Adaptive Node Resurgence in Wireless Sensor-Actuator Networks

T Sridepa[#1], G Vijaybaskar [*2]

 # PGP College of Arts and Science, Namakkal, Tamil Nadu, India
** Assistant Professor, Department of Computer Science & Applications, PGP College of Arts and Science, Namakkal, Tamil Nadu, India*

***Abstract*— In wireless sensor-actor networks, sensors probe their surroundings and forward their data to actor nodes. Actors collaboratively respond to achieve predefined application mission. Since actors have to coordinate their operation, it is necessary to maintain a strongly connected network topology at all times. Moreover, the length of the inter-actor communication paths may be constrained to meet latency requirements. However, a failure of an actor may cause the network to partition into disjoint blocks and would, thus, violate such a connectivity goal. One of the effective recovery methodologies is to autonomously reposition a subset of the actor nodes to restore connectivity. Contemporary recovery schemes either impose high node relocation overhead or extend some of the inter-actor data paths. We overcome these short comings and present a Least-Disruptive topology Repair (LeDiR) algorithm with GSO. LeDiR + GSO relies on the local view of a node about the network to relocate the least number of nodes and ensure that the glow seeks for the neighbor set in the local-decision range.**

***Keywords*— Sensor-actor networks, LeDiR algorithm, GS Optimization, GSO Local range estimation.**

I. INTRODUCTION

Recent advances in wireless communications and electronics have enabled the development of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate over short distance. Wireless sensor networks (WSNs) are becoming a rapidly developing area in both research and application. Although it was originally driven by military applications, WSNs are now a day's being investigated and applied in many different civilian applications like vehicle tracking, forest surveillance, earthquake observation, biomedical or health care applications. A WSN consists of a large number of sensor nodes scattered in the region of interest to acquire some physical data. The sensor nodes have the capabilities of sensing, processing, and communicating. They operate in an unattended environment, with limited computational and sensing capabilities. Coverage is an important issue in WSN and is related to energy saving, connectivity, and network reconfiguration. It mainly addresses how to deploy the sensor nodes to achieve sufficient coverage of the service area so that every point in the service area is monitored at least by one sensor node. The sensors serve as wireless data acquisition devices for the more powerful actor nodes that process the sensor readings and put forward an appropriate response.

II. PROBLEM STATEMENT

An Actors usually coordinate their motion so that they stay reachable to each other. However, a failure of an actor may cause the network to partition into disjoint blocks and would thus violate such a connectivity requirement. The remote setup in which WSAN's often serve makes the deployment [3] of additional resources to replace failed actors impractical, and repositioning of nodes becomes the best recovery option [1],[2]. In addition, tolerance of node failure cannot be orchestrated through a centralized scheme given the autonomous operation of the network.

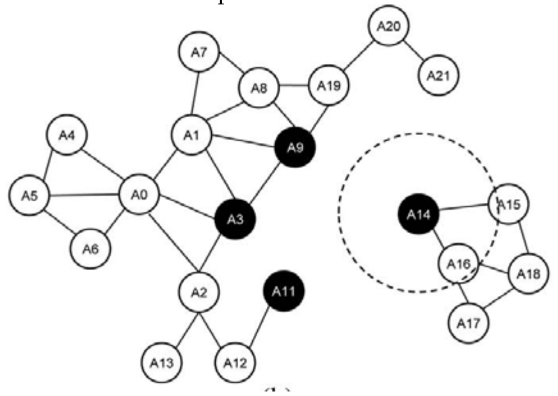


Fig 1.1 Node movement

On the other hand, distributed recovery will be very challenging since nodes in separate partitions will not be able to reach each other to coordinate the recovery process.

A. *Problem Definition:*

The transmission range of actors is finite and significantly less than the dimensions of the deployment area. We assume that the actors can move on demand to perform tasks on larger areas or to enhance the inter-actor connectivity [1]. Given the application-based interaction, an actor is assumed to know how many actors are there in the network. It is assumed that a sensor node can reach at least one actor over multihop paths and will not be affected if the actors have to change their positions. In prior work relies on maintaining one- or two-hop neighbor lists and predetermines some criteria for the node's involvement in the recovery. However, one-hop-based schemes often impose high node repositioning overhead, and the repaired inter-actor topology [4] using two-hop schemes may differ significantly from its prefailure status. In deterministic deployment [3], the sensor nodes are deployed in pre-determined locations so that the service area is completely covered and the network connectivity is maintained [1]. The impact of 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.

III. RELATED WORK

Node mobility has been exploited in various wireless networks (e.g., WSNs, WSANs, MANETs) in order to improve performance metrics such as network lifetime, throughput, coverage and connectivity.

Mobility helps improving dependability of these networks in several ways. For instance, failed nodes can be replaced with the other spare nodes (actors) [5] by moving the spare nodes to their locations. Similarly, if the network is partitioned, mobility can be exploited to restore connectivity by moving one or multiple nodes to the selected locations. Recently, the aforementioned dependability issues have been studied extensively in the context of WSANs. The main focus of these works was to deal with individual failures and restore the connectivity of the actuators [5] or the connectivity of the WSN with the sink node.

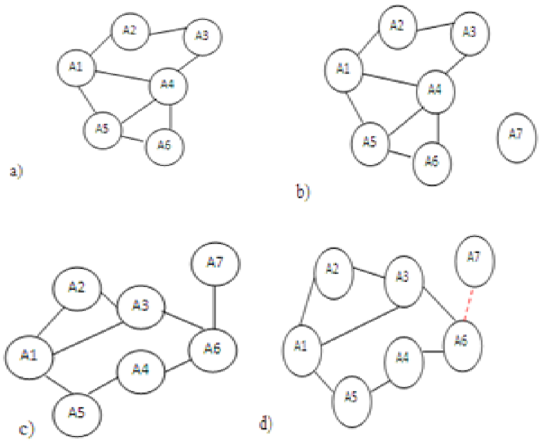


Fig 1.2 Mobility of nodes improving network connectivity and determine the failure nodes

A) NODE A1-A6 b) Node A7 c) A7 new Node connected with network group

B) d) Node A6 failure.

Connectivity of the WSAN is defined as follows: Let Xi = [xi1 + xi2] be the position coordinates of actuator (actor) i = 1, 2, ., n, after it is relocated. Connectivity between actuator i and j are determined by the following variable,

$$y_{ij} = \begin{cases} 1, & \|\mathbf{x}^i - \mathbf{x}^j\| \le r, \\ 0, & \text{otherwise.} \end{cases}$$

The appropriate actuators (actor) can be relocated to those locations afterwards and thus the network becomes connected again.

A. *Adaptive Node Recovery*

The main idea of this category of recovery schemes is to reposition some of the healthy nodes in the network to reinstate strong connectivity. For example, both Distributed Actor Recovery Algorithm (DARA) [1] and Partition Detection and Recovery Algorithm (PADRA) [2] require 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. DARA pursues a probabilistic scheme 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 the faulty node. The BC selection criterion is based on the least node degree and physical proximity to the faulty node. The relocation procedure is recursively applied to handle any disconnected children. In other words, cascaded movement is used to sustain network connectivity. It directly moves to the location of the failed node; instead, a cascaded motion is pursued to share the burden. Each node can estimate the cost by applying where *Rs* is the number of responses received by the searching node and Time to live (TTL) is the initial time to live value that used by the searching nodes. TTL also represents the maximum hops among all responses. All the searching node needs to estimate its cost.

$$Cost_{Cas} = dFS \times TTL \times Rs$$

B. *Adaptive Node Recovery Process:*

1) *Failure Detection:* If an actor AF is damaged or stops functioning, e.g., due to the transmission throughout failure, connectivity which should have been sent by AF periodically will cease. The interruption by each of the AF's neighbour's, ANi, as an indication of its failure. The recovery procedure will be executed regardless of whether AF is a critical node, i.e., a cut-vertex or not.

2) *Searching for non-cut vertices:* Each neighbour's in Step 1 and not within r/2 distance from AF starts a search process looking for the nearest non-cut vertex node, where r is the communication range. Each searching node broadcasts a search process containing several entities, such as its own ID, the failed node ID, Time-To-Live (TTL), which denotes the maximum number of hops the search spans. Each node that receives such a request sends a feedback if it is a non-cut vertex node. It also decrements the TTL value and forward the request if it does not reach zero. If two searching nodes are neighbour's, they will receive requests from each other. In this case, we can let each node discard the searching request from the other node or accept and forward the request. The connectivity restoration as a node placement problem on a network group and reposition the deployed nodes to meet varying requirements on the intersegment traffic.

3) *Least-Disruptive topology repair (LeDiR):* As mentioned earlier, LeDiR is a reactive scheme that

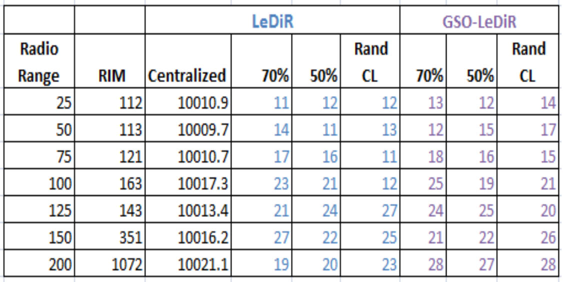
opts to restore connectivity while imposing the least travel overhead and in a distributed manner. LeDiR is to restore connectivity without extending the length of the shortest path among nodes compared to the pre-failure topology. The main idea for LeDiR is to pursue block movement instead of individual nodes in cascade. To limit the recovery overhead, in terms of the distance that the nodes collectivity travel, LeDiR identifies the smallest among the disjoint blocks. LeDiR limits the relocation to nodes in the smallest disjoint block to reduce the recovery overhead. The smallest block is the one with the least number of nodes and would be identified by finding the reachable set of nodes for every direct neighbor of the failed node and then picking the set with the fewest nodes is a challenging function.

IV. PROPOSED METHODOLOGY

A. *GSO based Least-Disruptive Topology repair*

The intelligent actions performed by a large number of relatively simple individuals. The independent individual follows some simple rules to coordinate operations. The movement of a glowworm operation is decided by the intensity possessed by its neighbors. Each glowworm is attracted towards the brighter glow of other glowworms in the neighborhood and decides to move towards one of its neighbors which have higher intensity.

TABLE 1.1 NUMBER OF SENT MESSAGES WITH VARYING ACTOR RADIO RANGE

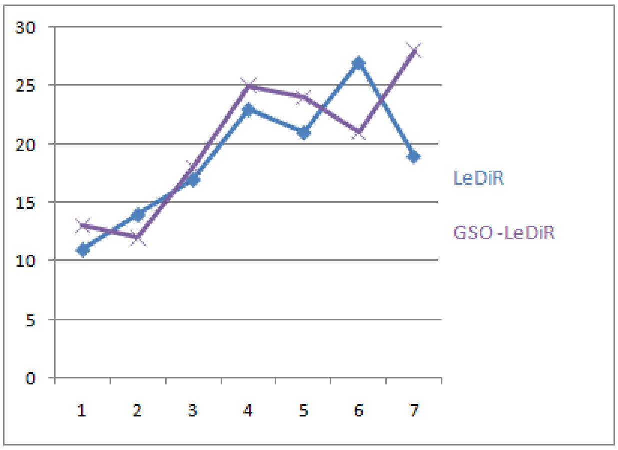


			LeDiR			GSO-LeDiR		
Radio Range	RIM	Centralized	70%	50%	Rand CL	70%	50%	Rand CL
25	112	10010.9	11	12	12	13	12	14
50	113	10009.7	14	11	13	12	15	17
75	121	10010.7	17	16	11	18	16	15
100	163	10017.3	23	21	12	25	19	21
125	143	10013.4	21	24	27	24	25	20
150	351	10016.2	27	22	25	21	22	26
200	1072	10021.1	19	20	23	28	27	28

Therefore the glowworms to partition into disjoint subgroups which represent the luciferin level of glowworm ‘i’ at time‘t’. For each glowworm ‘i’, the probability of movement towards a neighbor ‘j’ is given by,

$$p_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{k \in N_i(t)} l_k(t) - l_i(t)}$$

LeDiR involves only one-hop neighbor’s of F, which is denoted hereafter as Neighbors (F), glowworm in the process of node block selection and moves only the group node to Neighbors (F) that belongs to Bs.



Glowworm has decides to move towards one of its neighbors which have higher intensity. Conversely, GS-LeDiR leverages the available route discovery process and does not impose pre-failure messaging overhead.

V. CONCLUSION

In this paper, we have proposed a sensor deployment scheme based on glowworm swarm optimization, which can maximize the coverage of the sensors with limited movement after an initial random deployment. Thus the probability for multiple nodes to fail at the same time is very small and would not be a concern for LeDiR. The LeDiR tends to shrink the smallest block inward toward the failed node; it may unite with glowworm to navigate better Node selection. It considering such a problem with collocated node failure is more complex and challenging in nature. In the future, we plan to investigate this issue. Our future plan also includes factoring in coverage and ongoing application tasks in the recovery process and developing a test bed for evaluating the various failure recovery schemes.

REFERENCES

[1] *A.Abbasi, M.Younis, and K. Akkaya, “Movement-assisted connectivity restoration in wireless sensor and actor networks,” IEEE Trans. Parallel Distrib. Syst., vol. 20, no. 9, pp. 1366–1379, Sep. 2009*.

[2] *M. Younis and R. Waknis, “Connectivity restoration in wireless sensor networks using Steiner tree approximations,” in Proc. IEEE GLOBECOM, Miami, FL, Dec. 2010, pp. 1–5*.

[3] R.-S. Chang and S.-H. Wang, “Self-deployment by density control in sensor networks,” IEEE Trans. Veh. Technol., vol. 57, no. 3, pp. 1745– 1755, May 2008.

[4] K. Dasgupta, M. Kukreja, and K. Kalpakis, “Topology-aware placement and role assignment for energy-efficient information gathering in sensor networks,” in Proc. 8th ISCC, Kemer-Antalya, Turkey, Jun. 2003, pp. 341–348.

[5] M. Sir, I. Senturk, E. Sisikoglu, and K. Akkaya, “An optimization based approach for connecting partitioned mobile sensor/actuator networks,” in Proc. 3rd Int. Workshop WiSARN, Shanghai, China, Apr. 2011, pp. 525–530. (2002) The IEEE website. [Online]. Available: http://www.ieee.org/