Comparing community structure identification

J. Stat. Mech. (2005) P09008

(http://iopscience.iop.org/1742-5468/2005/09/P09008)

Journal of Statistical Mechanics: Theory and Experiment
An IOP and SISSA journal

Comparing community structure identification

Leon Danon[1,2], **Albert Díaz-Guilera**[1], **Jordi Duch**[2] **and Alex Arenas**[2]

[1] Departament de Fisica Fonamental, Universitat de Barcelona, Marti i Franques 1, 08086 Barcelona, Spain
[2] Departament d'Enginyeria Informàtica i Matemàtiques, Universitat Rovira i Virgili, Campus Sescelades, 43007 Tarragona, Spain
E-mail: leon.danon@urv.net, ldanon@ffn.ub.es, albert.diaz@ub.edu, jordi.duch@urv.net and alexandre.arenas@urv.net

Received 15 August 2005
Accepted 5 September 2005
Published 26 September 2005

Online at stacks.iop.org/JSTAT/2005/P09008
doi:10.1088/1742-5468/2005/09/P09008

Abstract. We compare recent approaches to community structure identification in terms of sensitivity and computational cost. The recently proposed modularity measure is revisited and the performance of the methods as applied to *ad hoc* networks with known community structure, is compared. We find that the most accurate methods tend to be more computationally expensive, and that both aspects need to be considered when choosing a method for practical purposes. The work is intended as an introduction as well as a proposal for a standard benchmark test of community detection methods.

Keywords: analysis of algorithms, network dynamics

Contents

1. Introduction

The study of complex networks has received an enormous amount of attention from the scientific community in recent years [1]–[6]. Physicists in particular have become interested in the study of networks describing the topologies of a wide variety of systems, such as the world wide web, social and communication networks, biochemical networks and many more. An important open problem is the analysis of modular structure found in many networks [7]. Distinct modules or communities within networks can loosely be defined as subsets of nodes which are more densely linked, when compared to the rest of the network. Such communities have been observed in different kinds of networks, most notably in social networks, but also in networks of other origin such as metabolic or economic networks [8]–[11]. As a result, the problem of identification of communities has been the focus of many recent efforts.

Community detection in large networks is potentially very useful. Nodes belonging to a tight-knit community are more than likely to have other properties in common. For instance, in the world wide web, community analysis has uncovered thematic clusters [12, 13]. In biochemical or neural networks, communities may be functional groups [14], and separating the network into such groups could simplify functional analysis considerably.

The problem of community detection is quite challenging and has been the subject of discussion in various disciplines. A simpler version of this problem, the graph bi-partitioning problem (GBP), has been the topic of study in the realm of computer science for decades. Here, one looks to separate the graph into two densely connected communities of equal size, which are connected with the minimum number of links. This is an NP-complete problem[3] [16]; however, several methods have been proposed to reduce the complexity of the task [17]–[20]. In real complex networks we often have no idea how many communities we wish to discover, but in general it is more than two. This makes

[3] In computational complexity theory, NP ('Non-deterministic Polynomial time') is the set of decision problems solvable in polynomial time on a non-deterministic Turing machine. NP-complete problems are the most difficult problems in NP.

doi:10.1088/1742-5468/2005/09/P09008

the process all the more costly. What is more, communities may also be hierarchical; that is, communities may be further divided into sub-communities and so on [21]–[24].

Nevertheless, many attempts to tackle these problems have been proposed recently. The proposed methods vary considerably in terms of approach and application, which makes them difficult to compare. Community identification is potentially very useful, and researchers from a number of fields may be interested in using one or several of the methods for their own purposes. But which? In order for the reader to be able to make an informed decision as to which method is most appropriate for which purpose, we distil information from the literature and compare the performance of those methods which lend themselves to objective comparison.

To this end, this paper is organized as follows. In section 2 we revisit the modularity measure designed to evaluate how good a particular partition of a network is. Then, we describe how to measure the sensitivity of the various methods and suggest the use of a more accurate representation of algorithm sensitivity based on information theory. We then compare the methods from a computational cost perspective and compare their sensitivity when applied to ad hoc networks with community structure. Finally, we suggest appropriate choices of community identification methods for a few different problems.

2. Evaluating community identification

A question that has been raised in recent years is how a given partition of a network into communities can be evaluated. A simple approach that has become widely accepted was proposed in [25]. It is based on the intuitive idea that random networks do not exhibit community structure. Let us imagine that we have an arbitrary network and an arbitrary partition of that network into n_c communities. It is then possible to define an $n_\mathrm{c} \times n_\mathrm{c}$ size matrix $\mathbf{e}$ where the elements e_{ij} represent the fraction of total links starting at a node in partition i and ending at a node in partition j. Then, the sum of any row (or column) of $\mathbf{e}$, $a_i = \sum_j e_{ij}$ corresponds to the fraction of links connected to i.

If the network does not exhibit community structure, or if the partitions are allocated without any regard to the underlying structure, the expected value of the fraction of links within partitions can be estimated. It is simply the probability that a link begins at a node in i, a_i, multiplied by the fraction of links that end at a node in i, a_i. So the expected number of intra-community links is just $a_i a_i$. On the other hand, we know that the *real* fraction of links exclusively within a partition is e_{ii}. So, we can compare the two directly and sum over all the partitions in the graph.

$$Q \equiv \sum_i (e_{ii} - a_i^2). \tag{1}$$

This is a measure known as *modularity*. As an example, let us consider a network comprised of n_c fully connected components with no links between them. If we then have n_c partitions, corresponding exactly to the components, modularity will have a value of $1 - 1/n_\mathrm{c}$. As n_c gets large, this value tends to 1. On the other hand, for particularly ‘bad’ partitions, for example, when all the nodes are in a community of their own, the value of modularity can take negative values. This is due to the fact that when nodes are alone in partitions there can be no internal links. To avoid this issue, Massen and Doye propose an alternative measure [26].

It is tempting to think that random networks exhibit very small values of modularity. As Guimerà *et al* show, this is not the case [27]. It is possible to find a partition which not only has a nonzero value of modularity for random networks of finite size, but that this value is quite high; for example, a network of 128 nodes and 1024 links has a maximum modularity of 0.208. This suggests that these networks that cannot have a modular structure actually appear to have one due to fluctuations.

3. Comparative evaluation

The methods that have been presented recently are extremely varied, and are based on a range of different ideas. In a longer article, we describe the methods in more detail and classify them according to the type of approach they present [28]. Also, the full description of each can be found in the respective references. Here we concentrate on comparing the methods in terms of performance. In order for the reader to be able to compare the algorithms, both in terms of their speed and sensitivity, we would like to present a qualitative comparison for all the methods presented until now. However, this is not possible as they are very varied, both conceptually and in their applications.

One way that has been employed to test sensitivity in many cases is to see how well a particular method performs when applied to ad hoc networks with a well known, fixed community structure [25]. Such networks are typically generated with $n = 128$ nodes, split into four communities containing 32 nodes each. Pairs of nodes belonging to the same community are linked with probability p_{in}, whereas pairs belonging to different communities are joined with probability p_{out}. The value of p_{out} is taken so that the average number of links a node has to members of any other community, z_{out}, can be controlled. While p_{out} (and therefore z_{out}) is varied freely, the value of p_{in} is chosen to keep the total average node degree, k, constant, and set to 16. As z_{out} is increased from zero, the communities become more and more diffuse and harder to identify (see figure 1). Since the 'real' community structure is well known in this case, it is possible to measure the number of nodes correctly classified by the method of community identification.

In [24], the author describes a method to calculate this value. The largest group found within each of the four 'real' communities is considered correctly classified. If more than one original community is clustered together by the algorithm, all nodes in that cluster are considered incorrectly classified. For example, for the case when z_{out}/k is small, if a method finds three communities, two of which correspond exactly to two original communities, and a third, which corresponds to the other two clustered together, this measure would consider half the nodes correctly classified. As the author notes, this measure is quite harsh, and some nodes which one may consider to be correctly clustered are not counted. On the other end of the spectrum, as z_{out}/k becomes large, and the networks become essentially random networks, this method rewards the identification of smaller clusters found within each of the original communities, which could be misleading.

We suggest that a more discriminatory measure is more appropriate, and propose the use of the *normalized mutual information* measure, as described in [29, 30]. It is based on defining a *confusion matrix* $\mathbf{N}$, where the rows correspond to the 'real' communities, and the columns correspond to the 'found' communities. The element of $\mathbf{N}$, N_{ij}, is the number

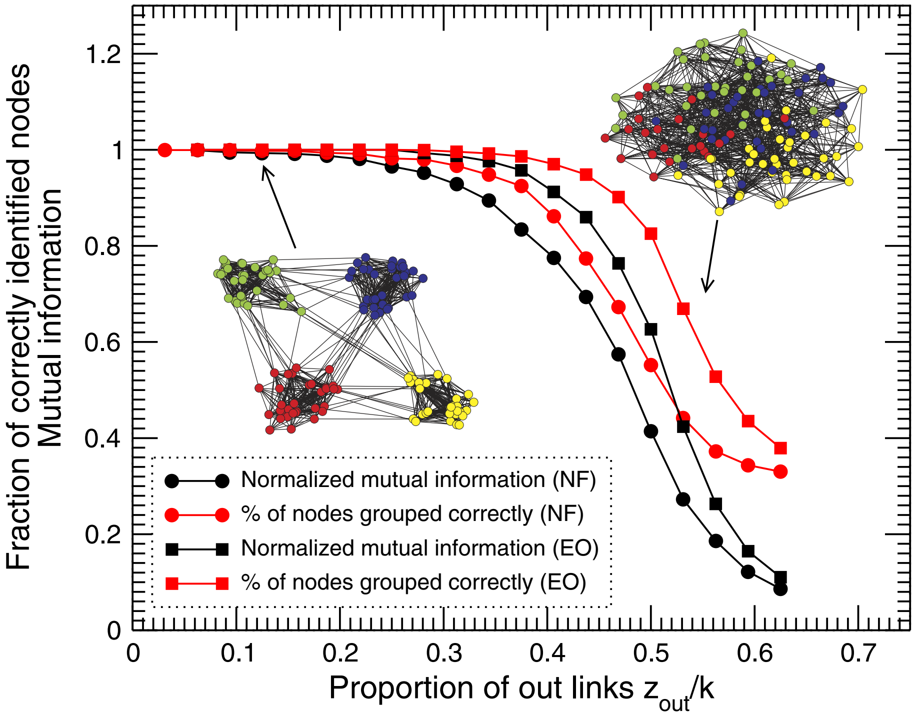

Figure 1. Algorithm sensitivity as applied to ad hoc networks with $n = 128$, the network divided into four communities of 32 nodes each and total average degree $z_{\rm out}$ fixed to 16. For low $z_{\rm out}/k$ the communities are easily distinguished. For higher $z_{\rm out}/k$ this becomes more complicated. Both measures of comparing original communities to ones found by the detection method are shown. The normalized mutual information measure is more discriminatory and appears more sensitive to errors in the community identification procedure. The results are shown for Newman's fast algorithm [24] and the extremal optimization algorithm [31].

of nodes in the real community i that appear in the found community j. A measure of similarity between the partitions, based on information theory, is then

$$I(A,B) = \frac{-2\sum_{i=1}^{c_A}\sum_{j=1}^{c_B} N_{ij}\log(N_{ij}N/N_{i.}N_{.j})}{\sum_{i=1}^{c_A} N_{i.}\log(N_{i.}/N) + \sum_{j=1}^{c_B} N_{.j}\log(N_{.j}/N)} \tag{2}$$

where the number of real communities is denoted c_A and the number of found communities is denoted c_B, the sum over row i of matrix N_{ij} is denoted $N_{i.}$ and the sum over column j is denoted $N_{.j}$.

If the found partitions are identical to the real communities, then $I(A,B)$ takes its maximum value of 1. If the partition found by the algorithm is totally independent of the real partition, for example when the entire network is found to be one community, $I(A,B) = 0$.

Both measures of accuracy give a good idea of how a method performs. However, the measure we propose for use here is more representative of sensitivity if the performance is dubious, since it measures the amount of information correctly extracted by the algorithm

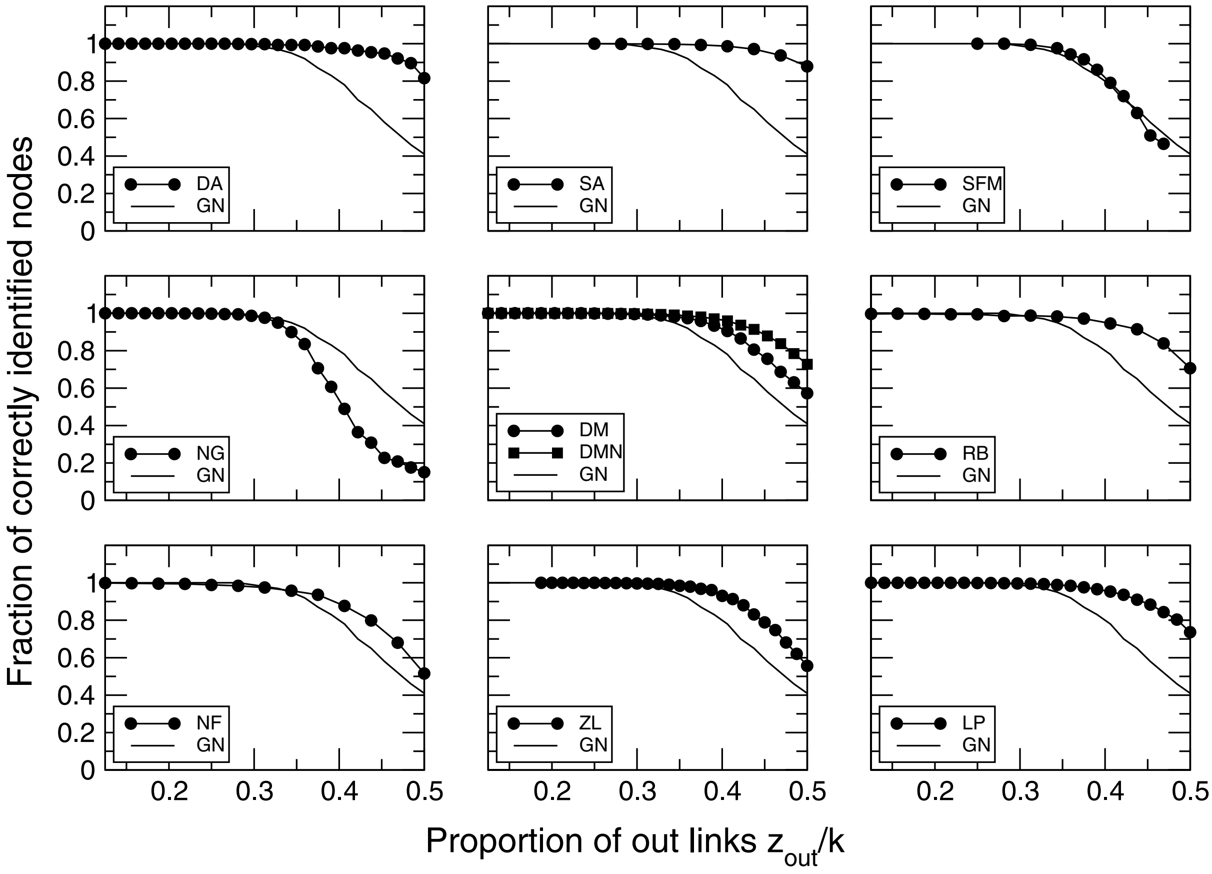

Figure 2. Comparing algorithm sensitivity using ad hoc networks with predetermined community structure. The x-axis is the proportion of connections to outside communities z_{out}/k and the y-axis is the fraction of nodes correctly identified by the method measure as described in [24]. The labels here correspond to the different methods and are listed in table 1.

explicitly. As an example, for small z_{out}, where two original communities are clustered together by the algorithm, this measure does not punish the algorithm as severely, taking into account the ability to extract at least some information about the community structure. On the other hand, for large z_{out}, this method is able to detect that the clusters found by the algorithm have little to do with the original communities, and $I(A, B) \to 0$.

In figure 2 we show the sensitivity of all methods we have been able to gather. The percentage of correctly identified nodes is calculated using the method described in [24], since this is the method employed by the various authors. We can see that accuracy varies in a similar way across the different methods as z_{out} increases and the communities become more diffuse. So, it remains difficult to compare the performance by looking at the methods separately, even with a reference performance.

To summarize the large amount of information, in figure 3 we plot the fraction of correctly identified nodes for only three values of z_{out} (6, 7 and 8), corresponding to $z_{\mathrm{out}}/k = 0.375$, 0.4375 and 0.5 respectively, for each method. From this we can see that most of the methods perform very well for $z_{\mathrm{out}} = 6$ ($z_{\mathrm{out}}/k = 0.375$), and even for

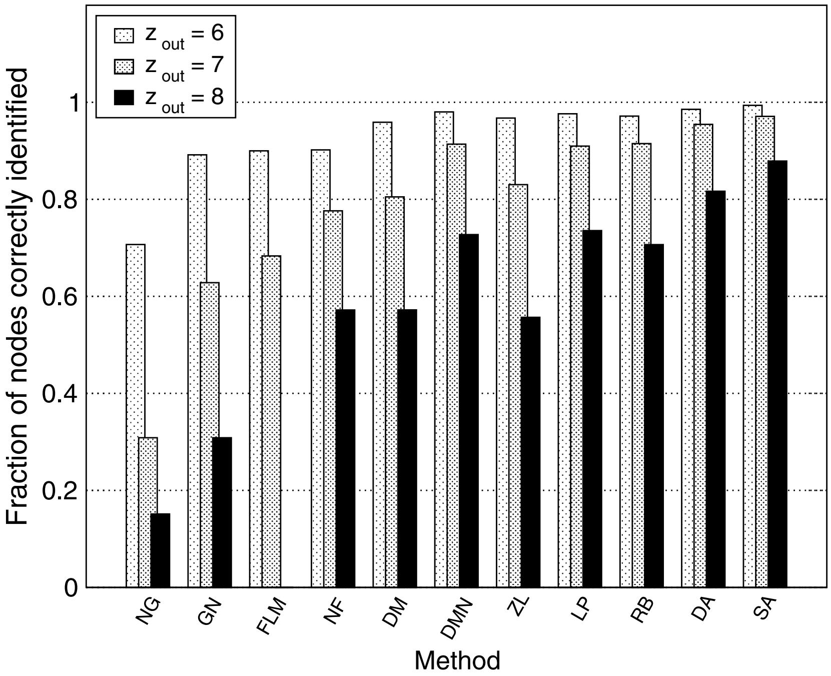

Figure 3. The fraction of correctly identified nodes at three specific values of z_{out}, 6, 7 and 8 for all available methods and for networks with fixed $k = 16$. Note that for the FLM method, the data for $z_{\mathrm{out}} = 8$ were not available. Here we can see that most of the methods are very good at finding the 'correct' community structure for values of z_{out} up to 6. At $z_{\mathrm{out}} = 7$ some methods begin to falter, but most still identify more than half of the nodes correctly. At $z_{\mathrm{out}} = 8$, when on average half the links are external, two methods are still able to identify over 80% of the nodes correctly.

$z_{\mathrm{out}} = 7$ ($z_{\mathrm{out}}/k = 0.4375$) most can identify more than half the nodes correctly. For $z_{\mathrm{out}} = 8$ ($z_{\mathrm{out}}/k = 0.5$) two methods are still able to identify more than 80% of the nodes correctly[4].

While accuracy is an essential consideration when choosing a method, it is just as important to consider the computational effort needed to perform the analysis [42]. For some of the approaches described in the literature, we have collected estimates of how the cost scales with network observables. For networks with n nodes and m links, the methods scale between $\mathrm{O}(m + n)$ for the fastest, and $\mathrm{O}(\exp(n))$ for the slowest (table 1). Such diversity is due to the different approaches taken by the authors. The faster methods tend to be approximate and less accurate, while the slower methods have other advantages (see [28] for a more detailed discussion). Differences in speed only become important when dealing with larger networks.

[4] One might expect that as the proportion of out links approaches 0.5 community structure no longer exists. However, since the external links are distributed among the other three communities, individual nodes remain more strongly connected to their own community than to other communities, even at this high value of z_{out}/k.

Table 1. Table summarizing how the computational cost of different approaches scales with number of nodes n, number of links m and average degree $\langle k \rangle$ [42]. The labels shown here are used in figures 2 and 3.

Author	Ref.	Label	Order
Eckmann and Moses	[13]	EM	$O(m\langle k^2\rangle)$
Zhou and Lipowsky	[14]	ZL	$O(n^3)$
Latapy and Pons	[15]	LP	$O(n^3)$
Newman	[24]	NF	$O(n\log^2 n)$
Newman and Girvan	[25]	NG	$O(m^2n)$
Girvan and Newman	[32]	GN	$O(n^2m)$
Guimera *et al*	[27, 43]	SA	Parameter dependent
Duch and Arenas	[31]	DA	$O(n^2\log n)$
Fortunato *et al*	[33]	FLM	$O(n^4)$
Radicchi *et al*	[34]	RCCLP	$O(n^2)$
Donetti and Muñoz	[35, 36]	DM/DMN	$O(n^3)$
Bagrow and Bollt	[37]	BB	$O(n^3)$
Capocci *et al*	[38]	CSCC	$O(n^2)$
Wu and Huberman	[39]	WH	$O(n+m)$
Palla *et al*	[40]	PK	$O(\exp(n))$
Reichardt and Bornholdt	[41]	RB	Parameter dependent

4. Choosing an algorithm

One has to take many factors into account when choosing an algorithm to use. The above comparison ought to give the reader an idea as to which algorithm is most appropriate for a given problem. In many cases, a compromise must be reached between accuracy and running time, especially for larger networks. To clarify this further, here are a few examples of real networks, and our suggestion for the appropriate community identification algorithm.

Say we want to analyse a relatively small network, for example the metabolic network of the worm *Caenorhabditis elegans*, which has 453 nodes. Since the network is small, and current desktop computer technology is reasonably fast, the speed of the algorithm should pose no restriction, and one is free to chose the slower, more accurate methods. In this case the simulated annealing (SA) method would be the most appropriate choice, since it gives the most accurate partitions, especially if the system is allowed to cool slowly (see [27, 26, 43] for more details).

Larger networks, with the number of nodes in the order of 10^5, become intractable with the more accurate methods. For example, when attempting to study the community structure of the actor collaboration network with 374 511 nodes, we estimate that the SA algorithm would take a few months of uninterrupted computation. However, a reasonable implementation of the fast algorithm would be able to perform this analysis in just a few hours [44], making it the appropriate choice, even if its accuracy is not the best.

Let us consider an intermediate sized network such as the Pretty Good Privacy (PGP) web of trust social network [45], containing 10 680 nodes. Although the SA algorithm would run in a reasonable time, it may be a better choice to compromise and employ a faster running algorithm. The EO method is not quite as accurate as SA, but the saving in computational effort for a network of this size is considerable. It is more accurate than the fast algorithm, however, and so this would make it a better choice.

5. Conclusion

In this work we have given a brief overview and comparison of the modern approaches to community identification in complex networks. A large amount of knowledge has been collected in the field, and real progress has been made, both in the identification of communities and their characterization. Some questions do remain open, and it is these that we would suggest for further study. Despite these efforts, the cost involved in computing communities in complex network remains significant. The fastest algorithm runs in linear time, but this particular method needs *a priori* knowledge of the number of expected communities, and assumes that all communities are of similar size [39]. At present, the fastest method for finding an unknown number of communities of unknown sizes has a cost which scales as $\mathrm{O}(n \log^2 n)$ with network size. While this makes the analysis of extremely large networks feasible, this algorithm does not guarantee that the partition found is the best possible one. Other algorithms which are more computationally expensive have other merits, such as accuracy or the ability to identify overlapping communities. So, when choosing a method one must consider carefully the context of its use. Ideally, one would like to have a method which guarantees accuracy and is fast at the same time, but finding such a method is challenging. The search for faster and more accurate methods is an important one and we would suggest this for further study.

Acknowledgments

The authors are grateful to Luca Donetti, Haijun Zhou, Mark Newman, Santo Fortunato, Jörg Reichardt, Claudio Castellano, Matthieu Latapy, Jean-Pierre Eckmann and Roger Guimera for providing their data and Sam Seaver for useful comments. This work has been supported by DGES of the Spanish Government Grant No. BFM-2003-08258 and EC-FET Open Project No. IST-2001-33555. LD gratefully acknowledges the funding of Generalitat de Catalunya.

References

[1] Barabási A L and Albert R, 2002 *Rev. Mod. Phys.* **74** 47
[2] Newman M E J, 2003 *SIAM Rev.* **45** 167
[3] Dorogovtsev S N and Mendes J F F, 2003 *Evolution of Networks: From Biological Nets to the Internet and WWW* (Oxford: Oxford University Press)
[4] Strogatz S H, 2001 *Nature* **410** 268
[5] Bornholdt S and Schuster H G (ed), 2002 *Handbook of Graphs and Networks—From the Genome to the Internet* (Berlin: Wiley–VCH)
[6] Pastor-Satorras R, Rubí M and Díaz-Guilera A (ed), 2003 *Statistical Mechanics of Complex Networks* (Berlin: Springer)
[7] Newman M E J, 2004 *Eur. Phys. J.* B **38** 321
[8] Boss M, Elsinger H, Summer M and Thurner S, 2003 *Preprint* cond-mat/0309582
[9] Ravasz E, Somera A L, Mongru D A, Olvai Z N and Barabási A L, 2002 *Science* **297** 1551
[10] Guimerà R and Amaral L A N, 2005 *Nature* **433** 895
[11] Holme P, Huss M and Jeong H, 2003 *Bioinformatics* **19** 532
[12] Flake G W, Lawrence S, Giles C L and Coetzee F M, 2002 *IEEE Comput.* **35** 66
[13] Eckmann J-P and Moses E, 2002 *Proc. Nat. Acad. Sci.* **99** 5825

[14] Zhou H and Lipowsky R, 2004 *Lecture Notes Comput. Sci.* **3038** 1062

[15] Latapy M and Pons P, 2004 *Preprint* cond-mat/0412568

[16] Garey M R and Johnson D S, 1979 *Computers and Intractability, A Guide to the Theory of NP-Completeness* (New York: Freeman)

[17] Kernighan B W and Lin S, 1970 *Bell Syst. Tech. J.* **49** 291

[18] Fiedler M, 1973 *Czech. Math. J.* **23** 298

[19] Boettcher S and Percus A G, 2001 *Phys. Rev.* E **64** 026114

[20] Pothen A, Simon H and Liou K-P, 1990 *SIAM J. Matrix Anal. Appl.* **11** 430

[21] Guimerà R, Danon L, Díaz-Guilera A, Giralt F and Arenas A, 2003 *Phys. Rev.* E **68** 065103

[22] Gleiser P and Danon L, 2003 *Adv. Complex Syst.* **6** 565

[23] Arenas A, Danon L, Díaz-Guilera A, Gleiser P M and Guimerà R, 2004 *Eur. Phys. J.* B **38** 373

[24] Newman M E J, 2004 *Phys. Rev.* E **69** 066133

[25] Newman M E J and Girvan M, 2004 *Phys. Rev.* E **69** 026113

[26] Massen C P and Doye J P K, 2005 *Phys. Rev.* E **71** 046101

[27] Guimerà R, Sales M and Amaral L A N, 2004 *Phys. Rev.* E **70** 025101

[28] Danon L, Duch J, Arenas A and Díaz-Guilera A, 2005 *COSIN Book* at press [cond-mat/0505245]

[29] Kuncheva L I and Hadjitodorov S T, *Systems*, 2004 *Man and Cybernetics, 2004 IEEE Int. Conf.* vol 2, p 1214

[30] Fred A L N and Jain A K, 2003 *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* (Piscataway, NJ: IEEE) pp II-128–33

[31] Duch J and Arenas A, 2005 *Phys. Rev.* E **72** 027104

[32] Girvan M and Newman M E J, 2002 *Proc. Nat. Acad. Sci.* **99** 7821

[33] Fortunato S, Latora V and Marchiori M, 2004 *Phys. Rev.* E **70** 056104

[34] Radicchi F, Castellano C, Cecconi F, Loreto V and Parisi D, 2004 *Proc. Nat. Acad. Sci.* **101** 2658

[35] Donetti L and Muñoz M A, 2004 *J. Stat. Mech.* P10012

[36] Donetti L and Muñoz M A, 2005 *Preprint* physics/0504059

[37] Bagrow J P and Bollt E M, 2004 *Preprint* cond-mat/0412482

[38] Capocci A, Servedio V, Colaiori F and Caldarelli G, 2004 *Lecture Notes Comput. Sci.* **3243** 181

[39] Wu F and Huberman B, 2004 *Eur. Phys. J.* B **38** 331

[40] Palla G, Derenyi I, Farkas I and Vicsek T, 2005 *Nature* **435** 814

[41] Reichardt J and Bornholdt S, 2004 *Phys. Rev. Lett.* **93** 218701

[42] Dijkstra E W, 1976 *A Discipline of Programming* (Englewood Cliffs, NJ: Prentice-Hall)

[43] Guimerà R and Amaral L A N, 2005 *J. Stat. Mech.* P02001

[44] Clauset A, Newman M E J and Moore C, 2004 *Phys. Rev.* E **70** 066111

[45] Guardiola X, Guimerà R, Arenas A, Díaz-Guilera A, Streib D and Amaral L A N, 2002 *Preprint* cond-mat/0206240