Modelling Critical Node Attacks in MANETs

Dongsheng Zhang, James G. Sterbenz

To cite this version:

Dongsheng Zhang, James G. Sterbenz. Modelling Critical Node Attacks in MANETs. 7th International Workshop on Self-Organizing Systems (IWSOS), May 2013, Palma de Mallorca, Spain. pp.127-138, 10.1007/978-3-642-54140-7_11 . hal-01291506

HAL Id: hal-01291506
https://inria.hal.science/hal-01291506

Submitted on 21 Mar 2016

Modelling Critical Node Attacks in MANETs

Dongsheng Zhang* and James P.G. Sterbenz*†

*Information and Telecommunication Technology Center
Department of Electrical Engineering and Computer Science
The University of Kansas, Lawrence, KS 66045, USA
{dzhang, jpgs}@ittc.ku.edu
http://www.ittc.ku.edu/resilinets
†School of Computing and Communications, InfoLab21
Lancaster University, Lancaster LA1 4YW, UK
jpgs@comp.lancs.ac.uk

Abstract. MANETs (mobile ad hoc networks) operate in a self-organised and decentralised way. Attacks against nodes that are highly relied to relay traffic could result in a wide range of service outage. A comprehensive model that could enhance the understanding of network behaviour under attacks is important to the design and construction of resilient self-organising networks. Previously, we modelled MANETs as an aggregation of time-varying graphs into a static weighted graph, in which the weights represent link availability of pairwise nodes. Centrality metrics were used to measure node significance but might not always be optimal. In this paper, we define a new metric called *criticality*[1] that can capture node significance more accurately than centrality metrics. We demonstrate that attacks based on criticality have greater impact on network performance than centrality-based attacks in real-time MANETs.

Keywords: graph theory, MANET, network resilience, challenge modelling, centrality, criticality

1 Introduction and Motivation

MANETs have the merit of quick and flexible self-organisation and have been utilised in various scenarios, such as vehicular ad hoc networks and wireless sensor networks. With the increasing deployment of MANETs in commercial and military uses, it becomes vital to design and construct a resilient and survivable MANET. Because of the peer-to-peer and multi-hop properties of MANET communications, challenges against certain nodes might cause the partitioning of the network. By strengthening specific critical nodes such as increasing the transmission range or recharging the battery, the whole network could be more resilient under attacks and challenges. Furthermore, due to node mobility, unpredictably long delay, and channel fading of wireless environment [19], MANETs suffer from dynamic connectivity that increases the complexity of modelling.

[1] our definition is different from the critically k-connected graph defined in [12]

In our previous work, MANETs are modelled as time-varying graphs and pairwise node interactions are aggregated within specific time windows [21]. Dynamic MANETs can be represented as static weighted graphs, in which the weight refers to link availability. The adversary is assumed to have complete knowledge about the network. Centrality metrics can be used to identify the relative significance of each node. However, research in SNA (social network analysis) showed that the removal of high centrality nodes might not necessarily cause maximal loss of network connectivity [4]. Articulation points whose removal increases the number of connected components could lead to maximum degradation of overall network performance. The CNPs (critical node problems), which are generally defined as the detection of a subset of nodes whose removal disconnects the graph maximally, have been widely studied in SNA [2]. Instead of using weighted centrality metrics to indicate node significance, we provide a more accurate selection of nodes whose removal could have a higher impact on the network. Network simulations are performed to verify how attacks against nodes selected by this approach could impact overall network performance.

The rest of the paper is arranged as follows. In Section 2, we introduce background and related work about wireless network challenge modelling, centrality metrics, and CNPs. In Section 3, we illustrate the difference between critical node detection in weighted and unweighted graphs. We describe how to detect critical nodes in weighted graph and model malicious attacks based on node criticality in Section 4. In Section 5, we exploit simulations to verify our approach using several examples with plots showing network performance under various types of attacks. Finally, we summarise our work and mention the steps for future research in Section 6.

2 Background and Related Work

Understanding network challenges that are inherent in the self-organising networks is essential to construct a resilient and survivable network [18]. Simulation tools can be utilised to examine network performance under various attacks and challenges [14]. Centrality metrics can be used to measure relative node significance. However, those nodes whose removal could partition the topology might be more vital to the whole network.

2.1 Network Challenge Modelling

A simulation framework that evaluates realistic challenges in wired networks has been developed [6]. Due to the dynamics and channel properties of wireless networks, techniques used to improve the disruption tolerance and network reliability for wired networks are not enough in the wireless context [19]. In order to capture the time-varying characteristics of MANETs, temporal graph metrics used in SNA take into account topology evolutions over time [20]. However, they are not applicable to real-time MANETs since traditional MANET routing protocols do not allow data transmission if there is no route between source and destination at the time of sending, which makes metrics such as temporal path ineffective. Temporal network robustness is used to measure how communication of a given time-varying network is affected by random attacks [16]; however, it does not address the impact of critical node attacks that could result in higher degradation of network performance.

2.2 Centrality Metrics

Centrality metrics (degree, betweenness, and closeness) have been used to measure comparative node importance in both weighted and unweighted graphs in SNA [9, 15]. Degree centrality indicates the node communication ability within its neighbours and the disadvantage is that it only captures the relation between adjacent nodes and fails to take into account global topological properties. Metrics that can capture global properties include betweenness and closeness. Betweenness is defined as the frequency that a node falls on the shortest paths and closeness is defined as the inverse of the sum of the shortest paths [9]. In order to calculate node betweenness and closeness in a weighted graph, the weights need to be inverted to represent link cost instead of link availability [13]. Betweenness measures the degree to which a node enables communication between all node pairs. Closeness measures the extent to which node communications capabilities are independent of other nodes. However, they might not always be effective to definitively indicate the structural importance of each node, since those nodes whose removal could cause most damage on the network are not necessarily the nodes with high centrality values [4]. Examples will be presented in Section 3 to illustrate the difference.

2.3 Critical Node Problems

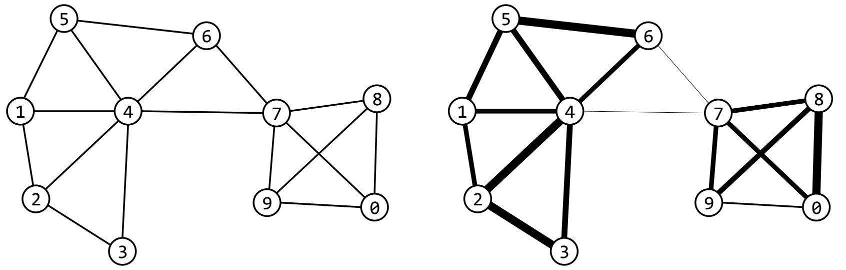
Vulnerability assessment in cases of potential malicious attacks is critical to network resilience design. A framework that models the network as a connected directed graph can evaluate network vulnerability by investigating how many nodes are required to be removed so that network connectivity can be degraded to a desired level [8]. A general graph-theoretical formulation of this problem is removing a certain number of nodes in a graph to maximize the impairment against overall network connectivity, which falls under CNPs. The CNPs are known as $\mathcal{NP}$-hard on general graphs [2]. Heuristics, branch and cut algorithms, and dynamic programming algorithms have been proposed to solve CNPs; however, all of them put certain constraints on graph structures such as trees, series-parallel graphs, or sparse graphs [2, 7, 17]. As far as we know, no effective approximation algorithms for the weighted graph CNPs have been proposed. Critical node behaviour has been studied using network simulations by only considering discrete static connected topologies [10, 11] and cannot be extended to general self-organising and dynamic MANETs.

3 Critical Nodes in Unweighted and Weighted Graphs

CNPs deal with the detection of one or multiple nodes whose removal would result in minimal pairwise connectivity. Two examples are given to show the relationship between nodes with high centrality values and the most critical nodes in unweighted and weighted graphs.

Figure 1a presents a 10-node unweighted graph topology, in which node 4 has the highest degree, betweenness, and closeness centrality values. However, the deletion of node 7 instead of node 4 from the graph would partition the network and result in lower connectivity within the rest of the graph. In contrast, the network is still connected after deleting node 4. The impact of removing node 4 is of no significant difference to the removal of any other nodes except node 7.

The case is more complex for critical nodes detection in a weighted graph. Figure 1b has the same network topology as Figure 1a except that each link is associated with certain weight. Assuming that the weights of link $(4,7)$ and $(6,7)$ are 0.001, whereas weights of all other links are far higher than 0.001. The removal of node 7 has trivial impact on the entire network, since node 7 is weakly connected to node 4 and 6 compared to other links before being removed. Even though the removal of node 7 partitions the network, the significance of the articulation point for an unweighted graph cannot be directly extended to weighted graphs. In the next section, we propose a method that can handle critical nodes detection approximately in a weighted graph.



(a) Critical nodes in unweighted graph (b) Critical nodes in weighted graph

Fig. 1. Critical node problems for unweighted and weighted graphs

4 Modelling Approach

In our modelling, two wireless nodes are assumed to be adjacent if they are within the transmission range of each other (without interference) and are connected if they can be reached via multihop links. Symmetric communication between nodes is assumed and an undirected graph is sufficient to model the network. Previously, we have modelled the dynamics of the MANETs by aggregating network topologies into a weighted graph in which the weight represents link availability, that is, the fraction of time that nodes are adjacent given the dynamic self-organisation of the network [21]. Based on this weighted link availability model, we propose a new method to detect critical nodes.

4.1 Constructing Link Availability Graphs

Figure 2 presents a scenario of MANET topologies at six consecutive time steps and Figure 3 shows the aggregated representation as an adjacency matrix. In Table 4.1, three centrality metrics are calculated based on the weighted adjacency matrix and they give different indications of node significance. Node 5 has the highest degree of 2.66; node 4 has highest betweenness of 0.4; node 3 has the highest closeness of 0.409. We define *criticality* as the inverse of the sum of pair-wise path availability after the removal of certain number of nodes. In this case, if only one node will be attacked, the removal of node 5 will impact the network most heavily with the rest of the graph having the minimum connectivity. The detailed algorithm for calculating node criticality is described in Section 4.2.

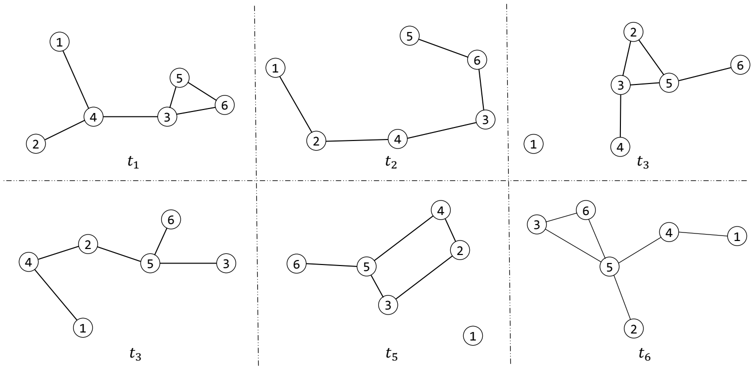
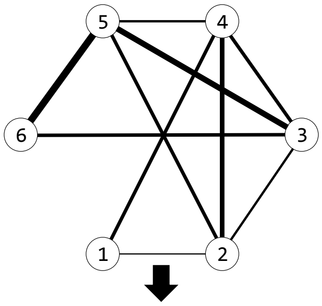


Fig. 2. MANET topologies at six consecutive time steps



$$\begin{bmatrix} 0.00 & 0.17 & 0.00 & 0.50 & 0.00 & 0.00 \\ 0.17 & 0.00 & 0.33 & 0.67 & 0.50 & 0.00 \\ 0.00 & 0.33 & 0.00 & 0.50 & 0.83 & 0.50 \\ 0.50 & 0.67 & 0.50 & 0.00 & 0.33 & 0.00 \\ 0.00 & 0.50 & 0.83 & 0.33 & 0.00 & 1.00 \\ 0.00 & 0.00 & 0.50 & 0.00 & 1.00 & 0.00 \end{bmatrix}$$

Fig. 3. Weighted link availability graph and adjacency matrix

Table 1. Single node attack priorities based on centrality and criticality values

Node	Degree	Betweenness	Closeness	Criticality
1	0.67	0.0	0.243	0.082
2	1.67	0.0	0.384	0.097
3	2.16	0.2	**0.409**	0.111
4	2.00	**0.4**	0.399	0.112
5	**2.66**	0.1	0.408	**0.138**
6	1.50	0.0	0.313	0.108

4.2 Measuring Network Connectivity of Weighted Graphs

In an unweighted graph, we measure the graph connectivity as the sum of all possible traffic flows between pairwise nodes within each component. For example, in Figure 1a, after the deletion of node 7, the original graph is split into two components containing 6 and 3 nodes respectively. Hence, the total possible traffic flows after deletion is $6 \times 5 + 3 \times 2 = 36$. However, in our weighted graph model, network connectivity cannot be simply measured in the same way as in unweighted graphs since each link is associated with a value $A(l_i)$, $(0 \leq A(l_i) \leq 1)$ as its link availability. Such a weighted graph can be treated as a complex system model. Path availability for a series and parallel model can be respectively calculated as:

$$A_s(P) = \prod A(l_i) \tag{1}$$

$$A_p(P) = 1 - \prod [1 - A(l_i)] \tag{2}$$

where path $P = (l_1, l_2, ..., l_i)$ [3]. For a general weighted graph that cannot be reduced to a series-parallel model, the number of possible states is non-polynomial and the numerical availability of the system is too complex to compute even after deleting each possible set of nodes. We measure network connectivity by applying following approximations.

Approximation 1: Only the strongest path of all possible paths (if there exists one) for a pair of nodes will be selected. Equation 1 can be used to calculate path availability between node pairs. The disadvantage is that if the aggregated graph is close to fully-connected with each link associating with similar weight, the selected path might fail to represent the actual connectivity between node pairs.

Approximation 2: The *hopcut*, which is the maximum number of hops in considered paths, is set to a certain number to shrink the size of candidate paths, as the number of all possible paths for a pair of node in a graph of order n (n vertices) could be as high as $n!$. The average multihop path availability tends to decrease with a growing number of hops as the value of $A(l_i)$ is less than 1. Hopcut can be set approximately according to the diameter of the graph. Simulation results will be given in next section about how the hopcut approximation affects the accuracy of critical nodes detection.

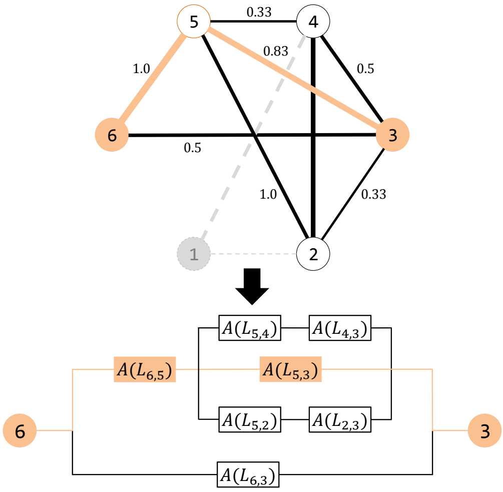


Fig. 4. Path selection between node 3 and 6

Algorithm 1 Overall network availability of a weighted graph $G = (V, E_w)$

```
$A_{\text{sum}} = 0$ {initialise the sum of path availability for all node pairs}
for $s$ in $V$ do
  for $d$ in $V$ do
    if $s$!=$d$ then
      $A_{\max} = 0$ {initialise maximum path availability between a specific node pair}
      for $P$ in all paths within hop count $h$ between node $s$ and $d$ do
        $A_P = 1$ {initialise current path availability}
        for $e$ in $P$ do
          $A_P = A_P \times W(e)$ {multiply availability of all the links in the path}
        end for
        if $A_P > A_{\max}$ then
          $A_{\max} = A_P$ {select the path with highest availability}
        end if
      end for
    end if
    $A_{\text{sum}} = A_{\text{sum}} + A_{\max}$ {add up path availability for each pair of nodes}
  end for
end for
return $A_{\text{sum}}$
```

An example that illustrates the algorithm is shown in Figure 4. Node 1 and all the links incident to it are removed due to attack. We need to calculate path availability for pairwise nodes. Consider node pair 3 and 6. The hopcut is set as 3. Hence, all the paths (less than 4 hops) between node 6 and 3 are $\{6, 5, 3\}$, $\{6, 5, 4, 3\}$, $\{6, 5, 2, 3\}$, and $\{6, 3\}$ with $A(l_{6,5}) \times A(l_{5,3})$ yielding a highest path availability as $1.0 \times 0.83 = 0.83$. Repeat the same process for other node pairs. Algorithm 1 describes how to calculate pairwise path availability for a general weighted link availability model. The algorithm complexity depends on the weighted graph structure. For an n-node MANET with hopcut set as h, the complexity of Algorithm 1 ranges from $O(n^2)$ if the aggregated graph is a tree to $O(n^2h!)$ if the aggregated graph is a complete graph. For the weighted graph model that we use in the paper, the graph structure becomes more full-mesh-like with larger aggregation window sizes of the MANET topologies. However, the difference among node significance becomes trivial in a full-mesh-like network and attacks toward any node would have similar impact on the network. Hence, in order to study how the removal nodes of high centrality and criticality could impact the network, we are less interested in full-mesh-like graphs that are more computationally complex to measure their connectivity.

4.3 Detecting Critical Nodes

Instead of using centrality metrics as significance indicators, we directly look into the most critical nodes in the network. It is known that finding critical nodes in a general graph is $\mathcal{NP}$-hard [2]. We propose two approximations that are specific to our link availability model to simplify the algorithm.

Approximation 1: Due to the mobility of MANETs and frequent changes of routing tables, it takes a certain amount of time to populate updated routing information on all nodes. Too short a contact duration between nodes might not allow the traffic transmission. Therefore we can simplify the weighted model by deleting links that are lower than a certain threshold.

Approximation 2: Since centrality metrics are related to the relative significance of each node, instead of considering all nodes as potential candidates of critical nodes, we only consider those nodes with high centrality values. The main purpose of this paper is to present a simulation model for challenges against MANETs. Even though we apply the above approximations, the computational complexity for graphs with a large number of nodes is still high. In our simulation, the number of nodes is set as 20 and the maximum critical node set size is set as 8. We will not provide a rigorous proof of how close this approximation is to the optimal solution, but we have simulation results in the next section to show that the removal of critical nodes detected using our approach does have higher impact on the whole network. The procedures of detecting k-critical nodes of a weighted graph G of order n are presented as follows:

Step 1 *Calculate the centrality metrics (degree, betweenness, and closeness) for each node in the graph and store the sorted node list in D, B, C*

Step 2 *Let L be the critical node list and add those nodes with k highest centrality values into L*

$$L = D(0:k) \cup B(0:k) \cup C(0:k) \tag{3}$$

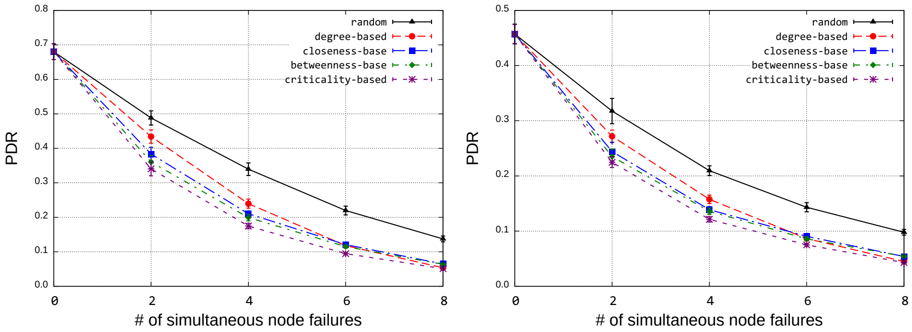
where $D(0:k)$ denotes the first k elements in list D. Remove the duplicate nodes in L.

Step 3 *Let the size of list L be S, then there are $\binom{S}{k}$ different potential critical node sets. Let N be one of the potential critical node set, and remove nodes in N and all the edges that are incident to them from graph G. Calculate the pairwise link availability for the rest of graph as $A(G)$. Iterate the same process on each case in $\binom{S}{k}$.*

Step 4 *The set of critical nodes whose removal result in the lowest $A(G)$ is selected as the critical node set.*

5 Simulation Analysis

In this section, we employ network simulation to examine the impact of attacks based on different metrics on the network performance. We use the network simulator ns-3.16 as our simulation tool [1]. Constant bit rate UDP traffic is generated at steady state during the simulation and all simulations are averaged over 10 runs. The Gauss-Markov mobility model is used to represent node motion patterns [5]. Functional verification of how different parameters such as node velocity, number of nodes, and routing protocols could influence base scenario network performance without attacks was done in our previous work [21]. In the real world, the placement of network resources must be balanced to the optimised resilience and deployment [18]. Due to space constraints, the simulation parameters used in this paper are limited to 20 nodes, 6 neighbour count, and [5, 10] m/s node velocity range. We examine the impact of a different number of simultaneous node failures, hopcut used in path availability approximation, and windows size that determines the granularity of topology aggregation. For the centrality-based attacks, each metric will be recalculated adaptively after the removal of other nodes. AODV (ad hoc on-demand distance vector) and DSDV (destination-sequenced distance vector) routing protocols are used so that we can have both reactive and proactive routing protocols. PDR (packet delivery ratio) is used to measure the network performance under attacks.

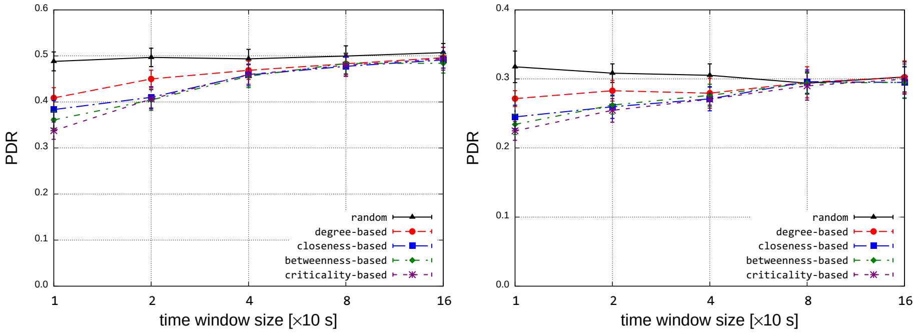


(a) AODV

(b) DSDV

Fig. 5. Network performance with increasing number of node failures

In Figure 5, network performance with increasing number of simultaneous node failures arising from random or malicious attacks is examined. All centrality- and criticality-based attacks have apparently higher impact on the network performance than random node failures as expected. Generally, criticality-based attacks result in a lower bound of network performance than centrality-based attacks due to the inaccuracy of node significance predicted by centrality values for certain topologies. With the increase of the number of simultaneous node failures, the difference between centrality- and criticality-based attacks shrinks. This can be explained as when the number of node failures increases, the network is partitioned into multiple components of small order and the difference of node significance become minor. In addition, the degree metric has a more accurate indication of node significance for a relatively large number of simultaneous node failures, whereas betweenness and closeness predict node significance better with a small number of simultaneous node failures.

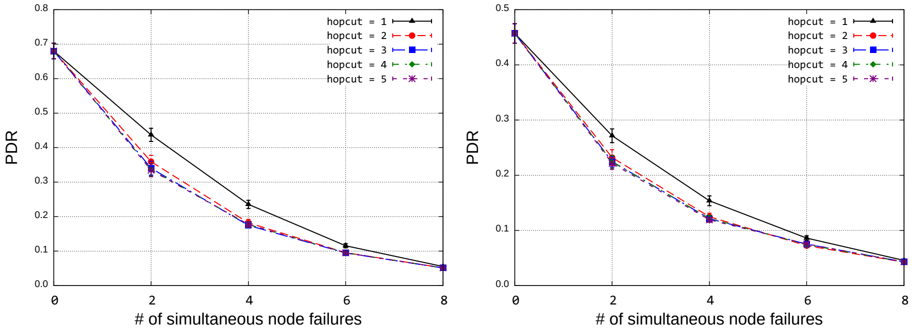


(a) AODV

(b) DSDV

Fig. 6. Network performance with increasing window sizes

Figure 6 shows the influence of topology aggregation granularity. As the time window increases, the difference of relative significance between nodes becomes trivial. When time window size is 10 s, the criticality-based attack has the lowest PDR and degree-based attack has the highest PDR of all centrality based attacks. When time window size is 160 s, PDRs under all different type of attacks converge to approximately the same value. This can be explained as the distribution of pairwise node interactions becomes even among all nodes and nodes will have similar centrality and criticality values under the Gauss-Markov mobility model given a long enough time window, that is, MANET routing is constantly re-optimising the network with moving nodes. As we can see, both centrality and criticality metrics might not be able capture relative node significance accurately in an almost fully-connected graph with evenly assigned weights. Figure 7 shows different approximations of maximum number of hops considered in calculating path availability. There are almost no difference for 3, 4, and 5 hopcut, which means that most communications between pairwise nodes happen within 3 hops for this specific simulation scenario.



(a) AODV

(b) DSDV

Fig. 7. Network performance with increasing hopcut

6 Conclusion and Future Work

In this paper, we conducted a detailed examination of how centrality metrics can be used to indicate node significance in MANETs. We proposed a novel *criticality* approach to measure connectivity of weighted graphs with the weight ranging from 0 to 1. We provided an approximate algorithm to find the critical node subset in MANETs. We demonstrated that critical node attacks impact network performance more than attacks based on centrality values. Future work will include more accurate scalable heuristics to detect any number of critical nodes in weighted graphs of larger size. The impact of network size and density on the detection of critical nodes will also be studied.

Acknowledgments

We would like to acknowledge Egemen K. Çetinkaya for the discussions and comments about this paper and other members of the ResiliNets group for discussions on this work. This research was supported in part by NSF FIND (Future Internet Design) Program under grant CNS-0626918 (Postmodern Internet Architecture) and by NSF grant CNS-1050226 (Multilayer Network Resilience Analysis and Experimentation on GENI).

References

1. The ns-3 network simulator. `http://www.nsnam.org` (July 2009)
2. Arulselvan, A., Commander, C.W., Elefteriadou, L., Pardalos, P.M.: Detecting critical nodes in sparse graphs. Computers and Operations Research 36(7), 2193–2200 (2009)
3. Billinton, R., Allan, R.: Reliability Evaluation of Engineering Systems. Plenum Press London (1983)
4. Borgatti, S.P.: Identifying sets of key players in a social network. Comput. Math. Organ. Theory 12(1), 21–34 (April 2006)

5. Camp, T., Boleng, J., Davies, V.: A survey of mobility models for ad hoc network research. Wireless Communications and Mobile Computing 2(5), 483–502 (2002), `http://dx.doi.org/10.1002/wcm.72`
6. Çetinkaya, E.K., Broyles, D., Dandekar, A., Srinivasan, S., Sterbenz, J.P.G.: Modelling Communication Network Challenges for Future Internet Resilience, Survivability, and Disruption Tolerance: A Simulation-Based Approach. Springer Telecommunication Systems pp. 1–16 (2011), published online: 21 September 2011
7. Di Summa, M., Grosso, A., Locatelli, M.: Branch and cut algorithms for detecting critical nodes in undirected graphs. Computational Optimization and Applications pp. 1–32 (2012)
8. Dinh, T., Xuan, Y., Thai, M., Park, E., Znati, T.: On approximation of new optimization methods for assessing network vulnerability. In: Proceedings of the IEEE Conference on Computer Communications (INFOCOM). pp. 1–9. IEEE (2010)
9. Freeman, L.: Centrality in social networks conceptual clarification. Social networks 1(3), 215–239 (1979)
10. Kim, T.H., Tipper, D., Krishnamurthy, P., Swindlehurst, A.: Improving the topological resilience of mobile ad hoc networks. In: 7th International Workshop on Design of Reliable Communication Networks (DRCN). pp. 191–197 (October 2009)
11. Kim, T., Tipper, D., Krishnamurthy, P.: Connectivity and critical point behavior in mobile ad hoc and sensor networks. In: IEEE Symposium onComputers and Communications ISCC. pp. 153–158. IEEE (2009)
12. Kriesell, M.: Minimal connectivity. In: Beineke, L.W., Wilson, R.J. (eds.) Topics in Structural Graph Theory, pp. 71–99. Cambridge University Press (2012)
13. Newman, M.E.J.: Scientific collaboration networks. ii. shortest paths, weighted networks, and centrality. Phys. Rev. E 64(1), 16132 (2001)
14. Nicol, D.M., Sanders, W.H., Trivedi, K.S.: Model-based evaluation: From dependability to security. IEEE Transactions on Dependable and Secure Computing 01(1), 48–65 (2004)
15. Opsahl, T., Agneessens, F., Skvoretz, J.: Node centrality in weighted networks: Generalizing degree and shortest paths. Social Networks 32(3), 245–251 (2010)
16. Scellato, S., Leontiadis, I., Mascolo, C., Basu, P., Zafer, M.: Evaluating temporal robustness of mobile networks. IEEE Transactions on Mobile Computing 12(1), 105–117 (January 2013)
17. Shen, S., Smith, J.C.: Polynomial-time algorithms for solving a class of critical node problems on trees and series-parallel graphs. Networks 60(2), 103–119 (2012)
18. Sterbenz, J.P.G., Hutchison, D., Çetinkaya, E.K., Jabbar, A., Rohrer, J.P., Schöller, M., Smith, P.: Resilience and survivability in communication networks: Strategies, principles, and survey of disciplines. Computer Networks 54(8), 1245–1265 (2010)
19. Sterbenz, J.P.G., Krishnan, R., Hain, R.R., Jackson, A.W., Levin, D., Ramanathan, R., Zao, J.: Survivable mobile wireless networks: issues, challenges, and research directions. In: Proceedings of the 3rd ACM workshop on Wireless Security (WiSE). pp. 31–40. Atlanta, GA (2002)
20. Tang, J., Musolesi, M., Mascolo, C., Latora, V.: Temporal distance metrics for social network analysis. In: Proceedings of the 2nd ACM workshop on Online social networks. pp. 31–36 (2009)
21. Zhang, D., Gogi, S.A., Broyles, D.S., Çetinkaya, E.K., Sterbenz, J.P.: Modelling Attacks and Challenges to Wireless Networks. In: Proceedings of the 4th IEEE/IFIP International Workshop on Reliable Networks Design and Modeling (RNDM). pp. 806–812. St. Petersburg (October 2012)