Available online at www.sciencedirect.com
ScienceDirect
Journal of Theoretical Biology 247 (2007) 331–336
Journal of Theoretical Biology
www.elsevier.com/locate/yjtbi

Do scale-free regulatory networks allow more expression than random ones?

Miguel A. Fortuna[a,*], Carlos J. Melián[a,b]

[a]Integrative Ecology Group, Estación Biológica de Doñana, CSIC, Avda. Ma Luisa s/n, 41013, E-41080 Sevilla, Spain
[b]National Center for Ecological Analysis and Synthesis, University of California, 735 State St., Suite 300, Santa Barbara, CA 93101, USA

Received 23 October 2006; received in revised form 13 March 2007; accepted 13 March 2007
Available online 21 March 2007

Abstract

In this paper, we compile the network of software packages with regulatory interactions (dependences and conflicts) from Debian GNU/Linux operating system and use it as an analogy for a gene regulatory network. Using a trace-back algorithm we assemble networks from the pool of packages with both scale-free (real data) and exponential (null model) topologies. We record the maximum number of packages that can be functionally installed in the system (i.e., the active network size). We show that scale-free regulatory networks allow a larger active network size than random ones. This result might have implications for the number of expressed genes at steady state. Small genomes with scale-free regulatory topologies could allow much more expression than large genomes with exponential topologies. This may have implications for the dynamics, robustness and evolution of genomes.

Keywords: Complex networks; Gene networks; Network assembly; Regulatory interactions; Transcriptional regulatory networks

1. Introduction

In the last years an increasing number of systems have been described as networks (i.e., a set of nodes connected between them by links) and represented as graphs (e.g., Strogatz, 2001; Albert and Barabási, 2002; Newman, 2003). Physical and social systems such as the World Wide Web (Albert et al., 1999; Huberman and Adamic, 1999), the Internet (Doyle et al., 2005), the worldwide air transportation network (Guimerá and Amaral, 2004; Guimerá et al., 2005), networks of acquaintance or other connections between individuals (Newman et al., 2002; Liben-Nowell et al., 2005), scientific collaboration networks (Newman, 2001; Barabási et al., 2002), and the network of human sexual contacts (Liljeros et al., 2001) are all examples of different systems studied under the network approach.

In addition, biological systems such as food webs (Paine, 1966; Cohen, 1978; Pimm, 1982), plant–animal mutualistic networks (Bascompte et al., 2003; Jordano et al., 2003), metabolic networks (Jeong et al., 2000; Ravasz et al., 2002), protein networks (Jeong et al., 2001; Giot et al., 2003; LaCount et al., 2005), and gene regulatory networks (Davidson et al., 2002; Luscombe et al., 2004), have also been explored using graph-theory methods. Perhaps the most challenging of such biological networks is that governing gene expression in a cell.

In a genome, thousands of genes direct the formation of proteins, including transcription factors that can activate or inhibit the transcription of genes to give mRNAs. Since these transcription factors are themselves products of genes, the ultimate effect is that genes regulates each other's expression as a part of gene regulatory networks (Davidson, 2001; Guelzim et al., 2002; Lee et al., 2002; Albert, 2005). The patterns of regulatory interactions at genomic scale (in which genes can affect each other's expression) are becoming increasingly resolved (Davidson et al., 2002; Guelzim et al., 2002; Lee et al., 2002; Stuart et al., 2003; Luscombe et al., 2004).

Recent evidence from whole-genome sequence suggests that organismal complexity arises much more from the elaborate regulation of gene expression than by the genome size itself (Knight, 2002; Levine and Tjian, 2003). In this context, previous results on small subsets of genes (Albert

*Corresponding author. Tel.: +34 954 23 23 40; fax: +34 954 62 11 25.
E-mail address: fortuna@ebd.csic.es (M.A. Fortuna).

0022-5193/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.jtbi.2007.03.017

and Othmer, 2003) have shown that the robustness of the network is depending on the topology (i.e., the distribution of the number of interactions a gene participates in) and the signature of regulatory interactions (i.e., whether the interaction activates or inhibits a gene). The effects of the topology of regulatory interactions on gene expression in large networks are, however, difficult to asses because the interaction signature is only known for a small subset of genes (Davidson et al., 2002; Guelzim et al., 2002; Lee et al., 2002; Albert and Othmer, 2003; Luscombe et al., 2004, see, however, Madan Babu et al., 2006).

2. Methods and results

In the present study we compiled the network of software packages of Debian GNU/Linux operating system along with their dependence and conflict interactions with the aim of shedding some light on the effect of the regulatory network structure on the number of active transcriptors. The interactions between software packages we consider to be regulatory interactions in the sense that they may or may not allow the installation of packages in the system. On the one hand, the package i depends (k^{dep}) of the package j when j has to be installed for i work (i.e., j activates i because i needs j to work). On the other hand, the package i has a conflict (k^{con}) with the package j when i does not work if j is installed in the system (i.e., j inhibits i). It does not necessarily mean that the package j also has a conflict with the package i (sometimes the package j is an improved version of the package i in a way that if i is already installed in the system then j improves it, but if j is installed then it already contains i and the later cannot be installed). Because links are directed we can find packages with ingoing and outgoing links (k_{in} and k_{out}, respectively). In a detailed picture of the network, we can identify all node types as a function of their k_{in} and k_{out} interactions. On the one hand, packages with just $k_{in}^{dep}>0$, packages with just $k_{in}^{con}>0$, and packages with both $k_{in}^{dep}>0$ and $k_{in}^{con}>0$, if they depend or have a conflict with other packages, or both, respectively. On the other hand, packages with $k_{out}^{dep}>0$, packages with $k_{out}^{con}>0$, and packages with $k_{out}^{dep}>0$ and $k_{out}^{con}>0$, if other packages depend or enter into conflict with them, or both, respectively.

To clarify the relationship between a regulatory gene network and the dependence network of software packages we must simplify the former. A gene network has two types of nodes, which correspond to transcription factors and the genes encoding them, and two types of directed links, which correspond to transcriptional regulation and translation (Lee et al., 2002). For simplicity, transcription factors are often combined with the genes encoding them (thus all nodes correspond to genes), and transcription and translation are condensed to one link (the assumption being if any of both processes happens, the other occurs too; see Albert, 2005). The nodes representing target genes that do not encode transcription factors become sinks (the above described packages with $k_{out}=0$) while non-transcriptionally regulated transcription factors correspond to sources ($k_{in}=0$). If the gene i encodes a transcriptional factor that activates the transcription of the mRNA of the gene j it will be said that the gene i activates the gene j, and if the gene i encodes a transcriptional factor that inhibits the transcription of the mRNA of the gene j it will be said that the gene i inhibits the gene j. These types of regulatory interactions are quite analogous to dependences k^{dep} and conflicts k^{con} in the network of software packages. Hence, if a gene i has $k_{in}^{dep}>0$ interactions it means that a k number of genes are needed to activate it. In the same way, if a gene i has $k_{out}^{dep}>0$ interactions it means that the gene i encodes a k number of transcriptional factors that activate other genes. Similarly, inhibition is analogous to conflict, k^{con}.

Let us now assume that the rules governing the transcription of a gene are determined by a Boolean function of the state of its transcriptional activators and inhibitors (Kauffman, 1969; Albert and Othmer, 2003). Transcription will only begin if the activators are expressed and the inhibitors are not (Kauffman, 1969). The effect of transcriptional activators and inhibitors is never additive, but rather inhibitors are dominant. The states of the nodes evolve in discrete time steps under several rules to a steady state in all nodes (Albert and Othmer, 2003). Each steady state or fixed point has a specific number of active and inactive transcriptors. The total number of active genes in each steady state represents the active network size. After n replicates of the network, the frequency of each steady state represents the distribution of the active network size (see Li et al., 2004, table 1).

Although we have defined the similarities between transcriptional and dependence networks, we should point out that there are some particularities of gene networks that preclude a full comparison of the two types of networks. Specifically, the self-degradation processes, the complex dynamics of activator and repressor, and the feedback circuits in which some genes are embedded make a perfect comparison difficult. In the Boolean network model, and in real gene networks, in addition to fixed points, cyclic attractors may also exist (Kauffman, 1969). This is not the case for the dependence network of software packages, in which a steady state of installed packages is reached once no more packages can be installed without entering into conflict with the previously installed packages. Another important difference is that in the Boolean network model the set of genes that are expressed in the attractors may be very different from the set of genes that were originally expressed in the initial condition. In contrast, in the dependence network of software packages all the installed packages (expressed genes) are retained throughout time, so that at the end all the packages that were originally installed remain installed. The analogy we can obtain, however, is the similarity of the final states in both types of networks. The total number of active genes in gene networks or

packages installed in software networks makes possible the comparison. At this level, does the number of activated nodes depend on the structure of regulatory interactions? Or in a more specific gene context, could small genomes with scale-free regulatory topologies allow much more gene expression than large genomes with exponential topologies?

To test the effect of the topology of a large regulatory network on the maximum number of activated nodes (hereafter active network size) we develop a null model (see Null model section) that (1) preserves the total number of dependences and conflicts as in the real network, and (2) maintains statistically the frequency of packages with different combinations of ingoing and outgoing interactions for dependences and conflicts (Fig. 1), but forcing them to an exponential degree distribution (Fig. 2a,c). The degree distribution $P(k)$ gives the fraction of nodes that have degree k and in directed and signed networks (as in this case) is measured for both ingoing (k_{in}) and outgoing (k_{out}) links for dependences (k^{dep}) and conflicts (k^{con}). An exponential degree distribution implies that nodes have a well-defined average number of links. A power-law degree distribution, on the other hand, indicates a much higher variability in the number of links per node. That is, the bulk of nodes have a few links, but a few nodes are much more connected than expected by chance. The rules of the null model avoid (1) dependence loops (i.e., if the package i depends of the package j, the package j or whatever it depends on cannot depend of the package i), and (2) contradictory links (i.e., if the package i depends on the package j, the package j or whatever it depends on cannot have a conflict with the package i).

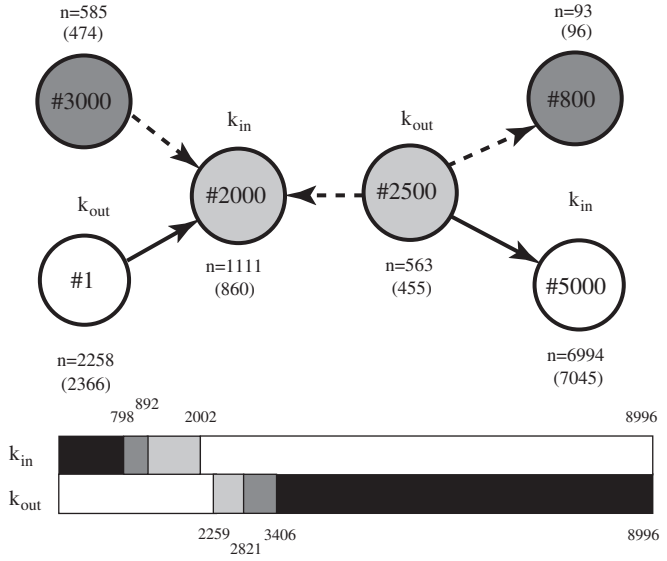

Fig. 1. Hypothetical graph illustrating the type of packages as a function of their k_{in} (number of incoming edges per node) and k_{out} (number of outgoing edges per node) and types of interactions (solid arrows represent dependences k^{dep} (number of dependences per node), and dotted arrows conflicts k^{con} (number of conflicts per node)). Packages with $k_{in}^{dep}>0$ (e.g., package number 5000), $k_{in}^{con}>0$ (e.g., package number 800) or both (e.g., package number 2000) mean that they depend or have a conflict with other packages, or both, respectively. Packages with $k_{out}^{dep}>0$ (e.g., package number 1) or $k_{out}^{con}>0$ (e.g., package number 3000) or both (e.g., package number 2500) mean that other packages depend or enter into conflict with them, or both, respectively. The total number of packages with each type of incoming and outgoing link in the network is n (in brackets the average value after 1000 replicates of the null model). Intervals in the horizontal bars correspond to the number of each type of package in the null model: packages with $k_{in}^{con}>0$ or $k_{out}^{con}>0$ (dark gray); packages with $k_{in}^{dep}>0$ and/or $k_{out}^{dep}>0$ (white); packages with $k_{in}^{dep}>0$ and $k_{in}^{con}>0$, or $k_{out}^{dep}>0$ and $k_{out}^{con}>0$ (light gray); and packages with $k_{in}=0$ or $k_{out}=0$ (black, not shown in the above graph).

We assembled 1000 replicates (dependence networks) from both real data (power-law degree distribution) and data from the null model (exponential degree distribution, see Fig. 2a,c) using a trace-back algorithm (see Trace-back algorithm section) and recorded the active network size in each replicate. Note that as a function of the assembly temporal sequence, each replicate from real data and data from the null model has a different number of packages installed. In this way we obtain the frequency distribution of the active network size from both real data and data from the null model (Fig. 2b). The frequency distribution of the active network size from data of the null model is significantly smaller than from the real data (Fig. 2b).

Our results suggest that genomes with scale-free regulatory topologies could allow a higher number of expressed genes at the steady state than genomes with exponential topologies. Rewiring connections instead of increasing the number of genes seems to be an alternative mechanism to enhance the expression of the network (Knight, 2002; Stuart et al., 2003; Luscombe et al., 2004). Recently, Mochizuki (2005) has showed that the diversity of cell states does not increase with either gene number or links number, but is instead highly influenced by the number of regulatory genes. This indicates that increases in the number of genes may not be the direct force driving the evolution of variety of cell types. The present study offers also a framework to explore the real ratios of activating and inhibiting interactions in large gene networks when large databases become available.

3. Data set

The regulatory network described here is composed by the binary i386 packages belonging to the sections main, contrib and non-free of the recently obsolete stable release of Debian distribution (3.0, alias *Woody*), available from the US Debian Server (http://www.us.debian.org/releases/woody/). It includes 8996 nodes (packages), and 31,904 regulatory interactions (30,003 dependences and 1901 conflicts).

4. Null model

We first coded the packages as a function of the type of link. For k_{in} interactions: (1) packages with $k_{in}^{dep}=0$ and $k_{in}^{con}=0$, from 1 to 798 (798 packages without k_{in}

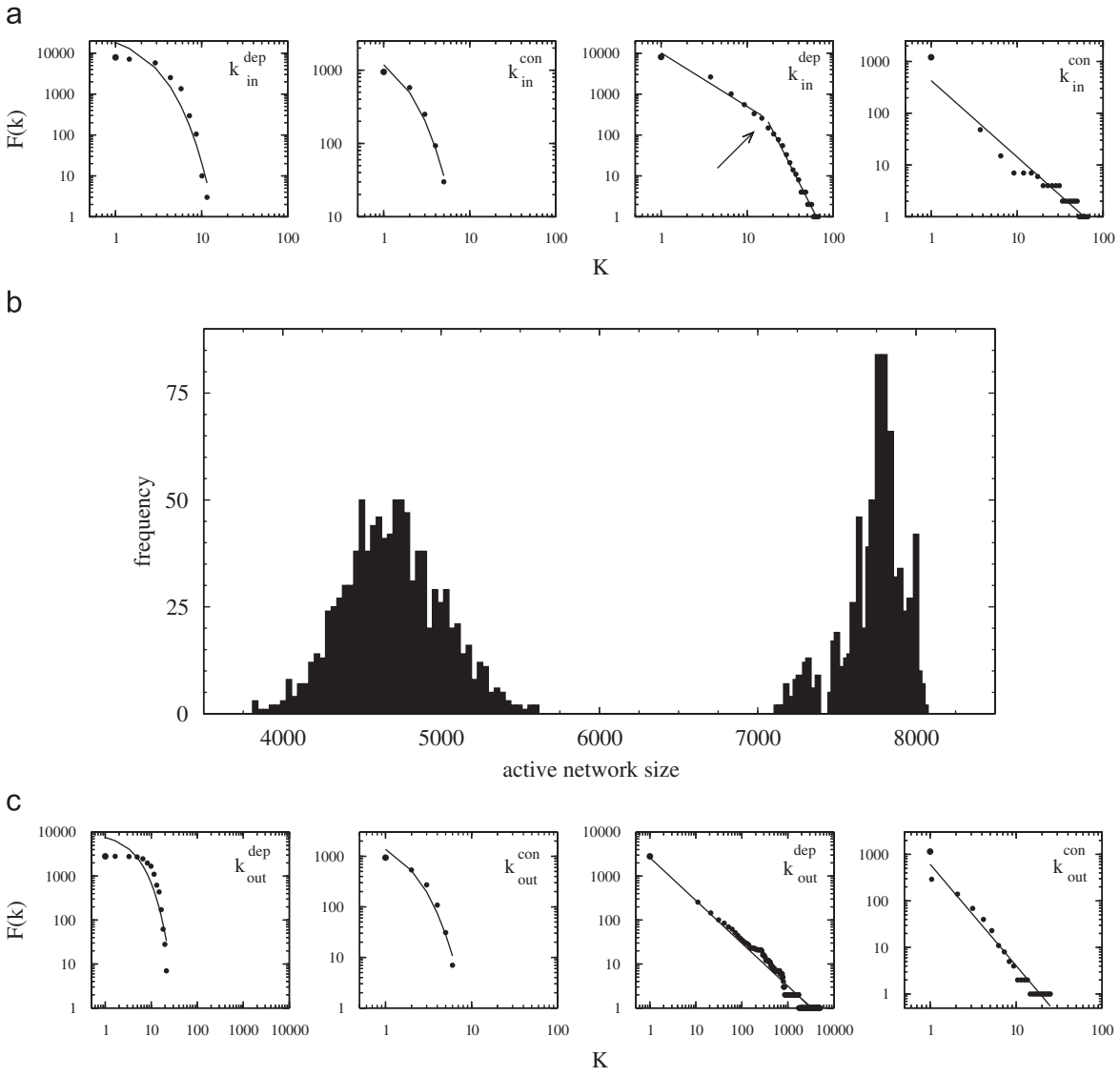

Fig. 2. (a) Cumulative k_{in} degree distributions of the null model (left side) and real data (right side). All degree distributions are marginally significant for both the null model (k_{in}^{dep}, $n = 7894$; k_{in}^{con}, $n = 944$), and real data (k_{in}^{dep}, $n = 8105$; k_{in}^{con}, $n = 1204$), decaying exponentially ($P = 0.07$ and 0.07, respectively) for the null model, and as a power law for real data ($P = 0.1$ for the first regression, and $P = 0.1$ for the second with a breakpoint in $k = 15$ (solid arrow), and $P = 0.07$, respectively). The degree distribution of the null model represents the average value for 10 replicates. (b) The frequency distribution of the active network size differs from a normal distribution for real data (right, Jarque-Bera test, $P < 0.05$, with an average active network size of 7647 packages) and does not differ from a normal distribution for the null model (left, Jarque-Bera test $P = 0.2$, with an average network size of 4750 packages). No replicate from the null model distribution is equal or higher than any replicate from the real data distribution ($P < 0.0001$). (c) Cumulative k_{out} degree distributions of the null model (left side) and real data (right side). Degree distributions for the null model are significant (k_{out}^{dep}, $n = 2821$), and marginally significant (k_{out}^{con}, $n = 941$), decaying exponentially in both cases ($P < 0.05$ and 0.09, respectively). Degree distributions for real data are significant (k_{out}^{dep}, $n = 2821$), and marginally significant (k_{out}^{dep}, $n = 1148$), decaying in both cases as a power law ($P < 0.05$ and 0.08, respectively). The degree distribution of the null model represents the average value for 10 replicates.

interactions as in the real network); (2) packages just with $k_{in}^{con} > 0$, from 799 to 891 (93 packages as in the real network); (3) packages with $k_{in}^{con} > 0$ and $k_{in}^{dep} > 0$, from 892 to 2002 (1111 packages as in the real network); and (4) packages just with $k_{in}^{dep} > 0$, from 2003 to 8996 (6994 packages as in the real network). For k_{out} interactions: (1) packages just with $k_{out}^{dep} > 0$, from 1 to 2258 (2258 packages as in the real network); (2) packages with $k_{out}^{dep} > 0$ and $k_{out}^{con} > 0$, from 2259 to 2821 (563 packages as in the real network); (3) packages just with $k_{out}^{con} > 0$, from 2822 to 3406 (585 packages as in the real network); and (4) packages with $k_{out}^{dep} = 0$ and $k_{out}^{con} = 0$, from 3407 to 8996 (5590 packages without k_{out} interactions as in the real network). Now we choose randomly a package i and a package j for each dependence (30,003) and conflict (1901) in the following way: (1) if the link is a dependence, we randomly choose a package i from the range of packages with $k_{in}^{dep} > 0$ (range 892–8996) and a package j from the range of packages with $k_{out}^{dep} > 0$ (range 1–2821) according with the following rule: the code of the package i must be bigger

than the code of the package j. (2) if the link is a conflict, we randomly choose a package i from the range of packages with $k_{in}^{con}>0$ (range 799–2002) and a package j from the range of packages with $k_{out}^{con}>0$ (range 2259–3406) according with the following rule: the code of the package i must be smaller than the code of the package j. Both conditions avoid dependence and conflict loops and also contradictory links.

The correspondence between the number of packages of each type in the real network and the average values from the null model after 1000 replicates is shown in Fig. 1.

5. Trace-back algorithm

Trace-back algorithm selects randomly a package, checks dependences and conflicts of this package with the rest of packages of the network, and whether they are installed or not in the network. If the package has a conflict with an already installed one, it is discarded and never will be part of the network. If there are no conflicts with installed packages, the algorithm checks whether some of the packages on which it depends directly or indirectly (by successive dependences) has been discarded or has a conflict with an already installed package. If so, is discarded too. Otherwise, is installed with all packages on which it depends directly as well as indirectly. It continues until no more packages are available to be included (i.e., packages excluded by the assembly temporal sequence due to their conflicts with packages already installed). Before starting each replicate, we have automatically installed the 100 packages considered basic to the system works (base-packages section in the URL indicated in Data set).

Acknowledgments

We thank Jordi Bascompte, Pedro Jordano, Peter Buston, and Miguel A. Rodríguez for helpful discussions that have contributed largely to improve the ms. M.A.F. was supported by the Spanish Ministry of Science and Education (Fellowship BES-2004-6682). C.J.M. was supported by the Spanish Ministry of Science and Education (Fellowship FP-2000-6137) and a Postdoctoral Fellowship at the National Center for Ecological Analysis and Synthesis, a Center funded by NSF (Grant #DEB-0553768), the University of California, Santa Barbara, and the State of California.

Appendix A. Supplementary data

Supplementary data[1] associated with this article can be found in the online version at 10.1016/j.jtbi.2007.03.017.

[1]The network used is available as a text file, and both the null model applied and the trace-back algorithm developed are available as a MatLab code.

References

Albert, R., 2005. Scale-free networks in cell biology. J. Cell Sci. 118, 4947–4957.

Albert, R., Barabási, A.-L., 2002. Statistical mechanics of complex networks. Rev. Mod. Phys. 74, 47–97.

Albert, R., Othmer, H.G., 2003. The topology of the regulatory interactions predicts the expression pattern of the segment polarity in *Drosophila melanogaster*. J. Theor. Biol. 223, 1–18.

Albert, R., Jeong, H., Barabási, A.-L., 1999. Diameter of the world-wide web. Nature 401, 130–131.

Barabási, A.-L., Jeong, H., Ravasz, E., Néda, Z., et al., 2002. Evolution of the social network of scientific collaborations. Physica A 311, 590–614.

Bascompte, J., Jordano, P., Melián, C.J., Olesen, J.M., 2003. The nested assembly of plant–animal mutualistic networks. Proc. Natl Acad. Sci. USA 100, 9383–9387.

Cohen, J.E., 1978. Food Webs and Niche Space. Princeton University Press, Princeton.

Davidson, E.H., 2001. Genomic Regulatory Systems; Development and Evolution. Academic Press, San Diego, CA.

Davidson, E.H., Rast, J.P., Oliveri, P., Ransick, A., et al., 2002. A genomic regulatory network for development. Science 295, 1669–1678.

Doyle, J.C., Alderson, D.L., Li, L., Low, S., et al., 2005. The robust yet fragile nature of the Internet. Proc. Natl Acad. Sci. USA 102, 14497–14502.

Giot, L., Bader, J.S., Brouwer, C., Chaudhuri, A., et al., 2003. A protein interaction map of *Drosophila melanogaster*. Science 302, 1727–1736.

Guelzim, N., Bottani, S., Bourgine, P., Képes, F., 2002. Topological and causal structure of the yeast transcriptional regulatory network. Nat. Genet. 31, 60–63.

Guimerá, R., Amaral, L.A.N., 2004. Modeling the world-wide airport network. Eur. Phys. J. B 38, 381–385.

Guimerá, R., Mossa, S., Turtschi, A., Amaral, L.A.N., 2005. The world-wide air transportation network: anomalous centrality, community structure, and cities' global roles. Proc. Natl Acad. Sci. USA 102, 7794–7799.

Huberman, B.A., Adamic, L.A., 1999. Growth dynamics of the World-Wide Web. Nature 401, 131.

Jeong, H., Tombor, B., Albert, R., Oltvai, Z.N., Barabási, A.-L., 2000. The large-scale organization of metabolic networks. Nature 407, 651–654.

Jeong, H., Mason, S.P., Barabási, A.-L., Oltvai, Z.N., 2001. Lethality and centrality in protein networks. Nature 411, 41–42.

Jordano, P., Bascompte, J., Olesen, J.M., 2003. Invariant properties in coevolutionary networks of plant–animal interactions. Ecol. Lett. 6, 69–81.

Kauffman, S.A., 1969. Metabolic stability and epigenesis in randomly constructed genetics nets. J. Theor. Biol. 22, 437–467.

Knight, J., 2002. All genomes great and small. Nature 417, 374–376.

LaCount, D.J., Vignali, M., Chettier, R., Phansalkar, A., et al., 2005. A protein interaction network of the malaria parasite *Plasmodium falciparum*. Nature 438, 103–107.

Lee, T.I., Rinaldi, N.J., Robert, F., Odom, D.T., et al., 2002. Transcriptional regulatory networks in *Saccharomyces cerevisiae*. Science 298, 799–804.

Levine, M., Tjian, R., 2003. Transcription regulation and animal diversity. Nature 424, 147–151.

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.

Liben-Nowell, D., Novak, J., Kumar, R., Raghavan, P., Tomkins, A., 2005. Geographic routing in social networks. Proc. Natl Acad. Sci. USA 102, 11623–11628.

Liljeros, F., Edling, C.R., Amaral, L.A.N., Stanley, H.E., Aberg, Y., 2001. The web of human sexual contacts. Nature 411, 907–908.

Luscombe, N.M., Madan Babu, M., Yu, H., Snyder, M., et al., 2004. Genomic analysis of regulatory network dynamics reveals large topological changes. Nature 431, 308–312.

Madan Babu, M., Teichmann, S.A., Aravind, L., 2006. Evolutionary dynamics of prokaryotic transcriptional regulatory networks. J. Mol. Biol. 358, 614–633.

Mochizuki, A., 2005. An analytical study of the number of steady states in gene regulatory networks. J. Theor. Biol. 236, 291–310.

Newman, M.E.J., 2001. The structure of scientific collaboration networks. Proc. Natl Acad. Sci. USA 98, 404–409.

Newman, M.E.J., 2003. The structure and function of complex networks. SIAM Rev. 45, 167–256.

Newman, M.E.J., Watts, D.J., Strogatz, S.H., 2002. Random graph models of social networks. Proc. Natl Acad. Sci. USA 99, 2566–2572.

Paine, R.T., 1966. Food web complexity and species diversity. Am. Nat. 100, 65–75.

Pimm, S.L., 1982. Food Webs. Chapman & Hall, London.

Ravasz, E., Somera, A.L., Mongru, D.A., Oltvai, Z.N., Barabási, A.-L., 2002. Hierarchical organization of modularity in metabolic networks. Science 297, 1551–1555.

Strogatz, S.H., 2001. Exploring complex networks. Nature 410, 268–276.

Stuart, J.M., Segal, E., Koller, D., Kim, S.K., 2003. A gene-coexpression network for global discovery of conserved genetic modules. Science 302, 249–255.