New Journal of Physics
The open–access journal for physics

Directed network modules

Gergely Palla[1], Illés J Farkas[1,2], Péter Pollner[1], Imre Derényi[2] and Tamás Vicsek[1,2]

[1] Statistical and Biological Physics Research Group of HAS, Pázmány Péter Sétány 1A, Budapest, H-1117 Hungary
[2] Department of Biological Physics, Eötvös University, Pázmány Péter Sétány 1A, Budapest, H-1117 Hungary
E-mail: pallag@angel.elte.hu

New Journal of Physics **9** (2007) 186
Received 15 February 2007
Published 28 June 2007
Online at http://www.njp.org/
doi:10.1088/1367-2630/9/6/186

Abstract. A search technique locating network modules, i.e. internally densely connected groups of nodes in *directed networks* is introduced by extending the clique percolation method originally proposed for undirected networks. After giving a suitable definition for directed modules we investigate their percolation transition in the Erdős–Rényi graph both analytically and numerically. We also analyse four real-world directed networks, including Google's own web-pages, an email network, a word association graph and the transcriptional regulatory network of the yeast *Saccharomyces cerevisiae*. The obtained directed modules are validated by additional information available for the nodes. We find that directed modules of real-world graphs inherently overlap and the investigated networks can be classified into two major groups in terms of the overlaps between the modules. Accordingly, in the word-association network and Google's web-pages, overlaps are likely to contain in-hubs, whereas the modules in the email and transcriptional regulatory network tend to overlap via out-hubs.

PII: S1367-2630(07)44249-5

Contents

1. Introduction

A widespread approach to the analysis of complex natural, social and technological phenomena is to assemble the participating molecules, individuals or electronic devices and their interactions into a network (nodes and links) and to infer functional characteristics of the entire system from this static web of connections [1, 2]. This approach is rooted in, among other things, statistical physics, where often the thermodynamic limit ($N \to \infty$, where N is the number of nodes) is considered, and the overall (large-scale) structure of connections is studied rather than the details at the level of nodes and links. Accordingly, over the past few years, several broadly studied *large-scale* properties of real-world webs have been uncovered, e.g. a low average distance combined with a high average clustering coefficient [3], the broad (scale-free) distribution of node degree (number of connections of a node) [4]–[7] and various signatures of hierarchical/modular organization [8, 9]. In addition, detailed analyses of the *small-scale* behaviour of the same complex webs have revealed overrepresented local structures: graph motifs [10, 11], i.e. small groups of nodes (typically of size 3–5) with specifically arranged connections among them. The identified small- and large-scale properties are both closely related to the dynamical behaviour of the corresponding complex system. Nodes with many connections (hubs) often have a central role in traffic [12], while motifs act as building blocks performing distinct basic information processing tasks [13].

The *intermediate-scale* substructures in networks (units larger than motifs), made up of vertices more densely connected to each other than to the rest of the network, are often referred

to as communities, modules, clusters or cohesive groups [14]–[21] with no widely accepted, unique definition. In the various types of networks these groups can represent, e.g. communities of people [14, 22, 23], functional units in biology [8, 24] and sets of tightly coupled stocks or industrial sectors in economy [25]. A reliable method to pinpoint network modules has many potential industrial applications, e.g. it can help service providers (phone, banking, web, etc) identify meaningful groups of customers (users), or support biomedical researchers in their search for individual target molecules and novel protein complex targets [26, 27]. In addition, modules and also some small subgraphs, are appropriate for 'coarse-graining' complex networks: each module/subgraph can be represented as a node and two such nodes can be linked, if the corresponding modules/subgraphs are connected (or overlap) [19, 28, 29].

The key requirements towards network module search techniques [19, 30, 31] are that they should be local, based on link density and error-tolerant (the removal or insertion of a link may alter only nearby modules). Furthermore, as dense groups in real-world graphs often overlap with each other, the module finding methods should allow overlaps between the groups. For example, in a social web each person belongs to several groups (family, colleagues and friends), in a protein interaction network each protein participates in multiple complexes [32] and a large portion of web-pages are classified under multiple categories [33]. Prohibiting overlaps during module identification strongly increases the percentage of false negative co-classified pairs. As an example, in a social web a group of colleagues might end up in different modules, each corresponding to their families and, in this case, the network module corresponding to their work unit is bound to become lost.

A recent link-density based approach to module finding, fulfilling the above requirements, is provided by the clique percolation method (CPM) [19, 34]. In this approach, the definition of the modules is based on k-cliques (complete subgraphs of size k in which each node is connected to every other node). A k-clique is a sub-graph with maximal possible link density, therefore it is a good starting point for defining modules. However, a method accepting only complete sub-graphs as modules would be too restrictive. Therefore, k-cliques are 'loosened up' in the following way. Two k-cliques are said to be adjacent if they share $k-1$ nodes (or in other words, if they differ only in a single node), and a module is defined as the union of k-cliques that can be reached from each other through a series of adjacent k-cliques. Such modules can be best visualized with the help of a k-clique template (an object isomorphic to a complete graph of k vertices). Such a template can be placed onto any k-clique in the graph, and rolled to an adjacent k-clique by relocating one of its vertices and keeping its other $k-1$ vertices fixed. Thus, the k-clique modules (k-clique communities) of a graph are all those sub-graphs that can be fully explored by rolling a k-clique template in them, but cannot be left by this template, as illustrated in figure 1. The algorithm used for the implementation of this technique is very efficient for most real networks, and provides the full list of overlapping modules in a short amount of time [19, 35].

A common shortcoming of current module finding methods is that they ignore the possible *directionality* of the links during the analysis of a network. The direction of a single link in most real network signals either the direction of some kind of flow (e.g. the flow of information, energy), or the asymmetry of the relation between the nodes (e.g. a superior–inferior relation). Consequently, nodes possessing mostly incoming links are expected to play a very different role in the network (or within the modules they belong to) from those possessing mostly outgoing links or from those having a similar amount of both kinds of links. Therefore, as a first attempt to take into consideration the directionality of links, we propose a simple measure for the nodes

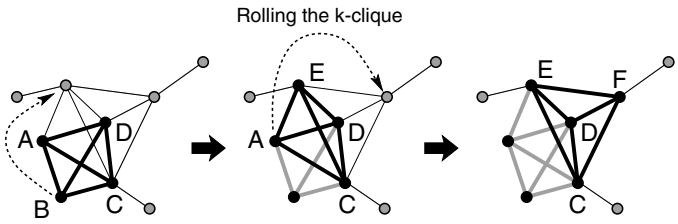

Figure 1. Illustration of the CPM [19, 34] with k-clique template rolling in a small undirected graph for $k = 4$. Initially the template is placed on A-B-C-D (left panel) and it is 'rolled' on to the subgraph A-C-D-E (middle panel). The position of the k-clique template is marked with thick black lines and black nodes, whereas the already visited links are represented by thick grey lines and dark grey nodes. Observe that in each step only one of the nodes is moved and the two 4-cliques (before and after rolling) share $k - 1 = 3$ nodes. At the final step (right panel) the template reaches the subgraph C-D-E-F, and the set of nodes visited during the process (A-B-C-D-E-F) are considered as a module identified by the CPM at $k = 4$.

within the modules to characterize their roles in terms of the numbers of their incoming and outgoing links.

At the same time the consideration of directionality in modules raises the question of whether a module searching algorithm that inherently takes into account the directionality of links is more suitable for directed networks than the original undirected algorithms. Along this idea, we define the notion of directed k-cliques (in which the configuration of the directed links has to meet certain criteria), and propose a restricted version of CPM (denoted as CPMd), in which only directed k-cliques can be used for the identification of modules. We apply this method to several networks: first, we examine the percolation transition of the directed k-cliques in the Erdős–Rényi (ER) random graph [36], then move on to study the directed modular structure of four real-world networks, including a word-association network, Google's web-pages, an email network, and the transcriptional regulatory graph of yeast. The identified directed modules are verified with the help of additional information (protein functional annotations, web-page names, and word usage frequencies) about the nodes.

2. Definitions

In undirected graphs a pair of nodes is either connected or not, whereas in a directed graph the same pair, (A,B), can be connected in three ways: either by a 'single link' as (i) A→B and (ii) A←B or by a 'double link' as (iii) A⇌B. Multiple links (i.e. more than one link between A and B in the same direction) and self-links (such as A→A) are not allowed. In the following we first define a simple measure for comparing nodes within a module based on the directionality of their links, then introduce the concept of directed k-cliques, the fundamental objects of our directed module finding approach.

2.1. Comparing the nodes according to their relative out-degree

A natural and simple approach to relate nodes in a module to each other is to compare the number of their incoming and outgoing links connected to other members in the module. For example, a node having only out-neighbours amongst the members of the module can be viewed a 'source' or a 'top-node', whereas a node with only incoming links from these members is a 'drain' or a 'bottom-node'. Most nodes, however, fall somewhere between these two extremes. To quantify this property, we introduce the *relative in-degree* and *relative out-degree* of node i in module α as

$$D^{\alpha}_{i,\mathrm{in}} \equiv \frac{d^{\alpha}_{i,\mathrm{in}}}{d^{\alpha}_{i,\mathrm{in}} + d^{\alpha}_{i,\mathrm{out}}}, \tag{1a}$$

$$D^{\alpha}_{i,\mathrm{out}} \equiv \frac{d^{\alpha}_{i,\mathrm{out}}}{d^{\alpha}_{i,\mathrm{in}} + d^{\alpha}_{i,\mathrm{out}}}, \tag{1b}$$

where $d^{\alpha}_{i,\mathrm{in}}$ and $d^{\alpha}_{i,\mathrm{out}}$ denote the number of in-neighbours and out-neighbours amongst the other nodes in the module, respectively. Obviously the values of both $D^{\alpha}_{i,\mathrm{out}}$ and $D^{\alpha}_{i,\mathrm{in}}$ are in the range between 0 and 1, and the relation $D^{\alpha}_{i,\mathrm{in}} + D^{\alpha}_{i,\mathrm{out}} = 1$ holds. For weighted networks, (1*a*, 1*b*) can be replaced by the *relative in-strength* and *relative out-strength* defined as

$$W^{\alpha}_{i,\mathrm{in}} \equiv \frac{w^{\alpha}_{i,\mathrm{in}}}{w^{\alpha}_{i,\mathrm{in}} + w^{\alpha}_{i,\mathrm{out}}}, \tag{2a}$$

$$W^{\alpha}_{i,\mathrm{out}} \equiv \frac{w^{\alpha}_{i,\mathrm{out}}}{w^{\alpha}_{i,\mathrm{in}} + w^{\alpha}_{i,\mathrm{out}}}, \tag{2b}$$

where $w^{\alpha}_{i,\mathrm{out}}$ and $w^{\alpha}_{i,\mathrm{out}}$ denote the aggregated weight of out-going and incoming connections with other members in the module α.

2.2. Directed k-cliques and the directed CPMd

In a complete sub-graph of size k the $k(k-1)/2$ links can be directed in $3^{k(k-1)/2}$ ways. Since the undirected CPM treats these alternatives as identical, introducing link directions allows a large variety of possible rules for defining directed modules. A natural concept, however, is to aim for 'directed modules' preserving some kind of directedness as a whole, rather than just being a collection of nodes connected by directed edges.

Therefore, we replace the k-cliques (the fundamental objects of the CPM) by *directed k-cliques*, which are defined as complete sub-graphs of size k in which an ordering can be made such that between any pair of nodes there is a directed link pointing from the node with the higher order towards the lower one. Since the presence of double links usually leads to multiple possibilities to order the nodes in a way fulfilling the above requirement, for simplicity we first concentrate on directed k-cliques with no double links. In this case, the higher the order of a node, the more out-neighbours it has in the k-clique (see illustration in figure 2*a*). Thus, the *restricted out-degree* of a node in the k-clique (the number of its out-neighbours in the k-clique,

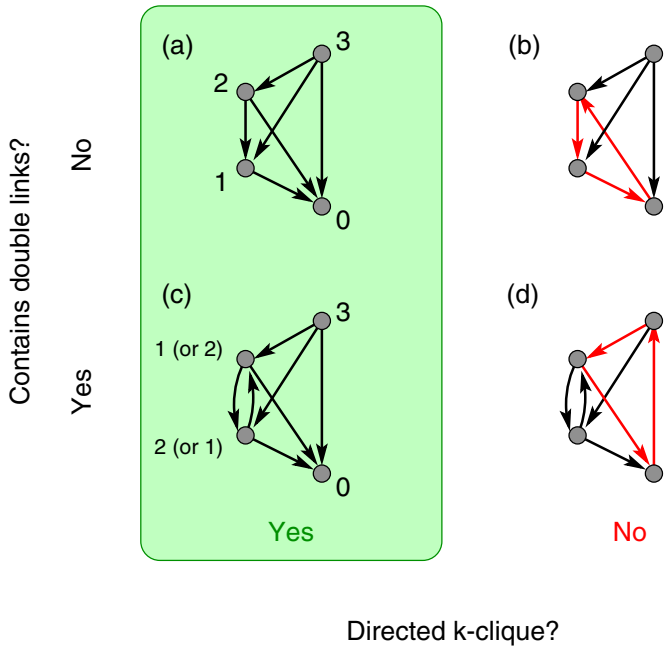

Figure 2. Groups of nodes forming a directed k-clique (a, c) and groups (b, d) that do not. (a) A directed k-clique without double links. The index of each node corresponds to its order (which is equivalent to number of its out-links) within the directed k-clique. (b) A complete sub-graph without double links, but not accepted as a directed k-clique, because it contains a directed loop. (c) A directed k-clique with a double link. Note that the order of the nodes depends on which link is deleted from the double link. (d) Double link in a complete sub-graph that is not a directed k-clique. It is not possible to remove a link from the double link in a way that all directed loops disappear.

ranging from 0 to $k-1$) can be assigned as its order. From this, it can be seen easily (for details see appendix A) that the condition for a k-clique with no double links to qualify as a directed k-clique is equivalent to the following three conditions.

1. Any directed link in the k-clique points from a node with a higher order (larger restricted out-degree) to a node with a lower order.
2. The k-clique contains no directed loops (where a 'directed loop' is a closed directed path).
3. The restricted out-degree of each node in the k-clique is different.

The overall directionality of such an object naturally follows the ordering of the nodes: the node with highest order is the one which has only out-neighbours, and can be viewed as the 'source' or 'top'-node of the k-clique, whereas the node with lowest order has only incoming links from the others, and corresponds to a 'drain' or 'bottom' node.

None of the above three conditions holds in the presence of double links: directed loops appear in the k-clique, the restricted out-degree of at least two nodes in the k-clique becomes the same (see appendix A), and we can find directed links pointing in the direction of increasing order. However, based on the ordering of the nodes, it is always possible to eliminate the double

links (by removing all links that point towards higher order) from a directed k-clique in such a way that the remaining single links fulfil all three conditions. See figure 2(c) as an example.

The k-clique adjacency is defined similarly to the undirected case: two directed k-cliques are adjacent if they share $k-1$ nodes. The directed k-clique modules (the CPMd modules) arise as the union of directed k-cliques that can be reached from each other through a series of k-clique adjacency. The k-clique template rolling picture can be applied to illustrate the CPMd modules in the same fashion as in the undirected case. The searching algorithm locating the CPMd modules is described in appendix B.

We note that the above definition of a directed k-clique is just one possibility among many others. Natural choices that also impose some kind of directionality on the k-clique include e.g. the requirement that at least one of the nodes should have out-links (or in-links) towards (from) all the other $k-1$ nodes, or the requirement that the nodes could be divided into two non-empty sets such that each node in the first set has an out-link towards each node in the second set (resembling directed hyper-edges). Our particular choice was motivated, on the one hand, by the fact that it is more restrictive than the others (providing a more specific tool to investigate the effects of directionality) and, on the other hand, by our finding that for most real world networks even such a restricted definition results in directed modules that are notably similar to the undirected ones (see section 4.5).

3. Percolation transition in the directed ER graph

The concept of (undirected) random graphs was introduced by Erdős and Rényi [36] in the 1950s in a simple model consisting of N nodes and connecting every pair of nodes independently with the same probability p. Even though real networks differ from this simple model in many aspects, the ER graph remains still of great interest, since such a graph can serve both as a test bed for checking all sorts of new ideas concerning complex networks in general, and as a prototype of random graphs to which all other random graphs can be compared.

Perhaps the most conspicuous early result on the ER graphs was related to the percolation transition taking place at $p = 1/N$. The appearance of a *giant component* in a network, which is also referred to as the *percolating component*, results in a dramatic change in the overall topological features of the graph and has been in the centre of interest for other networks as well. In a more general framework, one can also address the question of k-clique percolation in the ER graph. Simple theoretical arguments as well as numerical simulations [34] show that the critical linking probability of k-clique percolation is $p_c^{\text{undir}} = [(k-1)N]^{-1/(k-1)}$. In this section we carry out a similar analysis concerning the percolation transition of directed k-cliques in the directed ER graph.

3.1. Derivation of the critical point

The directed equivalent of the ER graph consists of N nodes providing $N(N-1)$ possible 'places' for the directed links, and these are filled independently with uniform probability p, producing on average $M \simeq N(N-1)p$ edges. (Note that in the original undirected ER graph there are only $N(N-1)/2$ possibilities to introduce an edge, therefore, at linking probability p, there are only $M \simeq N(N-1)p/2$ connections.) The critical linking probability p_c decreases with increasing

N, and converges to zero as $N \to \infty$. We restrict ourselves to the large N limit, and evaluate p_c to leading order only. Let us suppose that we approach the critical point from below: the directed k-cliques do not assemble yet into a giant module, we can find only small, isolated modules and the system is dispersed. In terms of our k-clique template rolling picture this means that when trying to explore the directed percolation clusters by rolling such a template on them, we must stop the rolling after a few steps as we run out of unexplored adjacent directed k-cliques.

One can estimate p_c from the condition that at the critical point the average number of yet unexplored directed k-cliques adjacent to the k-clique we have just reached becomes equal to one. (This makes it possible to roll our template on and on for a long time.) Since we are going to evaluate p_c to leading order only, we can neglect the possibility of rolling our k-clique template using double edges between the same nodes: when reaching a directed k-clique, the minimal number of further edges that must be present to enable the continuation of the template rolling is $k-1$. The probability of such a case is therefore proportional to p^{k-1}. Even though it is not forbidden in the first place to continue using double edges as well, each double edge in the new directed k-clique we are going to roll onto multiplies the probability by p. In other words, the probability of rolling further to a k-clique containing one double edge is smaller by a factor of p, the probability of rolling further to a k-clique containing two double edges is smaller by a factor of p^2, etc.

During the branching process exploring a directed k-clique percolation cluster, at the point when we are about to roll our template further on, we can choose the next node for relocation in $k-1$ different ways, which can then be relocated to approximately N places. If there were no restrictions for the directioning of the links inside a directed k-clique, then the $k-1$ new links connecting the new node to this $k-1$ shared nodes could be directed in 2^{k-1} ways. However, the new directed k-clique has to fulfil the three conditions detailed in section 2.2 as well, therefore the actual number of allowed configurations is much smaller. The rank of the new node in the new directed k-clique can be chosen in k ways: the $k-1$ nodes shared with the previous k-clique are already ordered, and we can 'insert' the new node to any place in this hierarchy. By fixing the order of the new node we fix the direction of the new links as well, therefore we can allow only k different configuration for the directionality of these links. By combining these factors together, the condition for reaching the critical point of the percolation transition can be written as

$$p_c{}^{k-1} N(k-1)k = 1, \tag{3}$$

from which we gain

$$p_c^{\text{theor}} = [Nk(k-1)]^{-1/(k-1)} = p_c^{\text{undir}}/k^{k-1}, \tag{4}$$

for the theoretical prediction of the critical edge probability. Note that in the limiting case of $k = 2$ (the directed edge percolation), the $p_c^{\text{theor}} = p_c^{\text{undir}}/2$ relation holds, which is consistent with the $2:1$ ratio for the number of links in the directed and undirected ER graph respectively.

3.2. Numerical simulations

There are two plausible choices to measure the size of the largest directed k-clique percolation cluster. The most natural one, which we denote by N^*, is the number of nodes belonging to this

cluster. We can also define an *order parameter* associated with this choice as the relative size of this cluster:

$$\Phi = N^*/N. \tag{5}$$

The other choice is the number $\mathcal{N}^*$ of directed k-cliques of the largest directed k-clique percolation cluster. The associated order parameter is again the relative size of this cluster:

$$\Psi = \mathcal{N}^*/\mathcal{N}, \tag{6}$$

where $\mathcal{N}$ denotes the total number of directed k-cliques in the graph. In figure 3(a) and (b), we display Φ and Ψ as functions of p/p_c^{theor}, where the directed k-clique size is $k = 4$, and the system size varies between $N = 50$ and $N = 1600$. The order parameter Φ converges to a step function for increasing system sizes, whereas Ψ converges to a limit function (which is 0 for $p/p_c(k) < 1$ and grows continuously to 1 above $p/p_c(k) = 1$). We have evaluated the transition point numerically as well, by computing the second moment of the distribution of $\mathcal{N}_i$ values, excluding the largest one, $\mathcal{N}_1 = \mathcal{N}^*$:

$$\chi = \sum_{i>1} (\mathcal{N}_i/\mathcal{N})^2. \tag{7}$$

Note that this quantity is analogous to the percolation susceptibility. Both below and above the transition point the $\mathcal{N}_i$ ($i > 1$) values follow an exponential distribution, and only at p_c do they have a power-law distribution. Thus, χ is maximal at the numerical transition point, p_c^{num}. In figure 3(c) we show χ calculated for the curves shown in figure 3(b), as the function of p/p_c^{theor}. In order to check the theoretical prediction for the critical point obtained in (4) we have carried out a finite-size scaling analysis of the numerical results. In figure 3(d) we show the ratio $p_c^{\text{num}}/p_c^{\text{theor}}$ as a function of $1/N$. Indeed, for large systems, the above ratio converges to one roughly as $1 + cN^{-1/2}$.

4. Results for real-world graphs

In this section we study the directed modular structure of four real-world networks ranging from a word association graph through Google's web-pages to email and transcription regulatory networks. When applied to real networks, the CPMd method has two parameters: the k-clique size k, and (if the network is weighted) a weight threshold w^* (links weaker than w^* are ignored). Changing the threshold is like changing the resolution (as in a microscope) with which the modular structure is investigated: by increasing w^* the modules start to shrink and fall apart. A very similar effect can be observed by changing the value of k as well: increasing k makes the modules smaller and more isolated from each other, but at the same time, each module becomes more cohesive. When we are interested in the modular structure around a particular node, it is advisable to scan through some ranges of k and w^*, and monitor how the obtained modules change. Meanwhile, when analysing the modular structure of the entire network, the criterion used to fix these parameters is based on finding a modular structure as highly structured as possible [19]. This can be achieved by tuning the parameters just below the critical point of the percolation transition. In this way we ensure that we find as many modules as possible, without the negative effect of having a giant module that would smear out the details of the modular structure by merging (and making invisible) many smaller modules. The technical details of the extraction of the directed k-clique modules are described in appendix B.

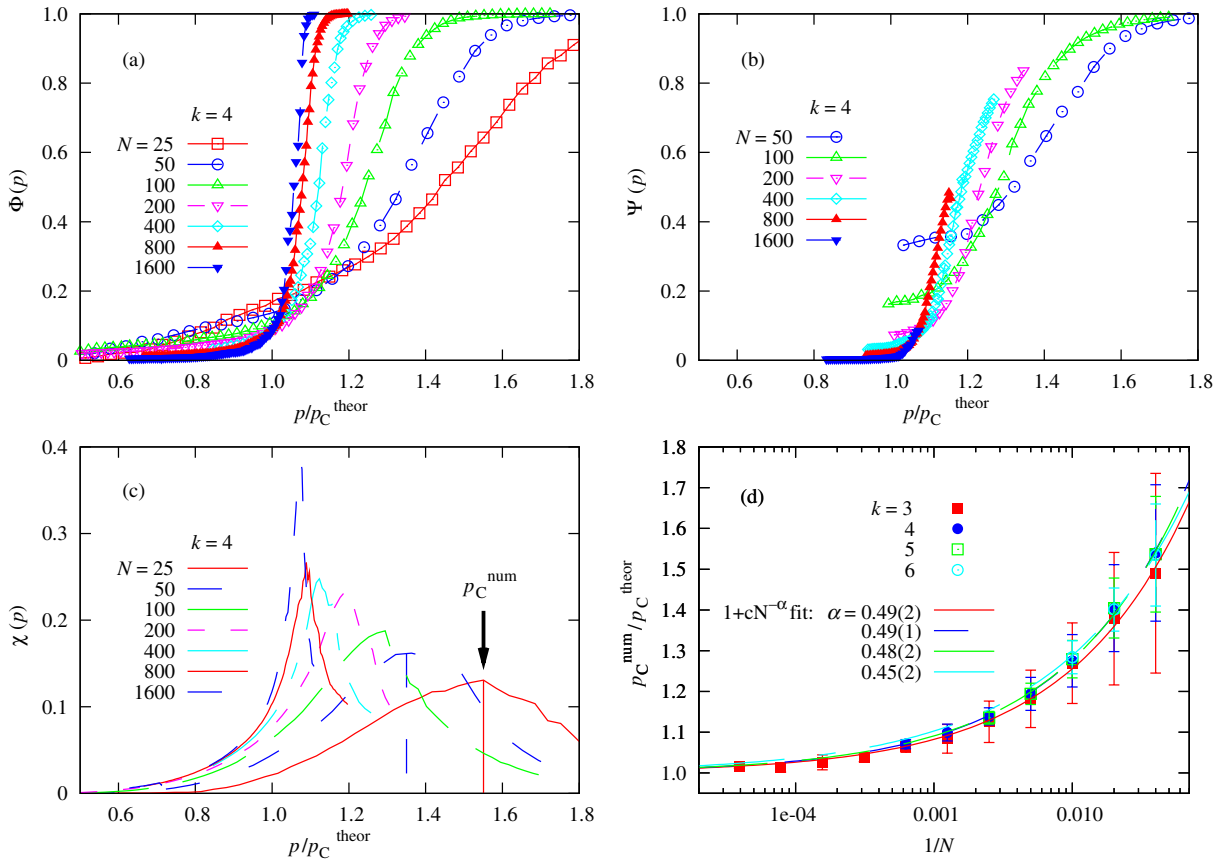

Figure 3. Numerical results for directed k-clique percolation in ER-graphs. In each sub-figure, points show an average over 4–100 simulations depending on system size. (a) The order parameter Φ (the number of nodes in the largest percolation cluster divided by N) as a function of p/p_c^{theor}, where p_c^{theor} was obtained from equation (4). (b) The order parameter Ψ (the number of directed k-cliques in the largest percolation cluster divided by the total number of directed k-cliques) as a function of p/p_c^{theor}. (c) The numerically determined value for the critical linking probability, p_c^{num}, defined as the average location of the maximum of $\chi(p)$, playing the role of the normalized percolation susceptibility (see equation (7)). (d) Verification of the theoretical prediction for the critical point. The $p_c^{\text{num}}/p_c^{\text{theor}}$ ratio converges to one for large N.

4.1. Word association graph

We examined the directed network obtained from the South Florida Free Association norms list (containing 10 617 nodes and 63 788 links), where the weight of a directed link from one word to another indicates the frequency that the people in the survey associated the end point of the link with its start point [37]. For illustration in figure 4, we show the (colour coded) modules of the word 'GOLD' obtained at $k = 4$ and $w^* = 0.023$, with the overlaps emphasized in red. According to its different meanings, this word participates in four, strongly internally connected modules. Beside the node labels we display the relative out-strength of the nodes in

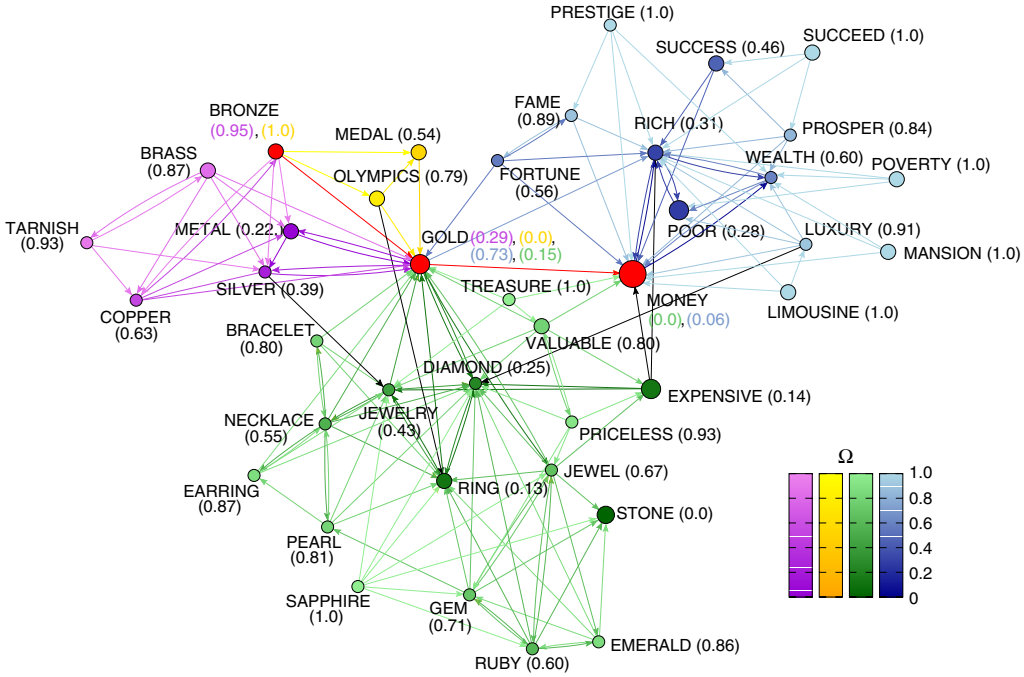

Figure 4. The directed modules of the word 'GOLD' at $k = 4$, $w^* = 0.023$ in the word association network [37]. The modules are colour coded and the overlaps between the modules are displayed in red. The size of each node is proportional to the number of modules it participates in (some of them are not shown in this figure). Beside the name of the nodes we display their $W^{\alpha}_{i,\mathrm{out}} = w^{\alpha}_{i,\mathrm{out}}/(w^{\alpha}_{i,\mathrm{in}} + w^{\alpha}_{i,\mathrm{out}})$ values as well. Nodes with high W (e.g. 'SAPPHIRE') usually correspond to special, rarely used words, whereas nodes with low relative out-degree (e.g. 'MONEY') are very common.

the modules using (2*b*). Apparently, nodes with a special/particular meaning (e.g. 'SAPPHIRE') tend to get high relative out-strength whereas commonly used words with general meaning (e.g. 'MONEY') have low relative out-strength. Thus, it seems that the overall directionality of the modules is from special words towards more general words. To make this observation more quantitative, we measured the number of hits obtained for the different words appearing in the network using the Google search engine. In figure 5 we show the scatter plot of the number of hits as a function of the relative out-strength of the members of the modules obtained at the optimal $k = 4$, $w^* = 0.016$ parameters. The decreasing tendency of the number of hits with increasing $W^{\alpha}_{i,\mathrm{out}}$ signals that words with higher relative out-strength (i.e. having mainly out-neighbours) are usually less frequently used than words with lower relative out-strength (i.e. having mainly in-neighbours).

4.2. Google's web-pages

In addition to being a prominent means of information retrieval, Google provides its own documents as well: usage notes, feature and product descriptions, etc. The map of hyper-links among Google's own web-pages offers a unique insight into how one of the major search portals

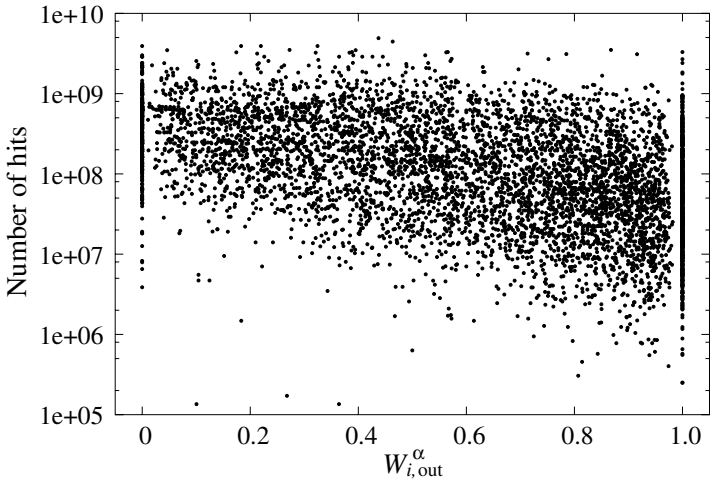

Figure 5. The number of hits obtained from Google for module members as a function of their relative out-strength in the word association network. The number of hits decreases with increasing $W^{\alpha}_{i,\mathrm{out}}$, therefore, frequently used words are likely to obtain a low relative out-strength.

arranges online content and thereby helps and guides our browsing. Excluding dynamic content and catalogues, we downloaded with our robot[3] 15 763 web-pages (nodes) and 171 206 directed links among them. For the current analysis, we excluded international pages and nodes farther than 3 steps from the start node, http://www.google.com, and obtained a graph with 946 nodes and 1817 links. Figure 6 shows three of the many overlapping directed modules identified by the CPMd in this network at $k = 6$. Apparently each of the identified overlapping modules in figure 6 is a group of internally densely connected nodes organised around a well-defined topic (jobs, accounts and enterprise solutions).

An interesting feature of Google's directed modules is that they *share their in-hubs, but not their out-hubs.* (By 'in-hub' we mean nodes with outstanding in-degree, whereas 'out-hub' stands for nodes with outstanding out-degree.) This structure enhances browsing efficiency. Having visited a particular, 'outlying' page of a module, one can quickly return to a node in the core of the same module. Then, due to the strong overlaps among the cores, one can quickly jump over to a new topic, i.e. the web-pages of another module. In summary, our ability to browse efficiently and hierarchically Google's web-pages is enhanced by the facts that modules overlap via their in-hubs.

4.3. Email network

A very common type of directed social networks is the one defined by messages and information flow (directed links) among individuals (nodes). To 'measure' such a social network, Ebel and Bornholdt [38] processed the directed network defined by the emails of students at the University of Kiel during a period of 112 days. We analysed both the entire data set and its subset containing only emails between internal addresses (students). The full network contains 57 158 nodes and

[3] The robot together with the directed graph of Google's web-pages can be downloaded from http://www.CFinder.org.

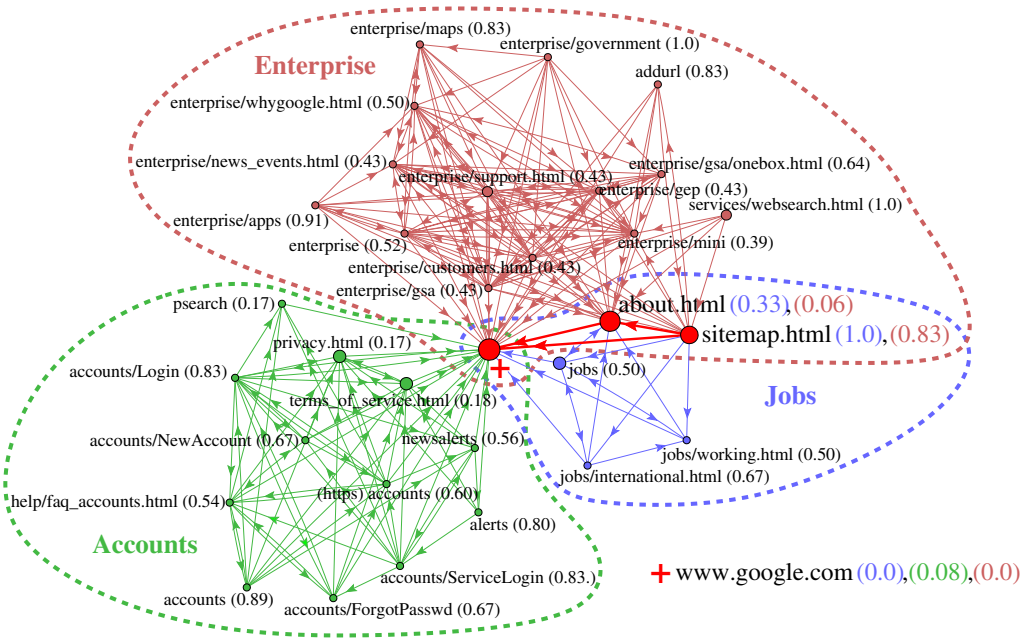

Figure 6. Three of the overlapping directed modules identified by CPMd in the directed net of Google's static pages at $k = 6$. These modules overlap with several further ones not shown in the figure; the size of each node is proportional to the number of its modules. The nodes and links of the three modules are coloured brown, green and blue, while their overlaps, i.e. nodes contained by more than one of these three modules, are red. The node marked with a + sign at centre is the starting page, http://www.google.com, and the names of the other nodes are their URLs without this prefix. The D values of the module members are marked beside the node labels. Observe that each module contains a number of nodes with many incoming links (a 'core'), some of which are in the overlaps. See text for further details and figure 9 for a detailed analysis of hubs and overlaps.

103 701 links, while the 1267 internal addresses (nodes) are connected by 1659 links. Figure 7 shows the directed modules in these two networks. Observe that even among the relatively small number of internal emails modules, overlaps do appear, e.g. node 5886 at the centre. In the full e-mail data set, external addresses have both the highest degrees (number of connections) and the largest numbers of modules they participate in. In contrast to e.g. the Google's web-pages, nodes with the largest out-degrees participate in a high number of modules.

4.4. The transcriptional regulatory network in yeast

In a cell the transcription of a gene is influenced (regulated) by one or more proteins called transcription factors (TES). This regulatory relationship is most often represented as a directed link pointing from the regulating protein (source node) to the protein of the regulated gene (target node). Recent experimental and computational techniques [39, 40] have enabled the genome-wide mapping of transcription regulatory relationships in the yeast, *S. cerevisiae.*

In figure 8, we display the obtained directed modules for $k = 3$. As an example, for some of the modules the most significant common functions of their participating proteins have been

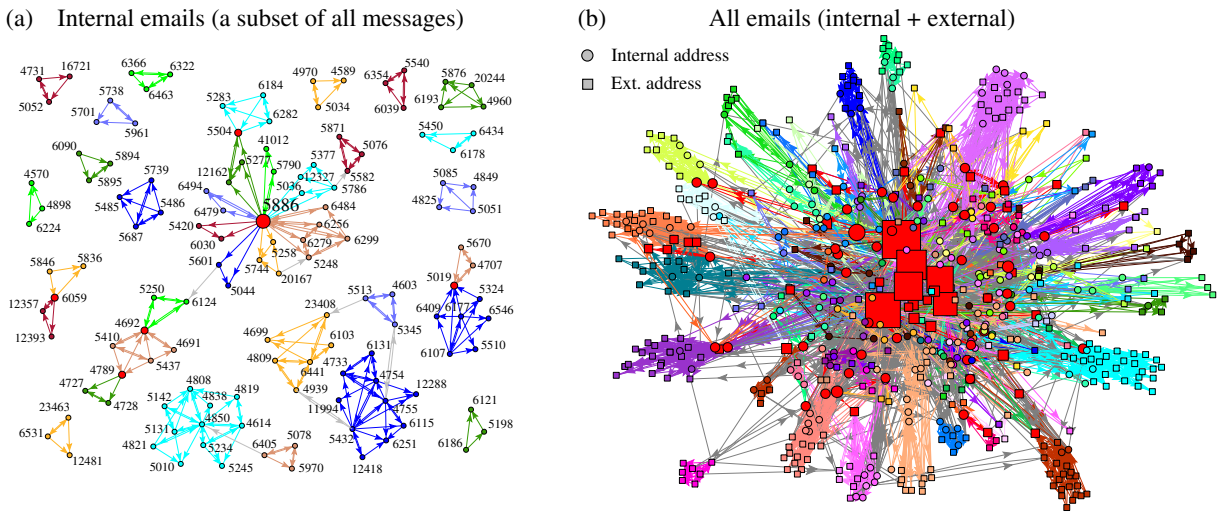

Figure 7. All directed modules in a network of student emails at the University of Kiel during a period of 112 days (data from [38]). On the left (a) only the graph of internal emails (between students of the university) is analysed, while on the right (b) internal and external messages are both included. Circles and boxes show internal and external email addresses, respectively, and the size of a node is proportional to the number of its modules. The largest nodes, i.e. those with the highest membership number, have significantly more outgoing than incoming links, meaning that in this email network modules share their out-hubs. See also figure 9. The optimal k-clique size parameter values are $k = 3$ (a) and $k = 4$ (b).

identified from the Gene Ontology protein function annotation database [41] with the search tool GO TermFinder [42]. The list of regulatory interactions was obtained from [40]. Most protein modules in figure 8 are arranged around a small number of large out-hubs, the major TFs, each of which regulates a large portion of all target genes in the module. Overlaps between the modules occur either through the TFs, e.g. via the nodes Met4 and Gcn4 in the bottom left part of the figure, or via large groups of regulated (target) genes, see, e.g. the red nodes at the 'interface' between the yellow and brown modules in the upper part of the figure. Hence, from the point of view of directed modules, the transcription regulation network is organized in a similar way to the email network, and an opposite way to Google's web-pages (and the word association network).

4.5. Comparison between CPMd and CPM

For each studied network, by ignoring the directionality of the links, we located the CPM communities as well. In the case of the word association network, where links are weighted as well, the weight of the undirected counterpart of a double link was defined as the sum of the corresponding two weights. Due to this difference in the weights as well as in the definition of modules, the optimal weight threshold was slightly different in the CPM approach.

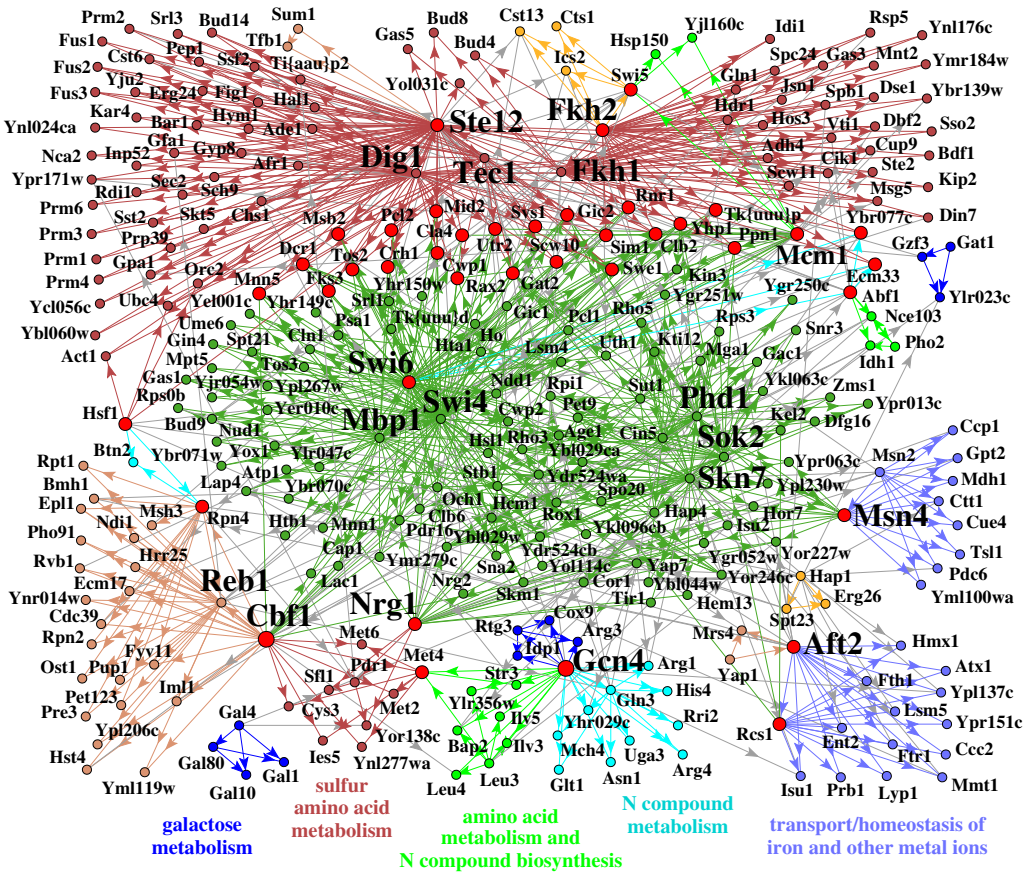

Figure 8. The directed modules of the web of transcription regulatory interactions in baker's yeast ($k = 3$). Each node shows one gene (and its protein) and a directed link stands for a transcription regulatory interaction between a protein and the target gene. Modules (communities) are coloured and overlaps are red. The overlapping nodes are mostly out-hubs. Group functions have been identified by GO TermFinder [42].

Surprisingly, in spite of the restrictions of the CPMd compared to CPM (and in the case of the word association network, the difference in link weights), about 70% of the modules were the same in the two approaches for the word association network and Google's web-pages, whereas this ratio turned out to be even higher (around 90%) for the email network and the transcription regulatory graph. Furthermore, for the rest of the directed modules one could find a relatively similar undirected module in most cases. This shows that the original CPM approach to the identification of modules is quite robust, our restrictions introduced in the CPMd leave the majority of the undirected modules intact.

4.6. Classification of real-world networks: modules are connected by in-hubs or out-hubs

An important aspect of network motifs (overrepresented small sub-graphs with a given structure) is that complex networks can be classified based on their motif significance profile (a pattern of motif usage) [11]. In a somewhat similar approach, here we classify the four investigated real-world webs into two major groups based on the overlaps of their directed modules.

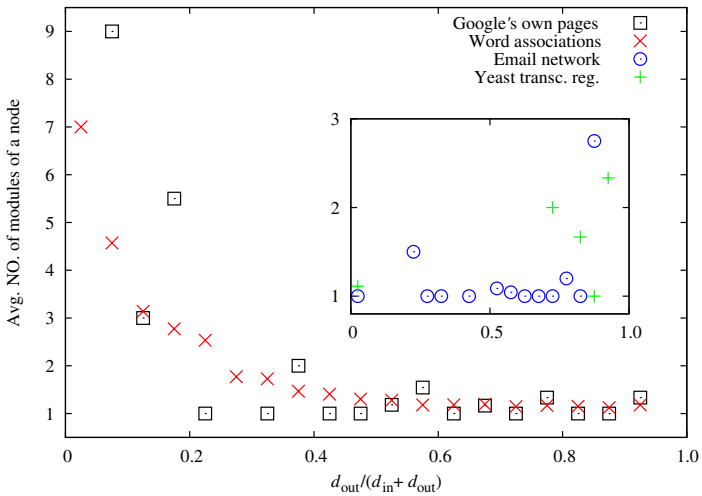

Figure 9. The average membership of a node versus its $d_{out}/(d_{in} + d_{out})$ ratio. This function is a growing (decreasing) one, if the modules are more likely to overlap via in-hubs (out-hubs).

Interestingly, the way that the out-hubs and in-hubs of the network are arranged within its directed modules is different among the various types of networks. To directly compare the studied networks from this aspect, in figure 9 we show the average number of modules of the nodes as a function of their relative out-degree $D_{i,\text{out}} \equiv d_{i,\text{out}}/(d_{i,\text{in}} + d_{i,\text{out}})$ ratio. Apparently, the modules in the word association network and Google's web-pages are connected by in-hubs: nodes contained by a large number of modules have a small $D_{i,\text{out}}$. In contrast, in the email network and the transcription regulatory graph of yeast the overlaps are more likely to contain out-hubs than in-hubs.

The plausible reason for the observed difference between the investigated networks is that overlaps contain hubs with increased likelihood in the first place, and the two kinds of hubs occur in the networks with different probabilities. In the word-association network and Google's web-pages in-hubs are more frequent: the number of words we associate to a cue word and the number of hyper-links that appear on a web-page is more or less constant, however a word with a general meaning or an important (general) web-page can appear as the target for many links. In contrast, we are more likely to find out-hubs than in-hubs in the email network and the transcription regulatory graph. The time spent on sending an email does not depend on the number of recipients, whereas reading a large number of incoming emails can take a lot of time, therefore being an in-hub in the email network is disadvantageous and in-hubs are rare. Similarly, in case of the transcription regulatory graph the number of TFs that can regulate a given protein is more or less constant, whereas a single TF can regulate many other proteins in parallel, therefore, out-hubs are much more frequent than in-hubs.

5. Summary and conclusions

We examined the directionality of network modules. To compare and order the nodes in a module, we introduced the relative out-degree, measuring the relative weight of the out-links of

a member to other nodes in the module. We developed a specific module finding algorithm for directed networks as well, based on the k-clique percolation approach. Even though the CPM can be extended to any kind of directed k-cliques (containing an arbitrary set of directed links), here we concentrated on the most plausible choice which allows a straightforward theoretical and numerical analysis. Following a simple branching procedure, we have derived the critical point of the directed k-clique percolation in the ER graph in the large N limit. The theoretical prediction was justified by numerical simulations. We have also studied the directed modular structure of real-world networks including a word association graph, Google's web-pages, an email network and the transcription regulatory network of yeast. The obtained modules were validated by additional information (annotations) for the members. The nodes contained in the overlaps between the modules enabled us to classify the examined networks in two major groups: the modules in the word association graph and Google's web-pages are likely to be connected by in-hubs, whereas the overlaps in the email network and the transcription regulatory network are more likely to contain out-hubs.

Acknowledgments

We thank the partial support of the Hungarian National Science Fund (OTKA T034995, K068669, PD048422) and the National Research and Technological Office (NKTH, CellCom RET).

Appendix A

In this appendix we show that for k-cliques with no double links, the following three statements are equivalent.

(i) Any directed link in the k-clique points from a node with a higher order (larger restricted out-degree) to a node with a lower order.
(ii) The k-clique contains no directed loops.
(iii) The restricted out-degree of each node in the k-clique is different.

(The restricted out-degree of a node is equal to the number of its out-neighbours in the k-clique).

(ii)→(iii): *If loops are absent, then all the members have different restricted out-degrees.*
If there are no loops, then there must be a node in the k-clique having all in-neighbours amongst the other members, since otherwise we could hop from node to node following a directed link inside the k-clique forever, (which would mean that it does contain at least one loop). If we reversed the direction of all links inside the k-clique we would not induce any loops, and therefore, this 'reversed' configuration would have a member with only incoming links from the others as well. From this it follows that there must be also a node in the k-clique with only out links towards the other nodes. By removing this node we obtain a $(k-1)$-clique in which directed loops are absent. Similarly to the previous case, this $(k-1)$-clique must have a node with only out-neighbours amongst the other members of the $(k-1)$-clique. By removing this node as well, we arrive at a $(k-2)$-clique containing no loops. And so on, by subsequently removing the node with only out-neighbours at each step we iterate over all nodes, and obviously the restricted out-degree of

the removed node is decreased by one at each step, hence all nodes have different numbers of out-links inside the k-clique.

(ii)→(i): *If loops are absent, then the links point from higher restricted out-degrees values towards lower ones.*
The above process showing (ii)→(iii) also reveals that the links inside a k-clique with no loops are always pointing from a node with a higher restricted out-degree towards a node with less out-links inside the k-clique.

(iii)→(ii): *If all nodes have different numbers of out-neighbours inside the k-clique, then directed loops are absent.*
The possible number of out-neighbours a node can have inside a k-clique falls in a range between 0 and $k-1$, therefore, if all nodes have different numbers of out-neighbours, then all of these possible values must actually appear in the k-clique. Since double links are absent, the node with $k-1$ out-links cannot have any incoming links from the other members, therefore, it is surely not part of any directed loops inside the k-clique. The node with $k-2$ out links has only a single incoming link, starting at the node with only out-links. Therefore, this node cannot be part of any directed loops either. Similarly, the node with $k-3$ out links has two incoming links, both starting at nodes that have been already shown to be excluded from any directed loops (the nodes with $k-1$ and $k-2$ out-neighbours, respectively). Thus, the node with $k-3$ out-neighbours amongst the other members 'inherits' this property (to be excluded from directed loops inside the k-clique) as well. And so on, by subsequently scanning the nodes in decreasing order of their restricted out-degrees, at each step all the incoming links to the node under investigation come from previously examined members that were shown to be excluded from loops, therefore, the investigated node cannot be part of any loops either.

(i)→(iii): *If each directed link points from a node with a higher restricted out-degree to a node with a lower one, then the restricted out-degree of each node in the k-clique is different.*

This statement is almost trivial, since if any pair of nodes had the same restricted out-degree, then the link connecting them would point in the direction of constant restricted out-degree.

For k-cliques with double links none of the three statement can hold. The presence of loops is trivial: a double link is already equivalent of a closed directed path. Furthermore, both constituents of a double link cannot point in the direction of decreasing order simultaneously. Therefore, we only have to prove that (iii) cannot be true either, i.e. for k-cliques with double links their members cannot have all different numbers of out-neighbours amongst the other nodes in the k-clique. The total number of links, m, inside a k-clique can be written as

$$m = \sum_{q=0}^{k-1} q n_q, \tag{A.1}$$

where q runs over the possible number of out-neighbours, and n_q is the number of members with the given restricted out-degree. When all the members have different number of out-links, $n_q = 1$ for all possible q values, and thus, $m = k(k-1)/2$, which is exactly the number of links in a k-clique with no double links. However, in presence of double links $m > k(k-1)/2$, therefore, at least one of the n_q values in (A.1) must be larger than one, meaning that there are nodes in the k-clique with equal restricted out-degrees.

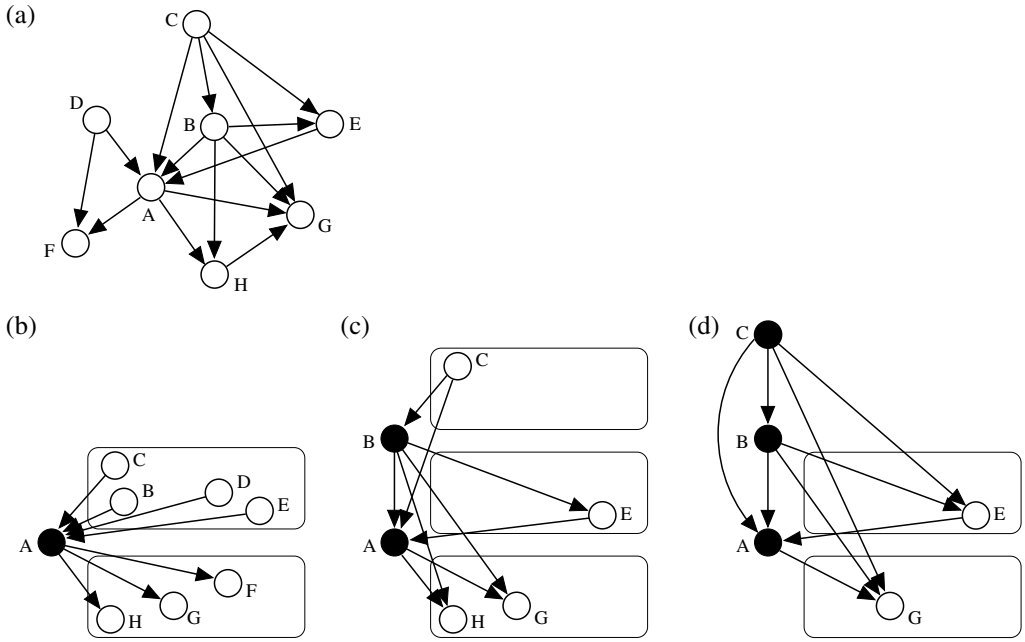

Figure B.1. Illustration of the directed clique search. (a) The neighbourhood of node A in a hypothetical directed network. (b) The initial state of the directed clique extraction algorithm: the in-neighbours of A are above A, whereas its out-neighbours are below it. (c) Node B is picked from the in-neighbours and is placed above A. Nodes D and F are not neighbours of B, therefore they are removed from the containers. Furthermore, a new container is introduced holding node E, which is in-between B and A in the hierarchy. (d) Node C is picked from the top container and is placed above B, node H is removed from the bottom container as it is not linked to C.

Appendix B

In this section we briefly describe our algorithm for extracting the CPMd modules in networks. Since any subgraph of a directed k-clique is a directed k-clique as well (with a smaller k value), an efficient way to extract the directed k-clique modules of a network is to first find all *directed cliques* first: a directed clique is a maximal directed k-clique, i.e. it is not part of an even larger directed k-clique. A CPMd module of a given k is equivalent of the union of directed cliques of size larger or equal to k, which can be reached from each other through overlaps of size larger or equal to $k - 1$.

We extracted the directed cliques using the following iteration

1. find all directed cliques of a given node,
2. remove the node and its links from the network.

To find the directed cliques of a given node, A, we use a back-tracing algorithm based on the hierarchical properties of the directed cliques. At the initial step we construct two containers, one for the in-neighbours and one for the out-neighbours of A. The hierarchy of the system at this point is illustrated in figure B.1(b): the in-neighbours are at the top, the out-neighbours are at the

bottom, and the node A itself is in-between them. Next we take a node from the in-neighbours (or the out-neighbours), this node and A form a directed 2-clique. We place the node above (or below) A, and filter the remaining nodes in the containers so that for both nodes in the newly formed 2-clique it is true that

1. the members in the containers above the node in the hierarchy are all in-neighbours of node A,
2. the members in the containers below the node in the hierarchy are all out-neighbours of node A.

If necessary, we may introduce a new container as well, e.g. in figure B.1(c), by picking node B from the top container, the node E which is an out-neighbour of B and an in-neighbour of A is placed in a container in-between B and A in the hierarchy. This way when picking the next node from any of the containers, its rank in the hierarchy inside the forming directed clique coincides with the rank of its container with respect to the already selected nodes. For example, when picking node C in the example shown in figure B.1, it is placed above node B. By recursively picking new nodes from the containers, filtering the containers and introducing new containers we build up a directed clique. (The extraction of the clique ends when all containers become empty.)

Our algorithm scales similarly to the original CPM (see the supplementary information of [19]). Since the determination of the full set of cliques of a graph is widely believed to be a non-polynomial problem, the extraction of the directed cliques is non-polynomial as well. In spite of this, in real networks our algorithm proves to be quite efficient. Our experience shows that the required CPU time depends on the structure of the input data very strongly, therefore, in general no closed formula can be given even to estimate the system size dependence. As an illustration of the computational speed, however, we note that a complete analysis of the word-association network with over 70 000 links takes less than 5 min on a PC. By extracting the directed modules of this system at different link-weight thresholds, the time dependence of the algorithm could be fitted with $t = AM^{B\ln(M)}$ where t denotes the time needed by our program, M stands for the number of edges, and A and B are fitting parameters.

References

[1] Barabási A-L and Albert R 2002 *Rev. Mod. Phys.* **74** 47
[2] Dorogovtsev S N and Mendes J F F 2003 *Evolution of Networks: From Biological Nets to the Internet and WWW* (Oxford: Oxford University Press)
[3] Watts D J and Strogatz S H 1998 *Nature* **393** 440
[4] Faloutsos M, Faloutsos P and Faloutsos C 1999 *Comput. Commun. Rev.* **29** 251
[5] Barabási A-L and Albert R 1999 *Science* **286** 509
[6] Boccaletti S *et al* 2006 *Phys. Rep.* **424** 175
[7] Jeong H, Tombor B, Albert R, Oltvai Z N and Barabási A-L 2000 *Nature* **407** 651
[8] Ravasz E, Somera A L, Mongru D A, Oltvai Z N and Barabási A-L 2002 *Science* **297** 1551
[9] Han J J *et al* 2004 *Nature* **430** 88
[10] Milo R *et al* 2002 *Science* **298** 824
[11] Milo R *et al* 2004 *Science* **303** 1538
[12] Guimerá R, Mossa S, Turtschi A and Amaral L A N 2005 *Proc. Natl Acad. Sci. USA* **102** 7794

[13] Mangan S and Alon U 2003 *Proc. Natl Acad. Sci. USA* **100** 11980
[14] Scott J 2000 *Social Network Analysis: A Handbook* 2nd edn (London: Sage)
[15] Shiffrin R M and Börner K 2004 *Proc. Natl Acad. Sci. USA* (Suppl 1)**101** 5183
[16] Everitt B S 1953 *Cluster Analysis* 3rd edn (London: Edward Arnold)
[17] Knudsen S 2004 *A Guide to Analysis of DNA Microarray Data* 2nd edn (Wiley-Liss)
[18] Newman M E J 2004 *Eur. Phys. J.* B **38** 321
[19] Palla G, Derényi I, Farkas I and Vicsek T 2005 *Nature* **435** 814
[20] Girvan M and Newman M E J 2002 *Proc. Natl Acad. Sci. USA* **99** 7821
[21] Rives A W and Galitski T 2003 *Proc. Natl Acad. Sci. USA* **100** 1128
[22] Watts D J, Dodds P S and Newman M E J 2002 *Science* **296** 1302
[23] Palla G, Barabási A-L and Vicsek T 2007 *Nature* **446** 664
[24] Spirin V and Mirny L A 2003 *Proc. Natl Acad. Sci. USA* **100** 12123
[25] Onnela J-P, Chakraborti A, Kaski K, Kertész J and Kanto A 2003 *Phys. Rev.* E **68** 056110
[26] Krogan N J *et al* 2006 *Nature* **440** 637
[27] Antonov A V and Mewes H W 2006 *J. Mol. Biol.* **363** 289
[28] Song C, Havlin S and Makse H A 2005 *Nature* **433** 392
[29] Pollner P, Palla G and Vicsek T 2006 *Europhys. Lett.* **73** 478
[30] Kosub S 2005 *Local density Network Analysis (Lecture Notes in Computer Science* vol 3418) ed U Brandes and T Erlebach (Berlin: Springer) chapter 6, pp 112–42
[31] Newman M E J 2006 *Proc. Natl Acad. Sci. USA* **103** 8577
[32] Guldener U *et al* 2005 *Nucleic Acids Res.* **33** D364
[33] Open Directory Project online at http://www.dmoz.org
[34] Derényi I, Palla G and Vicsek T 2005 *Phys. Rev. Lett.* **94** 160202
[35] Adamcsek B, Palla G, Farkas I J, Derényi I and Vicsek T 2006 *Bioinformatics* **22** 1021
[36] Erdős P and Rényi A 1960 *Publ. Math. Inst. Hung. Acad. Sci.* **5** 17
[37] Nelson D L, McEvoy C L and Schreiber T A *The University of South Florida word association, rhyme, and word fragment norms* online at http://www.usf.edu/FreeAssociation/
[38] Ebel H, Mielsch L-I and Bornholdt S 2002 *Phys. Rev.* E **66** 035103
[39] Harbison C T *et al* 2004 *Nature* **431** 99
[40] Blais A and Dynlacht B D 2005 *Genes Dev.* **19** 1499
[41] The Gene Ontology Consortium 2000 *Nat. Genetics* **25** 25
[42] Boyle E I *et al* 2004 *Bioinformatics* **20** 3710