Community Cut-off Attack on Malicious Networks

Iveta Dirgová Luptáková(✉) and Jiří Pospíchal

Faculty of Natural Sciences, University of SS. Cyril and Methodius in Trnava, 917 01 Trnava, Slovak Republic
iveta.dirgova@ucm.sk, jiri.pospichal@gmail.com

Abstract. This paper aims to provide an efficient algorithm for quick disabling of known malicious network by sequential removal or incapacitation of their nodes. The nodes are selected for deletion in such a sequence, that the network is swiftly separated into small disjoined parts. We propose using a community detection based on random walks. For all the divisions of the found communities into two separate sets we create bigraphs defined by the edge set with each edge's node in different community and use Koenig's theorem to find the best vertex cut (set of vertices to be deleted). This community detection and their separation is used recursively on a currently maximal component of the network. The effectiveness of our algorithm is tested on both real-world and model networks by quantifying network robustness measure R based on the size of maximum component. Its results compare favorably against standard centrality based attack strategies.

Keywords: Complex networks · Attack strategy · Community detection · Minimum vertex cut · Bigraph · Koenig's theorem · Betweenness

1 Introduction

Social, biological or technological networks and networking is ordinarily regarded desirable or at worst neutral, as an effective way to communicate and distribute knowledge or resources [2, 25]. Typically, attacks on networks have been studied from the defender's point of view, where the goal was to design a topology of network in such a way that would ensure its robustness against either random failures or intentional attacks [1, 18].

However, there are many cases, where the goal of a network is malevolent or its result unintentionally harmful. As examples can serve terrorist networks, malevolent fake-news spreading networks, the spread of disease, cascading failures in power networks, malnets used in DDoS attack [22, 32, 33] or spreading worms and viruses, dark nets distributing child pornography or involved in other criminal activities, etc. [7, 8, 10–12, 26, 38].

To study such networks in general, their structure is narrowed to graphs composed of nodes and edges. The only difference from an ordinary graph is then in their topological properties. Unlike ordinary graphs, the real world networks tend to have a

© Springer International Publishing AG 2017
A. Kravets et al. (Eds.): CIT&DS 2017, CCIS 754, pp. 697–708, 2017.
DOI: 10.1007/978-3-319-65551-2_50

few nodes with a very high degree (the degree distribution has a "heavy tail"), a high clustering coefficient expressing the fact that neighbors of a node are likely connected themselves, and a community structure [2, 25].

In order to shut down or incapacitate such networks, one can formally delete edges or nodes, sequentially or at once. In graph theory, removing a smallest possible set of nodes at once to split the graph roughly into halves is called balanced minimum vertex cut problem (also known as the vertex separator problem), it belongs to graph partitioning problems and it is NP-hard [14]. Another approach related to graph partitioning is community detection, but while in graph partitioning the number of partitions is set in advance, in most community detection algorithms the number and the size of the communities is determined from network's topology [2].

In our case, we shall consider sequential deletion of nodes. There exist various strategies for selecting nodes for deletion, mainly based on local centrality measures like vertex degree or its betweenness. Sequences of vertices with decreasing values of degrees or betweenness [15] can be calculated at once for the complete network (initial degree/betweenness distribution), or recalculated iteratively when the vertex with maximum value is deleted. The iterative strategies (sequence targeted attack or recalculated centrality) for degree/betweenness (RD/RB) require more computational resources, but provide better results [19], therefore they will be used as reference attack strategies.

Measures, how to calculate the incapacitation of the network, are mainly based on average length of paths between vertices, degrees of remaining nodes or size of connected components of the network [13, 36]. In our paper, we use network robustness measure R based on the size of a maximum component after each node deletion for quantifying the attack effectiveness of the whole sequence [17, 31, 34, 35], defined as:

$$R = \frac{1}{N+1} \sum_{Q=0}^{N} S(Q) \tag{1}$$

where N is the initial number of nodes in the network, $S(Q)$ is the fraction of nodes in the largest connected component after deletion of $Q = Nq$ nodes and q is a current fraction of deleted nodes vs. N.

At the core of our attack strategy is Module-Based Attack, its idea originating from [16], later developed by [9] and further by [30]. We have adapted their strategy by merging the found communities into such two supersets, which have the best ratio of size of the removed superset against the number of removed bridges, which we recalculate iteratively.

There are many methods for community detection, based on various principles. Edge betweenness uses the number of shortest paths through it in [23]. Fast greedy uses local optimization in [6]. Node labels are updated by majority voting in the neighborhood of the node in [28]. The leading non-negative eigenvector of the modularity matrix is calculated by [24]. The modularity measure and a hierarchical approach are used in Louvain method [4]. The modularity maximization is transformed into an integer programming problem in [5]. Spin-glass model and simulated annealing is used by [29], and random walks are utilized in "walktrap" [27]. We have tried all the methods with the equally good best results provided by edge betweenness [23], optimal [5], spinglass [29]

and walktrap [27]. In further calculations, we shall use a community detection method walktrap with reasonable robustness and runtime complexity [39].

2 Community Cut-off Attack Strategy

The module- or community-based attack strategy, which was studied by da Cunha [30], was defined as partially simultaneous attack, which first finds nodes in intercommunity connections and then removes them. The selection of the nodes to be removed is determined only on the bases of their betweenness and on sequential removal of the intercommunity node, whose counterparts on the other side were already removed. The strategy, according to [30], is as follows:

1. Extract communities using a heuristic detection algorithm.
2. Make a list of the nodes that participate in intercommunity connections.
3. Sort the list in descending betweenness centrality order.
4. Delete nodes one by one, starting from the first in the list.
5. When a node from a link between two communities is deleted, its counterpart is removed from the list, unless it also participates in other intercommunity connections.
6. The attack is always restricted to the largest connected component of the network. In other words, if at some point the next node in the list does not belong to the remaining largest connected component that node is skipped.

While this strategy is effective, sometimes it is not optimal. To illustrate the problem, due to a scarcity of benchmark malicious networks, we shall show it on a network of the "contiguous USA graph". Its nodes represent the contiguous 48 states of the United States plus the District of Columbia (DC) and its edges connect pairs of states (plus DC) that are connected by at least one drivable road [20, 37], see Fig. 1. The nodes representing states are labeled by their postal abbreviations and clustered into 7 communities by the walktrap algorithm [27], the clusters are shown by the colored regions. The community attack specified above would start to attack the nodes MO, PA, KY, TN, NY, NE, …. In the lower part of the figure are betweenness values of the nodes. The deletion would start with nodes of values 403, 328, 321, 256, 253, 180,.. so after 6 deletions only nodes VT, NH, ME, MA, CT, RI would be cut off. The size of the largest component would be 4 times repeatedly diminished by 1, only the deletion of the fifth node NY would cut-of 6 nodes. In minimization of R value, we are trying to maximize the following ratio:

$$\frac{\text{number_of_cut} - \text{off_nodes} + \text{number_of_deleted_nodes}}{\text{number_of_deleted_nodes}} \tag{2}$$

In this case, the ratio is (6 + 6)/6 = 2, which means, that in average for one deleted node we cut of one other node from the largest component. After deletion of 6 nodes, the size of the largest connected component is 49−12 = 37. If we look at the network, just by deleting NY node marked by red framed square we can cut off 6 other nodes, VT, NH, ME, MA, CT, and RI, which gives the ratio 7. By further deletion of nodes

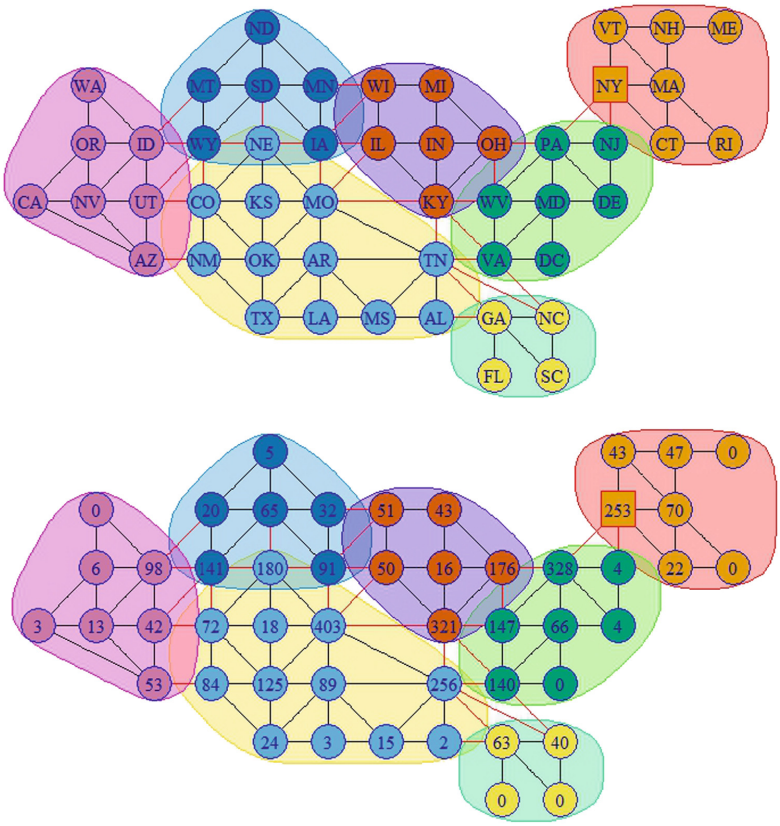

Fig. 1. Contiguous USA neighboring states, clustered by walktrap algorithm. The lower figure shows betweenness values of nodes.

MN, IA, MO, TN and AL we can cut of further 16 nodes, with a ratio (16 + 5)/5 = 4.2. After deletion of just 6 nodes, the size of the largest connected component is only 49−(6 + 16 + 6) = 21. The algorithm of [30] could therefore certainly be improved. The Fig. 2 shows the partial results after deletion of only 6 nodes, on the left for the algorithm [30] resulting in the largest component with 37 nodes, on the right improvement with the largest component of 19 nodes (improved even comparing to original community cut off by iterative recalculating of communities after deletion of each node).

However, this brings us to the question, how to decide, which communities should be cut-off from the rest and which of the nodes incident with intercommunity edges should be deleted?

When the number of communities is reasonably small, the answer is simply combinatorial enumeration. In our case, if we label the initially found 7 communities shown in Fig. 1 by A-G, we can partition the set of the communities into all possible

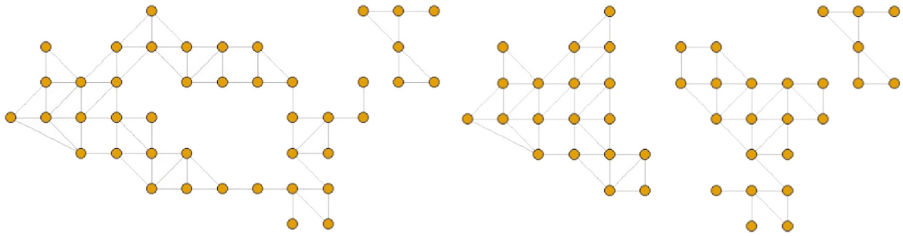

Fig. 2. Contiguous USA neighboring states, after deletion of 6 nodes. The left figure shows the result of the antecedent algorithm [30], the second result of the new one.

couples of proper subsets {{A},{B,C,D,E,F,G}}, {{B},{A,C,D,E,F,G}}, …, {{G}{A, B,C,D,E,F}}, {{A,B},{C,D,E,F,G}}, {{A,C},{B,D,E,F,G}},…, {{F,G}{A,B,C,D, E}}, {{A,B,C},{D,E,F,G}}, …, {{E,F,G},{A,B,C,D}}, which makes 63 partitions. However, from these we can omit those partitions, which do not induce connected networks. If, say, communities A and B are not connected by an edge, then there is not much point in analyzing which nodes are best to be cut off from {{A,B},{C,D,E,F,G}}, since it would be in most cases already covered by {{A},{B,C,D,E,F,G}}, {{B},{A,C, D,E,F,G}}. In our case, it will reduce the number of partitions to analyze just to 21, see Fig. 3. Occasionally, there may exist a few cases, when e.g. a node from a subset {C,D, E,F,G} could be connected both to A and B. For instance, cutting of {A} from {B,C,D, E,F,G} would require deletion of 2 nodes, cutting of {B} from {A,C,D,E,F,G} would require deletion of 2 nodes, but cutting of {A,B} from {C,D,E,F,G} would require only 3 nodes, since one in C is connected to both A and B, even though A is not connected with B. Analyzing these cases in optimization would require too much computational overload, so we choose to ignore them. Their inclusion would only slightly shift the order, in which the communities are cut off.

In order to separate two parts of a partition, we take edges with one end in each part of partition. These edges induce a bipartite graph.

Hopcroft–Karp algorithm takes as input a bipartite graph and produces as output a maximum cardinality matching – a maximum set of edges when no two edges share a node. It runs in $O\left(|E|\sqrt{N}\right)$ time in the worst case, typical time is linear. Finding nodes to delete to disconnect the bipartite graph means to find a minimum vertex cover from the matching. By application of König's theorem [3, 21] it can be done in logarithmic time. When more possible minimum vertex cover sets exist, we take the one, which has more nodes in the larger community, so that the size of resulting largest connected component is further diminished.

Compared with the algorithm [30], communities are extracted anew after each removal of a node. This does not have to be the case, when we order the gains for partitions in decreasing order and delete all the nodes of their minimum vertex covers in that order. This would substantially accelerate the computation, but only slightly increase the resulting R value. Similarly, if the number of found communities would be too large in order to analyze all the partitions, it can be artificially decreased by merging smallest most connected communities. Nevertheless, in our calculations we did not find it necessary.

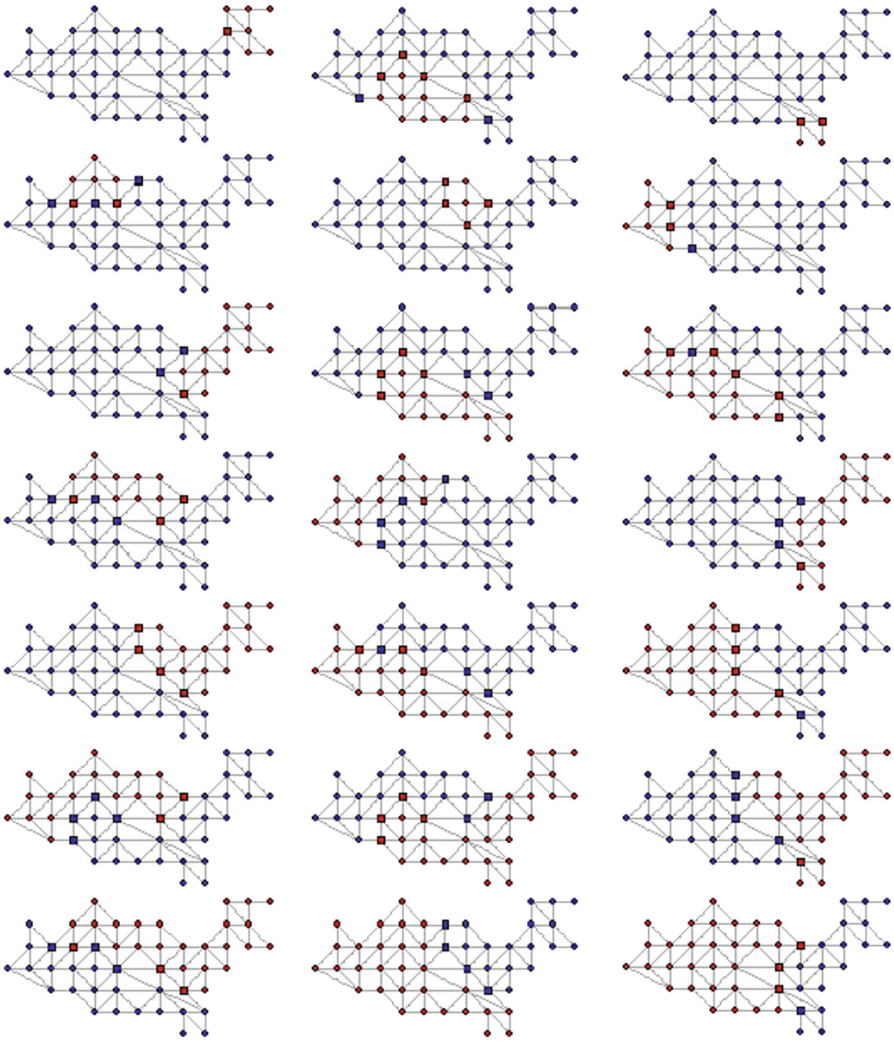

Fig. 3. Contiguous USA neighboring states, all possible partitions of communities into two subsets (when each subset induces a connected graph) colored red and blue, nodes to be deleted to separate the partitions are marked by squares. (Color figure online)

On the other hand, if we would like to get even better results of R value, in the following pseudocode we can skip the removing partitions with an unconnected graph induced by its subset of nodes, slightly improving results at a great computational cost. Further, finding communities anew after deleting each node is advantageous most of the time, but not always. The algorithm might be improved by remembering old minimum vertex covers with their gains and compare the new gains with the old ones.

The pseudocode of the adapted modular based attack strategy is as follows:

```
Program AdaptedModuleAttack:
BEGIN
INPUT network
WHILE number of edges in network>0
    Find largest connected component CC
    IF CC is tree THEN
        Delete its node with maximum betweenness
    ELSE
        Extract communities in CC using a heuristic detection algorithm.
        IF only one community is found THEN
           delete its node with maximum betweenness,
        ELSE
           Generate all possible partitions of the communities in two
           nonempty subsets.
           FOR each partition:
             IF either set the nodes from communities in the two subsets
             does not induce a connected component, THEN
                Remove the partition
             ELSE
                Define both subsets of communities as new communities, find
                edges that are incident with nodes from both communities
                Create a bipartite graph induced by these edges
                Using Hopcroft-Karp algorithm and König's theorem, find
                the minimum vertex cover for the bipartite graph, which
                has more nodes in the larger community
                Calculate gain of the partition from eq. (2), when number
                of deleted nodes equals the cardinality of the minimum
                vertex cover and the number of cut-off nodes is the number
                of nodes in the currently smaller of the two partition
                communities after eventual deletion of the minimum vertex
                cover
             ENDIF
           ENDFOR
           Find partition with maximum gain; delete the node with maximum
           degree from its minimum vertex cover.
        ENDIF
    ENDIF
    Calculate S(Q) from eq. (1)
ENDWHILE
Print network robustness measure R and plot S(Q) sequence
END
```

3 Experimental Results

The effectiveness of the adapted algorithm (AMBA - Adapted Module Based Attack) was compared against iterative strategies recalculating degree (RD) and betweenness (RB) on selected networks. Since we do not have access to the code of [30], it would not be fair to compare our adapted algorithm against our version of their code and the published results of the original module-based attack [30] are unfortunately only in graphical form. However, there is no reason why our adapted algorithm should be worse.

Apart of the contiguous USA graph described previously, we tested attack strategies to attack Barabási–Albert (BA) preferential attachment model, generated by starting with a triangle and adding each time a node together with 3 edges. Another test

Table 1. *R*-values for various combinations of attacks and networks

Network	ContiguousUSA	Barabási–Albert	Geometric random
Attack strategy	R	R	R
RD	0.2873	0.2525	0.3866
RB	0.1820	0.2294	0.2792
AMBA	0.1751	0.2431	0.2564

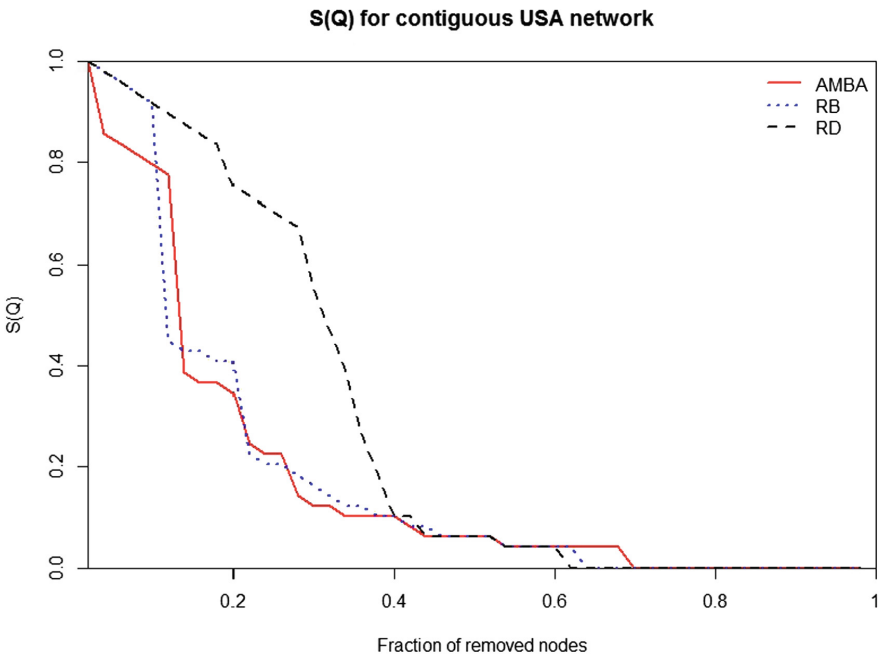

Fig. 4. S(Q) values for RD, RB and our AMBA attacks on contiguous USA neighboring states network. (Color figure online)

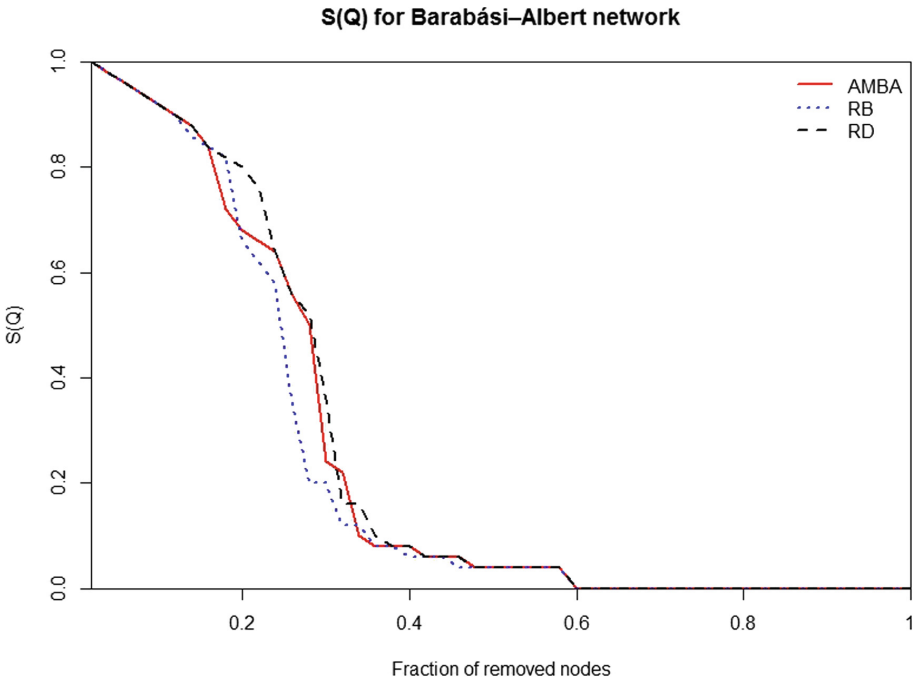

Fig. 5. S(Q) values for RD, RB and our AMBA attacks on Barabási–Albert network. (Color figure online)

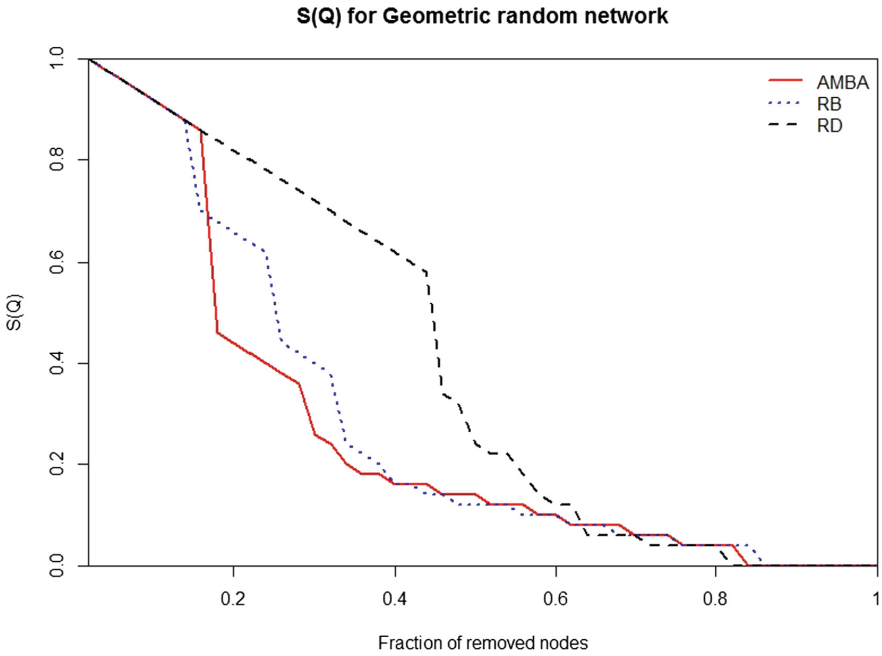

Fig. 6. S(Q) values for RD, RB and our AMBA attacks on Geometric random network. (Color figure online)

involved a geometric random (GR) network model where random points on a square will be connected with an undirected edge if they are closer to each other in Euclidean norm than a given radius (0.3 in our sample). We used randomly generated networks with 50 nodes.

Results of R values for the attack strategies and networks can be found in Table 1. The S(Q) values for all the tested strategies against the fraction q of removed nodes are shown for the contiguous USA networks in Fig. 4, and for the BA networks in Fig. 5. For the GR network the resulting S(Q) values are presented in Fig. 6.

4 Conclusions

We have adapted the module-based attack [30] and demonstrated, where and why the adapted algorithm gets better results. Figures 4, 5 and 6 show the results of our algorithm in a graphical form, compared to reference strategies RD and RB. A lower curve represents a network broken into smaller separated pieces, which corresponds to a better result. Our approach shown by a red line got better results for two of the three cases, and only for Barabási–Albert network, our results are slightly worse than RB approach but still better than RD approach. Our improvements come with a computational cost. Nevertheless, if we would bound the number of communities, finding minimum cover set of bipartite graph of vertices from the borders of communities is fast, so the increased computational cost should not increase the overall complexity of the module-based attack approach.

The choice, if to use our adapted algorithm, depends on how soon the results are needed and what type and how big is the attacked network. In case of cascading failures of a power grid or wi-fi malnet, the original algorithm providing less optimal results faster might be more suitable. In case of terrorist networks, immunization of disease, or any network based at least partially on real world Euclidean distances, which support emergence of modular architecture, our approach providing better results should justify the higher computational cost.

References

1. Albert, R., Jeong, H., Barabási, A.L.: Error and attack tolerance of complex networks. Nature **406**(6794), 378–382 (2000)
2. Barabási, A.-L., Pósfai M.: Network Science. Cambridge University Press, Cambridge (2016). http://barabasi.com/networksciencebook. Accessed 7 Apr 2017
3. Bipartite minimum vertex cover. http://tryalgo.org/en/matching/2016/08/05/konig/
4. Blondel, V.D., Guillaume, J.L., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech: Theory Exp. **2008**(10), P10008 (2008)
5. Brandes, U., Delling, D., Gaertler, M., Gorke, R., Hoefer, M., Nikoloski, Z., Wagner, D.: On modularity clustering. IEEE Trans. Knowl. Data Eng. **20**(2), 172–188 (2008)
6. Clauset, A., Newman, M.E., Moore, C.: Finding community structure in very large networks. Phys. Rev. E **70**(6), 066111 (2004)

7. Cohen, R., Erez, K., Ben-Avraham, D., Havlin, S.: Breakdown of the Internet under intentional attack. Phys. Rev. Lett. **86**(16), 3682 (2001)
8. Cohen, R., Havlin, S., Ben-Avraham, D.: Efficient immunization strategies for computer networks and populations. Phys. Rev. Lett. **91**(24), 247901 (2003)
9. De Meo, P., Ferrara, E., Fiumara, G., Provetti, A.: On Facebook, most ties are weak. Commun. ACM **57**(11), 78–84 (2014)
10. D'Orsogna, M.R., Perc, M.: Statistical physics of crime: A review. Phys. life Rev. **12**, 1–21 (2015)
11. Duijn, P.A., Kashirin, V., Sloot, P.M.: The relative ineffectiveness of criminal network disruption. Sci. Rep. **4**, 4238 (2014)
12. Duijn, P.A.C., Klerks, P.P.H.M.: Social network analysis applied to criminal networks: recent developments in Dutch law enforcement. In: Masys, A. (ed.) Networks and Network Analysis for Defence and Security. LNCS. Springer, Cham (2014). doi:10.1007/978-3-319-04147-6_6
13. Ellens, W., Kooij, R.E.: Graph measures and network robustness. arXiv preprint arXiv:1311.5064 (2013)
14. Feige, U., Hajiaghayi, M., Lee, J.R.: Improved approximation algorithms for minimum weight vertex separators. SIAM J. Comput. **38**(2), 629–657 (2008)
15. Freeman, L.: A set of measures of centrality based on betweenness. Sociometry **40**, 35–41 (1977)
16. Granovetter, M.S.: The strength of weak ties. Am. J. Sociol. **78**(6), 1360–1380 (1973)
17. Hong, C., He, N., Lordan, O., Liang, B., Yin, N.: Efficient calculation of the robustness measure for complex networks. Phys. A: Stat. Mech. its Appl. **478**, 63–68 (2017)
18. Huang, X., Gao, J., Buldyrev, S.V., Havlin, S., Stanley, H.E.: Robustness of interdependent networks under targeted attack. Phys. Rev. E **83**(6), 065101 (2011)
19. Iyer, S., Killingback, T., Sundaram, B., Wang, Z.: Attack robustness and centrality of complex networks. PLoS ONE **8**(4), e59613 (2013)
20. Knuth, D.E.: The Art of Computer Programming, Volume 4, Fascicle 0: Introduction to Combinatorial Functions and Boolean Functions. Addison-Wesley, Upper Saddle River (2008). p. 15
21. Kőnig, D.: Gráfok és mátrixok. Matematikai és Fizikai Lapok **38**, 116–119 (1931)
22. Korytar, M., Gabriska, D.: Integrated security levels and analysis of their implications to the maintenance. J. Appl. Math. Stat. Inf. **10**(2), 33–42 (2014)
23. Newman, M., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E **69**, 026113 (2004)
24. Newman, M.: Finding community structure using the eigenvectors of matrices. Phys. Rev. E **74**, 036104 (2006)
25. Newman, M.: Networks An Introduction. Oxford University Press, Cambridge (2010)
26. Pastor-Satorras, R., Vespignani, A.: Immunization of complex networks. Phys. Rev. E **65**(3), 036104 (2002)
27. Pons, P., Latapy, M.: Computing communities in large networks using random walks. In: Yolum, P., Güngör, T., Gürgen, F., Özturan, C. (eds.) International Symposium on Computer and Information Sciences, LNCS, vol. 3733, pp. 284–293. Springer, Berlin Heidelberg (2005)
28. Raghavan, U.N., Albert, R., Kumara, S.: Near linear time algorithm to detect community structures in large-scale networks. Phys. Rev. E **76**, 036106 (2007)
29. Reichardt, J., Bornholdt, S.: Statistical mechanics of community detection. Phys. Rev. E **74**, 016110 (2006)

30. da Cunha, B.R., González-Avella, J.C., Gonçalves, S.: Fast fragmentation of networks using module-based attacks. PLoS ONE **10**(11), e0142824 (2015). doi:10.1371/journal.pone.0142824
31. Schneider, C.M., Moreira, A.A., Andrade, J.S., Havlin, S., Herrmann, H.J.: Mitigation of malicious attacks on networks. Proc. Natl. Acad. Sci. **108**(10), 3838–3841 (2011)
32. Šimon, M., Huraj, L., Čerňanský, M.: Performance evaluations of IPTables firewall solutions under DDoS attacks. J. Appl. Math. Stat. Inf. **11**(2), 35–45 (2015)
33. Šimon, M., Huraj, L., Hosťovecký, M.: IPv6 network DDoS attack with P2P grid. In: Kravets, A., Shcherbakov, M., Kultsova, M., Shabalina, O. (eds.) Creativity in Intelligent Technologies and Data Science. Communications in Computer and Information Science, vol. 535. Springer, Cham (2015). doi:10.1007/978-3-319-23766-4_32. (pp. 407–415)
34. Tambouratzis, T., Souliou, D., Chalikias, M., Gregoriades, A.: Maximising accuracy and efficiency of traffic accident prediction combining information mining with computational intelligence approaches and decision trees. J. Artif. Intell. Soft Comput. Res. **4**(1), 31–42 (2014)
35. Tambouratzis, T., Souliou, D., Chalikias, M., Gregoriades, A.: Combining probabilistic neural networks and decision trees for maximally accurate and efficient accident prediction. In: The 2010 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. IEEE (2010)
36. Van Mieghem, P., Doerr, C., Wang, H., Hernandez, J.M., Hutchison, D., Karaliopoulos, M., Kooij, R.E.: A framework for computing topological network robustness. Delft University of Technology, report20101218 (2010)
37. Weisstein, E.W.: Contiguous USA Graph. From MathWorld–A Wolfram Web Resource. http://mathworld.wolfram.com/ContiguousUSAGraph.html
38. Xu, J., Chen, H.: The topology of dark networks. Commun. ACM **51**(10), 58–65 (2008)
39. Yang, Z., Algesheimer, R., Tessone, C.J.: A comparative analysis of community detection algorithms on artificial networks. Sci. Rep. **6**, 30750 (2016). doi:10.1038/srep30750