network. The vertical axis shows the number of networks in logarithmic scale. Inset: Zoom to show the plot for the lowest values of the number of edges.

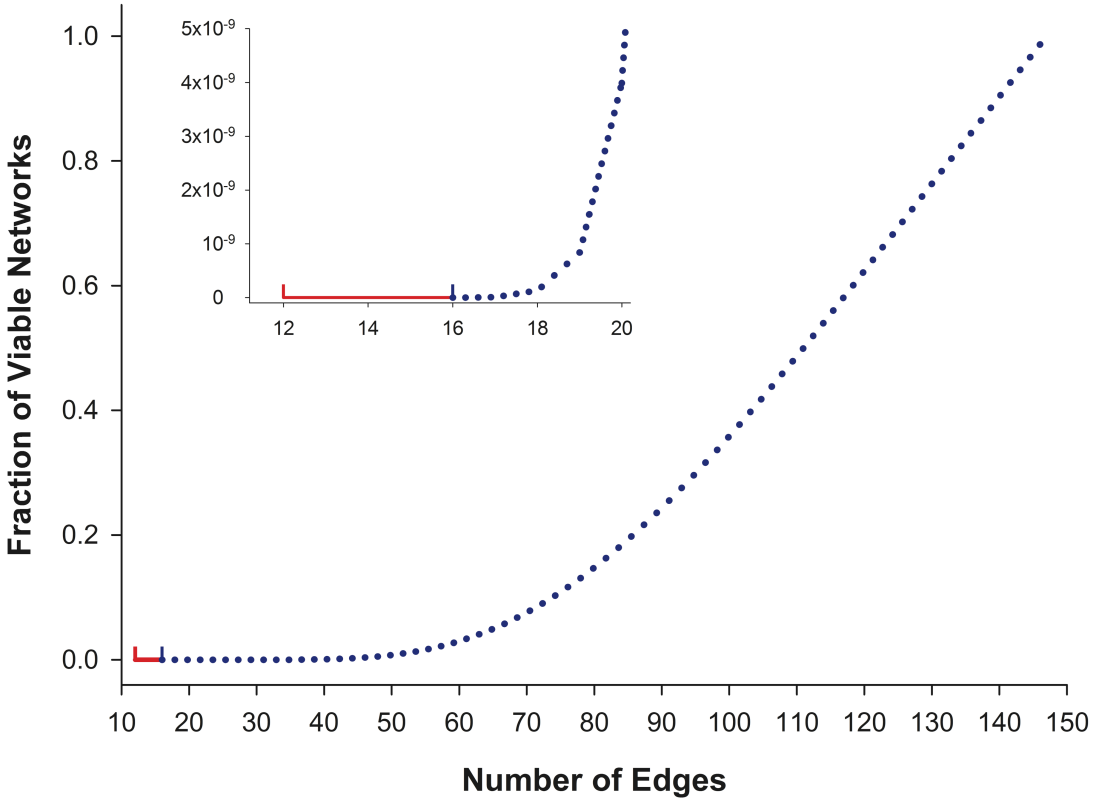

Fig.S2 Ratio of the number of networks in the *CP* ensemble versus that in the *C* ensemble as a function of number of edges in the network. Inset: Zoom to show the plot for the lowest values of number of edges.

The counting of networks in the *C* and *CP* ensembles

Since the number of networks in our ensembles is astronomical, it is not possible to count them by explicit enumeration. Nevertheless, in the cases of *C* and *CP*, it is possible to use algebraic methods to obtain the exact sizes of these sets. In practice, the calculation requires a lot of tedious operations, so we have implemented them in a Mathematica notebook (Supplemental Information, File S1). We explain here the logic of the algorithm.

To encode the set of interactions of a network of 15 genes, it is necessary and sufficient to specify the list of incoming edges for each gene. In practice, we use a binary 15 by 15 matrix A, hereafter referred to as the incidence matrix; its entry A_{ij} (in row *i* and column *j*) is 1 if there is an edge from node *j* to node *i*, otherwise it is 0. For computational convenience, it is possible to encode each line of 15 bits into a number from 0 to 2^{15}-1 using its binary code.

As a pedagogical exercise, let us first count all possible incidence matrices having exactly M non-zero entries. The calculation is based on treating that constraint using generating functions in a "variable" X whose power is the number of entries set to 1. Specifically, for each line *i*, define the polynomial in the variable X

$$P(X) = n_0 X^0 + n_1 X^1 + \ldots + n_{15} X^{15}$$

where n_k is the number of ways of having exactly *k* of the entries on that line set to 1 if one ignores any constraint on the total number of edges; for this illustrative case, n_k is just the binomial coefficient associated with choosing *k* elements among 15. Now taking the product over all rows *i*, we obtain a polynomial which is the "generating function" of the number of incidence matrices

$$Z(X) = \prod_{i=1}^{15} P_i(X)$$

Indeed, when expanding Z in powers of X, we obtain

$$Z(X) = \prod_p n(p) X^p$$

where n(p) is the number of incidence matrices having a total of p entries set to 1. Solving the present exercise then boils down to first computing Z(X) as a product of known polynomials and then extracting its coefficient n(M) which multiplies X^M.

The counting of the number of networks in *C* can be tackled by generalizing in a non-trivial way the previous calculation. The main source of difficulty comes from the constraint that the total number of edges outgoing from a given leaf node is fixed (set to 1). To deal with this, we introduce a variable for each leaf node: let U, V, and W be these three variables so that for instance the power of V gives the number of times an edge leaves the second leaf node. The polynomial P_i to consider now is a function of 4 variables, X, U, V, and W: it is defined as the generating function for all ways to specify input edges to gene *i* while respecting all constraints in *C* except the one on the *total* number of edges. For instance the coefficient of U W X^k in P_i is the number of ways to specify *k* inputs to gene *i* when using one edge from leaf node 1, zero from leaf node 2, and one from leaf node 3. Because in fact each leaf node has one output, there are no terms of degree 2 or higher for U, V, or W. The computation of this polynomial P_i of four variables must also take into account the constraint that there must be at least one input ($k \geq 1$) and if there is a self interaction, there must be at least one other input. We have determined all coefficients of each P_i within a Mathematica code. The next step is to compute the product of all these P_i, thereby constructing the generating function Z(X,U,V,W) for counting networks in *C*. When performing the products of the P_i, we take advantage of the fact that terms of degree 2 or higher for U, V and W are forbidden so Mathematica sets them to zero on the fly. Furthermore, since the leaf nodes have no inputs, their P_i is equal to 1 and so can be omitted in the product defining Z(X,U,V,W). The number of networks in *C* is then obtained by extracting the coefficient of U V W X^M in Z(X,U,V,W).

The computation of the size of *CP* follows the same strategy. The key point is that because the phenotypic viability constraint corresponds to imposing 10 steady states on the gene expression patterns, one has in fact a list of independent constraints, 10 for each gene, that can be incorporated separately on each P_i(X,U,V,W). For a given *i*, the Mathematica code in fact scans all values of *k* from 1 to 15 and enumerates all possible choices of *k* inputs to gene *i*; it then checks whether the constraints of C are satisfied, and if they are it checks whether one can have a Boolean function taking the inputs to the correct output for all 10 steady states. This last step considers the structure of the truth table for the Boolean function and verifies whether the 10 constraints lead to any contradiction, *i.e.*, whether identical inputs are supposed to produce different outputs. Once the P_i are computed in this way, the rest of the algorithm proceeds as for *C*.

In principle, the same approach could be used to count the number of elements in the *CD* and *CDP* ensembles. Because the out-degree of every gene is now imposed just as it was for the leaf nodes in *C* and *CP*, it is necessary to introduce one variable for each gene. Our formalism thus requires handling polynomials of 15 variables. The Mathematica code is able to produce the P_i but they involve a huge number of terms. Then in practice it is not possible to compute the product of these polynomials, preventing one from extracting the desired coefficient. Note that even without any phenotypic constraint, this counting problem is just too difficult and except for small networks, only approximate counting methods are practical.

Table S1 Comparison of frequency of 1, 2 and 3-node subgraphs in the Arabidopsis network with ensembles *C*, *CP*, *CD* and *CDP*. The table lists the mean and standard deviation of the frequency of subgraphs in each ensemble. The table gives the Z-scores for each subgraph in the Arabidopsis network when benchmarked against the ensembles *C*, *CP*, *CD* and *CDP*.

Subgraph		Arabidopsis network	Z-score w.r.t. *C*	Z-score w.r.t. *CP*	Z-score w.r.t. *CD*	Z-score w.r.t. *CDP*	*C*		*CP*		*CD*		*CDP*	
							Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.
1–node	Self Edge A → A	4	0,310	-1,218	0,117	-1,600	3,532	1,512	5,783	1,464	3,861	1,188	5,565	0,978
2–node	Mutual Edge A ↔ B	10	2,379	2,771	1,826	2,297	5,807	1,763	5,253	1,713	7,335	1,459	6,872	1,362
3–node	A ← B → C	5	-2,146	-2,099	-1,542	-1,845	14,724	4,531	14,398	4,478	9,369	2,833	10,632	3,052
	A → B ← C	19	-0,432	-0,232	-1,360	-1,047	21,356	5,457	20,208	5,201	24,991	4,406	23,504	4,302
	A → B → C	18	-2,217	-2,126	-0,978	-0,922	36,425	8,310	35,535	8,249	24,394	6,540	23,849	6,342
	A ↔ B ← C	23	1,612	2,029	0,114	0,357	14,895	5,029	13,302	4,781	22,455	4,794	21,382	4,532
	A ↔ B → C	14	0,364	0,539	1,446	1,280	12,459	4,238	11,744	4,185	9,520	3,098	9,798	3,281
	A ↔ B ↔ C	6	1,642	1,983	1,183	1,469	2,474	2,147	2,108	1,963	3,338	2,250	2,906	2,106
	A → B → C, A → C	8	-0,766	-0,513	-0,470	-0,253	12,053	5,289	10,403	4,689	9,860	3,960	8,895	3,534
	A ← B ← C, A → C	0	-1,698	-1,668	-1,271	-1,141	4,143	2,440	3,798	2,277	2,169	1,706	1,790	1,569
	A ← B → C, A ↔ C	4	1,061	1,604	-0,414	-0,302	2,395	1,513	1,864	1,332	4,829	2,002	4,546	1,811
	A → B ← C, A ↔ C	4	0,979	1,396	0,855	1,176	2,479	1,553	2,034	1,408	2,624	1,609	2,293	1,452
	A → B → C, A ↔ C	2	-1,364	-1,055	-1,826	-1,544	4,961	2,170	4,075	1,966	6,242	2,323	5,343	2,165
	A → B ↔ C, A ↔ C	7	3,276	4,212	0,882	1,260	1,886	1,561	1,414	1,326	5,079	2,178	4,586	1,916
	A ↔ B ↔ C, A ↔ C	2	5,418	6,755	1,618	1,532	0,116	0,348	0,078	0,285	0,682	0,815	0,752	0,814

Table S2 Comparison of average clustering coefficient in the Arabidopsis network with ensembles *C*, *CP*, *CD* and *CDP*. The table lists the mean and standard deviation of the average clustering coefficient in each ensemble. The table gives the Z-scores for average clustering coefficient in the *Arabidopsis* network when benchmarked against the ensembles *C*, *CP*, *CD* and *CDP*.

	Arabidopsis network	Z-score w.r.t. *C*	Z-score w.r.t. *CP*	Z-score w.r.t. *CD*	Z-score w.r.t. *CDP*	*C*		*CP*		*CD*		*CDP*	
						Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.
Average Clustering Coefficient	0,6265	2,4362	2,7369	0,3784	0,5271	0,4672	0,0654	0,4383	0,0688	0,5974	0,0768	0,5837	0,0812

Table S3 Comparison of 4 directed assortativity coefficients in the first and fourth quartiles of ensembles *CP* and *CDP* where quartiles are based on robustness of networks in each ensemble. The table lists the mean and standard deviation of the 4 directed assortativity coefficients in each ensemble quartile.

Assortativity coefficient	Arabidopsis network	*CP* Q1		*CP* Q4		*CDP* Q1		*CDP* Q4	
		Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.
(out,in)	-0,248	-0,086	0,125	-0,080	0,126	-0,254	0,085	-0,254	0,084
(out,out)	-0,115	0,035	0,131	0,044	0,130	-0,066	0,107	-0,059	0,107
(in,out)	-0,113	-0,015	0,132	-0,014	0,134	-0,011	0,109	-0,038	0,110
(in,in)	-0,048	0,006	0,131	0,005	0,131	-0,029	0,082	-0,049	0,082

Table S4 Comparison of frequency of 1, 2 and 3-node subgraphs in the first and fourth quartiles of ensembles *CP* and *CDP* where quartiles are based on robustness of networks in each ensemble. The table lists the mean and standard deviation of the frequency of subgraphs in each ensemble quartile.

Subgraph		Arabidopsis network	*CP* Q1		*CP* Q4		*CDP* Q1		*CDP* Q4	
			Mean	S.D.	Mean	S.D.	Mean	S.D.	Mean	S.D.
1–node	Self Edge A → A	4	5,577	1,444	6,003	1,466	5,412	0,997	5,710	0,942
2–node	Mutual Edge A ↔ B	10	5,299	1,731	5,207	1,692	6,973	1,352	6,782	1,368
3–node	A ← B → C	5	14,342	4,508	14,491	4,465	10,862	3,100	10,458	3,009
	A → B ← C	19	20,578	5,330	19,767	5,038	23,268	4,243	23,726	4,347
	A → B → C	18	35,357	8,240	35,605	8,170	23,484	6,221	24,128	6,439
	A ↔ B ← C	23	13,569	4,857	13,023	4,680	21,870	4,527	20,944	4,519
	A ↔ B → C	14	11,665	4,164	11,806	4,168	9,712	3,296	9,873	3,276
	A ↔ B ↔ C	6	2,142	2,006	2,078	1,924	2,856	2,104	2,961	2,132
	A → B → C, A → C	8	10,705	4,783	10,133	4,584	8,635	3,443	9,137	3,610
	A ← B ← C, A → C	0	3,782	2,291	3,789	2,242	1,787	1,566	1,802	1,573
	A ← B → C, A ↔ C	4	1,957	1,366	1,779	1,300	4,589	1,822	4,468	1,789
	A → B ← C, A ↔ C	4	2,099	1,437	1,984	1,388	2,214	1,426	2,339	1,456
	A → B → C, A ↔ C	2	4,148	1,986	3,998	1,949	5,367	2,172	5,322	2,170
	A → B ↔ C, A ↔ C	7	1,493	1,374	1,350	1,276	4,789	1,932	4,409	1,887
	A ↔ B ↔ C, A ↔ C	2	0,089	0,305	0,070	0,265	0,859	0,860	0,663	0,763

Computation of p-values

Given a specific network such as the *Arabidopsis* one, we wish to test the hypothesis H_0 that it is typical of a benchmark ensemble E of networks generated *in silico*. (In practice, E is taken as C, CP, CD or CDP.) If the network does seem atypical, one rejects H_0 but doing so there is a small chance that in fact one is doing so erroneously. The probability of erroneously rejecting H_0 is the p-value, and is calculated as follows. A summary statistic S is used for the test. Within H_0, one computes P(S), the distribution of S, for instance by simulation (using the networks in the benchmark ensemble), and one also determines S*, the value of the statistic for the specific network considered. Then the right-sided p-value associated with rejecting H_0 is the probability that the random variable S, distributed according to P(S), will be larger than S*. Similarly the left-sided p-value is the probability that S will be smaller than S* while the two-sided p-value is twice the minimum of the left and right-sided p-values.

When P(S) is Gaussian or expected to be so, it is convenient to consider the Z-score of S*. It is defined as the difference between S* and the mean of P(S), divided by the standard deviation of P. It gives the deviation from the mean in units of standard deviation. From it, a p-value can also be extracted reliably as long as P(S) is Gaussian.

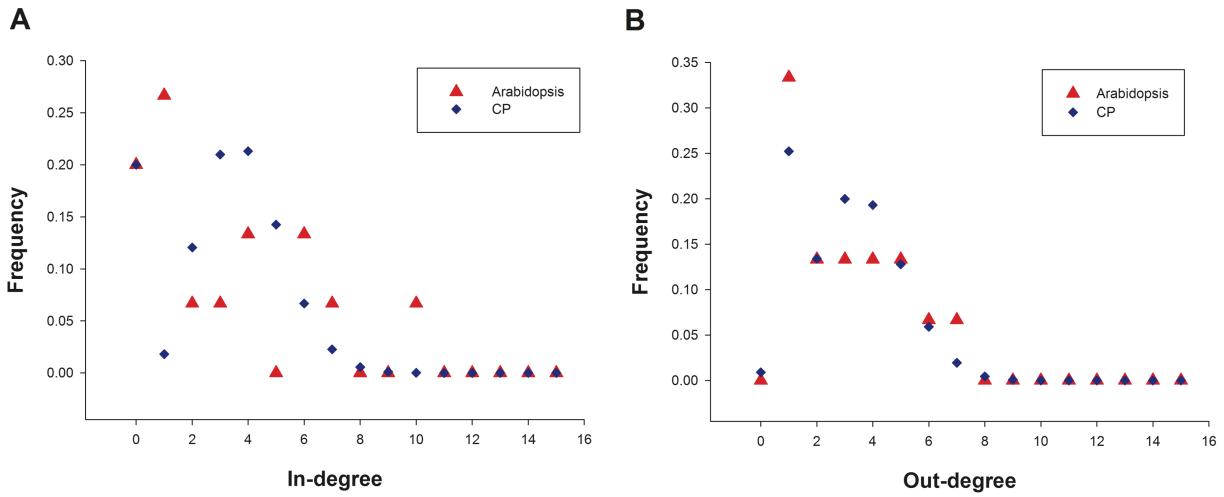

Fig.S3 Degree distributions for the *Arabidopsis* reference network and networks in the *CP* ensemble. (A) In-degree distribution. (B) Out-degree distribution.

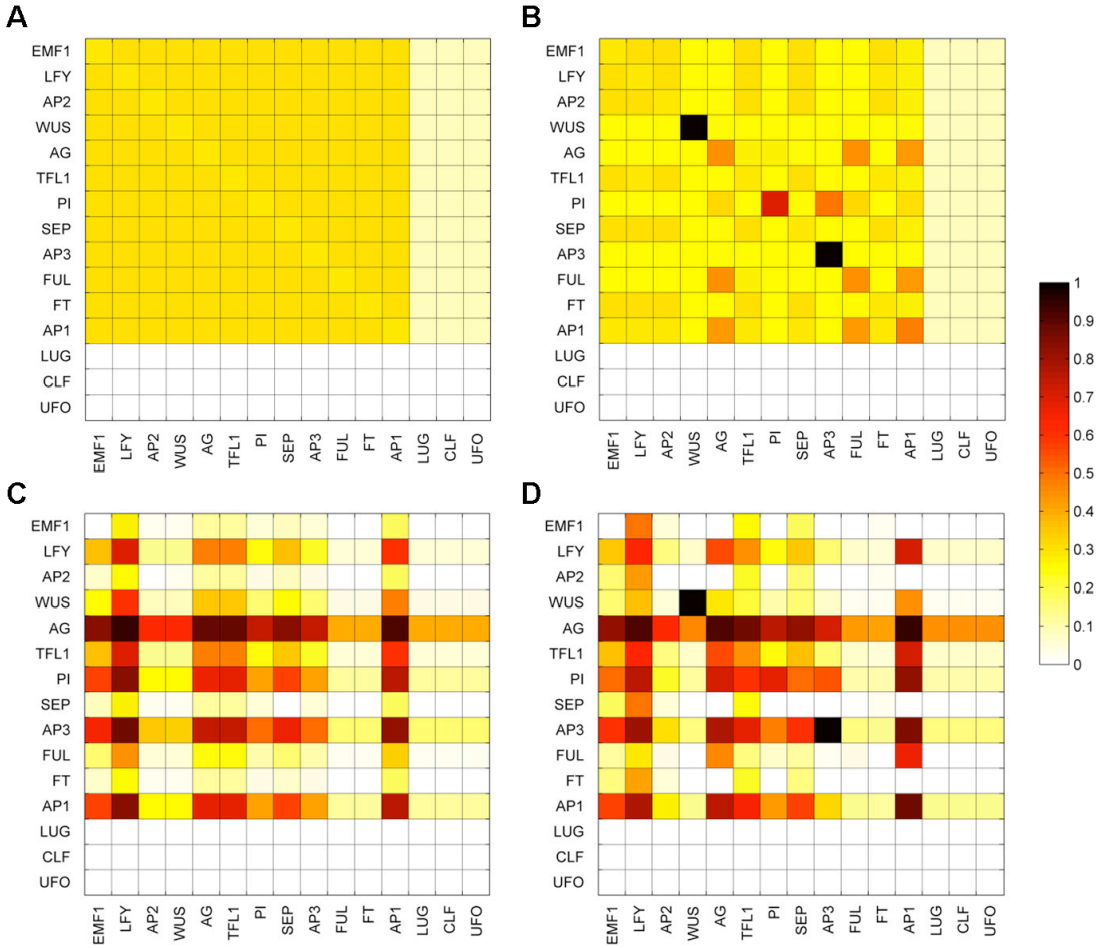

Fig.S4 Edge usage in the four ensembles. The incidence matrix captures the presence or absence of an edge in a directed network. If there is a directed edge from node *j* to *i* then the (i,j)th entry of the incidence matrix is non-zero. Edge usage is the frequency with which a given edge is realized across sampled networks for an ensemble. Heat maps show the edge usage for networks in ensembles (A) *C*, (B) *CP*, (C) *CD* and (D) *CDP*. To reduce statistical noise, over 2.10^6 networks were sampled for case (A).

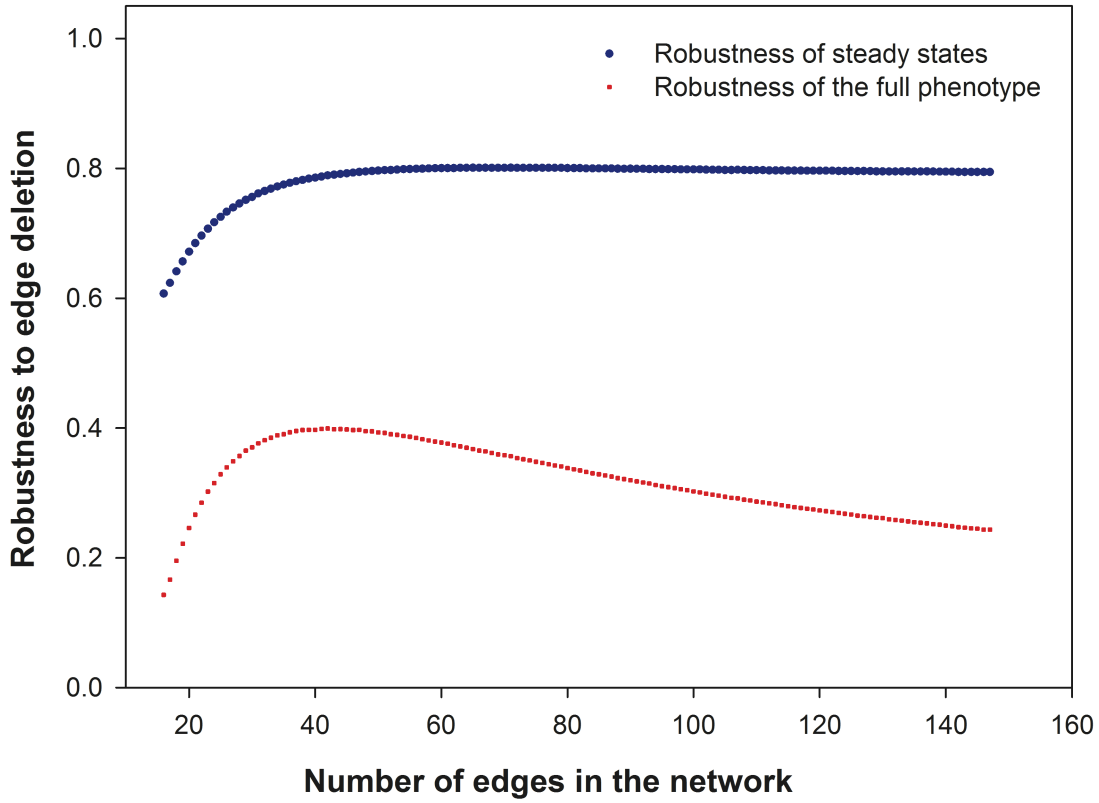

Fig.S5 Average robustness to random edge deletion as a function of the number of edges in the network for the *CP ensemble. The blue dots represent the probability for a steady state to survive after random edge deletion and the red dots represent the probability to maintain phenotypic viability after random edge deletion.*