ELSEVIER

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 19 (2013) 1167 – 1172

Procedia Computer Science

International Workshop on Survivable and Robust Optimal Networks

Localized Algorithm for Segregation of Critical/Non-critical Nodes in Mobile Ad Hoc and Sensor Networks

Muhammad Imran[a],a*, Mohamed A. Alnuem[b] Mahmoud S. Fayed[b], Atif Alamri[b]

[a]*Deanship of e-Transactions and Communication, King Saud University, Riyadh, Saudi Arabia*
[b]*College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia*

Abstract

Timely segregation of connectivity-centric critical/non-critical nodes is extremely crucial in mobile ad hoc and sensor networks to assess network vulnerabilities against critical node failures and provide precautionary means for survivability. This paper presents a localized algorithm for segregation of critical/non-critical nodes (LASCNN) that opts to distinguish critical/non-critical nodes to the network connectivity based on limited topology information. Each node establishes and maintains a k-hop connection list and employ LASCNN to determine whether it is critical/non-critical. Based on the list, LASCNN marks a node as critical if its k-hop neighbor's become disconnected without the node, non-critical otherwise. Simulation experiments demonstrate the scalability of LASCNN and shows the performance is quite competitive compared to a scheme with global network information. The accuracy of LASCNN in determining critical nodes is 87% (1-hop) and 93% (2-hop) and non-critical nodes 91% (1-hop) and 93% (2-hop).

Selection and peer-review under responsibility of Elhadi M. Shakshuki

Keywords: Mobile ad hoc and sensor networks; localized and distributed algorithms, segregation, critical/non-critical nodes

1. Introduction

Mobile ad hoc and sensor networks (MAHSNs) are captivating significant attention from research community because of their suitability for enormous range of applications. Nodes in these networks are expected to establish and maintain a connected wireless network and operate autonomously in unattended setups. Due to ad hoc nature, limited energy and transmission range nodes have to rely on multihop communication i.e., a node forwards a message to the next node on the path towards the destination. The nodes organize themselves because there is no central entity to manage the network. Therefore, maintaining network connectivity is extremely desirable in order for the network to stay functional.

Nonetheless, random deployment, frequent mobility and network dynamics (e.g., sleep/wakeup) inherently introduce critical i.e., cut vertex nodes to the network connectivity. Nodes in these networks are

* Corresponding author. Tel.: +966-1-4696332;
E-mail address: cimran@ksu.edu.sa.

Selection and peer-review under responsibility of Elhadi M. Shakshuki
doi:10.1016/j.procs.2013.06.166

susceptible to physical damage and attacks; therefore, failure of nodes is not surprising. Failure of a critical node partitions the network into disjoint segments and may disrupt network operation [1]. The network stays connected despite the loss of a non-critical node. Thus, it is imperative to distinguish critical/non-critical nodes to the network connectivity. The autonomous and self-organizing nature of these large-scale dynamic networks requires localized and distributed schemes. Moreover, nodes with limited resources should use agile and lightweight procedure and avoid sophisticated diagnostics. Most of the published schemes are centralized and require globalized network state information for determining critical nodes. Centralized schemes require maintaining up-to-date network wide information that may not be practical for large-scale dynamic networks. Moreover, it becomes almost impossible to identify critical nodes at regular intervals due to frequent topology changes. Furthermore, none of existing localized scheme determines non-critical nodes. The prime objective is to demonstrate the feasibility of localized schemes in accurately segregating critical/non-critical nodes in-advance for MAHSNs.

This paper presents a fully distributed and localized algorithm for segregation of critical/non-critical nodes (LASCNN) that opts to distinguish connectivity-centric critical/non-critical nodes to assess the network vulnerability against critical node failures and provide precautionary means for survivability. Nodes establish and maintain a connection list based on localized information (i.e. k-hop) and employ LASCNN to process the list in determining critical/non-critical nodes. LASCNN examines the list and mark the node as critical if its k-hop neighbors become disconnected without the node, non-critical otherwise. Simulation experiments validate the effectiveness of LASCNN in determining critical/non-critical nodes with high accuracy. Simulation results indicate that the performance of LASCNN is scalable and quite competitive compared to centralized schemes with globalized network information.

The rest of the paper is organized as follows. We elaborate system model and problem statement in Section 2. Section 3 provides insights to works related. The proposed LASCNN algorithm is presented in Section 4. Section 5 contains the results and analysis. The concluding remarks are presented in Section 6.

2. System Model and Problem Statement

LASCNN is applicable to MAHSNs, where nodes are randomly deployed in an area of interest and are assumed to be able to move around. After deployment, nodes discover each other through *hello* messages and form a connected network. Nodes have limited resources in terms of energy and computation. Nodes are assumed to acquire their positions through GPS or any other localization techniques and periodically update their status (ID, location, etc.) to their neighbors. The communication range (R_c) of a node is limited and refers to a maximum Euclidean distance that its radio can reach and is equal for all nodes.

The network vulnerability primarily depends on the position of nodes in the network topology. For example failure of a critical node, such as *N4* in Fig. 1, will divide the network into three disjoint segments. Meanwhile, loss of non-critical nodes, such as *N17* or *N18* does not affect network connectivity. To identify critical nodes, the published schemes can be categorized based on maintaining topology information into globalized and localized. Globalized schemes require maintaining network wide information. However, maintaining such a level of information is infeasible for large-scale dynamic networks. On the other hand, localized schemes only require k-hop information. We argue that localized schemes better suits MAHSNs because of their distinguished characteristics such as size and dynamic nature. LASCNN belongs to this category.

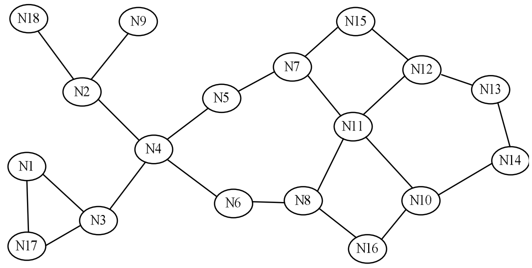

Fig. 1: An example of a connected MAHSN segment

3. Related Work

Identifying cut vertices in graph theory has been extensively investigated in the literature [2, 3].

However, these algorithms are difficult or impossible to be adopted for MAHSNs. These algorithms can be categorized into centralized and distributed based on execution. Centralized algorithms [4, 5] required to maintain entire topology information which involves very high communication overhead, and thus may not be suitable for large-scale dynamic networks. On the other hand, a globalized distributed implementation of centralized algorithm based on depth first search (DFS) does not require network wide information at any node and can be used for MAHSNs. For example, DDFS (Distributed Depth-First Search) and CAM [6] build a DFS tree in a distributed fashion. However, results [7] show that these algorithms are not suitable for MAHSNs due to long time execution and high communication cost. Few localized and distributed algorithms to determine critical nodes in MAHSNs have been proposed in [7, 8]. These algorithms only determine critical nodes and do not provide survivability mechanism by identifying non-critical nodes. Moreover, the accuracy of determining critical nodes is less.

Partitioning detection and connectivity restoration schemes can be categorized into pre-partitioning, post-partitioning and hybrid. Pre-partitioning algorithms [7, 8] provides network vulnerability assessment by identifying critical nodes. However, our proposed LASCNN algorithm inherently dangles survivability mechanism by accurately segregating non-critical nodes that can be used for partitioning avoidance and connectivity restoration. Post-partitioning algorithms [9] assess the impact of the node failure on network connectivity and execute connectivity restoration. Hybrid algorithms [10-12] segregate critical/non-critical nodes in advance and provide a survivability mechanism against node failures. A semi-hybrid scheme in [13] execute a connectivity restoration in case of a critical node failure. However, none of the connectivity restoration scheme aims at accurately segregating critical/non-critical nodes.

4. Localized Algorithm for Segregation of Critical/Non-critical Nodes

The distributed and localized algorithms are more suitable for large-scale dynamic networks such as MAHSNs because they can quickly distinguish critical/non-critical nodes with far less computation and communication overhead. LASCNN checks connectivity of neighbors for each node based on localized information to determine if the node is critical/non-critical. The details of LASCNN are as follows:

4.1. Establish and maintain connection list

Unlike globalized schemes, LASCNN requires each node to establish and maintain a k-hop neighbor connection list. To avoid unnecessary computation overhead in maintaining adjacency matrix, each node only maintains a duplication-free pair-wise connection list *(ConnList)* based on k-hop information. After discovering each other, nodes establish a 1-hop and 2-hop *ConnList*. The 1-hop *ConnList* contains the direct neighbors of a node in the form of a pair. For example, a 1-hop *ConnList* of node *N11* for the network segment of Fig. 1 is shown in Table 1 where each row represents an undirected connection between pair of nodes. Similarly, 2-hop connections can be acquired through 1-hop neighbors. These are the connections between 1-hop neighbors, between a 1-hop and a 2-hop neighbors, and possibly connection between 2-hop neighbors can be determined based on the position of nodes. Table 1 shows a 2-hop *ConnList* of *N11* for the similar network segment. For example, *N11* can determine the connection between its 2-hop neighbors *N13* and *N14* based on their position. Similarly, generalizing k-hop *ConnList* will contain connections between k-hop neighbors, between a k-hop and a (k-1)-hop neighbors and if some k-hop nodes are connected.

Table 1: 1-hop and 2-hop *ConnList* of *N11*

1-hop	**2-hop**	
N7 ↔ N11	N7 ↔ N11	N7 ↔ N15
N8 ↔ N11	N8 ↔ N11	N8 ↔ N16
N10 ↔ N11	N10 ↔ N11	N10 ↔ N14
N11 ↔ N12	N11 ↔ N12	N10 ↔ N16
	N5 ↔ N7	N12 ↔ N13
	N6 ↔ N8	N12 ↔ N15
		N13 ↔ N14

To accurately segregate critical/non-critical nodes, nodes periodically exchange status update messages with neighbors to reflect topology changes. Upon receiving a message, a node forwards the message to (k-1)-hop neighbors. For example, while maintaining 2-hop *ConnList*, *N12* will update status of *N11* to *N13*. A noticeable point is that all nodes need to maintain their *ConnList* based on status updates; therefore, it becomes infeasible to maintain more topology information in large-scale dynamic networks.

4.2. LASCNN algorithm

Once all nodes establish a k-hop connection list (e.g., 1-hop and 2-hop), each node independently process its list to determine whether it is critical/non-critical. Basically, a node *N* needs to determine from the k-hop *ConnList* that whether its k-hop neighbors remain connected or not. If k-hop neighbors of *N* remains connected without it, *N* is critical, non-critical otherwise. While processing *ConnList*, each node populates a k-hop connected neighbors list *ConnNeighbors* that contains entries of connected neighbors for the node. The detail process of LASCNN is as follows:

If there is only one connection in the *ConnList* then the node is non-critical because absence of such node will not affect network connectivity. In case of more connections, *N* start processing its list from first connection i.e., *ActiveConn*. The processing of *ActiveConn* only proceeds if *N* is not involved in the connection pair; otherwise, the next connection becomes the *ActiveConn*. The node pair in *ActiveConn* will be inserted in the *ConnNeighbors* if it is empty. Otherwise, if there is a common node in *ActiveConn* and *ConnNeighbors*, the other node will be put into *ConnNeighbors* if it is not already there. The next connection will become the *ActiveConn* and will be processed in the same manner until all the connections are processed. The next iteration of processing *ConnList* will commence if size of *ConnNeighbors* increased in the current iteration. In this way, all the connected neighbors of *N* will become part *ConnNeighbors*. If the size of k-hop *ConnNeighbors* is less than the number of k-hop neighbors of *N*, *N* is critical, non-critical otherwise.

LASCNN works independently of the input size and works similar for 1-hop, 2-hop and k-hop information. The algorithm will separately determine 1-hop, 2-hop and k-hop critical/non-critical nodes. For example, processing 1-hop *ConnList* of *N11* results in an empty *ConnNeighbors* list that indicates that it is 1-hop critical. On the other hand, *N11* is 2-hop non-critical because all its 2-hop neighbors remain connected after processing its 2-hop *ConnList*. A noticeable point is that LASCNN can accurately determine non-critical nodes with only k-hop information. This is really important while providing survivability mechanism. Moreover, LASCNN identify all critical nodes that are globally critical. However, it may determine some nodes as critical while they are not indeed critical. This is understandable because it is almost impossible for a node to know the long alternate paths across the network. Considering the merits of localized algorithms and the fact that none of the critical node is missed, such a category of approaches fits well and the reduced accuracy is not a major concern. Pseudo code of LASCNN is omitted due to space limitation.

5. Results and Analysis

The accuracy of LASCNN is validated through simulation experiments. Experiments involve randomly generated connected topologies in an area of 1000m × 600m with varying network densities and radio ranges. The network densities have been set to 20, 40, 60, 80, 100, 150, 200 and 250. The radio range "*r*" of nodes is changed among 25, 50, 75, 100 and 125. For each experiment, results of 15 different topologies are averaged and reported. All results are subject to 90% confidence interval analysis and stays within 6%-10% of the sample mean. To prove the effectiveness and efficiency, we compare LASCNN to that with globalized knowledge which is perfectly accurate in segregating critical/non-critical nodes. Following metrics are used to assess the performance:

- The *detection ratio* is the percentage of nodes that are detected critical/non-critical: This metric indicates the effectiveness of the localized algorithms in determining critical/non-critical nodes.
- The *accuracy of determining critical/non-critical nodes*: This metric gauges the accuracy of localized algorithms in detecting critical/non-critical nodes.

We have used the following parameters to vary the network configuration in the experiments:

- The *network density (N)* affects the node count and the degree of network connectivity.
- The *radio range (r)* is the reachability of nodes and affects the network connectivity.

Detection ratio (critical/non-critical nodes): The results for the detection ratio of critical (C) and non-critical (NC) nodes are shown in Fig. 2 (a-b) and Fig. 2 (c-d) respectively. Both Fig. 2 (a-b) clearly shows that the performance of LASCNN remains consistent and is not much affected while varying *N* and *r* which indicates scalability. Moreover, LASCNN performance in determining critical nodes is quite competitive compared with the globalized scheme. As expected, the ratio of determining critical nodes hone with the more network information. However, some nodes with localized information determine themselves as critical while they are not cut-vertices globally. More importantly, no critical node is missed. The number of critical nodes decreases with the increase in *N* and *r*. This is because connectivity of the network improves in both the cases, and thus alternative paths become available. Fig. 2 (c-d) shows the scalability of LASCNN while determining non-critical nodes. Moreover, LASCNN performance in identifying non-critical nodes is very close to a globalized scheme. This is because a node locally determined as non-critical is indeed globally non-critical. Fig. 2 (c) indicates that the performance of LASCNN almost matches with the globalized for sparse and dense topologies. In both the cases, alternative paths are shorter and are known locally. The performance of LASCNN slightly degrades with the long radio range as is evident from Fig. 2 (d). This is because it is almost impossible to discover long alternative paths through localized information.

Accuracy of determining critical/non-critical nodes: Fig. 2 (e-h) report on the accuracy of determining critical (C) and non-critical (NC) nodes respectively. Both the Fig. 2 (e) and (f) suggests that some nodes are inaccurately determined as critical while they are not globally cut-vertices. More importantly, no critical node is missed, and hence, such an inaccuracy is not a major concern compared to the overhead of maintaining network wide information at each node. Fig. 2 (e) shows that the maximum inaccuracy of determining critical nodes with 2-hop and 1-hop information is only 7% and 13% respectively. Similar observation can be made for Fig. 2 (f). The results in Fig. (e-f) indicate that the inaccuracy of determining critical nodes also depends on topology, density and radio range. Fig. 2 (g-h) shows the accuracy of determining non-critical nodes while varying the *N* and *r*. Both the Fig. 2 (g-h) clearly indicates that non-

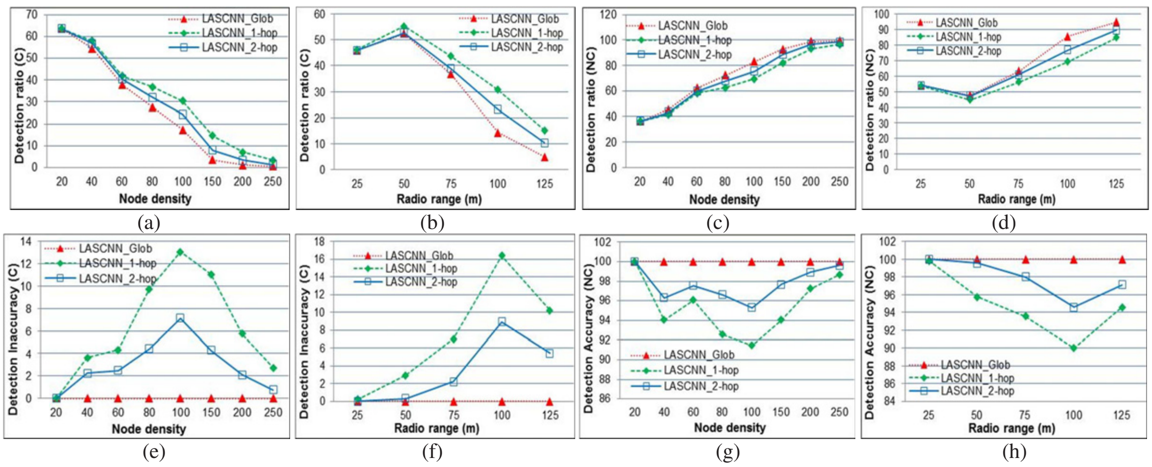

Fig. 2: Detection ratio of critical nodes, while varying *N* (a) and *r* (b). Detection ratio of non-critical nodes, as a function of *N* in (c) and *r* in (d). Detection inaccuracy of critical nodes, as a function of *N* in (e) and *r* in (f). Detection accuracy of non-critical nodes, while changing network density (a) and radio range (b).

critical nodes determined based on 1-hop and 2-hop information are indeed non-cut vertices globally as well. In other words, our localized scheme determines 91% of the non-critical nodes with 1-hop and 93% with 2-hop as is evident from Fig. 2 (g). Fig. 2 (h) provides similar results while varying *"r"*.

6. Conclusion and Future Work

This paper has presented LASCNN, a localized and distributed algorithm to segregate connectivity-centric critical/non-critical nodes. LASCNN requires each node to establish and maintain a k-hop neighbor connection list and process the neighbor's connection to check their connectivity. A node is determined as non-critical, if its neighbors stay connected, critical otherwise. LASCNN is validated through extensive simulations. Simulation results have confirmed the effectiveness and efficiency of LASCNN. We plan to validate LASCNN on prototype network of mobile robots.

References

[1] M. Imran, M. Younis, A. M. Said, and H. Hasbullah, "Partitioning Detection and Connectivity Restoration Algorithm for Wireless Sensor and Actor Networks," in *the Proceedings of the 8th IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC-2010)*, Hong Kong, China, 2010, pp. 200-207.
[2] D. B. West, *Introduction to Graph Theory (2nd Edition)*: Prentice Hall, 2001.
[3] F.Buckley and M.Lewinter, *A Friendly Introduction to Graph Theory*: Prentice-Hall, 2003.
[4] M. Duque-Anton, F. Bruyaux, and P. Semal, "Measuring the Survivability of a Network: Connectivity and Rest-Connectivity," European Transactions on Telecommunications, vol. 11 (2), pp. 149-159, 2000.
[5] D. Goyal and J. J. Caffery, "Partitioning Avoidance in Mobile Ad Hoc Networks Using Network Survivability Concepts," in the Proceedings of the 7th International Symposium on Computers and Communications (ISCC'02), Giardini Naxos, Italy, 2002, pp. 553 - 558
[6] X. Liu, L. Xiao, A. Kreling, and Y. Liu, "Optimizing overlay topology by reducing cut vertices," in in the Proceedings of the 2006 international workshop on Network and operating systems support for digital audio and video, Rhode Island, USA, 2006.
[7] S. Xiong and J. Li, "An Efficient Algorithm for Cut Vertex Detection in Wireless Sensor Networks," in the proceedings of the 2010 IEEE 30th International Conference on Distributed Computing Systems (ICDCS) Genova, Italy, 2010, pp. 368-377.
[8] M. Jorgic, I. Stojmenovic, M. Hauspie, and D. Simplot-ryl, "Localized algorithms for detection of critical nodes and links for connectivity in ad hoc networks," in the Proceedings of the 3rd Annual IFIP Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net), Bodrum, Turkey, 2004, pp. 360-371.
[9] A. A. Abbasi, M. Younis, and K. Akkaya, "Movement-Assisted Connectivity Restoration in Wireless Sensor and Actor Networks," IEEE Transactions on Parallel and Distributed Systems, vol. 20 (9), pp. 1366-1379, 2009.
[10] M. Imran, M. Younis, A. M. Said, and H. Hasbullah, "Localized Motion-based Connectivity Restoration Algorithms for Wireless Sensor and Actor Networks," Journal of Network and Computer Applications, vol. 35 (2), pp. 844-856, 2012.
[11] M. Imran and N. A. Zafar, "Formal Specification and Validation of a Hybrid Connectivity Restoration Algorithm for Wireless Sensor and Actor Networks," Sensors, vol. 12 (9), pp. 11754-11781, 2012.
[12] K. Akkaya, F. Senel, A. Thimmapuram, and S. Uludag, "Distributed Recovery from Network Partitioning in Movable Sensor/Actor Networks via Controlled Mobility," IEEE Transactions on Computers, vol. 59 (2), pp. 258-271, 2010.
[13] M. Imran, M. Younis, N. Haider, and M. A. Alnuem, "Resource efficient connectivity restoration algorithm for mobile sensor/actor networks," EURASIP Journal on Wireless Communications and Networking, vol. Nov, 2012.