Error and attack tolerance of layered complex networks

Maciej Kurant and Patrick Thiran
Ecole Polytechnique Fédérale de Lausanne (EPFL), CH-1015, Lausanne, Switzerland

Patric Hagmann
Department of Radiology, Lausanne University Hospital (CHUV), Rue du Bugnon, 46, CH-1011 Lausanne, Switzerland

(Received 21 January 2007; published 9 August 2007)

Many complex systems may be described by not one but a number of complex networks mapped on each other in a multi-layer structure. Because of the interactions and dependencies between these layers, the state of a single layer does not necessarily reflect well the state of the entire system. In this paper we study the robustness of five examples of two-layer complex systems: three real-life data sets in the fields of communication (the Internet), transportation (the European railway system), and biology (the human brain), and two models based on random graphs. In order to cover the whole range of features specific to these systems, we focus on two extreme policies of system's response to failures, no rerouting and full rerouting. Our main finding is that multi-layer systems are much more vulnerable to errors and intentional attacks than they appear from a single layer perspective.

DOI: 10.1103/PhysRevE.76.026103 PACS number(s): 89.75.Hc, 89.20.Hh, 89.40.Bb, 89.75.Fb

I. INTRODUCTION

The robustness of a complex system can be defined by its behavior under stress. There are two general categories of such stress: *errors*–failures of randomly chosen components, and *attacks*–failures of components that play a vital role in the system. Recently, many complex systems have been successfully described in terms of complex networks [1]. These graphs may differ greatly in their response to failures. For instance, "scale-free" networks (i.e., networks whose node degree distribution is heavy-tailed [2]), such as the World Wide Web, Internet, protein networks, ecological networks, or cellular networks, exhibit remarkable robustness to errors but, at the same time, they are very vulnerable to attacks such as the removal of the most highly connected nodes [3–6]. Subsequent studies of other attack strategies [7,8], cascading failures [9,10], defensive strategies [9,11–14], and vulnerability of weighted networks [15] gave us valuable insight into the robustness of complex networks treated as distinct objects. Many such networks, however, are only a part of larger systems, where a number of coexisting topologies interact and depend on each other [16]. For example, in the Internet, a graph formed by an application (such as WWW or peer-to-peer) is mapped onto the IP network that is, in turn, mapped onto a physical mesh of cables and optical fibers. The topology at each layer is different. Similarly, it is convenient to view a transportation network as a two-layer system, with a network of traffic demands mapped onto the physical infrastructure. This layered view sheds new light on the tolerance to errors and attacks of many complex systems. In this paper we show that what is observed at a single layer does not necessarily reflect well the state of the entire system. On the contrary, a tiny, seemingly harmless (from one-layer perspective) disruption of the lower layer graph may destroy a substantial part of the upper layer graph making the whole system useless in practice.

A framework for the analysis of layered complex networks was recently introduced in Ref. [16]. In a two-layer case, the system consists of a weighted logical graph, $G^{\lambda}=(V^{\lambda},E^{\lambda})$, and the underlying physical graph, $G^{\phi}=(V^{\phi},E^{\phi})$. The logical nodes, are a subset of physical nodes, $V^{\lambda}\subset V^{\phi}$. Every logical edge, $e^{\lambda}=(u^{\lambda},v^{\lambda})$, is mapped on the physical graph as a physical path, $M(e^{\lambda})$, connecting the nodes u^{ϕ} and v^{ϕ}, corresponding to u^{λ} and v^{λ}.

This layered framework allows us to study the robustness of the entire system. Because logical edges are mapped as physical paths that are usually longer than one hop, many physical links serve more than one logical edge (see Fig. 1). The failure of such a physical link affects all logical edges that are mapped on it. In other words, failures at the physical layer propagate to the logical layer and multiply. Moreover, the resulting failures at the logical layer are strongly correlated in time and space. These three features make the response of a layered system to failures much more complex than what is observed at a single layer.

II. DATA SETS

In order to gain more insight into the problem, we study the behavior under stress of three examples of large layered systems in fields as different as transportation, communication, and biology. In addition, we consider two artificial systems based on classic and power-law random graphs. Below we present an overview of these five data sets in Table I and describe each of them.

A. Railway

Our first data set, called Railway, is the European railway system. It is extracted from timetables of 60 775 trains in central Europe with the algorithm described in Ref. [17]. The resulting physical graph reflects the real infrastructure that consists of 4853 nodes (stations) and 5765 edges (rail tracks). The logical graph contains 7038 edges, each connecting the first and the last station of a train. The logical edge weight is the number of trains following the same route.

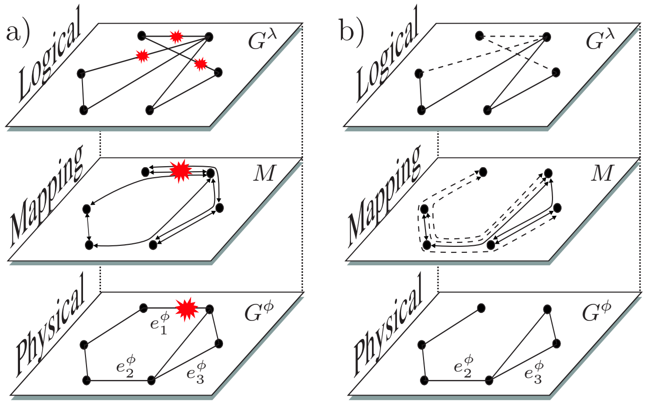

FIG. 1. (Color online) (a) Illustration of failure propagation, multiplication, and correlation in a two-layer system. A single failure in the physical graph results in three correlated failures in the logical graph. (b) The system after a failure of e_1^{ϕ}. The dashed lines are valid only under the "full rerouting" scenario.

The route itself is the mapping of this edge on the physical graph.

B. Gnutella

The second data set, called Gnutella, is an example of a large peer-to-peer (P2P) application in the Internet. In a P2P system, the links between users are virtual and are usually created independently of the underlying Internet structure, thus forming a very different topology. Due to its immense size and dynamics, the existing maps of the Internet at the IP level (i.e., where the nodes and IP routers are hosts) are very incomplete. Therefore, we focus on its aggregated version, where each node is an autonomous system (AS–usually an Internet service provider), and where edges reflect the connections between the ASes. The topology of the AS-level Internet is well known thanks to numerous Internet mapping projects such as DIMES [18] and CAIDA [19]. For our physical graph we take the 09/2004 topology provided by CAIDA, which consists of 16 911 nodes and 37 849 edges. For the logical graph we take a snapshot of the Gnutella P2P network collected in September 2004 by the crawler developed in Ref. [20]. It consists of around one million users, connected by several million links. In order to obtain the AS-level version of this network, we translated the IP addresses of the users into the corresponding AS numbers. All users with the same AS number are grouped in a single node of the logical graph, and all links connecting the same pair of ASes become one logical edge of weight equal to the number of contributing links. As a result, we obtain an AS-level logical graph of Gnutella with 1214 nodes and 31 193 edges. The mapping of each logical edge is defined by the shortest path in the physical graph connecting its end-nodes.

C. Brain

Our third data set, called Brain, is a millimetric scale map of the structural connectivity of the entire human brain. It was inferred from a diffusion magnetic resonance imaging (MRI) scan with the approach described in Ref. [21]. This methodology partitions the brain gray and white matter into a set of compact regions of comparable size. There are 1013 regions in the gray matter and 3432 regions in the white matter. Every region becomes a node in the physical graph (i.e., $|V^{\phi}|=4445$) and every gray matter region becomes a node in the logical graph. The logical edges, E^{λ}, in this data set reflect the fiber tracts (bundles of axons) connecting different gray matter regions. Each such tract, e^{λ}, traverses the white matter; the sequence of white matter regions on its path defines the mapping, $M(e^{\lambda})$. At the physical layer, two nodes are connected by a physical edge, e^{ϕ}, if they appear to be directly connected (i.e., they are consecutive in the sequence of regions) in at least one mapping, $M(e^{\lambda})$. By this procedure, we obtain a two-layer structure, where the logical graph consists of the gray matter to gray matter connections in the brain and is mapped on the physical layer that reflects the axonal wiring used to establish these long-range connections.

D. Two artificial models: ER on ER and BA on ER

As a reference point, we also study two artificial systems. ER on ER is the classic unweighted Erdös-Rényi (ER) random graph on top of another ER graph of the same number of nodes. We consider only the largest connected components of these graphs. The nodes in the logical and the physical layers are randomly paired. BA on ER is constructed in the same way, except that the logical graph is now the Barabási-Albert (BA) power-law random graph [2].

III. NO REROUTING VERSUS FULL REROUTING POLICY

Many real-life systems have mechanisms to partially or fully recover from failures. For instance, the Internet consists of several (seven) layers that are specified in the ISO/OSI network model [22] (in practical implementations usually not all the layers can be distinguished). Some of these layers, e.g., the network layer with the IP protocol, attempt to find an alternative path around a failing link or node. This requires, among other things, the physical graph to be connected. Finding an appropriate detour might be more difficult

TABLE I. Two-layer systems analyzed in this article: Railway–train traffic flows on top of the railway network of central Europe; Gnutella–Gnutella P2P network on top of the AS level Internet; Brain–long distance gray matter to gray matter axonal connections in the human brain on top of the 3D lattice covering the white matter; ER on ER–classic Erdös-Rényi (ER) random graph on top of another ER graph; BA on ER–Barabási-Albert (BA) power-law random graph on top of the ER graph. $\langle l\rangle$ is the average shortest path length; $\langle m\rangle$ is the average mapping length.

Data set	$\|V^{\phi}\|$	$\|E^{\phi}\|$	$\langle l\rangle$	$\|V^{\lambda}\|$	$\|E^{\lambda}\|$	$\langle m\rangle$
Railway	4853	5765	53.8	2509	7038	9.9
Gnutella	16 911	37 849	3.7	1214	31 193	2.8
Brain	4445	20 967	9.1	1013	15 369	10.3
ER on ER	2000	4000	5.7	2000	10 000	5.7
BA on ER	2000	4000	5.7	2000	10 000	5.7

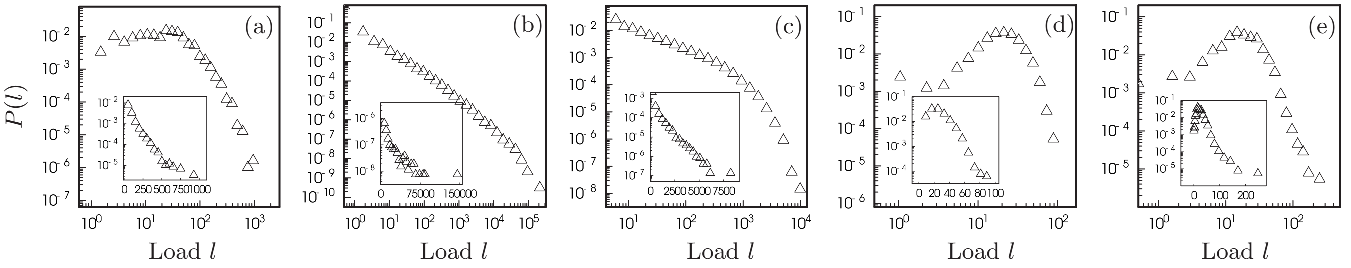

FIG. 2. Edge load distribution in the five studied systems: Railway (a), Brain (b), Gnutella (c), ER on ER (d), and BA on ER (e). The main plots are in $\log_{10}$–$\log_{10}$ scale ($\log_{10}$-binned); the insets present the same distributions in $\log_{10}$-linear scale (linear-binned).

in railway networks because, for a train, its entire path is important, not only the end-points. Although it is sometimes possible to slightly change the itinerary of the train or to organize alternative means of transportation (e.g., a bus) around the failing section, the common practice is to halt all the trains that use it. In order to keep our analysis general and to cover the whole spectrum of possible situations, in this paper we study two extreme policies: no rerouting and full rerouting. In the former case we immediately delete all logical edges affected by a physical failure. In the latter case, we delete an affected logical edge, e^{λ}, only when there is no path in the physical graph, G^{ϕ}, between the end-nodes of e^{λ} (i.e., end-nodes of e^{λ} belong to different components of G^{ϕ}). Otherwise, the logical edge, e^{λ}, remains in the graph, and its mapping is updated by the shortest path in G^{ϕ}. Consider the example in Fig. 1. Under the no rerouting policy, three logical edges are removed after the failure of e_1^{ϕ} but as the physical graph G^{ϕ} is still connected, under the full rerouting policy these three logical links can be rerouted and thus remain in the logical graph.

By studying the two extreme policies, no rerouting and full rerouting, we also capture the specific features of our three real-life data sets. For instance, in the railway system each rail track has a limited capacity that cannot be exceeded. Therefore, even if we allow for rerouting, some routes will be forbidden due to a possible overload. In the Gnutella data set, the AS graph routing depends on the internal policy of involved ASes and peering relationships established between the ASes [23]. This results in routes that are not necessarily the shortest possible and makes some of the routes invalid. These additional constraints imposed on the Railway and Gnutella paths naturally limit below the full rerouting level of the performance of these systems. Finally, the brain also has some ability to reroute around broken connections by activating parallel pathways; this is called plasticity. For example, after a stroke in primary or secondary motor cortices, some limb functions can be recovered in the animal as well as in the human by recruiting alternative pathways [24,25]. However, these processes take substantial time. Therefore, the brain response can be described as moving from a no rerouting policy just after the insult to a partial rerouting policy during the recovery process.

In other words, all responses of real systems to physical failures are located somewhere between the no rerouting and the full rerouting policy.

IV. EDGE LOAD DISTRIBUTION

Before we simulate the effect of failures on our systems directly, we try to roughly predict what will happen by studying related distributions. In a layered system, every physical node or edge can be characterized by the load. The load, l, of the physical node, v^{ϕ}, or edge, e^{ϕ}, is the sum of weights of all the logical edges whose paths traverse v^{ϕ} (respectively, e^{ϕ}) [16]. The load becomes a very important parameter when we allow for failures in the system. Clearly, the higher the load of a failing physical component, the more it perturbs the logical layer. If the load is distributed evenly in the physical graph, a random failure will not be very different from an intentional attack. Conversely, if the load distribution is very uneven, the highly loaded parts become an obvious target for an efficient attack. In Fig. 2 we present the load distribution in the layered systems we study. In each case the distribution is broad and heavily right-skewed (except perhaps ER on ER). This means that there is a significant number of physical links that carry much more traffic than the other links. Consequently, we can anticipate that an attack targeted on the most loaded links will harm the system much more efficiently than a random error.

V. SIMULATION RESULTS

In this section we simulate the error and attack scenarios on the five studied systems. The results are presented in Fig. 3. Although the system responses differ in all five cases, they share a number of common features:

(1) *Attacks are much more harmful than errors*. For example, in Gnutella with no rerouting, half of the logical mass (total edge weight) is erased after 22% physical edges randomly fail, or after only 0.04% most loaded edges are attacked. Although under the full rerouting policy this difference is smaller, we still need about 60 times more random failures than attacks to achieve the same goal.

(2) *When the system is attacked, the logical graph is usually affected much faster than the physical graph*. For instance, in Gnutella, an attack (with or without rerouting) on 5% of the physical edges hardly affects the physical graph—the largest connected physical component covers almost the entire original graph. At the same time, this seemingly unharmful attack deletes more than 95% of logical edges! We obtain similar results when we consider the size of the largest connected component in the logical graph as the measure of robustness. (These results are not shown in Fig. 3 for better readability.)

(3) *The attack under the full rerouting policy affects the physical graph more than under no rerouting*. When rerouting is allowed, the logical edges are deleted only when the

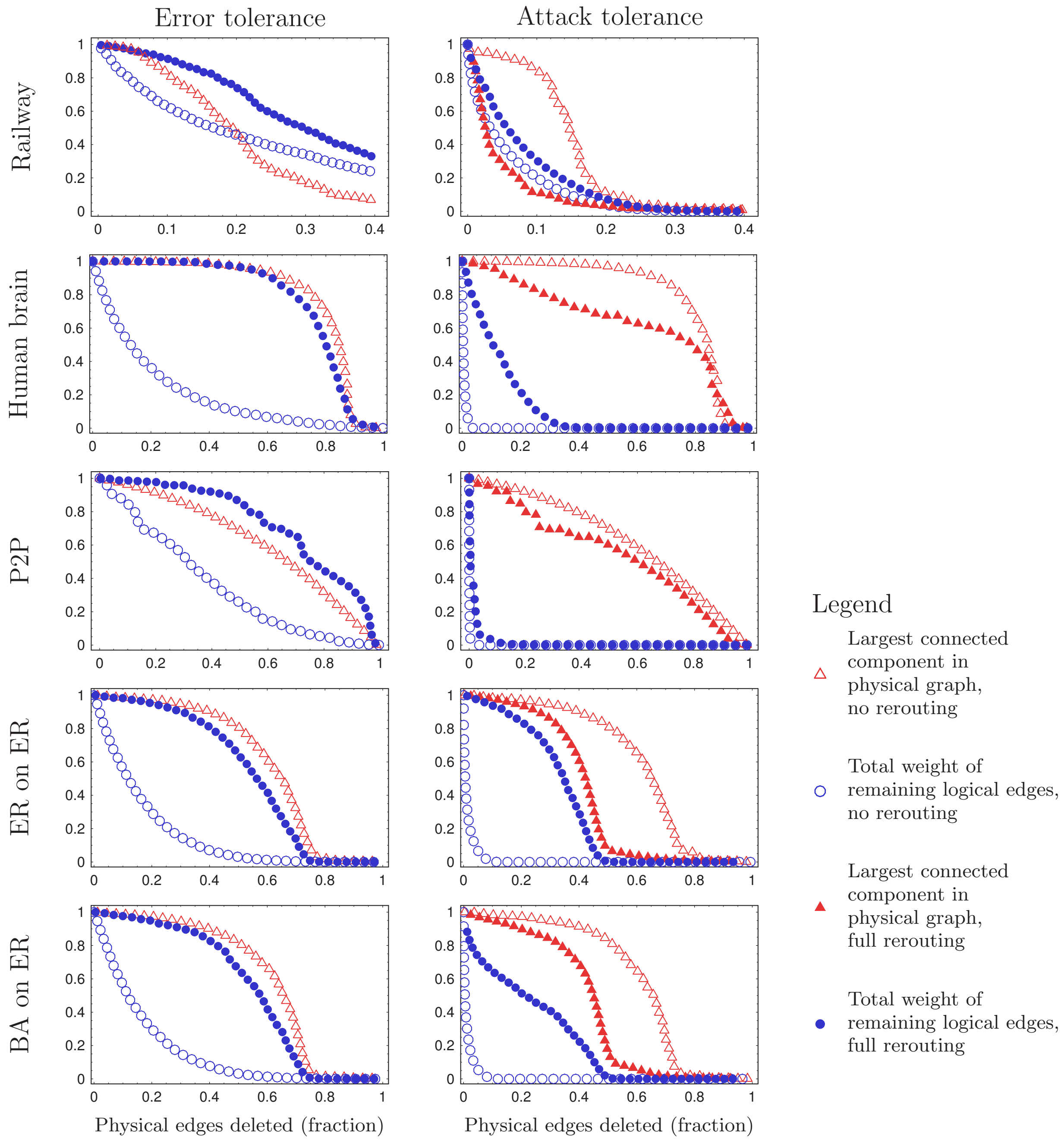

FIG. 3. (Color online) Error and attack tolerance of five layered systems (rows). The first three are real-life data sets; the last two are based on classic ER and power-law BA graphs. At each iteration we remove one physical edge, e^{ϕ}_{del}, either at random (error tolerance, left column), or by choosing the most loaded one (attack tolerance, right column). In both cases we observe the size of the largest connected component in the physical graph, G^{ϕ} (triangles), and the total weight of the remaining logical edges, (circles). Every logical edge, e^{λ}, whose mapping contains e^{ϕ}_{del} is deleted either directly (no rerouting, unfilled symbols), or only when there is no path in G^{ϕ} between the end-nodes of e^{λ} (full rerouting, filled symbols). The results for ER on ER and BA on ER are averaged over ten realizations. Note that, in the case of random errors, the largest connected physical component is not affected by the adopted policy (no-rerouting or full-rerouting). Therefore the filled- and empty-triangle curves coincide for all figures in the left column. We draw only the empty-triangle curves for clarity.

physical graph gets partitioned. This, in turn, effectively reduces the size of the largest connected physical component. This behavior can be explained by the example in Fig. 1. Initially, the physical edge e^{ϕ}_1 is used by three logical links. It is the most loaded edge in the physical graph and, hence, it is the first one removed by our attack. Now, under the no rerouting policy, three logical edges are deleted. In what remains, the most loaded physical edge is e^{ϕ}_3. This edge is removed in the second round of the attack, keeping the physical graph connected. In contrast, under the full rerouting policy, after the removal of e^{ϕ}_1 the three affected logical links are rerouted [see Fig. 1(b)]. Assuming that the new routes follow shortest paths, the physical edge e^{ϕ}_2 becomes the most loaded and is removed in the second round of the attack. This efficiently splits the physical graph into two components of three nodes each [27].

(4) *Rerouting does not always help much*. This is expressed by the proximity of the filled and unfilled circles under attack in Fig. 3 (see, e.g., Railway and Gnutella). As any real-life failure recovery policy falls between these two extremes (no rerouting and full rerouting), such systems are especially vulnerable to attacks.

(5) *A heterogeneous logical topology makes the system more vulnerable to attacks*. This can be observed in the two random-graph-based examples. As the node degree distribution of the BA graph follows a power-law, there is no typical node (or 'scale') and, hence, BA on ER is a system with a heterogeneous logical topology. In other words, there is a non-negligible probability of existence of hubs (nodes of a very high degree) in the BA graph. As we assume no correlation between the logical and physical node degrees, in the vicinity of such logical hubs the load of physical edges is usually high. This makes the BA on ER system vulnerable to attacks, which is reflected by the fast initial drop of both circle curves in the last subfigure in Fig. 3. In contrast, the degree distribution of the ER graph is concentrated around the average value making it much more homogeneous. So there are no hubs in the logical graph of ER on ER and the load is distributed more evenly. This can be observed in Fig. 2—with the same number of nodes and edges, the maximal load in BA on ER is roughly three times higher than in ER on ER. Consequently, ER on ER is more robust to attacks than BA on ER.

To conclude, the response of a multi-layer system to failures is much more complex than what is observed at a single layer. In particular, the logical layer is affected by physical attacks much faster than the physical layer is. This is very important because what ultimately counts in a multi-layer system is the upper-most (i.e., logical) layer; it directly reflects the service provided by the system to its users, such as trains, P2P application, and the long distance connections in the brain.

This work is only the first step towards understanding the behavior of layered systems under stress. There are numerous aspects that require further investigation: What is the effect of traffic locality, weight and load distribution, failure correlation, or topological properties at the two layers on the robustness of the system? Do there exist attacks even more efficient than the one proposed in this paper? Is it possible to significantly improve the resilience of a system, e.g., by adding a relatively small number of physical or logical edges? We are planning to address these issues in our future work.

[1] M. Newman, A.-L. Barabasi, and D. J. Watts, *The Structure and Dynamics of Networks* (Princeton University Press, Princeton, 2006).

[2] A. Barabási and R. Albert, Science **286**, 509 (1999).

[3] R. Albert, H. Jeong, and A.-L. Barabási, Nature **406**, 378 (2000).

[4] R. Cohen, K. Erez, D. ben-Avraham, and S. Havlin, Phys. Rev. Lett. **85**, 4626 (2000).

[5] R. Cohen, K. Erez, D. ben-Avraham, and S. Havlin, Phys. Rev. Lett. **86**, 3682 (2001).

[6] D. S. Callaway, M. E. J. Newman, S. H. Strogatz, and D. J. Watts, Phys. Rev. Lett. **85**, 5468 (2000).

[7] P. Holme and B. J. Kim, Phys. Rev. E **65**, 066109 (2002).

[8] L. K. Gallos, R. Cohen, P. Argyrakis, A. Bunde, and S. Havlin, Phys. Rev. Lett. **94**, 188701 (2005).

[9] A. E. Motter, Phys. Rev. Lett. **93**, 098701 (2004).

[10] L. Zhao, K. Park, and Y.-C. Lai, Phys. Rev. E **70**, 035101(R) (2004).

[11] L. da Fontoura Costa, Phys. Rev. E **69**, 066127 (2004).

[12] T. Tanizawa, G. Paul, R. Cohen, S. Halvin, and H. E. Stanley, Phys. Rev. E **71**, 047101 (2005).

[13] V. Latora and M. Marchiori, Phys. Rev. E **71**, 015103(R) (2005).

[14] M. Schäfer, J. Scholz, and M. Greiner, Phys. Rev. Lett. **96**, 108701 (2006).

[15] L. Dall'Asta, A. Barrat, M. Barthélemy, and A. Vespignani, eprint arXiv:physics/0603163 (2006).

[16] M. Kurant and P. Thiran, Phys. Rev. Lett. **96**, 138701 (2006).

[17] M. Kurant and P. Thiran, Phys. Rev. E **74**, 036114 (2006).

[18] http://www.netdimes.org

[19] http://www.caida.org/

[20] D. Stutzbach, R. Rejaie, and S. Sen, Proc. of IMC'05 (2005).

[21] P. Hagmann, M. Kurant, X. Gigandent, P. Thiran, V. J. Weeden, R. Meuli, and J. P. Thiran, *Mapping Human Whole-Brain Structural Networks with Diffusion MRI*, PloS ONE, 2007 Jul 4;2:e597.

[22] J. F. Kurose and K. W. Ross, *Computer Networking* (Addison Wesley, Boston, 2003).

[23] L. Gao, IEEE/ACM Transactions on Networking **9**, 733 (2001).

[24] R. Dijkhuizen *et al.*, Proc. Natl. Acad. Sci. U.S.A. **98**, 12766 (2001).

[25] R. Marshall *et al.*, Stroke **31**, 656 (2000).

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

[27] This phenomenon is similar in spirit to the clustering algorithm proposed by Newman [26]. There, at every iteration, the edge with the highest betweenness is deleted. (The *betweenness* of a vertex, or an edge is the fraction of shortest paths between all pairs of vertices in a network, that pass through it.) This results in physical graph partitions that correspond to its clusters (or communities). It can be viewed as a special case of our attack, i.e., assuming the logical topology a fully connected unweighted graph. However, as the real-life traffic patterns are much more heterogenous [16], the attack under full rerouting produces partitions that correspond to the high traffic cut-sets in the physical graph, rather than to communities.