IRACST – Engineering Science and Technology: An International Journal (ESTIJ),ISSN: 2250-3498
Vol.4, No.3, June 2014

A Distributed LeDiR algorithm for WSA Networks

R.J.Egusha,
PG Scholar,
Department of Electronics and Communication Engineering,
Dhanalakshmi Srinivasan college of Engineering, Coimbatore, India.
eguegusha@gmail.com

B.Mathivanan,
Assistant Professor,
Department of Electronics and Communication Engineering,
Dhanalakshmi Srinivasan college of Engineering, Coimbatore, India.
bbmathi@gmail.com

E.Konguvel,
Assistant Professor,
Department of Electronics and Communication Engineering,,
Dhanalakshmi Srinivasan college of Engineering, Coimbatore, India.
konguart08@gmail.com

Abstract: **A wireless sensor actor network is a set of multiple connected components. The sensor node will collect information from surroundings and send it to actor nodes. The actor node will respond based upon the different applications. The correct response from the actor node indicates strong connectivity between the nodes. Meanwhile, fast transfer of data may result in failure of connectivity termed as cut. To overcome this problem, DCD algorithm is proposed, which is distributed and asynchronous. The DCD algorithm is based on the iterative computation of the nodes and it is used to recover from a multiple node failure. The convergence rate of the underlying iterative scheme is independent of the size and structure of the network. This algorithm also enables fast detection of node failures. The results are obtained and then compared by detailed simulations.**

Keywords: Application, Cut, Detection, Nodes, Fast, wireless sensor actor network.

I. INTRODUCTION

Wireless sensor and actor networks (WSANs) refer to a group of sensors and actors linked by wireless medium to perform distributed sensing and actuation tasks [1]. In such a network, sensors gather information about the physical world, while actors take decisions and then perform appropriate actions upon the environment, which allows remote, automated interaction with the environment. The actor's failure on the network topology can be very limited, e.g., a leaf node, or significant if the failed actor is a cut vertex. A node (vertex) in a graph is a cut vertex if its removal, along with all its edges, produces a graph with more connected components (blocks) than the original graph .The meaning of the term actor differs from the more conventional notion of actuator .An actuator is a device to convert an electrical control signal to a physical action, and constitutes the mechanism by which an agent acts upon the physical environment .From the perspective considered in this project, however, an actor, besides being able to act on the environment by means of one or several actuators, is also a network entity that performs networking-related functionalities, i.e., receive, transmit, process, and relay data .For example, a robot may interact with the physical environment by means of several motors and servo-mechanisms (actuators) [2]. However, from a networking perspective, the robot constitutes a single entity, which is referred to as actor. Hence, the term actor embraces heterogeneous devices including robots, unmanned aerial vehicles (UAVs), and networked actuators such as water sprinklers, pan/tilt cameras, robotic arms, etc. Applications of wireless sensor and actor networks may include team of mobile robots that perceive the environment from multiple disparate viewpoints based on the data gathered by a sensor network, a smart parking system that redirects drivers to available parking spots, or a distributed heating, ventilating, and air conditioning (HVAC) system based on wireless sensors .A wireless sensor network (WSN) consists of spatially distributed autonomous sensors to monitor physical or environmental conditions, such as temperature, sound, pressure, etc. and to cooperatively pass their data through the network to a main location. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance; today such networks are used in many industrial and consumer applications, such as industrial process monitoring and control, machine health monitoring. The WSN is built of "nodes" – from a few to several hundreds or even thousands, where each node is connected to one (or sometimes several) sensors. The topology of the WSNs can vary from a simple star network to an advanced multi-hop wireless mesh network. A sensor network typically consists of hundreds, or even thousands, of small, low-cost nodes distributed over a wide area. For example, many military applications strive to avoid any centralized and fixed points of failure [3]. In fact, node failure is expected to be quite common due to the typically limited energy budget of the nodes that are powered by small batteries. Failure of a set of nodes will reduce the number of multi hop paths in the network.

II. RELATED WORK

A number of schemes have recently been proposed for restoring network connectivity in partitioned WSANs. All of these schemes have focused on reestablishing severed links without considering the effect on the length of pre-failure data paths. Some schemes recover the network by repositioning the existing nodes, whereas others carefully place additional relay nodes.. But a node failure in the network may lead to partition of network topology into two or more disjoint sets making the nodes unreachable. This node failure can be recovered by autonomous reposition of the subset of actor nodes to restore connectivity. It can be achieved by two algorithms such as RIM (Recovery through Inward Motion) and DARA (Distributed Actor Recovery Theorem). On the other hand, some work on sensor relocation focuses on metrics other than connectivity, e.g., coverage, and network longevity.

III. RECOVERY THROUGH INWARD MOTION

A Recovery through Inward Motion (RIM) is a distributed algorithm it restore the network connectivity after a node failure [4]. RIM restores the connectivity of a WSN through the efficient repositioning of some of its nodes. RIM is a localized scheme that limits the scope of the recovery process. The main idea is that when a node fails, its neighbors move inward toward its position so they can connect with each other. RIM is simple and effective. It employs a simple procedure that recovers from both serious and non-serious breaks in connectivity, without checking to see if the failed node is a cut vertex [5] .RIM triggers a local recovery process by relocating the neighbors of the lost node. RIM minimizes messaging overhead and reduces the distance that individual nodes travel during the recovery.

IV. DISTRIBUTED ACTOR RECOVERY ALGORITHM

A Distributed Actor Recovery Algorithm (DARA), which opts to efficiently restore the connectivity of an inter actor network to its pre-node-failure level. DARA is a localized scheme that avoids the involvement of every single actor in the network. The idea is to identify the least set of actors that should be repositioned in order to reestablish a particular level of connectivity [6] .DARA strives to localize the scope of the recovery process and minimize the movement overhead imposed on the involved actors. The main optimization objective of DARA is to minimize the total distance traveled by the involved actors in order to limit the overhead incurred by the movement. In addition, DARA strives to minimize the messaging costs in order to maintain scalability.

V. LEAST-DISRUPTIVE-TOPOLOGY-REPAIR ALGORITHM (LeDiR)

The goal for LeDiR is to restore the connectivity without extending the length of the shortest path among nodes compared to the pre-failure topology. Least Disruptive Topology Repair Algorithm overcomes the disadvantages of High node relocation overhead (because many nodes involve in the recovery process) and Extension of inter-actor data paths relative to its pre-failure status. This algorithm is used to detect and recover a single node failure [7]. It is a localized and distributed algorithm which makes use of the existing route discovery activities. This method relocates the smallest number of nodes and there is no extension of data paths .The LeDiR algorithm first detect the faulty node and finds whether it is a cut vertex by using the partially populated Shortest-path Routing Table (SRT) and invocate the recovery process. Then the block with least number of nodes are found out for relocation which limits the recovery overhead. Then the faulty node is replaced by the neighbor (parent node) of the failed node which belongs to the smallest block [8]. After that the child nodes which are directly connected to the parent node are also relocated thereby recovering the connectivity.

The figure1 shows that LeDiR restores the network connectivity after the failure of a cut vertex (critical node). (a) WSAN before a cut vertex fails. (b) WSAN topology after applying LeDiR.

The following parameters are used to vary the characteristics of the topology in the different experiments:

1) Number of deployed actors (*N*): This parameter affects the node density and the WSAN connectivity. Increasing *N* makes the WSAN topology highly connected.
2) Communication range (*r*): All actors are assumed to have the same communication range *r*. The value of *r* affects the initial WSAN topology. While a small *r* creates a sparse topology, a large *r* boosts the overall connectivity.

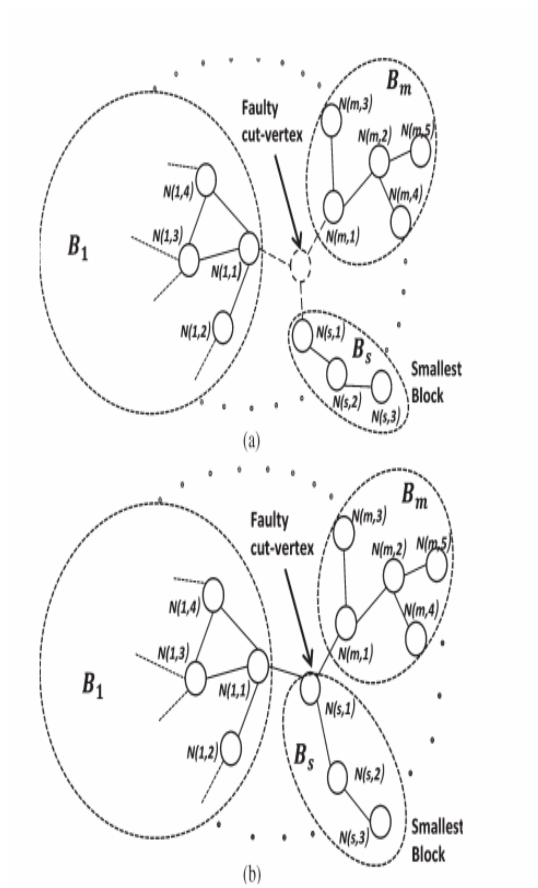

Figure 1 .LeDiR restores network connectivity.

VI. DISTRIBUTED ALGORITHM

A multiple node failure can be detected and recovered by distributed Algorithm. Monitoring large high spatial and temporal resolution of the networks and failure occur at the set of nodes will reduce the number of hop paths in the network. Such failures can cause a subset of nodes – that have not failed to become disconnected from the rest, resulting in a "cut" [9] .Two nodes are said to be disconnected if there is no path between them. Let us consider the problem of detecting cuts by the nodes of a wireless network. Assume that there is a specially designated node in the network, which we call as the source node. The source node is the base station that serves as an interface between the network and its users. When a node *u* is disconnected from the source, say that it is disconnected from Source event has occurred for *u*. When a cut occurs in the network that does not separate a node *u* from the source node say that Connected but a Cut Occurred Somewhere CCOS event has occurred for *u* .By cut detection it mean that 1) detection by each node when cut occurs, and 2) detection of CCOS events by the nodes close to a cut, and the approximate location of the cut. The location of a cut means the location of one or more active nodes that lie at the boundary of the cut and that are connected to the source. Nodes that detect the occurrence and approximate locations of the cuts can then alert the source node or the base station.

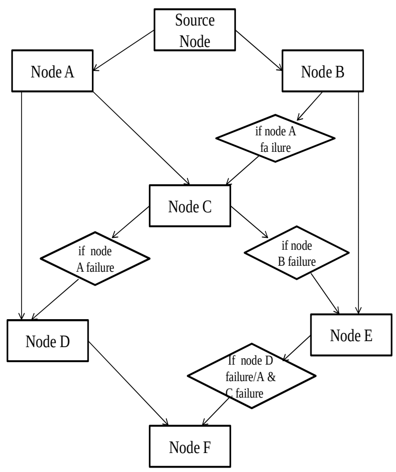

Figure 2: Activity Diagram

In the above diagram that will any failures of the node then it will be changed the root to the other to pass the information without having the lost of information or the delay of the information or delay of the message if the node fails. The information can pass the other root. The algorithm for detecting CCOS (Connected Cut Occurred Somewhere) events on finding a short path, and is partially inspired by the jamming detection [10] .When a node detects a large change in its local state as well as failure of one or more of its neighbors, and both of these events occur within a (predetermined) small time interval.

VII. IMPLEMENTATION

In this paper the proposed method consists of the following modules.

A. Distributed Cut Detection:
Distributed iterative computational step through which the nodes compute their electrical potentials. That will shows the where the cut will appear and solve.

B. Cut:
A wireless sensor network can get separated into multiple connected components due to the failure of some of its nodes which is called a “cut.”

C. Source node:
In this paper consider the problem of detecting cuts by the nodes of a wireless network. Since a cut may or may not separate a node from the source node, we distinguish between two distinct outcomes of a cut for a particular node.

D. Network separation:
Some nodes may separate from the network, that results the separated nodes cannot receive the data from the source node.

VIII. SIMULATION RESULTS

As mentioned earlier, LeDiR strives to restore network connectivity while minimizing the recovery overhead and maintaining the shortest path lengths at their pre-failure value. It groups the results into two sets first overhead related metrics and the second is path length validation metrics. It compare the performance of Distributed algorithm to Least-Disruptive-Topology-Repair Algorithm (LeDiR), Recovery through Inward Motion(RIM)and Distributed Actor Recovery Algorithm (DARA), which are the most effective published solutions for the tolerance of a multiple node failure in WSAN.

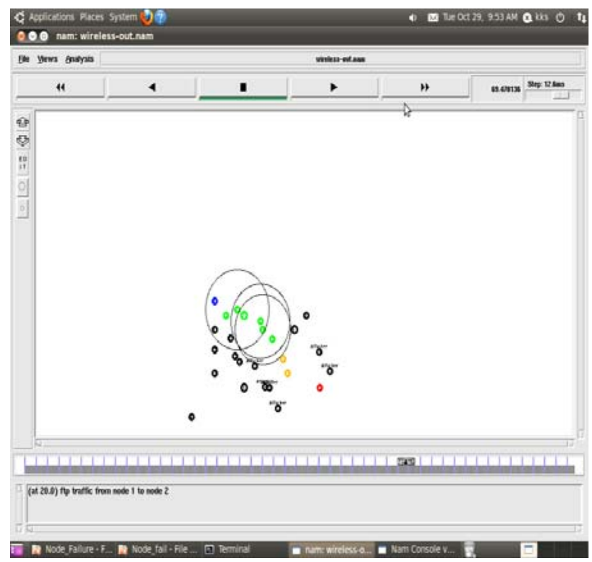

Figure 3a: Recover the node failure

The blue color node indicates the source node and the red color denotes the destination node .The green color nodes are responsible for choosing the path .Some of the nodes behaving as attackers in the network resulting in node failure.

The failure node can be detected and recovered in different topology. Number of extended paths per topology (average over 30 runs) after performing the recovery while varying the communication range (with N = 100).

Number of extended paths per topology (average over 30 runs) after performing the recovery while varying the network size (with r = 100).

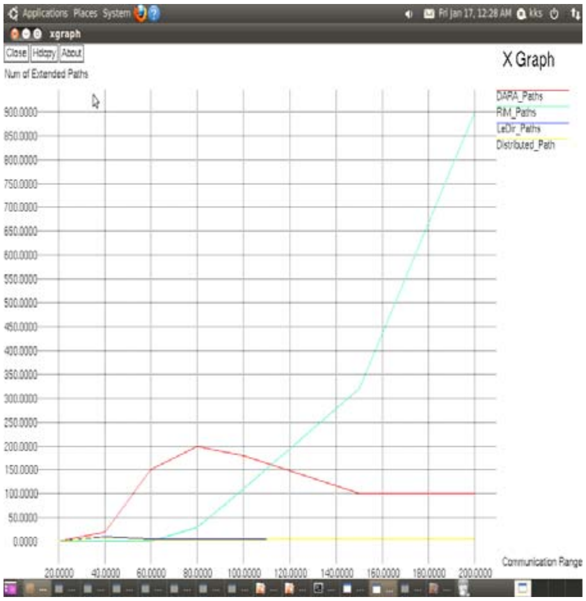

Figure 3b: Extended Paths Vs Communication Range

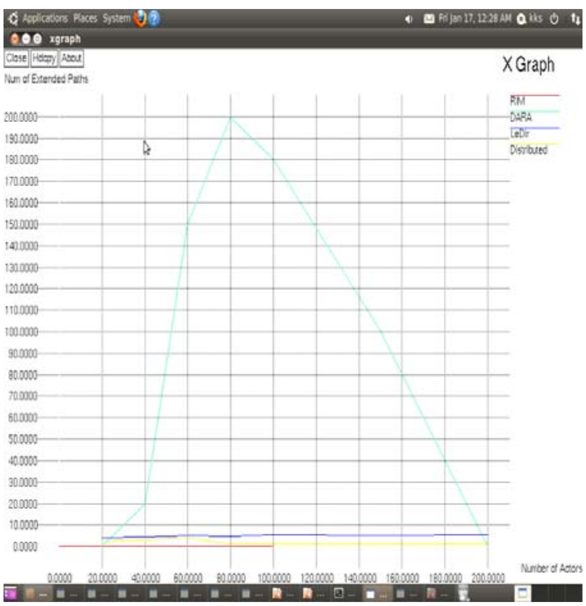

Figure 3c: Extended Paths Vs Actors

IX. CONCLUSION

In this paper a Distributed Algorithm is proposed for detecting a multiple node failure. Detecting cuts by the remaining nodes of a wireless sensor networks. Algorithm that allows every node to detect when the connectivity to a specially designated node has been lost and one or more nodes to detect the occurrence of the cut .The algorithm is distributed and asynchronous. Every node needs to communicate with only those nodes that are within its communication range. The algorithm is based on the iterative computation of a fictitious electrical potential of the nodes.

REFERENCES

[1]. I. F. Akyildiz and I. H. Kasimoglu, " Wireless sensor and actor networks :Research challenges," Ad Hoc Network, vol. 2, no. 4, pp. 351–367, Oct. 2004.

[2]. I. F. Akyildiz ,Y. Sankara subramaniam ,E. Cayirci, "Wireless sensor networks: a survey of ", Computer Networks, Vol. 38, pp. 393- 422 , 2002.

[3]. G .Cao ,G .Wang ,T. La Porta ,W. Zhang, " Sensor Relocation in Mobile Sensor Networks" in the Proceedings of 24th Annual IEEE Conference on Computer Communications (INFOCOM'05), Miami, FL, Mar. 2005.

[4]. D. Estrin ,J. Heidemann ,and N. Bulusu "GPS-less Low-cost Outdoor Localization for Very small Devices," IEEE Personal Communications , Vol. 7, No. 5, pp. 28–34, Oct. 2000.

[5]. M. Younis ,S. Lee ,S. Gupta , and K. Fisher, "A localized self-healing algorithm for networks of moveable sensor nodes," in Proc. IEEE GLOBECOM, New Orleans, LA, Nov. 2008, pp. 1–5.

[6]. M. Younis, and K. Akkaya , "A Distributed in Connectivity Restoration Algorithm in Wireless Sensor and Actor Networks," in the Proceedings of the 32nd IEEE Conf. on Local Computer Networks (LCN'07, Oct. 2007.

[7]. M. Younis ,A. Alfadhly and U. Baroudi, and, "Least distance movement recovery approach for large scale wireless sensor-actor networks," in Proc. Workshop FedSenS, Jul. 2011.

[8]. W. Zhang, G. Xue, and S. Misra, " The Fault-tolerant relay node placement in the wireless sensor networks: Problems and algorithms ," in Proc. 26th Annu. Joint Conf . INFOCOM , Anchorage , AK , May 2007 , pp. 1649–1657 .

[9]. S. Lee and M. Younis, "Recovery from the multiple simultaneous failures in the wireless sensor networks using minimum Steiner tree," J. Parallel Distributed Comput., vol. 70, no. 5, pp. 525–536, May 2010.

[10]. F. Senel, A.Akkaya, A.Thimmapuram, and S. Uludag, "Distributed recovery from network partitioning in movable sensor/actor networks controlled mobility," IEEE Trans Comput., vol. 59, no. 2, pp. 258-271, Feb. 2010.