Available online at www.sciencedirect.com

SCIENCE DIRECT®

Science of Computer Programming 53 (2004) 195–214

Science of Computer Programming

www.elsevier.com/locate/scico

Roles in networks

Geoffrey Canright*, Kenth Engø-Monsen

Telenor Research and Development, Snarøyveien 30, N-1331 Fornebu, Norway

Received 29 October 2003; accepted 30 December 2003

Available online 17 July 2004

Abstract

In this paper we offer a topology-driven (‘natural’) definition of subclusters of an undirected graph or network. In addition we find rules for assigning unique roles (from a small set of possible roles) to each node in the network. Our approach is based on the use of a ‘smooth’ index for well-connectedness (eigenvector centrality) which is computed for each node. This index, viewed as a height function, then guides the decomposition of the graph into regions (associated with local peaks of the index), and borders (valleys) between regions. We propose and compare two rules for assigning nodes to regions. We illustrate our approach with simple test graphs, and also by applying it to snapshots of the Gnutella peer-to-peer network from late 2001. This latter analysis suggests that our method implies novel ways of interpreting the notion of well-connectedness for a graph, as these snapshots represent very well connected networks. We argue that our approach is well suited for analyzing computer networks, towards the goal of enhancing their security.

Keywords: Graphs; Clusters; Centrality; Security

1. Introduction

Networks (graphs) are interesting objects. They have a great deal of structure, and yet at the same time are simple: they consist (in simplest form) only of nodes, connected by links. The abstract idea of a network (or graph—we use the terms interchangeably) is also highly useful in modeling structures observed in the world. Examples include: social networks, communications networks, the World Wide Web, metabolic and genetic

* Corresponding author.
E-mail addresses: geoffrey.canright@telenor.com (G. Canright), kenth.engo-monsen@telenor.com (K. Engø-Monsen).

0167-6423/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.scico.2003.12.008

networks in biological systems, food webs, ... In short, a *network* is a simple, nontrivial abstract structure, fascinating in its own right, and also highly relevant for many branches of science and technology.

System administration invariably involves managing a network, which is composed of multiple types of links. Examples include: the physical links between the machines, the logical links between users and files, and the social links between users. An important aspect of system administration is to ensure the free flow of needed information over the network, while at the same time inhibiting the flow of harmful or damaging information over this same network. The *structure of the network* plays a crucial role in the implementing of these two important (and partly conflicting) goals of system administration. Both goals involve the spreading of information over links of the network; hence both problems are strongly sensitive to the network structure. Because of this dependence, we feel that the understanding of network structure can be a valuable component of effective system administration.

For these reasons, networks merit serious study. A network is one of the simplest abstractions of structure that we can study; yet, understanding the structure of a network is a nontrivial undertaking. This question has received a great deal of attention in the last decade or so. Most of the measures of network structure that have been studied to date [8] take the form of 'whole-graph' properties—that is, scalar measures or distributions which apply to the graph as a whole, and are calculated using averaging. Examples of such whole-graph properties include the node degree distribution, the diameter or average path length, clustering coefficients, and the notion of 'small worlds' (which is based on the previous two).

Whole-graph properties are important and useful; however, they cannot be the complete answer to the question: How can we understand the structure of a network? Suppose, for example, we look at a small neighborhood, or even a single node, and wish to say something meaningful about the role that subgraph or node plays in the overall structure of the network. We can of course say where the single node lies on the node degree distribution. Similarly, we can compare the clustering coefficient of a neighborhood with that for the whole graph. What else can we say about small pieces of the whole?

The work of Kleinberg [7] gives a partial answer. Kleinberg considered a directed graph, defined two distinct types of roles for the nodes on the graph, and gave a way to calculate indices which quantify the degree to which each node plays the two types of role. That is, each node in a directed graph may be assigned an Authority score and a Hub score. It is important to note that these scores can be based solely on the *topology* of the graph—independent of any questions of content or meaning, or of any 'properties' of the nodes.

The names of these two role types convey their meaning. Nodes with high Authority are nodes which are important because they are pointed to by important nodes—in fact, by nodes with high Hub scores. And the latter obtain their high Hub scores by pointing to good Authority nodes. In short: Hubs point, and Authorities are pointed to. These ideas can be highly useful in analyzing the structure of the WWW: Authorities are likely good endpoints of an information search, while Hubs are useful in leading the search to the Authorities. It seems clear that similar roles arise in social networks: sometimes, one knows who has the needed information (the Authority); at other times, one needs to ask a good Hub.

Kleinberg's work gives us *indices* for each node in the network. These indices tell us useful information abut the role(s) the node plays in the network. Such information is quite distinct from whole-graph information; and yet it is still derived (at least as originally presented) purely from the topological structure of the graph.

Another aspect of a graph, which is again distinct from whole-graph properties, is the *community structure* of the graph. In the same paper, Kleinberg suggested a way to find such communities in graphs such as the Web graph. The mathematical tools used are basically the same as those used to find Hub/Authority scores—which means, among other things, that the decomposition of the graph into communities was also based purely on the structure of the graph, without invoking any knowledge or properties of the nodes or links. Furthermore, we note that decomposing a graph into subcommunities provides new information about the roles played by nodes: they may be members of community X; they may happen to lie in *no* community; they may be 'leaders' in some sense of their community, or they may lie on the 'edge'; and they may play an important role in linking multiple communities.

Hence we view the notion of community structure of a graph, and the question of roles of nodes and links in a graph, to be tightly related.

Many other works have addressed the same problem of how to find 'natural' communities in a directed graph such as the Web. In contrast, Girvan and Newman [5] have looked at this question for undirected graphs. Their basic approach is to define communities by first finding their 'boundaries'—specifically, by finding links with high 'betweenness', which, when removed, break the graph into subcommunities.

In this paper we will also focus attention on undirected graphs. Our goals are as follows. We wish to find a 'natural' means—that is, one based solely on the graph topology—for decomposing an undirected graph into communities. We also wish to define a set of roles for the nodes of the graph, such that each node is assigned one, and only one, role. That is, unlike Kleinberg, we want our roles to be binary (Yes/No) properties of nodes—and exclusive as well. The roles we will arrive at are: 'leader' of a community; member of a community; and two types of roles for nodes in the 'border set', i.e., nodes not belonging to any community.

Our approach is roughly dual to that of Girvan and Newman. We begin, not with the 'edges', but with the 'centres' of the communities. From this starting point, we work 'outwards' to find the members, and finally the border nodes. We do not claim that this set of roles is complete, in the sense that no others could be defined. (For example, it might be of interest to identify further substructure within each community.) However, our set of roles is complete and consistent, in the sense that our definitions allow a unique and unambiguous association of a single role with each node on the graph.

Our work draws inspiration from that of Kleinberg on directed graphs. On undirected graphs, however, the two role types (Hub and Authority) become the same (a type of centrality). Also, we seek Yes/No definitions of roles, rather than continuously varying indices. Yet, as we will see, the centrality index that emerges from applying Hub and Authority definitions to an undirected graph will provide the starting point from which we define communities, and thereafter roles.

1.1. Possible applications

How might these ideas be useful in analyzing real networks?

The utility for *social networks* seems clear [4]. It is obviously of interest to identify communities in a measured social network. The links may be of any type (as long as they are undirected) for our ideas to apply: friendship, collaboration, etc. The roles of ‘centre of a community’, ‘member of a community’, and ‘bridge between communities’ are also intuitively plausible for social networks. An example with a slightly different flavor is the network of sexual contacts. Here too these ideas may be quite useful, in work addressed at limiting the spread of sexually transmitted diseases: perhaps one would focus on the two complementary goals of (i) preventing infection of the central nodes of each community, and (ii) preventing the transmission of the disease across the bridging nodes.

We expect that there are useful and interesting applications of these ideas to *technological or communication networks* as well. Again, the only prerequisite is that the links be undirected. An obvious example is the Internet. One difficulty with technological networks is that the significance of the roles we define is less obvious—that is, what a role *implies* about a node depends on what kind of question one asks. One type of question for technological networks is similar to that from the sexual network mentioned above: How does one prevent the spreading of damage? And in this case the analogy seems useful. However, there are likely other types of questions about such networks that may be usefully illuminated by the methods presented here.

Finally, there are of course those networks that are both social and technological. Examples include the telephony graph; peer-to-peer networks [10] overlaid on the Internet; and the combined network of computers, files, and users that is the daily preoccupation of every system administrator. Here, once again, security seems an obvious application for these ideas: one wishes to identify nodes that should be given highest priority in protecting against viruses, for example. We note in this context that our method of analysis may be applied either to physical networks, or to logical networks which exist as overlay networks on top of the physical network. The important common aspect is the identification of links (physical or logical), over which information can flow.

Previous papers [3,13] have shown in more detail how to apply the analysis presented here to networked computers with many users. Here our main goal is to present in detail the definitions for the roles, and the logic behind these definitions. In addition, we will apply this analysis to the several snapshots of the Gnutella peer-to-peer network. Peer-to-peer networks [10] are hybrid social/technological networks that are self-managing—that is (excepting Napster, which is defunct), they build and maintain a structure without the help of any central node. Thus any communities and other types of structure that may be found in snapshots of the Gnutella network are formed from many local actions of members who lack any kind of view of the entire network. Nevertheless, we will see that Gnutella networks, at least at the times of the snapshots, are extremely well connected.

2. Roles in networks—the logic

The idea we wish to pursue is that ‘well-connectedness’ may be viewed as a *height function* over the discrete space (the graph). If our height function is smooth enough, then we can employ ideas appropriate for smooth surfaces over a continuous space. That is, we want to use a *topographical* picture to define regions in a network. Regions will correspond to 'mountains', with the centre of each region being the corresponding mountain top. Boundaries between regions will then be defined as those points failing to be uniquely associated with one mountain region.

2.1. The mathematics

We focus on 'smooth' functions over a discrete space. Again we draw as much insight as possible from the continuous case.

2.1.1. Harmonic functions and smoothness

Suppose the domain space is continuous. Then harmonic functions are the most smooth functions available. These functions are solutions to Laplace's equation,

$$\nabla^2 \phi = 0. \tag{2.1}$$

For a given space, one obtains different solutions to (2.1) from differing boundary conditions on ϕ.

We immediately identify some problems with the continuum picture. One problem is that there are no maxima (or minima) away from the boundary. Hence our topographic picture cannot work with such smooth functions: we will find no mountain tops not lying on the boundary. Furthermore, we are seeking a *natural* way of defining regions. Here 'natural' means guided as much as possible by the topology of the graph. Hence it is undesirable to have to assign values for our height function ϕ at the boundary—we would prefer that the topology solve that problem for us.

We can of course solve this last problem by setting ϕ = constant (for example, zero) at the boundary. That is, we just give the boundary some nominal reference value. This is 'natural' enough; however, we then get that ϕ = constant over the *entire* space, due to the averaging property of Laplace's equation.

The discrete version of Laplace's equation is

$$\mathbf{L}\phi = \mathbf{0}, \tag{2.2}$$

where $\mathbf{L} = \mathbf{K} - \mathbf{A}$ is the Laplacian matrix, $\mathbf{K} = \mathrm{Diag}(k_1, k_2, \ldots)$ is a diagonal matrix whose ith entry is the node degree k_i, and $\mathbf{A}$ is the adjacency matrix, with $A_{ij} = 1$ if there is a link from i to j, and 0 otherwise.

It is easy to see that the averaging property holds here also: solutions to (2.2) obey

$$\phi_i = \frac{1}{k_i} \sum_{j = nn \text{ of } i} \phi_j . \tag{2.3}$$

Here '*nn*' means 'near neighbor'. The discrete Laplace equation thus offers 'most smooth' functions for the discrete case; but it has all the problems seen for continuous harmonic functions, plus one more. The additional problem stems from the crucial fact that the specification of the boundary of a discrete space is not unique—in fact, there is no natural way to define such a boundary. We can of course take the (perhaps least arbitrary) assumption that none of the points are boundary points—all are to have their height determined by the graph structure—but then we get back the constant solution $\phi_i =$ constant.

2.1.2. Eigenvector centrality

A small change in the above picture solves all of its problems at once. The small change is as follows: we ask for a height function which obeys, instead of the averaging property (2.3), the following:

$$\phi_i = \frac{1}{\lambda} \sum_{j=nn \text{ of } i} \phi_j . \tag{2.4}$$

That is, instead of taking the strict average over all neighbors, we divide the neighbor sum by a constant λ, which is the same for all nodes. This equation can be written as

$$\mathbf{A}\phi = \lambda\phi, \tag{2.5}$$

where $\mathbf{A}$ is again the adjacency matrix. Now we have an eigenvalue equation, and our height function ϕ is an eigenvector of the adjacency matrix. We want in fact the eigenvector which is the stable iterative solution of (2.4), because we want height to signify 'well-connectedness'. That is, (2.4) encodes the idea that node i's well-connectedness is determined, to within a scale constant λ, by that of all of i's neighbors. Iterating this requirement, from any starting point, will give the principal eigenvector of the adjacency matrix. This eigenvector gives the stable, self-consistent solution of (2.4); it also has the property that it is positive semidefinite, since $\mathbf{A}$ is.

Thus we have a simple mathematical definition of well-connectedness on an undirected graph. It is a definition known from sociology: it is due to Bonacich [2], and is termed 'eigenvector centrality' or EVC. We do not offer a rigorous definition of smoothness for this function. It is plausible that positive definite solutions to (2.4) are reasonably smooth: they are 'almost' averaging. Also, we find from experience (see below) that EVC is 'smooth enough' to give sensible results, when applied for the purpose of defining regions in a graph.

With this one modification, the problems we saw above with Laplace's equation (discrete or otherwise) are no longer present. EVC can have local maxima away from the boundary. In fact, since it measures well-connectedness, local maxima of EVC tend to lie well away from any nodes that one might be tempted to call 'boundary nodes'. Furthermore, there is no need to define a boundary for the discrete case: all nodes may have EVC values determined by Eq. (2.4), with no values input as 'boundary conditions'.

Finally we recall our main goal: to assign a natural and unique role to each node in the network, based solely on the topology of the graph. As noted above, Kleinberg found two such roles for directed graphs: Hubs and Authorities. Hubs are naturally good at pointing to good Authorities; and Authorities are naturally good at being pointed to by good Hubs. We see already from these simple grammatical statements that the distinction between Hubs and Authorities vanishes when the arcs of the graph become undirected (so that 'pointing to' = 'being pointed at'). The mathematics gives the same result: for the undirected case, the adjacency matrix is symmetric, $\mathbf{A} = \mathbf{A}^{\mathrm{T}}$, and so the matrices defining Hubs and Authorities become the same.

In short, for undirected graphs, the two types of roles collapse to one. That one role (more precisely, an index quantifying the degree to which the node plays the role) is eigenvector centrality. The Hub operator $\mathbf{AA}^{\mathrm{T}}$ and the Authority operator $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ become simply $\mathbf{A}^2$, whose principal eigenvector is the same as that for $\mathbf{A}$.

Hence we find that two of the roles identified in Kleinberg's work with directed graphs become a single (type of) role for an undirected graph. This role type we call well-connectedness, or eigenvector centrality. We seek, however, further distinctions among the nodes of an undirected graph—in other words, multiple distinct roles, to which any given node may be assigned. These roles will be defined in the next section. Eigenvector centrality (EVC) will be our height function, and hence our starting point.

2.2. Definitions of the roles

Let us first make precise the difference between 'role type' and 'role'. We can associate real-valued indices or 'scores' with each node: Hub and Authority scores for the directed case, and EVC score for the undirected case. These are role types; in fact it is fair to say that all three scores represent some type of centrality. All nodes have some degree of centrality; and 'being central' is certainly a *type* of role. By *role*, however, we mean a binary (Yes/No) distinction applied to each node, so that each node receives a single Yes and hence is assigned a unique and unambiguous role. Centrality (a role type) will give us a smooth height function over the graph, allowing us to use topographic criteria to assign a role to each node.

2.2.1. Centres

We hold onto the picture of mountains, valleys, saddles, etc for our height function. Each mountain may be defined by its peak. The peak is a local maximum of the height function. Our first role is then the mountain peak.

Centre: *Any node which is a local maximum of the eigenvector centrality is a Centre.*

2.2.2. Regions

Each mountain top defines a mountain. Hence the number of Regions in the graph is equal to the number of centres. (Henceforth, except when roles are defined, we drop the capital letters; the meaning should be clear from context.) Regions are usually composed of more than one node; hence the role for a node cannot be a region, but rather a Region Member.

Region Member: *Each node that may be uniquely associated with a single Centre, according to an unambiguous rule, is a member of that Centre's Region, and hence a Region Member.*

It remains to specify the 'unambiguous rule'. We suggest two possibilities.

Rule 1 (Distance). *A node is associated with Centre C if it is closer (in number of hops) to C than to any other Centre C'.*

Rule 2 (Steepest Ascent). *For each node i, a steepest-ascent path starting at i will terminate at one (or more) Centres. If it terminates at a single Centre, then node i is associated with that Centre.*

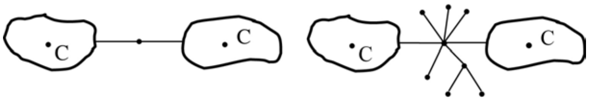

Fig. 2.1. Bridge Node (left) and Bridge Node and Danglers (right).

These rules are simply the discrete-domain version of the process of associating a part of the domain (base space) with each mountain top—hence defining each mountain. We are careful here to break our definition of region into two parts: the definition itself, which refers to a rule but does not specify it; and the rule. We do this because we feel that more than one rule is possible for the discrete case; and so we state the region definition in a way that captures the 'mountain' idea, but leaves the rule unspecified.

Both rules stated above satisfy the intuitively reasonable criterion that a centre's near neighbors should (in general) belong to its region. (It is, after all, the number and connectedness of a centre's neighbors that gives that centre its high EVC.) Both rules are also easy to implement in a simple iterative fashion—starting with the centres, and working outwards from them, 'coloring' nodes according to the regions (centres) they belong to. The steepest-ascent rule is, however, the rule which is the most faithful to our topographic picture.

2.2.3. Borders—between regions

On a continuous topographic surface there are points which lie *between* mountains, and belong to no unique mountain. It may happen that analogous points exist for the discrete case as well. Nodes which cannot be associated with any one mountain are assigned to the Border set.

Border Nodes: *Any node for which the unambiguous rule for Region membership gives more than one answer is a Border Node.*

Intuitively, we think of border nodes as 'connecting regions'. And yet, a bit more thought reveals that not all border nodes are equal in this regard. Some border nodes do indeed play an important role in connecting two or more regions: they lie on paths which connect the respective centres (hence regions). See the left panel of Fig. 2.1. Other nodes may be removed, without any loss in the degree of connection between the regions. See the right panel of Fig. 2.1. Hence we are motivated to define two distinct roles to the set of border nodes.

Bridge Node: *A Border Node which lies on at least one non-self-retracing path connecting two Centres is a Bridge Node.*

Dangler: *Any Border Node which is not a Bridge Node is a Dangler.*

Danglers of course may *inject* new information into the network; but they do not play a significant role in the *transport* of information between regions.

Finally, we wish to single out a class of *links* which play an important role in connecting regions. Our reason for doing so here is that the border set for the steepest-ascent rule is in general very small or zero. In this case we still wish to highlight those network elements which connect the regions. Hence we define:

Bridge Links: *Any link whose endpoints lie in two distinct Regions is a Bridge Link.*

Bridge links will occur for either region rule above.

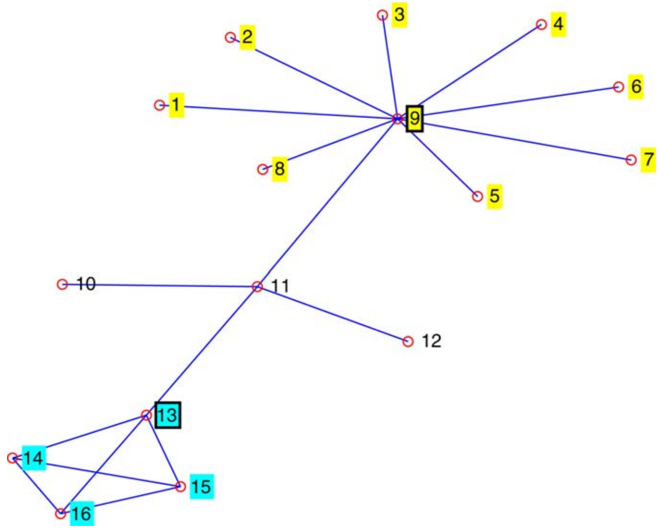

Fig. 3.1. A simple graph with two regions, as defined by Rule 1 (distance test).

One can imagine rules for defining regions which give 'fat' borders. For example, one could associate nodes with centres according to:

Rule 1′ (Distance with Cutoff). *A node is associated with Centre C if it is closer (in number of hops) to C than to any other Centre C′*, and *if its distance from C is not greater than h hops.*

'Fat' borders arise for such a rule since there could be many nodes which are farther than *h* hops from any centre. In general, 'fat' boundaries arise if we choose a rule designed to avoid the 'growing together' of regions from their respective centres. The distance to which growth is allowed could then be measured in hops (as in Rule 1′), or in decrements in EVC.

Boundaries according to Rule 1 are 'thin': essentially one node wide. Boundaries according to Rule 2 are even thinner: in general, they are 0 nodes wide, since it is rare that a node will have two or more steepest-ascent paths, leading to different local maxima.

3. Examples

We illustrate our method, and compare the two rules for defining regions, using some simple examples.

Fig. 3.1 shows a simple graph with two centres. The Border consists of three nodes. One (node 11) is a bridge node which clearly plays an essential role in connecting the two regions. The other two are danglers.

Applying Rule 2 to the same graph gives us Fig. 3.2. Here we see that the entire border has been 'swallowed' by the dominant centre (node 9). The rather peripheral role of nodes

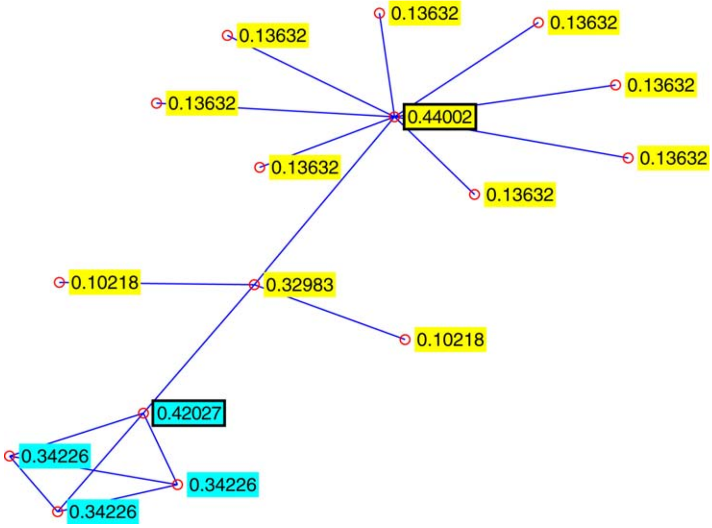

Fig. 3.2. The same graph as in Fig. 3.1, but defining the regions using Rule 2. EVC values for the nodes are also shown.

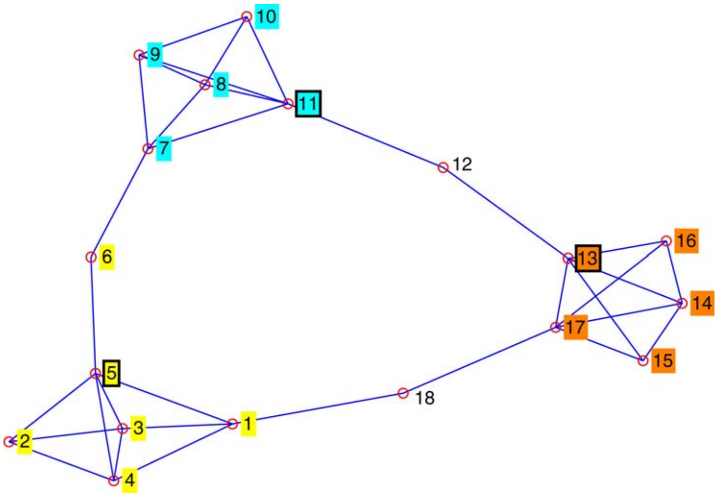

Fig. 3.3. A graph with three regions; Rule 1.

10 and 12—formerly classified as danglers—is now reflected in their distance (2 hops) from their centre (and of course in their low EVC).

Comparing these two figures thus confirms our expectations about the differences between the two rules: a border set, with or without danglers, is typically present with Rule 1, but absent with Rule 2. We see the same picture for a graph with three regions in Figs. 3.3 and 3.4.

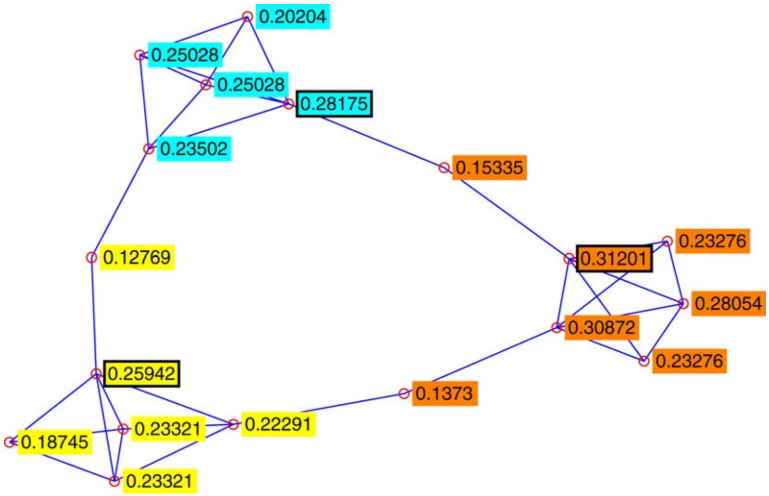

Fig. 3.4. The same graph as in Fig. 3.3, but defining the regions using Rule 2.

We note here that local *minima* of the EVC play, in general, rather uninteresting roles in the network. For example, in Figs. 3.1 and 3.2, the two dangler nodes, and all the nodes around the central light grey (yellow in the web version) node, are local minima of the EVC. They are, loosely speaking, nodes on the' edge' of the network. A local minimum of the EVC must be poorly connected in general. That is, one must abandon the picture of a local minimum as a low-lying node surrounded 'in all directions' by neighboring, higher-lying nodes—because there are likely very few such directions (neighbors) for nodes which are local minima of the EVC. Hence it is logical to *define* the 'edge of the network' as being precisely this set of nodes (local minima).

Bridge nodes, on the other hand, correspond to the topographic notion of 'saddles'. That is, they lie 'between' peaks; and there are usually lower-lying nodes (danglers), linked to the saddles, but in another 'direction'—so that, at a bridge node, as with a true saddle, one goes only uphill along a certain 'axis' or direction, and only downhill along another.

The bridges in Fig. 3.4 are local minima—but only because they have no links in any other 'direction' than that (uphill) joining the centres. That is, the saddles have no width, and hence lie on the edge of the network. The other local minima (nodes 2, 10, 15, and 16) are farthest from the 'action' (high connectivity) in the graph—they lie on its edge.

To illustrate the application of these ideas, we suppose that the nodes in Figs. 3.1 and 3.2 are users in a computer network, while the links are effective connections between users which allow information flow. Here we say 'effective' connections, because the links may not be direct: they may be mediated by files to which both users have read and write access [3]. We conclude immediately from the analysis that the user system is naturally composed of two main groups (light grey (yellow in the web version) and medium grey (blue in the web version)). Furthermore, node 9 is most central to the light grey (yellow in the web version) group, while node 13 is most central for the medium grey (blue in

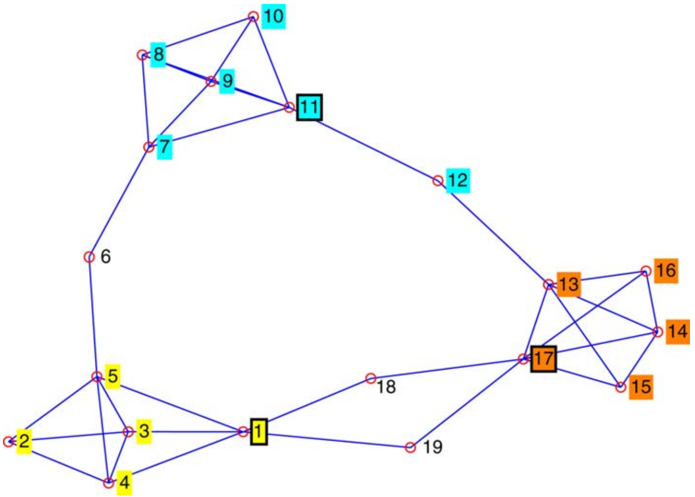

Fig. 3.5. A graph with three regions, and parallel bridge nodes; Rule 1.

the web version) group. Finally, node 11 is a bridge node which is crucial for the flow of information between the two groups.

Suppose further that we are concerned with security for this small system. Then we can immediately identify nodes 9, 13, and 11 as most urgently needing protection from whatever threats the system faces. Nodes 9 and 13 are to be protected because they are centres of their regions: if they are infected, then there is a high probability that their entire region will also be infected. Furthermore, we can give node 9 a higher priority for protection than node 13, since its region is larger. Finally, node 11 merits extra protection, since if it can be rendered immune to the threats, then these threats have no ready channel for spreading from one region to another.

Note that the use of Rule 2 does not single out any border nodes for special protection—even though node 11 clearly plays an important role in connecting the two regions. However, Rule 2 will identify the *link* between 11 and 13 as a bridge link. The obvious consequence of this is that the nodes on each end of each bridge link deserve special protective measures.

We can turn this problem on its head, by giving the administrator the problem of *spreading desired information* over this same small network. Our analysis then suggests an efficient strategy for doing so: one starts with the centres (nodes 9 and 13), and arranges for the desired information to be broadcast from there.

Finally, we illustrate another possible difference between the two rules in Figs. 3.5 and 3.6. Here the point is that the border (from Rule 1) not only vanishes when we apply Rule 2—it also moves: that is, node 12 belongs to the 'medium grey (blue in the web version)' region according to Rule 1, but to the 'dark grey (orange in the web version)' region by Rule 2.

It is of course to be expected that the distance rule and the steepest-ascent rule will give conflicting results for some nodes. An important point to be gleaned from Fig. 3.1

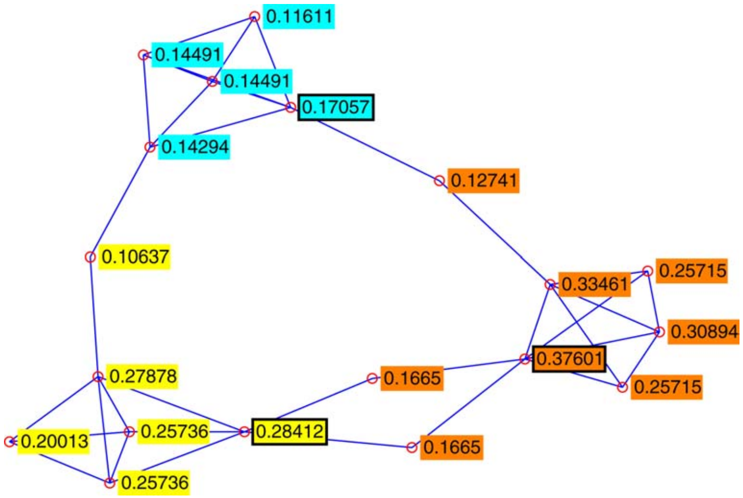

Fig. 3.6. The same graph as in Fig. 3.5, but defining the regions using Rule 2.

through 3.6 is that the general qualitative picture is rather insensitive to the choice of rule for defining regions. We expect this to be the case for most graphs. The choice of centres is independent of which rule is used; and these centres in turn exist precisely because they lie in a region of the graph that has some ‘weight’—that is, some number of nodes which are better connected to one another than to their ‘surroundings’. In short, we believe that the distinct rules, which ostensibly define regions, actually differ principally according to where they place the *boundaries* between regions—while the regions are in themselves rather stable objects.

4. Example—the Gnutella network

We have applied our method of analysis to a set of snapshots of the Gnutella peer-to-peer network. These snapshots [6] were taken in November and December of 2001, and consist of about 1000 nodes in one connected piece (ignoring very small disconnected pieces).

Peer-to-peer networks [10] are logical overlay networks based upon the Internet; they are largely or entirely self-organized. At the time these snapshots were taken, the Gnutella network used protocols calling for some, high-bandwidth, nodes to be ‘supernodes’. These nodes take responsibility for indexing and query handling on behalf of a set of ‘ordinary’ nodes associated with them. The ordinary nodes are encouraged to maintain a neighbor set of at least three or four neighbors. The supernodes may be expected to handle a higher number of neighbors, depending on their capacity.

From this simple picture one might expect a two-humped node degree distribution. Instead, we find that the node degree distribution is, to a good approximation, a power-law distribution (see Fig. 4.1 for an example). Power-law distributions are not uncommon when the network is self-organized as is the case here. Such graphs are extremely well

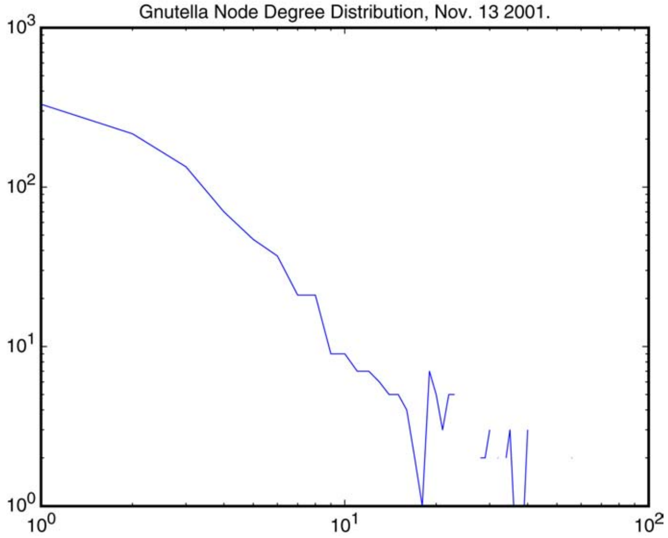

Fig. 4.1. Node degree distribution for the Gnutella network, November 13, 2001.

connected—for instance, they are highly resilient to random node deletion. In the context of this work, the term ‘well connected’ may be also taken to imply ‘few centres, with many links between the centres’. That is—in topographic language—local peaks are separated from one another by regions of low connectivity. A graph with few such regions will not support the existence of many local peaks of the centrality. This then is our expectation for the Gnutella networks that we analyzed.

We have applied both rules to seven Gnutella snapshots. The results are summarized in Table 4.1.

The most prominent qualitative result that we see is just that expected: there are very few regions in these well connected graphs. Five of the graphs have two regions, and the other two are composed of a single region. We view this in itself as a strong result: the ratio of regions to nodes is extremely small here, indicating that these graphs are very well connected.

Next we compare the two rules for defining region membership. Not surprisingly, Rule 1 (based on distance) is more ‘democratic’ than Rule 2 (based on height). That is, it will not be uncommon that nodes are closer (in hops) to a ‘weak’ (low EVC) centre than they are to a dominant centre; yet at the same time such nodes can be topographically part of the local peak associated with the more distant, but higher, centre. We see extreme cases of this in the last two snapshots. In each case there are on the order of 100 nodes which move from the weak region to the dominant one, on shifting from Rule 1 to Rule 2. The dominant region also acquires essentially all of the border set when Rule 2 is applied.

In short, in four out of five of the cases with two regions, the disparity in size between the two regions is increased by applying Rule 2 instead of Rule 1. (The exception is found in the second row of the table; here the size ratio of the two regions is 1.7 and 1.4, for Rules 1 and 2 respectively.)

Table 4.1
Results for a series of snapshots of the Gnutella peer-to-peer network from late 2001. Node numbers (which are arbitrary) are given in parentheses. The last three columns simply give the number of nodes in the corresponding sets

Date	# nodes	Top degree	Centres	Regions (Rule 1)	Border (Rule 1)	Regions (Rule 2)
13.11.01	992	(24):83	(24)	211	570	772
		(335):84	(335)	211		220
16.11.01	1008	(13):53	(13)	192	488	425
		(54):55	(54)	328		583
20.12.01 (1)	904	(105):77	(105)	904	0	904
20.12.01 (2)	1077	(56):125	(56)	1077	0	1077
		(259):125				
27.12.01 (1)	1095	(18):136	(18)	244	447	318
		(87):142	(87)	394		777
27.12.01 (2)	1026	(13):118	(13)	498	394	996
		(42):109	(97)	134		30
28.12.01	1050	(194):126	(194)	521	378	972
		(410):109	(518)	151		78

For the case of Rule 1, we see that the border set is very large—on the order of half of the nodes. We regard this as another sign of the well-connectedness of the graphs: those few distinct regions which are found are well connected to one another. For the case of Rule 2, this shows up in the large number of border *links* connecting the regions.

We have also created visualizations of the Gnutella graphs studied here, using the Archipelago visualization tool [13]. We show here the results for both rules, applied to one of the more extreme cases—that is, to the second graph for December 27 (sixth in Table 4.1). Archipelago, in its current version, places each centre at the geometric centre of a ring, and all member nodes for a region on the edge of the corresponding ring. Border nodes are placed in a smaller ring between regions. Border links are colored grey (red in the web version); other links are black.

Visualization of large graphs is a difficult problem [1]. Our method of analysis gives suggestions for visualization, but is far from solving the problem completely. We feel that the main points to be taken from Figs. 4.2 and 4.3 are the same as those taken from Table 4.1. (The graph is well connected, with many border elements; and Rule 2 yields a much more skewed partitioning of the nodes into regions than does Rule 1.) That is, these figures do not give new insight; but they do capture the insights gained from our analysis, and render them visually.

The Gnutella network is a logical network, overlaid on the Internet. There is thus no single system administrator responsible for this network. Furthermore, current peer-to-peer networks are quite large. Considering finally the fact that peer-to-peer networks have little or no central authority, we find that such networks are both largely unmanaged and difficult to manage.

Yet there is some degree of management, at least in the form of protocol distribution. Our analysis here implies that there can be ‘smart’ strategies for protocol update distribution—*if* one can map out the network topology, so that our analysis can be

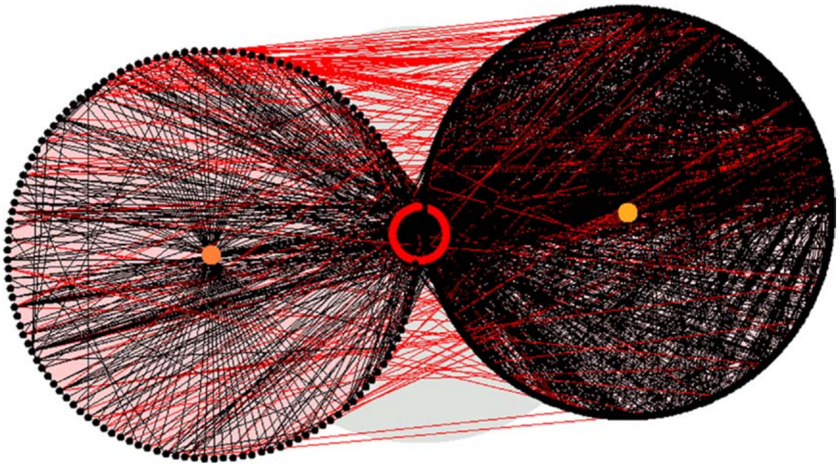

Fig. 4.2. Visualization of the December 27th Gnutella graph (sixth in Table 4.1), using the Archipelago software. Rule 1.

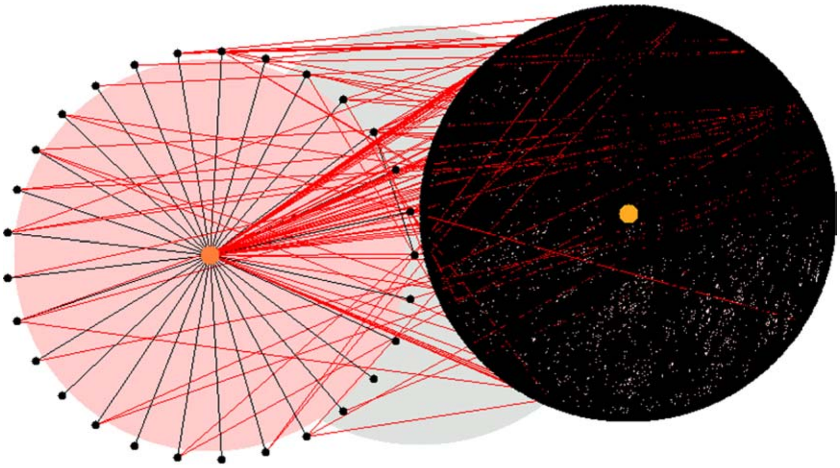

Fig. 4.3. Visualization of the December 27th Gnutella graph (sixth in Table 4.1), using the Archipelago software. Rule 2.

performed. That is, our method singles out central nodes as the best starting places for distribution.

Our analysis also gives a new interpretation of the term ‘well-connectedness’ for a network. That is, we find that these thousand-node networks are composed of one, or at most two, regions, with many bridges between the two regions for the latter case. The implication of this result is clear: it is very difficult to *prevent* the spreading of information over such a network. One can try to focus on the centres; but they are imbedded in a sea of other highly central nodes. And the bridges are too numerous to allow for the guarding of all of them. This conclusion is thus a negative one: we suggest that there is no readily practical way to protect such networks against undesired or harmful information. This conclusion is consistent with that of [12].

5. Summary, discussion, and future work

Our goal in this work has been to give some meaning, beyond whole-graph properties, to the term 'structure of a graph'. Our approach has been to seek a 'natural' (i.e., purely topology-driven) definition of the structure, in terms of regions (natural clusters), centres of these regions, and nodes (and links) in a border set lying between regions. Our basic criterion for defining a region (and its centre) has been well-connectedness, as measured by the 'smooth' graph function, eigenvector centrality or EVC. In addition to defining natural clusters of a graph, our approach also assigns a unique role to each node in the graph.

Our two rules for defining regions give qualitatively similar pictures for the graph structure as a whole, but rather different pictures in terms of which roles for nodes are present in the analysis. That is, Rule 1—associating nodes with regions based purely on their distance, in hops, from centres—places a significant number of nodes in the border set. These nodes in turn can be placed in two distinct roles: bridge nodes, and danglers (see Fig. 3.1). Rule 2 holds more closely to the 'topographic' spirit of our approach, associating nodes with centres to which they are linked by a steepest-path ascent. This rule normally (in the absence of special symmetry) places *no* nodes in the border set—such that, with Rule 2, the two roles in the border set (bridge nodes and danglers) are essentially excluded, and all nodes are either centres of a region, or members of a region.

One can imagine other rules for defining regions. We feel that the principal aspect of our approach is to identify centres first, and then let regions 'grow' outwards from these centres. Both of our rules fit this picture; but other rules do as well, and could be investigated. One might argue that the topographic rule (Rule 2) is most consistent with the spirit of our approach, relying as it does on a topographic picture of a graph, based on EVC. However, different rules may be better suited to different purposes for doing the structural analysis.

Our approach is dual to that of Girvan and Newman [5], in the sense that their approach defines regions by finding their boundaries, while ours finds first their centres. It would of course be very interesting to compare results for these two approaches; we plan to do so in future work. The Girvan/Newman approach also allows for a hierarchical decomposition of a graph, by breaking clusters into subclusters, etc. A similar hierarchical decomposition could also be done in our case, by eliminating border nodes and links, and applying our analysis to the resulting isolated regions.

We do not regard the set of roles that we have found as exhaustive. Since regions are defined by 'growing outwards' from centres, one can certainly quantify 'closeness' to the centre, for each region—in terms of distance (in the spirit of Rule 1), or 'height' (EVC) (in the spirit of Rule 2). As a very simple example, one can assign the role of 'Edge of the region' to those nodes which are connected to border elements (nodes or links). A different type of Edge role may be assigned to those nodes which are 'farthest' from the centre, but not linked to any border elements. For example, referring to Fig. 3.3, the first (linking) type of edge role may be assigned to nodes 11 and 7 (medium grey (blue in the web version)), 6 and 1 light grey (yellow in the web version), and 17 and 13 (dark grey (orange in the web version)). *All* of the other nodes, except centre node 5, also lie on the edge of their respective regions, by this definition. Of course, these regions are 'all edge' (and two of three centres lie on the edge) because the regions themselves are so small.

So far, we have only defined our approach, and given illustrative examples. An important task for future work is to test the sensitivity of our approach to small changes in graph topology. For example, Kleinberg's HITS algorithm has been shown to be, in some cases, highly sensitive to small topology changes [9]. The sensitivity occurs when the eigenvalue gap is small—so that small perturbations can lead to large mixing in of subdominant eigenvectors.

While reserving this question for future work, we offer a speculative comment here. A small eigenvalue gap is associated with poor mixing—i.e., slow convergence to the asymptotic distribution for a random walk. In very loose terms, such graphs are poorly connected. We feel then that, in such cases, sensitivity of the analyzed graph structure to small topology changes is reasonable, and not 'undesirable'. That is, the cluster structure of poorly connected graphs may be expected to be sensitive to small topology changes. This idea, of course, must be tested in future work. Also, our loose interpretation of the term 'well connected' needs to be sharpened.

In real networks, one is always subjected to the problem of incomplete information: it is often not possible to measure completely the topology of the network. Incomplete information means missing nodes or links—which may be regarded as a type of topology change, in the direction (usually) of making the network less well connected. The above discussion applies also to this kind of topology change: the analysis will be only weakly sensitive to this kind of error, when the network is well connected. In fact, there is almost certainly such error present in the Gnutella measurements which are analyzed here. However, even in the presence of such error, we find an extremely well connected network. Hence our qualitative conclusions should not be affected by this source of error.

For a poorly connected network, missing links can strongly affect the analysis. One example of this is the user/file system reported in [13]. This system was analyzed twice: first, ignoring links connecting students and faculty; and then including such links. In each case, the system was found to consist of a set of disconnected clusters. However, the nature and number of these clusters changed considerably with the inclusion of the student–faculty links. This is an extreme case—since the network is so poorly connected that it is disconnected—but it illustrates our point. In fact, the 'true' network is likely very poorly connected. However, there are also (likely) undetected links which render it connected, and hence, susceptible—albeit weakly—to system-wide infection.

A clear conclusion from the above discussion is that measurement errors (missing links and/or nodes) have much more effect in distorting the analysis when the network is poorly connected. Hence, in such cases, extra effort may be merited to try to detect all relevant elements (links and nodes).

We mention yet another direction for future work, namely directed graphs. Here we suggest one possible approach, which we plan to explore further. Kleinberg's approach gives *two* scalar height functions over a directed graph. That is, each node gets a Hub score and an Authority score. Hence one obvious generalization of our approach is to apply it *twice* for a directed graph—once for each score. This gives of course two distinct structural analyses for one graph—a feature which may or may not be desirable.

We note that the notion of ‘neighbor’ must be clarified before this suggestion may be implemented. That is, if we look at node i’s Hub score, and seek to relate that score to i’s neighbors, which nodes should we look at? Those pointing to i, those pointed to by i, or both? We suggest that neither set is appropriate: the Hub/Authority calculation propagates the two scores via iterations of a *two*-hop operator ($\mathbf{A}^{\mathrm{T}}\mathbf{A}$ and $\mathbf{A}\mathbf{A}^{\mathrm{T}}$, respectively). Hence (for example) the neighbors of a node i, viewed as a Hub, are those nodes which are linked to the node i by one application of $\mathbf{A}\mathbf{A}^{\mathrm{T}}$. In other words, these two compound operators (which are each symmetric) define two different undirected graphs: the Hub graph and the Authority graph. Our suggestion is then to apply our approach to each of these undirected graphs.

An interesting unanswered question is then how one can define a single, unique decomposition of a directed graph. It is not clear to us how to do so. Furthermore, it is not clear whether such a goal is more or less advantageous than the two-decomposition approach sketched here for directed graphs. One possible approach is to use the PageRank algorithm [11], which gives a single, authority-like score for each node on a directed graph, without resorting to compound operators. However, one then still needs a good definition of a node’s neighbors.

Finally we come to applications of our method. We have displayed one application here, in our analysis of the Gnutella network of late 2001. We believe that our picture of this peer-to-peer network offers valuable insight, which is not available from other approaches such as the node degree distribution. The principal insight is that the network—or at least the subgraph found in the snapshots—is so well connected that it is ‘barely’ decomposable into regions at all: some snapshots consist of a single region, while others consist of two regions which are well connected to one another. It would be highly interesting to study more recent, and possibly more complete, snapshots of peer-to-peer networks, in order to test if this picture still holds. Peer-to-peer networks are but one example of many self-organized social/technological networks, which manage without central guidance to build up highly robust and well connected structures.

Our approach has also been implemented, in an earlier and limited form, in the Archipelago [13] software for analyzing the bipartite graphs composed of users and files on a computer network. Here the analysis promises to be useful for the purpose of security management. Clearly, both highly central nodes, and bridges (links or nodes) can be singled out as deserving extra attention and care in the preventing of the spread of damage. The highly central nodes are most likely to infect their regions; and the bridges in turn must be guarded so that the infection does not spread from one region to others. There is, for example, only a small number of bridging elements in Fig. 3.5, or Fig. 3.6. Hence it would be practical to immunize these elements, and so ensure that any infection is isolated to a single region. For larger regions, it would also be practical to immunize the most central nodes in each region—prioritizing of course those regions with the greatest number of nodes. The Gnutella examples, on the other hand, are hard to protect, because they are *too* well connected. In our language, this means that there are many nodes in each region with roughly the same centrality; and that there are many bridges between regions (for those cases where there is more than one region).

Our method is applicable to many other types of graphs—in principle, to any graph which is undirected. The method is easily modified also to allow weights (other than 0 or 1) for the links between nodes. We believe that our method will prove to be useful in the analysis of *social networks*—which may (again) have a (positive) strength associated with each link.

Acknowledgements

We thank Mark Burgess and Michelle Girvan for helpful discussions, Mark Burgess and Tuva H. Stang for generating the pictures of the Gnutella networks, and Ken Stephenson for helpful discussions of discrete topography and of Dolly Parton. We acknowledge partial support by the Future & Emerging Technologies unit of the European Commission through Project BISON (IST-2001-38923).

References

[1] G.D. Batista, P. Eades, R. Tamassia, I.G. Tollis, Graph Drawing: Algorithms for the Visualization of Graphs, Prentice Hall, Upper Saddle River, New Jersey, 1999.

[2] P. Bonacich, Factoring and weighting approaches to status scores and clique identification, Journal of Mathematical Sociology 2 (1972) 113–120.

[3] M. Burgess, G. Canright, K. Engø, A graph theoretical model of computer security: from file access to social engineering, International Journal of Information Security (2003) (accepted for publication).

[4] G. Canright, K. Engø-Monsen, Å. Weltzien, Multiplex structure of the communications network in a small working group, Social Networks—An International Journal of Structural Analysis (2003) (submitted for publication).

[5] M. Girvan, M. Newman, Community structure in social and biological networks, Proceedings of the National Academy of Sciences of USA 99 (2002) 8271–8276.

[6] M. Jovanovic, private communication.

[7] J.M. Kleinberg, Authoritative sources in a hyperlinked environment, Journal of the ACM 46 (1999) 604–632.

[8] M. Newman, The structure and function of complex networks, SIAM Review 45 (2003) 167–256.

[9] A.Y. Ng, A.X. Zheng, M.I. Jordan, Stable algorithms for link analysis, in: Proc. 24th Annual Intl. ACM SIGIR Conference, ACM, 2001.

[10] A. Oram (Ed.), Peer-to-peer, Harnessing the Power of Disruptive Technologies, O'Reilly, Sebastopol, California, 2001.

[11] L. Page, S. Brin, R. Motwani, T. Winograd, The pagerank citation ranking: Bringing order to the web, Technical Report, Stanford Digital Library Technologies Project, 1998.

[12] R. Pastor-Satorras, A. Vespignani, Epidemic spreading in scale-free networks, Physical Review Letters 86 (2001) 3200–3203.

[13] T.H. Stang, F. Pourbayat, M. Burgess, G. Canright, K. Engø, Å. Weltzien, Archipelago: A network security analysis tool, in: Proceedings of the 17th Annual Large Installation Systems Administration Conference, LISA 2003, San Diego, CA, USA, October, 2003.