SI Text

1. What Makes a Map a Map?

A good way to think about the goal of our work is to think of an electronic map. We need new tools to analyze and represent the information embedded inside a complex network in a scalable manner, in the same manner that a map does. Current representations of, say, metabolic networks (Fig. 5) do not provide information about which are the relevant nodes and how the different nodes are organized. With our methods we are able to extract the relevant scales in the network, which opens the door to a scalable representation of complex systems.

2. Hierarchies in Complex Networks

We focus on the detection of "inclusion hierarchies" in complex networks, that is, those hierarchies in which a module at a given level splits into several submodules at the following level in the hierarchy. We assume that the organization of nodes into modules at any level is defined by the different connectivity densities between groups of nodes; so that, a group of nodes that are more interconnected between themselves than they are to other nodes in the network comprise a module in the network organization.

Additionally, in the same way that electronic maps work, we assume that information at different levels does not mix. That is, when we zoom within a certain state of the United States, we see a more local information about different cities included in that state. However, we do not see the information regarding boundaries of the state itself, because that information is not relevant for the organization at that scale. In the same spirit, we assume that the internal organization of a module *solely* depends on the network of connections within the module.

First, we formally define the concept of inclusion hierarchies in complex networks, and then we introduce a ensemble of networks (hierarchically nested random graphs) with which to test the accuracy if our algorithm at detecting the network organization.

2.1. Inclusion Hierarchies

Consider the ensemble of networks composed of N nodes, $\mathcal{N} = \{ n_i : i = 1, \ldots, N \}$, that hold membership in a set of nested groups, $\mathcal{G} =\{ g_{(k_1 \ldots k_\ell)} : \ell = 1, 2 \ldots \}$, where ℓ is the level at which the group is defined, and the labels $k_1 \ \ldots \ k_{\ell-1}$ indicate the groups at higher levels in the hierarchy within which the group is nested. For instance, group g_{111} is a group defined at $\ell = 3$

that is nested inside group g_{11} defined at $\ell = 2$, which in turn is a subgroup of group g_1 defined at $\ell = 1$.

Let $\mathcal{G}_i \subset \mathcal{G}$ be the set of groups in which node n_i holds membership. Here, we consider that node n_i holds membership in only one group per level, and that membership to groups follows a nested hierarchy. Therefore, for node n_i to hold membership in group g_{11}, node n_i must also hold membership in group g_1.

We assume that the probability p_{ij} of the edge (n_i, n_j) being present in a network is a function *solely* of the set of comemberships $\mathcal{M}_{ij} = \mathcal{G}_i \cap \mathcal{G}_j$ of the two nodes. Note that our assumptions imply that: *(i)* $\mathcal{M}_{ij}$ obeys transitivity, so that if $\mathcal{M}_{ij} = \mathcal{M}_{ik}$, then $\mathcal{M}_{ij} = \mathcal{M}_{jk}$; and *(ii)* node memberships in groups $\{\, g_{k_1 k_2} \,\}$ at the second level are uniquely and completely defined by the subnetwork of connections of all nodes holding membership in group g_{k_1}, that is, information at deeper levels in the hierarchy is totally decoupled from the information at higher levels in the hierarchy.

In the simplest scenario, p_{ij} is a nondecreasing function of the cardinality x of $\mathcal{M}_{ij}$, which implies that groups of nodes holding membership in the same groups will be more densely connected than a randomly selected group of nodes. This is precisely the underlying assumption in many algorithms aiming to detect the top level community structure of complex networks assuming a flat organization of the nodes (1, 2, 3).

2.2. Hierarchically Nested Random Graphs

Let us now introduce an ensemble of random networks which are constructed following hierarchical node membership assignment: hierarchically nested random graphs. We restrict our ensemble to networks with a homogeneous hierarchical organization of the nodes (see Section 3.4. for other kinds of hierarchical organization) that have the same degree distribution as Erdős-Rényi graphs (4).

To illustrate the model, consider a network composed of 640 nodes that hold membership in a set of groups $\mathcal{G}$ with a three-level homogeneous nested organization. We assign group memberships so that the number S_ℓ of nodes holding membership in each group for $\ell = 1, 2$, and 3 is $S_1 = 160$, $S_2 = 40$, and $S_3 = 10$, respectively. For $\ell = 1$, nodes can hold membership in one of four different groups $\{\, g_{k_1} : \ k_1 = 1, \ldots, 4 \,\}$. For $\ell = 2$, nodes holding membership in group g_{k_1} can hold membership in one of four groups $\{\, g_{k_1 k_2} : k_2 = 1, \ldots, 4 \,\}$. Finally, for $\ell = 3$, nodes holding membership in groups g_{k_1} and $g_{k_1 k_2}$ can hold membership in one of four groups $\{\, g_{k_1 k_2 k_3} : k_3 = 1, \ldots, 4 \,\}$.

In the example, we assign membership to groups so that the number of member nodes S_ℓ for groups defined at levels $\ell = 1, 2$, and 3 is $S_1 = 160$, $S_2 = 40$, and $S_3 = 10$, respectively.

Let $\mathcal{G}_i$ be the set of groups in which node n_i holds membership. Then, the probability that an edge between nodes (i, j) exists is a monotonically increasing function of the cardinality of $\mathcal{M}_{ij} = \mathcal{G}_i \cap \mathcal{G}_j$.

Let p_x be the probability of an edge existing between n_i and the set $\{n_j : ||\mathcal{M}_{ij}|| = x\}$;

then the expected number of connections is

$$k_x \;=\; \begin{cases} p_x(S_x - 1) & x > 0 \\ p_x S_x & x = 0 \end{cases} \quad , \tag{1}$$

where S_0 is the number of nodes that do not hold membership in any of the groups n_i holds membership in ($S_0 = 3 \times 160$). Therefore, the expected total average degree of n_i is $\overline{k} = \sum_x k_x$. We construct the model networks such that the ratio $\rho = k_{<\ell}/k_\ell$ is constant, where $k_{>\ell} = \sum_{\ell'<\ell} k_\ell$.

Given ρ and $\overline{k}$, the analytical expression for p_x is

$$p_{ij} \;=\; p_x \left\{ \begin{array}{ll} \frac{\rho^{\ell_{\mathrm{max}}-x}}{(1+\rho)^{\ell_{\mathrm{max}}-x+1}}\,\frac{\overline{k}}{S_x-1}, & x = 1, \ldots, \ell_{\mathrm{max}} \\ & \\ \frac{\rho^{\ell_{\mathrm{max}}-x}}{(1+\rho)^{\ell_{\mathrm{max}}-x}}\,\frac{\overline{k}}{S_x}, & \quad x = 0, \end{array} \right\} \;,$$

where $\ell_{\mathrm{max}} = 3$ in the example. Note that, by construction, there is a lower bound to the value of ρ since $p_{\ell_{\mathrm{max}}} = \overline{k}/(1+\rho)/(S_{\ell_{\mathrm{max}}} - 1) \leq 1$.

In the example network, for $\rho = 1$ and $\overline{k} = 16$, we obtain the following values for p_ℓ: for $\ell = 3$ we have modules of 10 nodes ($S_3 = 10$), so that $p_3 = 8/9$ and $k_3 = 8$; for $\ell = 2$, $S_2 = 40$, therefore, $p_2 = 4/39$ and $k_2 = 4$; finally, for $\ell = 1$, $p_1 = 2/159$ and $p_0 = 2/360$.

3. Uncovering the Organization of Complex Networks

We present here a detailed explanation of our method to extract the organization of complex networks. We impose that our method fulfills two requirements: *(i)* it must not be network specific, *(ii)* it must identify the different levels in the hierarchy as well as the composition of the modules at each level.

Our method consists in two main steps: *(i)* the construction of an affinity matrix between all pairs of nodes and *(ii)* the extraction of the hierarchical tree.

3.1. The Affinity Matrix

Our aim is to obtain a measure of the likelihood that two given nodes belong to the same module. Our strategy is to define a measure of similarity by looking at the network as a whole. Thus, a natural candidate is to study the modularity of the network.

Following Girvan and Newman, we define the modularity M of a partition P of a network as (5, 6)

$$M(P) = \sum_{i=1,m} \left[\frac{l_i}{L} - \left(\frac{d_i}{2L} \right)^2 \right] \quad , \tag{2}$$

where L is the total number of links, l_i is the number of links within module i, d_i is the sum of degrees of all the nodes inside module i, and the sum is over all the m modules in partition P.

Traditionally, the “best” partition is thought to be that for which the modularity has the largest value. One would argue then that the organization of a network into modules is that given by the “best” partition. However, for some model networks, the “best” partition of the network does not correspond to the top level organization (see Fig. 3*C*).

The answer to this problem comes from realizing that the task of finding the partition with the largest modularity is a hard optimization problem, since the modularity landscape is very rugged and has many local maxima (7, 8). Thus, there are many partitions that have large modularity and yet they may be quite different from one another.

In defining affinity between a pair of nodes, one wants nodes that are close in the hierarchy to have a large affinity and to be classified in the same module in partitions with large modularity. We formalize this requirement by defining the affinity A_{ij} of a pair of nodes (i, j) as the probability that the two nodes are classified in the same module for partitions P that are local maxima of the modularity landscape $\{\tilde{P} \in \mathcal{P}_{\max}\}$. Partition $\tilde{P}$ is a local maxima if neither the change of a single node from one community to another nor the merging of two communities yields a larger modularity. Following Stillinger and Weber’s decomposition of rugged landscapes (9), in computing A_{ij} we let partitions corresponding to local maxima whose basin of attraction $b(\tilde{P})$ is larger yield a larger contribution to A_{ij} (see Fig. 1*A*). That is, for a pair of nodes (i, j), the partition in each visited local maxima contributes $a_{ij}(\tilde{P}) = 0, 1$ to A_{ij}. Thus, if partition $\tilde{P}$ has a probability $b(\tilde{P})$ of being visited

$$A_{ij} = \sum_{\tilde{P} \in \mathcal{P}_{\max}} b(\tilde{P})\, a_{ij}(\tilde{P}). \qquad \textbf{(3)}$$

We find the local maxima of the modularity landscape by performing Monte Carlo simulations at zero temperature using the algorithm of Guimerà and Amaral (8). Specifically, we first generate a random partition of the nodes into modules. We then perform a hill-climbing search until we reach a local maximum of the modularity. After completing these two steps, we update a matrix recording the fraction of times each pair of nodes is classified in the same module. We repeat these steps until the matrix has converged to its asymptotic value (Fig. 1*B*). In fact, we obtain the same coclassification matrix regardless of whether we start each run from a different random configuration, or from the same configuration putting each node in a separate community. The number of runs necessary for convergence depends, in general, on both the size and the connectivity of the network. For all the networks we study, we find that a number of runs of the order of the size of the network is sufficient.

3.2. Uncovering the Hierarchical Tree

3.2.1. Statistical Significance of the Hierarchical Structure

In order to assess whether the network under analysis has an internal organization, one needs to compare the structure of its modularity landscape with that of an appropriate null model, which in this case is an ensemble of “equivalent” networks with no internal organization. These equivalent networks have the same number of nodes and an identical degree sequence as the network under consideration. Note that connections between nodes are drawn at random, so that these networks have no internal structure.

We construct the null model networks by randomizing the original network following a link swapping protocol (10, 11). Despite their having no structure, random structureless networks have many partitions with non-zero modularity (6); therefore, for a network to have a significant internal structure, the local maxima in the landscape must have larger modularities than the local maxima of the corresponding ensemble of equivalent random networks.

To quantify the significance of the internal structure of any network, we first compute the average modularity of the local maxima in the landscape

$$M_{\rm av} = \sum_{\tilde{P} \in \mathcal{P}_{\rm max}} \frac{b(\tilde{P})\, M(\tilde{P})}{||\mathcal{P}_{\rm max}||} \qquad \textbf{(4)}$$

Then, we generate r randomizations of the network ($r = 10$ for the results shown in the main manuscript), and we compute the average modularity $M^i_{\rm av}$ for the local maxima in the landscape of each randomized network $i = 1, \ldots, r$. For each randomization we compute the average over $m = 100$ local maxima. In virtue of the central limit theorem, the set of average modularities for the whole ensemble $\{\, M^i_{\rm av}\,\}$ is a normally distributed variable with mean $M_{\rm rand}$ and variance $\sigma^2_{M_{\rm rand}}$. For Gaussian variables, a simple way to quantify statistical significance is to compute the z-score (SI Fig. 6),

$$z = \frac{M_{\rm av} - M_{\rm rand}}{\sigma_{M_{\rm rand}}}\,. \qquad \textbf{(5)}$$

In our analysis, we have used threshold values for z of 1.645 and 2.3267, which correspond to significance levels of 5% and 1%, respectively. In the manuscript, we show results for the 1% significance level, since it is more accurate at finding the exact number of levels for hierarchically nested random graphs. Nevertheless, results at the 5% significance level are the same for all the real networks studied.

3.2.2. Ordering the Affinity Matrix

The goal of this procedure is to order the nodes in such a way that nodes with highest affinity are closest in the ordering. This is analogous to finding the ordering for which the largest elements in the affinity matrix A_{ij} are closest to the diagonal. To find an ordering close to the optimal ordering, we use simulated annealing (12) with a cost function that weights each element by its distance to the diagonal (13)

$$\mathcal{C} = \frac{1}{N} \sum_{i,j=1}^{N} A_{ij} |i - j|, \qquad \textbf{(6)}$$

where N is the order of the affinity matrix.

For every iteration in the simulated annealing search, we propose $\mathcal{O}(N^2)$ moves in which segments of contiguous nodes attempts to change positions in the ordering. We accept or reject each attempted move following a standard Metropolis algorithm. For each attempt, we randomly pick: (a) a segment of contiguous nodes and (b) a new position for the first node—the remaining nodes will be placed keeping the relative distance to the first node. The first node and its new position are picked from a uniform distribution; the width of the segment is picked from a Gaussian distribution whose variance depends linearly on both the temperature T and the size of the network N—for low temperatures only changes of single nodes are proposed. We compute the value of the cost function for the new order $\mathcal{C}'$ and we accept the change with probability $p = \exp\left[(\mathcal{C} - \mathcal{C}')/T\right]$.

We start from a random initial ordering at temperature T. After every iteration we decrease the temperature by a factor in the range $[0.95,\ 0.999]$. The process stops when $\mathcal{C}$ has not changed for 20 iterations.

This stage is the most time consuming. We have tried to use other faster algorithms to rank nodes in one dimension, such as classical scaling (14) and hierarchical clustering. Both methods, however, give a considerably less accurate ordering than our algorithm (SI Fig. 7), especially for real-world networks.

3.2.3. Uncovering the Top-Level of the Hierarchy

To identify the best partition at the top-level, we assume the following *ansatz* for the matrix structure: The matrix has a block-diagonal organization, with n boxes bound by $n-1$ break points b_s, for $s = 2, \ldots, n$, note that $b_1 = 1$. The matrix elements inside box s are assumed equal to A_0^s. In general, if $s \neq s'$, then $A_0^s \neq A_0^{s'}$. The matrix elements outside the boxes are assumed equal to B_0.

The goal then is to find the set of parameters $\{n, b_2, ..., b_n, B_0, A_0^1, ..., A_0^n\}$ that best fits the data. We quantify the closeness-of-fit of a given set of parameters by the sum $\mathcal{L}$ of the squared deviations of the data to the model

$$\mathcal{L} = \sum_{(i,j)} \left\{ \begin{array}{ll} (A_{ij} - A_0^s)^2 & \text{if } (i,j) \in \text{box } s \\ (A_{ij} - B_0)^2 & \quad\text{otherwise} \end{array} \right. \tag{7}$$

For a given set $(n, b_2, \ldots, b_n)$, the best estimators for $(B_0, A_0^1, \ldots, A_0^n)$ are as follows.

$$\hat{A}_0^s = \sum_{(i,j)\in box_s} \frac{A_{ij}}{N_s} \quad \text{where} \quad N_s = \sum_{(i,j)\in box_s} 1 \ \text{ and } \ s = 1, \ldots, n \tag{8}$$

$$\hat{B}_0 = \sum_{(i,j)\notin \bigcup_{s=1}^{n} box_k} \frac{A_{ij}}{N_B} \quad \text{where} \quad N_B = \sum_{(i,j)\notin \bigcup_{s=1}^{n} box_s} 1 \tag{9}$$

There are two challenges in finding the best set of remaining parameters. The first challenge is the extraordinarily large number of possible parameter combinations. For given n and set of break points, one can estimate the other parameters simply by calculating average values for the elements of the affinity matrix inside the boxes. The limiting step in considering all combinations of parameters is the supra-exponential increase with n of the number of combinations of break points.

To overcome this problem, we use a greedy algorithm for partitioning the matrix in the spirit of the segmentation algorithms used for time series analysis (15, 16). Specifically, we start by setting $n = 1$ and by computing $\mathcal{L}_1$ for a single box. Next, we assume $n = 2$, we calculate $\mathcal{L}$ for $b_2 = 2, ..., N-1$. We select the value of b_2 for which $\mathcal{L}$ is the smallest. We denote the minimum of $\mathcal{L}$ for $n = 2$ by $\mathcal{L}_2$.

We then consider the case $n = 3$. We do not consider all possible partitions of the matrix into three boxes but only those in which there are already two break points defined b_1 and b_2. We calculate the minimum value of $\mathcal{L}$ for $n = 3$ and denote it $\mathcal{L}_3$. If $\mathcal{L}_3 < \mathcal{L}_2$, we accept the new break point; otherwise, we stop. In principle, one should iterate the same procedure until $n = N$ and then select the best fit the set of parameters for which $\mathcal{L}$ is minimal: as $\mathcal{L}_{\text{min}} = \min\{\mathcal{L} \in \{\mathcal{L}_1, \ldots, \mathcal{L}_N\}\}$. In practice, for the type of matrices we analyze, $\mathcal{L}_x$ has a minimum for small values of x ($x/N << 1$); therefore, one does not need to explore the whole range of x to find the global minimum.

The second challenge arises from the fact that the partition one obtains following the greedy algorithm may over-fit the data. Indeed, it is a well-known fact that models with a larger number of parameters will yield better "fits" to data. However, those models will not necessarily provide better *descriptions* of the data, in same way that n data points that approximately obey a linear relationship will be reproduced without deviation by a polynomial of degree $n-1$. To avoid over-fitting, we use the Bayesian information criterion (BIC) (17) to obtain the set of boxes that better fits our model without introducing unnecessary parameters. BIC measures the likelihood of the model given the data set, so that, the model with a larger likelihood is taken to be the model that best describes the data. When the model parameters of the model have been estimated by least-squares minimization, the BIC is defined as

$$BIC = m \, \log\left(\frac{\mathcal{L}}{m}\right) \; + \; k \log(m), \tag{10}$$

where m is the total number of data points, $\mathcal{L}$ is the residual sum of squares, and k is the number of parameters. We select the set of boxes that yields the smallest BIC.

A different criterion that also balances the goodness-of-fit with the parsimony of the model is Akaike's information criterion (18), which measures how close the description given by the model is to the "real" description. When the model parameters of the model have been estimated by least-squares minimization, the AIC is defined as:

$$AIC = m \, \log\left(\frac{\mathcal{L}}{m}\right) \; + \; 2k. \tag{11}$$

Again, the model with lowest AIC is taken to provide the best description of the model. We prefer BIC over AIC because of how $\mathcal{L}$ behaves upon increasing the number of parameters (19).

In practice, both criteria give the same answer for the analyzed networks, which in the majority of cases is that the set of boxes corresponding to $\mathcal{L}_{\min}$ is the one with the lowest BIC and AIC. The reason for that is that the number of boxes n^* that we find for the best fit is much smaller than the total number of rows in the matrix N.

3.2.4. Construction of the Hierarchical Tree

The result of the application of the algorithm described in the two previous section is an affinity matrix that comprises n boxes, which can be identified with n modules. Let $\mathcal{V}_s$ be the set of nodes in module s. We define the affinity matrix $\mathbf{A}_s$ as the affinity of the nodes in $\mathcal{V}_s$.

As we say in the main text, we surmise that for inclusion hierarchies the information at deep levels in the hierarchy is decoupled from the information at levels higher in the hierarchy. Therefore, once the algorithm finds the modules at the top level in the hierarchy, in order to find the submodules in modules s, we only need to consider the subnetwork of connection between nodes in $\mathcal{V}_s$ in order to define $\mathbf{A}_s$. Then, iterate the same algorithm for each of the subnetworks defined by each of the n modules.

Accuracy of the Method: Application of the Method to Model Networks

3.3. Hierarchically Nested Random Graphs

In the main text, we have already shown the performance of our method for hierarchically nested random graphs. In figure 2*A*, we compare the performance of our method at detecting the organization of a three-level hierarchical random graph compared to other standard methods to measure node affinity (topological overlap) and for extracting the hierarchical organization given a measure of affinity (hierarchical clustering). In SI Fig. 8, we show the same analysis for a network for a hierarchically nested random graph with a "flat" organization of the nodes.

3.4. Other Model Networks

In the main text, we have already analyzed the performance of our method in finding the organization of the nodes in hierarchical random graphs (see Fig. 2*C*). These networks, as explained in Section 2.2. have, by construction, a homogeneous hierarchical organization of the nodes. Here, we analyze model networks without hierarchical organization and with a heterogeneous hierarchical organization to see whether our method can differentiate both classes of networks.

First, we analyze two networks which we know lack a community structure and, therefore, a hierarchical organization: the random graph (20) and the preferential attachment model (21). Second, we analyze a network which is hierarchical by construction but is constructed differently from the ones considered so far and has a heterogeneous structure (that is, nodes are organized into modules of different sizes and each module has a different number of modules).

As expected, for the first two model networks, our method does not detect any structure in the organization of the nodes. In both cases, we find that the most statistically significant partition is to group all the nodes in the same box (SI Fig. 9). Instead, for the networks with a built-in heterogeneous hierarchy, our method does identify the heterogeneity in the hierarchical organization of the different modules (SI Fig. 10).

4. Method Validation: Application of the Method to Real-World Networks

4.1. Air-Transportation Network

In Fig. 3, we show the results obtained with our method for the air-transportation network. The reason for using such network is that we have an *a priori* intuition of how the network is organized due to the politico-economical forces that have shaped the network. Our method is able to extract such organization without any external input; it uses just a collection of nodes and edges as shown in SI Fig. 11, in which the organization of the network is not apparent at all.

4.2. Technological and Social Networks

In SI Fig. 8, we show the results of applying our method to two real networks: the transistor implementation of an electronic circuit (A) and the e-mail exchange network of a Catalan university (B).

4.3. Metabolic Networks

In the main text, we show the hierarchical trees obtained the UCSD reconstruction of the metabolic network of *E. coli*. Here, we show the coclassification matrices and hierarchical trees obtained at the 1% significance level for the Ma-Zeng reconstruction of the metabolic network of *E. coli* (22), the reconstruction of the metabolic network of *E. coli* according to the KEGG database, and for the UCSD reconstruction of the metabolic network obtained for *H. pylori* (23). Note that both networks share common features with those shown in Fig. 4. Modules at the top level have heterogeneous sizes, some of them having an internal organization and some of them not.

References

1. Newman MEJ (2004) *Proc Natl Acad Sci USA* 101:5200–5205.

2. Guimerà R, Amaral LAN (2005) *Nature* 433:895–900.

3. Duch J, Arenas A (2005) *Phys Rev E* 72:art. no. 027104.

4. Watts DJ (1999) *Small Worlds: The Dynamics of Networks between Order and Randomness* (Princeton University Press).

5. Girvan M, Newman MEJ (2002) *Proc Natl Acad Sci USA* 99:7821–7826.

6. Guimerà R, Sales-Pardo M, Amaral LAN (2004) *Phys Rev E* 70:art. no. 025101.

7. Reichardt J, Bornholdt S (2004) *Phys Rev Lett* 93:art. no. 218701.

8. Guimerà R, Amaral LAN (2005) *J Stat Mech Theor Exp* art. no. P02001.

9. Stillinger FH, Weber TA (1982) *Phys Rev A* 25:978–989.

10. Maslov S, Sneppen K (2002) *Science* 296:910–913.

11. Guimerà R, Sales-Pardo M, Amaral LAN (2007) *Nature Phys* 3:63–69.

12. Kirkpatrick S, Gelatt CD, Vecchi MP (1983) *Science* 220:671–680.

13. Wasserman S, Faust K (1994) *Social Network Analysis* (Cambridge University Press, Cambridge, UK).

14. Everitt BS, Landau S, Leese M (2001) *Cluster Analysis* (Arnold Pub.).

15. Bernaola-Galvan P, Ivanov PC, Amaral LAN, Stanley HE (2001) *Phys Rev Lett* 87:168105.

16. Fukuda K, Stanley HE, Amaral LAN (2004) *Phys Rev E* 69:art. no. 021108.

17. Schwarz G (1978) *Ann Stat* 6:461–464.

18. Akaike H (1974) *IEEE Transactions on Automatic Control* 19:716–723.

19. Burnham KP, Anderson DR (2002) *Sociological Methods Research* 33:261–304.

20. Bollobás B (2001) *Random graphs* (Cambridge University Press, Cambridge, UK) 2nd ed.

21. Barabási AL, Albert R (1999) *Science* 286:509–512.

22. Ma H, Zeng AP (2003) *Bioinformatics* 19:270–277.

23. Thiele I, Vo TD, Price ND, Palsson BØ (2005) *J Bacteriol* 187:5818–5830.