RESEARCH ARTICLE OPEN ACCESS

Survey: Fault Tolerance Mechanism with Optimal Topological Changes in WSAN

Nisha M.Katre[1], Mangala S.Madankar[2], Punesh U.Tembhare[3]
Computer Science Department, RTM Nagpur University, Nagpur, India
[1]nishakatre44@gmail.com
[2]msmadankar@gmail.com
[3]punesh_tembhare@yahoo.com.

Abstract:
Wireless sensor and actor networks (WSANs) additionally employ actor nodes in the wireless sensor network (WSN) which process the sensed data and perform actions based on this collected data and Inter-actor coordination is required to get the good response. This suggests that the employed actors should form and maintain the connected inter-actor network. However, a failure of an actor may cause the network to partition into disjoint blocks and would thus violate such a connectivity requirement. In order to handle such failures, fault tolerance is done by either autonomously repositioning a subset of the sensor nodes to restore connectivity or addition of relay nodes. Here review is focused on different kinds of algorithms which can be used for the recovery process in WSAN with minimal topology changes.

Keywords— Fault tolerance, network recovery, topology management, wireless sensor actor network (WSAN)

I. INTRODUCTION

In WSAN there exist two types of nodes, sensors and actors. Sensors are inexpensive and highly constrained in energy and processing capacity and actors are more capable nodes with relatively more onboard energy supply and richer computation capability and communication resources. The impact of the actor's failure on the network topology can be limited, ex.a leaf node. 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 failure of an actor may cause the network to partition into disjoint blocks and would thus violate the connectivity requirement. To handle such fault tolerance various techniques are used like addition of relay node, node repositioning in centralized or distributed way.

Recovery through node repositioning and recovery by placement of relay nodes.

A. *Recovery through node repositioning*

The main idea is to reposition some of the healthy nodes in the network to reinstate strong connectivity. 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. For example, in Fig. 1, the network stays strongly connected after the loss of a leaf actor such as A_3 or a nonleaf node like A_6. Meanwhile, the failure of the cut vertex A_4 leaves nodes A_1, A_2, and A_8 isolated from the rest of the network. Further in this paper, the terms cut vertex and critical node will be used interchangeably. To handle the failure of a cut vertex node, two methodologies can be identified: 1) precautionary and 2) real-time restoration. The precautionary methodology handles fault tolerance by establishing a biconnected topology, where every pair of nodes A_i and A_j has two distinct paths with no common nodes other than A_i and A_j. Therefore, the network stays connected after a single node failure.

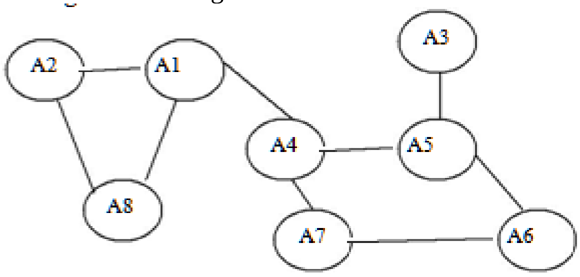

Fig 1: one-connected inter-actor network

The aim of this paper is to provide a survey of various fault tolerance algorithms used in WSAN. The paper is organized as follows: The overview of fault tolerance techniques is described in section 2 and finally Conclusion is done in section 3 at the end of paper.

II. AN OVERVIEW OF FAULT TOLERANCE SCHMES

A number of schemes have recently been proposed for restoring network connectivity in partitioned WSANs. All of these schemes have mainly categorized into two groups as

However, implementation such connectivity may require the deployment of a large number of actors and can thus be impractical due to the high cost. Also, it may constrain the mobility of actors and negatively affect application-level functionality. On contrary, real-time restoration implies a response only when a failure is detected. Real-time restoration better suits WSANs since they are asynchronous and reactive in nature, where it is difficult to predict the location and scope of the failure.

Some of the fault tolerance algorithms based on this technique are as follows.

1) Least-Distruptive topology Repair Algorithm(LeDiR)

LeDiR is generally designed for recover from a single node failure. The node failure may cause conflicting conditions for LeDiR to converge successfully. For multiple nodes failure at the same time, LeDiR tends to shrink the smallest block inward toward the failed node; it may negatively affect the node coverage. It cares for the path length between nodes [1]. This algorithm strives to restore network connectivity while minimizing the recovery overhead and maintaining the shortest path lengths at their prefailure value. This algorithm is almost insensitive to the variation in the communication range. LeDiR also works very well in dense networks and yields close to optimal performance even when nodes are partially aware of the network topology.

When a node fails, its neighbors will individually consult their possibly incomplete routing table to decide on the appropriate course of actions and define their role in the recovery if any. If the failed node is critical to the network connectivity, i.e., a node whose failure causes the network to partition into disjoint blocks, the neighbor that belongs to the smallest block reacts.

2) Partition Detection and Recovery Algorithm(PADRA)

It is connected dominating set (CDS) based partition detection and recovery algorithm [2]. The basic idea is to identify whether the failure of a node causes partitioning or not in advance. If such partitioning occurs then algorithm designates one of the neighboring nodes to initiate the connectivity restoration process. In this process consists of repositioning of a set of actors in order to restore the connectivity. In this restoration process is to localize the scope of the recovery process and optimize the movement overhead imposed on the involved actors.

PADRA handles the failure of any node in a connected wireless sensor actor network (WSAN). At the time of failure, CDS based technique of the inter-actor network can decide whether a node is a cut-vertex or not before the failure happens. Basically, if a node finds out that it is a cut-vertex, the closest dominatee/neighbor is delegated to perform failure recovery on behalf of that node. The failure recovery is done by determining the closest dominatee node and replacing it with the failed node in a cascaded manner so that the movement load is shared among the actors that sit on the path to the closest dominatee. This algorithm requires every node to maintain a list of their two-hop neighbors and determine the scope of the recovery by checking whether the failed node is a cut vertex. PADRA can find possible partitioning in advance and restore the connectivity and minimized node movement and message overhead.

3) Distributed Actor Recovery Algorithm(DARA)

DARA, a Distributed Actor Recovery Algorithm efficiently restores the connectivity of the interactor network that has been affected by the failure of an actor. Two variants of this algorithm are developed to address 1- and 2-connectivity requirements. The basic idea is to identify the least set of actors that should be repositioned in order to reestablish a particular level of connectivity. DARA focuses to localize the scope of the recovery process and minimize the movement overhead imposed on the involved actors.

DARA pursues a probabilistic mechanism to identify cut vertices. A best candidate (BC) is selected from the one-hop neighbors of the dead actor as a recovery initiator and to replace with the faulty node. The BC selection criterion is based on the least node degree and physical proximity to the faulty node [3]. The relocation procedure is recursively applied to handle any disconnected children. In other words, cascaded movement is used to sustain network connectivity. It does not care for the path length between nodes. It does not care for the path length between nodes.

The main optimization objective of DARA is to minimize the total distance traveled by the involved actors in order to limit the overhead by the movement. DARA strives to minimize the messaging costs and maintain scalability and entire recovery process is distributed and enabling the network to the self-heal without external supervision.

DARA generally describes two ways for WSANs, such as first one is DARA-1C and other is DARA-2C, are developed to address 1 and 2-connectivity. The goal of DARA-1C is to restore the network connectivity by exploiting the mobility of actor's nodes. The basic idea is to pick one of the neighbors of a failed node in order to replace it. DARA-1C strives to pick the neighboring actor and that will trigger the least number of cascaded movements. The total distance traveled by actors is minimized.

4) Recovery through Inword Motion(RIM)

In RIM, any lost link during the recovery will be reestablished through cascaded relocation of nodes. The collective effect seems like the network topology is shrinking inward. It reduces prefailure communication overhead [4]. Recovery through Inward Motion (RIM), a distributed algorithm is proposed to efficiently restore network connectivity after a node failure in the network. It is a distributed algorithm for Recovery through Inward Motion. RIM is a localized mechanism that limits the scope of the recovery process.

The basic idea is that if a node fails, then its neighbors move inward toward its position .The basic is that these neighbors are the ones directly impacted by the failure. They can reach to each other again, and the network connectivity is restored to its pre-failure status. The relocation procedure is recursively applied to handle any nodes that get disconnected when one of their neighbors moves. RIM is very 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. The recovery process is distributed and enabling the network to heal itself without external supervision.

TABLE 1: COMPARISON OF ANALYTICAL RESULT OF LeDiR, RIM, AND DARA [1]

Property	LeDiR	RIM	DARA
Maximum number of nodes to be involved	$\frac{N\ 1}{2}$	N-1	N-3
Maximum messages to be sent	$\frac{3N\ 1}{2}$	2N-1	5N-3
Maximum distance travelled by a nodes	r	r/2	r
Maximum distance travelled by all engaged nodes	$\frac{rN}{2}$	rN	rN

Table I provides a comparison of the analytical performance bounds for LeDiR [1] to those of DARA [3] and RIM [4]. As indicated by the table, LeDiR outperforms both baseline approaches when considering the recovery overhead at the network level in terms of the number of nodes participating in the recovery and the distance that these nodes collectively travel. RIM is better in terms of balancing the travel overhead on all nodes in the network yet matches LeDiR at the network level. It is very important to note that reiterate that unlike LeDiR, neither RIM nor DARA provides any guarantee on the internode path length.

5) Recovery through Inword Motion(RIM)

LDMR is a distributed recovery algorithm that exploits non cut-vertices in order to require the least travel distance from the engaged nodes in WSAN [5]. The recovery process starts with the search phase where each neighbor broadcasts a message containing several entries such as failed node ID, neighbor node ID and, Time-To-Live (TTL). Each neighbor chooses the best candidate among received responses based on a certain criteria (e.g. distance). The picked candidates replace the moved nodes without additional node relocation overhead. LDMR outperforms RIM for larger and sparse networks.

LDMR avoids the cascaded relocation by sending messages to find the replacement for the neighbors of F after they move It Reduced prefailure communication overhead. RIM provides better performance when the number of nodes is small. As the number of nodes increases, LDMR outperforms RIM.

B. Recovery through placement of relay nodes

It helps to reestablish connectivity (i.e., 1-vertex connectivity) using the least number of relays while ensuring a certain quality in the formed topology. Unlike contemporary methods that often form a minimum spanning tree among the isolated segments. Such a topology exhibits stronger connectivity than a minimum spanning tree and achieves better sensor coverage and enables balanced distribution of traffic load among the employed relays. For better robustness of the formed topology, the Spider Web approach is extended so that the final topology is guaranteed to be 2-vertex connected.

TABLE II: COMPARISON OF LeDiR, PADRA, DARA, RIM, and DORMS

Parameters	LeDiR	PADRA	DARA	RIM	DORMS
Node Repositioning techniques	Yes	Yes	Yes	Yes	No
Relay Node techniques	No	No	No	No	Yes
Shortest path length constraints consideration	Yes	Yes	Yes	No	No
Minimal topology changes consideration	Yes	No	No	No	No

The spider Web Heuristic optimizes the required number of relays nodes [7]. It requires minimum number of relays nodes. Spider Web and DORMS (Distributed algorithm for Optimized Relay node placement using Minimum Steiner tree) also strive to minimize the required number of relays in WSAN. Both Spider Web and DORMS deploy relays inwards toward the center of the deployment area. The former considers the segments situated at the perimeter and establishes a topology that resembles a spider web. DORMS initially forms a star topology with all segments connected through a relay placed at the center of the area. Then, adjacent branches are further optimized by forming a Steiner tree for connecting two segments and the center node to reduce the required relay count.

Table II provides a comparison of LeDiR, PADRA, DARA, RIM and DORMS. As indicated by the table, LeDiR performs with both path length constraint and minimal topology constraint.

III. CONCLUSION

In recent years, WSANs have growing attention due to their potential in many real life applications. In WSANs, reestablishing network connectivity after node failure without extending the length of data paths is serious problem. To deal with this problem several schemes are there. An overview of such schemes is provided in this paper. It is found that LeDiR algorithm is the only scheme that restores connectivity by careful repositioning of nodes so that optimal changes in the topology are made. Most of the fault tolerance schemes consider only single node failure considerations, but in real time scenario multiple node failure also occurs. Thus schemes based on multiple node failure considerations are needed for effective fault tolerance.

REFERENCES

[1] Ameer, Anabasis, Mohammed Youngish and Uthman Barodi "Recovering from Node Failure in Wireless Sensor-Actor Networks with Minimal Topology Changes", IEEE Transaction On Vehicular Technology, page no 256-271, Vol.62 Jan2013.

[2] A. Thimmapuram, K. Akkaya, F. Senel, and S. Uludag, "Distributed recovery of actor failures in wireless sensor and actor networks," in Proc. IEEE WCNC, Las Vegas, NV, Mar. 2008, pp. 2480–2485.

[3] M. Younis, and K. Akkaya "Movement-assisted connectivity restoration in wireless sensor and actor networks," IEEE Transaction on. Parallel Distributed. System. vol. 20, no. 9, pp. 1366–1379, Sep. 2009.

[4] M. Younis, S. Lee, S. Gupta, and K. Fisher, "A localized self-healing algorithm for networks of moveable sensor nodes,"IEEE GLOBECOM, New Orleans, November 2008.

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

[6] K.Akkaya and M. Younis, "Bio-inspired relay node placement heuristics for repairing damaged Wireless sensor networks," IEEE Transaction in May 2011.

[7] M. Younis, "Recovery from multiple simultaneous failures in wireless sensor networks using Minimum Steiner tree," J. Parallel., vol. 70, May 2010.