Volume 6, Issue 7, July 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Approaches for Network Partitioning and Recovery in WSAN

Ankur Gupta
Asst. Prof, C.S.E, RIMT, Mandi Gobindgarh,
Punjab Technical University, Jalandhar,
Punjab, India

Rajwinder Singh
M.Tech, C.S.E, RIMT, Mandi Gobindgarh,
Punjab Technical University, Jalandhar,
Punjab, India

Abstract : ***WSAN often known as Wireless Sensor and Actuator Networks that are collection of heterogeneous elements adopted a wireless network to perform distributed sensing and acting tasks. The realization of actuator networks needs to satisfy the requirements introduced by the wireless network. In WSANs, sensors are deployed in an area used to collect the information about physical world, transmits to the actuator nodes for further actions upon the environment, which allows the user to sense and act from a distance. But the question arises about the failure of actuator nodes. Several reasons occurs such as due of battery exhaustion, message traffic overhead, hardware failure and many more and these failures may result the connected network into several segments that reduce the capability as well as life of network.***

Keywords: Wireless sensor and actuator networks; Wireless sensor networks, Connectivity restoration, Network Failure.

I. INTRODUCTION

Wireless Sensor and Actuator Networks have been receiving a growing attention because of their suitability for critical applications. They are mainly used in critical domains where continuous communication is desired to achieve in such areas like battlefield, disaster management, rescue operations, fire operations, space exploration, and home security and so on. In WSAN, failures of one or more nodes may disturb the communication network, loss of communication links, partitioning of network, stopping the actuators. Therefore, WSAN's should be fault tolerable as well as self recoverable.The WSAN should follow self healing approach to the recover from network failures. The sensors in a network collaboratively operate in harsh environment and decreasing the cost of application. Sensors are battery operated and have limited processing capabilities. Once they deployed the form a network automatically in order to share data and provide communication to each other.

Such as Border protection, the nodes need to be stay reachable to each other for survivors and assess damages.

WSAN consist of low cost sensing nodes that are responsible for collecting data from their surroundings and reporting to Actuator nodes over communication links. In inter-Actuator networks, Actuators need to coordinate with each other to process the sensor's data and then execute the plan. The Actuator's should decide the best solution among all the solutions.

This paper is structured as follows: section II, presents a brief review of the Application-oriented fault detection and recovery algorithm for WSAN. Section III represents overview of the Segregation of Critical/Non-critical Nodes in Mobile Ad Hoc and Sensor Networks. Section IV covers the Partitioning detection and connectivity restoration algorithm. Section V presents a localized self healing algorithm. Section VI represents a distributed recovery algorithm. Section VII covers the Recovery approach from Network Partitioning. Section 8 concludes the paper with summary.

II. APPLICATION-ORIENTED FAULT DETECTION AND RECOVERY ALGORITHM

The WSAN maintains inter Actuator connectivity in order to reach application level services. Whenever Failure occurs it may partition the inter network into disjoint segments. This paper proposed an application-oriented fault detection and recovery algorithm to re establish the connectivity by identifying the critical Actuators and assign backup for them. A backup node detects the critical node failure and starts the recovery process by moving them to optimal position.

In AFDR, the recovery of nodes has three visions [1]:

a) Recovery should be distributed
b) Rapid recovery of nodes is desirable
c) Energy and message overhead of recovery should be minimized

AFDR determines the partitioning of network in advance by first determines the cut vertex and then backup nodes. The communication range of Actuators has two radios for sensor-Actuator and Actuator-Actuator communication. Requirements of AFDR approach are as follows:

2.1 Critical and Non-Critical Nodes

The non-critical and critical nodes are collectively present in network. Losing the non-critical nodes doesn't affect the connectivity but failure of critical nodes will partition the network into disjoint sub networks.

Most of the existing approaches for the recovery of nodes are purely reactive and can be of two flavors such as block movement i.e requires high pre failure connectivity and cascaded movement i.e needs network state information of nodes for maintenance.

2.2 Arrival message matrix

AFDR, maintains Arrival message matrix that is collection of connections that a candidate has; to determine whether the candidate is cut vertex or not[2].

The main purpose of AFDR is the selection of backup or alternate node and failure detection.

First the cut vertices are identified and then select backup node so that it reacts instantaneously to the failure of cut vertex and avoid further partitions.

2.3 1-Hop Neighbor List

In AFDR, the 1-hop neighbors list is published for the selection of backup node. Each node is determined whether it is suitable for backup or not. For the backup, the node with the smallest flow is selected. All neighbors exchange HEARTBEAT messages to update their status. AFDR defines optimal position for failure recovery; that location through which the backup node can communicate with surrounding sensors as many as possible.

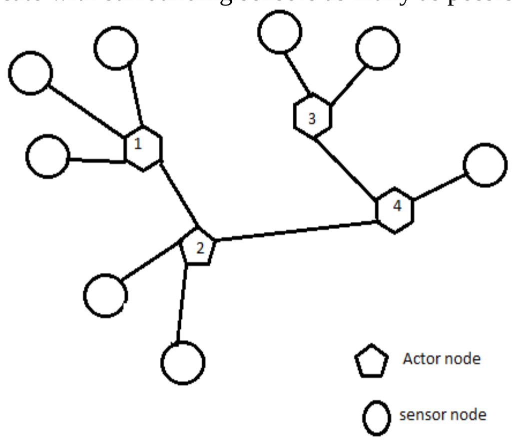

Fig I: A connected network with critical and non critical nodes

In the above figure, a connected topology consists of critical and non-critical nodes. The Actuator node 2 is a critical or cut vertex node of the network. If it fails then it partition the network into disjoint segments.

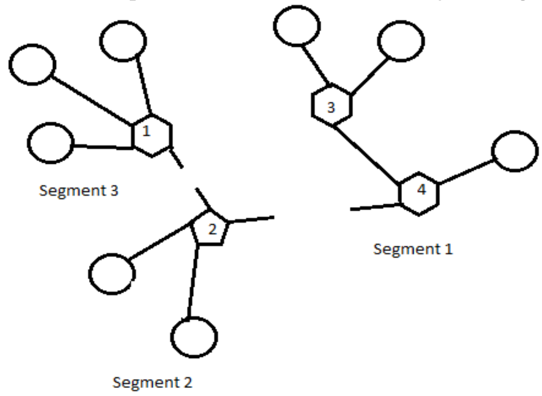

Fig II: Network partitioned into disjoint segments

III. LOCALIZED ALGORITHM FOR SEGREGATION OF CRITICAL/NON-CRITICAL NODES

The timely connectivity of centric critical/non-critical nodes is extremely crucial in MANET and sensor networks. Like MANET, the other adhoc network also requires the network connectivity as an essential constraint. Due to the dynamic nature, topology changes, limited energy and multi hop communication, MANET requires continuous network connectivity.

MANET'S are collection of nodes such as critical nodes and non-critical nodes. Critical nodes are found due to the random deployment, frequently movement of nodes and dynamic nature of connectivity. Whenever failure occurs, the critical node partitions the network into several segments and may slow the network operation [3]. The localized approach for the segregation of critical/non-critical nodes (LASCNN) is used to distinguish critical/non-critical nodes in the MANET. Constraints of LASCNN approach are as follows:

3.1 Centralized Scheme & Distributed Scheme

For determining the critical nodes, there are two types of schemes such as centralized and distributed [4,5]. In centralized scheme, up to date network wide information is required to maintain in advance, so this scheme isn't reliable for large scale dynamic networks. In distributed scheme, the nodes maintain a connection list based on k-hop information and employ the localized algorithm for segregation of critical/non critical nodes to analyze the list in determining critical/non critical nodes of network.

LASCNN checks the connectivity of nodes in the network based upon localized information. LASCNN algorithm for MASHN‘s is fast and quickly distinguish the faulty nodes with less computation and message overhead.

3.2 K-Hop Neighbor List

A connection list is maintained to determine critical/non critical nodes. If k-hop neighbors of node are still connected without it the node is critical otherwise not.

LASCNN works independently for 1 hop, 2 hop and k-hop information and identifies all critical nodes that are globally critical.

3.3 Detection & Restoration Schemes

The detection and restoration schemes can be flavored into three flavors like pre-partitioning that provides network assessment by identifying critical nodes in advance [2].

The post-partitioning algorithms react after the failure occurs in network [6] and execute restoration mechanism.

The hybrid algorithms separate critical and non-critical nodes in advance and start recovering the failure.

IV. PARTITIONING DETECTION AND CONNECTIVITY RESTORATION ALGORITHM

The inter Actuator network is a collection of critical/non critical nodes. Maintaining inter-Actuator connectivity becomes crucial in such situations where Actuators have to quickly change their topology. As Failure of critical Actuator partitions the inter-Actuator network into various segments, and disrupt the network operation.

Thus, a distributed partitioning detection and connectivity restoration algorithm called PCR is used to identify critical Actuators and designate appropriate backup nodes to handle their failure. A backup Actuator initiates a recovery process that may involve coordinated multi-Actuator relocation.

WSAN consist of number of sensor nodes that report directly to the Actuator nodes forming a connected topology. In this paper, The PCR (Partitioning detection and connectivity restoration) algorithm is hybrid in nature as it has two parts of implementation i.e. proactive approach and reactive approach. To find the critical Actuators of network proactive approach is followed with 1-hop positioned information. Once the critical Actuators are determined, appropriate backup nodes are found for replacement. Each backup node starts monitoring the primary node through HEARTBEAT messages.

4.1 Cut Vertex Detection

For identifying critical Actuators, several algorithms are adopted to identify cut vertices. These algorithms can be categorized into two categories:

Centralized algorithm requires each node to be aware of topology and involves huge message overhead.

Distributed algorithm detects CDS and requires 2-hop information. Therefore, PCR involves a simple localized cut vertex algorithm that only requires 1-hop information to detect critical nodes. The procedure runs on each node in a distributed manner to identify that a node is critical or not.

4.2 Backup Selection

The backup selection is a proactive approach of PCR. Once the primary nodes are identified the next step is to select an appropriate backup. The Actuators maintain 1-hop information to avoid extra overhead of communication. Though, backup Actuators are determined and notified before a failure of critical nodes. The first requirement of backup selection is that backup node should be non critical and the backup node should have least degree. The inter-Actuator distance degree close to backup node with least distance is selected to reduce the message overhead.

The backup node may be selected for more than one node in the network. PCR employs existing Actuator nodes to become backup. It doesn't require extra nodes for serving as backup.

Once the failure is detected, Actuator receives a backup message and starts monitoring the primary node through HEARTBEAT messages. Missing the number of consecutive HEARTBEAT messages indicates the failure of primary Actuator's node and initiates the recovery process.

4.3 Recovery Process

The recovery process is a reactive approach of PCR. If the backup node is non-critical Actuator then it simply replaces the primary node and recovery process would complete. But, if the backup node is also critical node then cascaded relocation is performed to avoid further partitions. The recovery process has three scenarios:

First, if a backup is a non-critical node then the scope of recovery will be fast with reduced message overhead. The backup node simply moves to the location of failed primary node and connect with new neighbors.

Second, when backup node is also critical node and cause further partitions if it moves to another location. In this case, the backup critical node inform to its own backup node so that the network stays connected.

Third, when the failed node and backup node, both are critical nodes and simultaneously serving one another. Then, the solution of this problem is cascaded relocation of nodes. In cascaded movement, the non-critical backup Actuator nodes replace the critical primary node and connectivity is restored.

V. LOCALIZED SELF HEALING ALGORITHM FOR MOVABLE SENSOR NODES

The wireless sensor networks (WSNs) deployed in search and rescue, battlefield, and other applications depends on inter-node interaction. The connectivity can be sustained through careful coordination, but the network can be partitioned if a node suddenly fails. Recovery through Inward Motion algorithm is effectively restore the network connectivity after a node failure. RIM initiates local recovery process by relocating the neighbors of the lost node.

The sensor nodes inform their neighbor Actuator nodes by moving but sudden node failures can disrupt the network. The loss of a node can break communication links in the network and nodes become unreachable. The worst case of network is that when the network may get partitioned into multiple disjoint segments. The sensors should be responsible for the recovery of failures.

5.1 Critical and Non-critical Nodes

It follows simple procedure to recover from both critical and non-critical failures in network without considering if the failed node is cut vertex. RIM applies to the network with collection of mobile sensor nodes either in a flat topology or hierarchical topology scheme.RIM assumes that nodes know their position relative to their neighbors but not exact neighbors.

The network connectivity depends on the failure node. If the failure node is leaf node then it doesn't affect the network connectivity but if the failure is critical or cut vertex node then it may partition the network into disjoint blocks.

5.2 1-Hop Neighbor Information

RIM approach aims to efficiently move nodes in order to restore the connectivity and require only 1-hop neighbor information. During network deployment, each node in a network broadcast a HELLO message to introduce itself to neighbors, and then build a 1-hop neighbor table to reflect the changes in the network. The table consists of node ID and its relative positions. Missing HEARTBEAT messages can detect the failed node.

RIM algorithm applies same recovery process to critical and non-critical nodes. the recovery start with 1-hop neighbor of failed node moving towards the failed node until they reach close to each other and again form the connected network[8].

VI. PARTITION DETECTION AND RECOVERY ALGORITHM

Wireless sensor and Actuator networks employs Actuator nodes within the wireless sensor network which can process the collected sensed data and perform certain actions based on the data. However, WSANs operate in harsh environments where Actuators can easily fail. Such failures can partition the inter-Actuator network into disjoint segments. To handle failures, a connected dominating set (CDS) based algorithm is proposed for partition detection and recovery. The algorithm identifies whether the failure causes partitioning of network or not in advance.

The wireless sensor network has some performance metrics such as network lifetime, throughput, coverage and connectivity. These metrics are important for the constant connectivity of topology.

PADRA (Partition Detection and Recovery Algorithm) is used to determine the partitioning in advance and provide self restore connectivity of case failures with minimum node movement and less communication overhead. Though, partitioning may occur by the failure of critical node that is serving as a gateway for multiple nodes. Such nodes are called critical nodes or cut vertex, so PADRA, determines whether the selected node is cut vertex or not in advance.

6.1 Connected Dominating Set Approach

This is done by utilizing connected dominating set of network. Greedy algorithm is used to detect the dominatee whose absence doesn't affect the connectivity of network. The nodes are replaced through cascaded movement where all the nodes of network are involved.

PADRA involves the recovery of only one node at a time. PADRA algorithm extends to MPADRA to recover multiple simultaneous nodes.

PADRA defines two goals that need to be achieved: First, whether a failure causes any partitioning. Second, if there is a partitioning, determine the set of movements to recover the failure.

To achieve the first goal, a CDS (connected dominating set) approach is followed to inform a particular node in advance whether partitioning will occur or not. If failure of node causes partitioning then set of nodes to be reorganized. The second goal is to examine such critical nodes and determine their movement. The movement of Actuator nodes is minimized to extend the lifetime of WSAN.

6.2 Depth First Search Approach

PADRA defines the approach DFS (Depth first search) [9] for determining cut-vertex or critical node in a network. This approach sends the flood of messages for the whole network, but it can be a costly approach as well as increase the message overhead.

6.3 DAI's Distributed Algorithm

PADRA uses DAI's distributed algorithm [10] to determine the CDS of inter-Actuator network. DAI's algorithm requires local information table that consist of 2-hop neighbors information. As a result, we need to calculate a dominator (critical node) or dominatee (non-critical node).

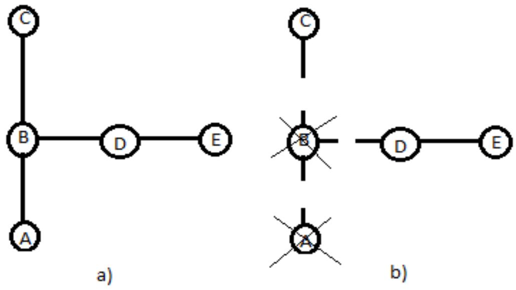

Fig III: Connected inter-Actuator topology

In the above figure, Actuator A is dominatee or non-critical node, failure of node A doesn't disrupt the network. The Actuator B is dominator or cut vertex, failure of node B partition the network into several partitions.
To maintain the connectivity of network, the alternate paths can be used during the failure. Since the detection of failure which causes the partitions of network, the alternate Actuator within the region is selected and replaces it with failed node to fix the connectivity of network, with fewer messages overhead.

6.4 Detection & Restoration Approach

As we know that, PADRA follows in advance approach for the detection of replacement nodes.
The main goal is to fix the connectivity of topology by determining the dominatee node or backup node with in topology in order to replace when failure occurs. Therefore, after the deployment of sensor and Actuator nodes in network, the CDS runs to determine the critical node or cut vertex. Each cut vertex node informs its backup node which will take care of its failure. In this way, the recovery time and overhead of network is minimized as the nodes will know in advance how to reach when failure occurs.
As we know that, to determine the closest dominatee, the greedy approach is followed. The closest dominatee among the neighbors of cut vertex with least node degree is selected as a backup node in case of failure of cut vertex. After the closest dominatee is determined, the relocation of nodes takes places. The movement of closest node will restore the connectivity as well as decrease the message overhead.

6.5 Cycling Problem

There is another problem occurs while movement of nodes in the network called cycling problem. PADRA also provides a method to detect the cycles of dominators in a network. To handle the cycles, an extra message ACK (acknowledgement) is to be sent before the movement of node. For replacement, the dominator should pick the dominatee which has not moved before.

VII. MULTIPLE PARTITION DETECTION AND RECOVERY ALGORITHM

In wireless networks, network connectivity needs to be maintained for exchange of data between sensor and Actuator nodes and perform data aggregation. However, connectivity can be lost due to certain failure of nodes. Even failure of single node may partition the network. The PADRA approach is followed to detect the partitions and then, restore the network connectivity. PADRA designates a backup node to restore the connectivity of network.
In such connected network, like MSN and WSAN failure of one or multiple nodes causes the disruption of multimode topology, which partition the network into disjoints sub networks, if alternate is not available for recovery.

7.1 Mutual Exclusion Approach

A distributed algorithm MPADRA introduces mutual exclusion approach to reserve the nodes in advance so that they cannot be moved for the recovery of other nodes.

7.2 Cut Vertex Determination

Like other approaches, DARA doesn't provide cut vertices of network. But MPADRA determines cut vertices in advance through CDS. There are n mobile nodes that are deployed in the area randomly and these nodes form a connected topology G.

7.3 Connected Dominating Set Approach

In the case of nodes, there are two goals need to be achieved. First, determine whether such failure causes partitioning within network. Second, if there is any partitioning then determine set of movements to restore the connectivity.

VIII. CONCLUSION

Approach	Capabilities
a) AFDR	Determines the partitioning of network in advance by first determines the cut vertex and then backup nodes.
b) LASCNN	Distinguish critical/ non critical nodes in MANET and works independently for 1hop, 2hop and k-hop.

c) PDCRA	Identify critical actuators and designate appropriate backup nodes to handle failures.
d) LSHA	Follows recovery through inward motion algorithm to restore network connectivity by relocating neighbors of lost node.
e) PADRA	Determine partitioning of network in advance and provide self restore connectivity with minimum node movement.
f) MPADRA	Introduces mutual exclusion approach to reserve nodes in advance so that they cannot be moved for recovery of other nodes.

REFERENCE

[1] K. Akkaya, A. Thimrnapuram, F. Senel, and S. Uludag, "Distributed recovery of Actuator failures in wireless sensor and Actuator networks," in Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '08), pp. 2480–2485, Los Vegas, Nev, USA, April 2008.

[2] X. Liu, L. Xiao, A. Kreling, and Y. Liu, "Optimizing overlay topology by reducing cut vertices," in Proceedings of the 16th Annual International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV '06), May 2006.

[3] M. Imran, M. Younis, A. M. Said, and H. Hasbullah, "Partitioning Detection and Connectivity Restoration Algorithm for Wireless Sensor and Actuator 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.

[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] A. A. Abbasi, M. Younis, and K. Akkaya, "Movement-Assisted Connectivity Restoration in Wireless Sensor and Actuator Networks," IEEE Transactions on Parallel and Distributed Systems, vol. 20 (9), pp. 1366-1379, 2009.

[7] F. Akyildiz and I. H. Kasimoglu, "Wireless Sensor and Actuator networks: Research Challenges," Elsevier Ad hoc Network Journal, Vol. 2, pp. 351- 367, 2004.

[8] M. Younis, S. Lee, S. Gupta, K. Fisher, "A Localized Self-healing Algorithm for Networks of Moveable Sensor Nodes," Proc. of the IEEE Global Telecommunications Conf. (Globecom'08), New Orleans, LA, Nov 2008.

[9] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms," Second Edition. MIT Press and McGraw-Hill, 2001.

[10] F. Dai and J. Wu, "An extended localized algorithms for connected dominating set formation in ad hoc wireless networks," IEEE Trans. On Parallel and Distributed Systems, vol. 15, no. 10, Oct. 2004.

