

## Research Article

# Quality of Service-Based Node Relocation Technique for Mobile Sensor Networks

**Adnan Anwar Awan,<sup>1</sup> Muhammad Amir Khan ,<sup>1</sup> Aqdas Naveed Malik,<sup>2</sup> Syed Ayaz Ali Shah,<sup>1</sup> Aamir Shahzad,<sup>1</sup> Babar Nazir,<sup>3</sup> Iftikhar Ahmed Khan ,<sup>3</sup> Waqas Jadoon,<sup>3</sup> Naveed Shahzad,<sup>1</sup> and Rab Nawaz Jadoon ,<sup>3,4</sup>**

<sup>1</sup>Electrical and Computer Engineering Department, COMSATS University Islamabad, Abbottabad Campus 22060, Islamabad, Pakistan

<sup>2</sup>Electronic Engineering Department, ISRA University, I-10 Markaz, Islamabad, Pakistan

<sup>3</sup>Department of Computer Science, COMSATS University Islamabad, Abbottabad Campus 22060, Islamabad, Pakistan

<sup>4</sup>School of Information Science and Technology, University of Science and Technology of China, Hefei 230000, China

Correspondence should be addressed to Muhammad Amir Khan; [amirkhancomsats@gmail.com](mailto:amirkhancomsats@gmail.com)

Received 19 April 2019; Revised 7 July 2019; Accepted 16 July 2019; Published 22 August 2019

Guest Editor: Dionisis Kandris

Copyright © 2019 Adnan Anwar Awan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks (WSNs) deployed in harsh and unfavorable environments become inoperable because of the failure of multiple sensor nodes. This results into the division of WSNs into small disjoint networks and causes stoppage of the transmission to the sink node. Furthermore, the internodal collaboration among sensor nodes also gets disturbed. Internodal connectivity is essential for the usefulness of WSNs. The arrangement of this connectivity could be setup at the time of network startup. If multiple sensor nodes fail, the tasks assigned to those nodes cannot be performed; hence, the objective of such WSNs will be compromised. Recently, different techniques for repositioning of sensor nodes to recover the connectivity have been proposed. Although capable to restore connectivity, these techniques do not focus on the coverage loss. The objective of this research is to provide a solution for both coverage and connectivity via an integrated approach. A novel technique to reposition neighbouring nodes for multinode failure is introduced. In this technique, neighbouring nodes of the failed nodes relocate themselves one by one and come back to their original location after some allocated time. Hence, it restores both prefailure connectivity and coverage. The simulations show our proposed technique outperforms other baseline techniques.

## 1. Introduction

WSNs gained global attention in the present times. The building block of WSNs is a sensor node. These sensors nodes can measure, gather, and sense the information from the external environment. This information can be used for decision making by the end users. There are various areas of interests for WSNs applications; for example, military surveillance and target tracking, providing relief in a natural disaster, monitoring biomedical health, and exploration of the hazardous environment and seismic sensing [1, 2].

The sensing nodes in WSNs have low power, memory, and computational and radio capabilities. On the contrary,

these sensors must perform variety of thermal, mechanical, chemical, biological, magnetic, and optical tasks to measure the environmental properties. Furthermore, these sensors must also communicate from hostile/harsh environment via attached radio to the base station. The major power source of the sensor is the battery. The secondary power supply may be equipped such as the solar panel to gather the power from the external environment. An actuator can be attached to sensors, depending on the type of sensor and application [3, 4].

Typically, WSNs have little infrastructure. They may be composed of few sensors to thousands of sensor nodes for the prescribed activities. The researchers classified WSNs

into two categories: the structured and the unstructured WSNs. In the unstructured WSNs, sensor nodes are densely deployed. These sensors can be deployed in the field in an ad hoc manner. Once the network is deployed, it remains unattended to perform reporting and monitoring functions. In unstructured WSNs, maintenance of networks such as failure detection and connectivity management is done in a reactive manner because of random deployment of sensor nodes. Whereas in structured WSNs, the preplanned approach may be used for some or all sensors nodes. The main advantage of structured WSNs is the deployment of a limited number of sensor nodes with lower management cost and network maintenance. The drawback is that the deployment of limited sensors in larger areas may result into an uncovered region [5].

WSNs have their own resource and design constraints. A design constraint depends on the monitoring environment and the type of application. For example, short-range of communication, limited power, lower bandwidth, and limited storage and processing memory could result into different designs. The environment in which a WSN will be deployed also plays a vital role in the formation of the network size and network topology and the scheme of deployment. Monitoring environment directly affects the network size. The indoor environment required a smaller number of nodes for the formation of the network in a limited area while more nodes are required to form a network in the outdoor environment to cover larger area. The ad hoc deployment is better in a hostile/out-of-range environment where communication among sensors becomes limited due to the obstacles in the environment [6].

Multiple sensors may fail simultaneously due to the hostile/harsh environment. In addition, large-scale node failure involving multiple nodes may occur causing many disjoint segments in a WSN. Connectivity restoration is a greater challenge in this case as compared to the failure of a single node problem. In many cases, simultaneous node failure is not contiguous, yet it is expressively a much difficult problem. To the best of the authors' knowledge, three types of approaches have been proposed in the recent research literature to reclaim multiple node failures.

The first approach suggests that to restore connectivity, the topology of the network may be rearranged by repositioning sensor nodes from the original positions of the WSN. Such techniques support self-healing of WSNs and are used in a distributive manner. The second approach recommends that multiple-relay sensor nodes should be deployed to reinstate the multiple dismember segments. In the third approach, data is collected by the help of mobile mules, which make trips to the critical areas and transfer that data from one partition to another in a WSN [7].

In context of the above sections, the key contributions of this paper are elaborated as follows:

- (i) As the first contribution, a technique to dynamically reposition the neighbouring nodes on multinode failure is proposed to enhance network performance.
- (ii) The second contribution is energy efficiency of the proposed algorithm. Here, common neighbouring

nodes of the failure node must select one failure node within their neighbourhood. Therefore, these nodes do not have to travel more than one node, and this saves energy.

- (iii) The third contribution is the better performance in QoS parameters like total distance travelled by nodes, numbers of messages exchanged within all nodes, average node relocation, and reduction in the percentage of field coverage as compared to other baseline techniques.

The remaining paper is organised as follows: Section 2 highlights the relevant related work in a summarized manner. Section 3 discusses the details of the proposed algorithm for WSNs. The energy model used is discussed in Section 4, whereas the proposed algorithm validation and the simulation procedure are presented in Section 5. Section 6 concludes the paper and proposes some possible future work.

## 2. Related Work

The failure of the multinode problem has been addressed many times in the literature in the recent past. To handle the multinode failure problem, the centralized approaches provided better solutions. These approaches handle problem by developing a recovery schedule for multinode failure by relocating nodes in WSNs. Among these approaches, in [7], the integer linear problem (ILP) is formulated for the recovery problem. In this work, a node's individual travelling distance and coverage loss caused by multinode failure were minimized by the optimization-based ILP that formed the connected topology. In this technique, the position of the node is assumed prior to the failure in WSNs. In [8], a technique grounded on the network flow transportation model is used where normally every single node should have the capability to go to each destination of WSNs. Here, the problem was solved by the polygon approximation model and the formulation of mixed-integer approach. This approach is better from the approach provided in [9] in terms of total distance travelled. However, for more than 30 nodes, this technique does not measure accurately and results in errors. In [10], an alternative employment technique is suggested to reduce the number of node locations for improving scalability. It is a practice to find the number of positions that promise connectivity if the relay sensors are deployed at these locations. As alternative relay sensors, the calculated locations are failed by the survival nodes in the WSN.

In PADRA (partition detection and recovery algorithm), repositioned sensors are recognized by the CDS (connected dominating set) of each partition and select the optimal nodes for recovery [11]. A greedy heuristic method is used to minimize the total travelled distance of the sensors. In this technique, the nearest dominating pair of nodes are picked, and the dominate sensor is relocated to the target position. The CDS of partitioned WSNs are updated until the connectivity is to be restored. The result of this technique shows that the heuristic method outperforms the solution and scales well as compared to [7, 8]. The distributed techniques

are proposed for the recovery of a single node, and these techniques avoid the conflict which occurs during the recovery. The study by Imran et al. [12] was an extended version of DCR (design of partitioning detection and connectivity restoration) which handles multinode failures [9]. This extended technique was known as RAM (recovery algorithm to handle multiple failures) and selected critical sensors and designated backup for these nodes. The key parameter which enabled the RAM was the backup of the selection processes that handled the multiple simultaneous node failures. The idea was to improve the backup node, criticality backup, and primary deterioration at the same time. As the criticality improved, it warranted the designating backups in the recursive fashion. To avoid the race condition, mutex is used during the relocation of nodes to prevent the new failure. This technique shows the tolerance of non-located multiple nodes and adjacent node failure.

Similar approach as described above is used to tolerate the multiple sensor failures that occur at different locations simultaneously [13]. This technique uses the PADRA algorithm in multiple positions. Two failure handlers are used: primary and secondary. In the primary handler, there is no prespecified time for the recovery, although the relocation of the nodes in a cascaded manner was done. The two processes of recovery are required for the relocation of the same nodes to create the race condition at two different positions. It is different than a single node failure, in which the neighbouring node of the failed node has a deterministic and reliable interpretation of the problem. It is important to make some assumptions like centre region knowledge, route knowledge, and camera availability. The major theme of the technique AUR (autonomous repair) is based on the relocation/repositioning towards the preknown area centre and as per assumptions. Vigorous nodes regrouped by AUR and moved them towards the deployment centre area and towards one another [14]. The AUR design principle is based on the connectivity modelling between neighbouring sensor nodes. AUR considers the localized repossession with sensors that interact with immediate neighbours. The failed nodes stretched the intrasegment topology. If the restoration of connectivity is not done, the block segment is moved towards the centre deployment area. The node density in the centre point was increased by moving the segments towards the centre. This ensures the reestablishment of the connectivity. A distributed technique is used in the minimum Steiner tree (DORMS) for optimized placement of the relay node, which assumes that the network centre must be known beforehand [15]. The difficulty is tolerated as the relay appointment problem, but relay nodes are selected among the survival nodes in the WSN partitions. Hence, to diminish the number of obligatory relays, DORMS enhanced as Steiner minimum tree (SMT). K-LCA is employed by DORMS, which reduced the number of the node for finding the topology [16].

Yet another technique is proposed to handle multiple node failures by getting the knowledge of full paths of nodes to sink in [17]. For determining the location of the damaged node, prefailure route information is utilized. The position of the nodes and their path towards sink are then collected

after the establishment of the path. The DARA (distributed actor recovery algorithm) calculates the probability to identify cut vertices and select the neighbour node for the failed node. It relocates the neighbour node based on the communication link number [18]. This technique is further enhanced by the proximity factor to the node failure and then using PADRA for the optimization of the cascaded relocation in intrapartition [19, 20].

The technique in [21] is based on incomplete sensor failure information. The assumption of this work is to equip nodes with cameras to collect topology facts (normally the numeral of end nodes). In this technique, the function of remuneration is established on a node degree, where options to increase the node degree level are available. In [22], the authors proposed geometric skeleton-based reconnection (GSR). The approach split the network into various logical segments. A group of nodes that have the maximum connectivity with other nodes is known as geometrical skeleton backbone (GS backbone). The record of all the skeletal backbone nodes is maintained by each segment. In the case of network segregation, each segment attempts to join the GS backbone. In this way, connectivity can be restored. In [23], HRSRT (hybrid recovery strategy established on random terrain) for restoration of connectivity is recommended for impaired WSNs. The realistic terrain influence of area of interest (AOI) is considered in this algorithm. The terrain is planned by plotting the AOI and dividing it into a grid of cells of equal size. Each cell is characterized by the weight (cell). The weight of each path  $\omega$  (Path) is also considered by adding the weight of each cell (cell) along path  $\omega$ . By (cell) and (Path), the thorough graph “Kni” is built by taking the least weight of paths between the segments. RTPP (random terrain-based path planning algorithm) is established by “Kni” for creating a tour  $T$  for connectivity restoration of mobile data collectors (MDCs). Hence, MDCs were responsible for the restoration of connectivity. The SACR (survivability-aware connectivity restoration) for segregated WSNs by using mobile nodes is presented in [24]. The levels of load data of segregated segments for connection of the separated segments of the segmented network were considered by the SACR technique. These isolated segments could be found between different disconnected segments. The location of inaccessible segments is found by a group of moveable nodes to reinstate connectivity. A relay partition is created for every isolated segment. In [25], a robot control strategy for the provision of a connected path from the AOI to the base station is proposed by the authors. The core objective of this algorithm is to find out smallest distance having a smaller number of hop counts for mobile robots with unbroken network connectivity. The proposed strategy encompasses two algorithms. The first algorithm is to recognize the nearest robot to the event area and assigns that robot to such location. Then, this algorithm quests for the nearest nonallocated robot to ask it to forward itself to the communication range of any of the connected section having allotted robot. The algorithm proceeds on till the network is completely connected. The second algorithm discovers out those locations, which have minimum hop count between the event area and the base station. A method

for cut-vertex or critical-node determination is done in [26]. This technique comprises two localized distributed algorithms to determine the states of nodes that they are either critical or noncritical. Two-hop local subgraph and connected dominating set (CDS) information for the detection of most of the critical and noncritical dominator nodes is used by the first algorithm. The second proposed algorithm uses a limited distributed depth-first search algorithm in unrecognized parts of the network without going over the whole network. This algorithm determines the states of all nodes by comprehensive test bed experiments. Simulation results show that, in the presence of a CDS, this algorithm discovers all critical nodes using low energy consumption. Table 1 provides the summary of protocols discussed in the related literature above.

### 3. Details of the Proposed Algorithm

*3.1. Problem Specification and Proposed Solution.* The loss of multiple sensor nodes because of the destruction, battery drainage, or another malfunctioning not only disturbs the overall coverage of the network but also limits the connectivity of WSNs. The method of the proposed procedure for the reinstatement of network connectivity and coverage is initiated in the same manner as  $C^3R$  [27]. Our proposed solution takes the assumption that all sensor nodes are independent from each other, i.e., each sensor node takes data from surroundings and uses other sensor nodes to send these data to the sink node. The said algorithm in [27] suggests that the sensor nodes involved in recovery shall take turns by moving to and fro from their original position to failed node position. This would result in prefailure coverage. The problem arises in the said algorithm when the neighbouring nodes take part in the recovery process fail. If a neighbouring node of a recovery node fails, then a recovery node must take care of both failed nodes, i.e., the previous one and the latest one. So, the concerned node has to travel to and fro towards both failed nodes. This may drain its battery very rapidly and make such recovery node as a failed node. If a recovery node itself failed, then there may be some other recovery neighbour nodes. They also must take care of previous and latest failed nodes, and soon they will also become failed nodes due to battery drainage. Due to this problem, coverage will be decreased. To solve this problem, an energy-efficient relocation of a neighbouring node for a multinode failure method has been proposed in this paper. This method restores the prefailure connectivity as well as coverage in the network. As specified earlier, the failure of multisensor nodes causing the network to be disjointed is the most challenging issue and of utmost importance. Our solution suggests that if the neighbouring node of recovery node fails, then the recovery node has to decide to take care of only one node, previous or latest on the basis of its distance towards the failed nodes. The concerned node will take care of the failed node with a shorter distance. There might be a situation when there are two recovery nodes having the same distance to the latest failed nodes, then the node with higher ID will take care of the latest node and the lesser ID will stick to the previous node failure. There might

be a case when the recovery node itself fails due to some reason and it has neighbouring nodes which are part of the previous node failed recovery process. In such situation, such nodes must decide whether they take care of the previous node failure or latest failed recovery node based on their distance and coverage from both nodes. So, they will take care of the node having a smaller distance with them. This technique will save a lot of energy as the recovery nodes do not need to travel to two nodes for taking care of them.

*3.2. Prefailure Operations.* A prefailure 1-hop neighbours' list is maintained by a node in our proposed technique. This list is generated as soon as nodes start off after deployment. For an introduction to its all neighbours, a "HELLO" message is broadcasted by each node in the network. Besides this, each node must know the location and neighbouring nodes' IDs. The proposed algorithm uses normal GPS coordinates for locations of neighbouring nodes. This information of neighbouring nodes' locations is needed for use in case of failure of some sensor nodes. HEARTBEAT messages are sent to all neighbours in a timely manner to confirm their liveness. Therefore, if a sensor node, let us say A, does not receive its predetermined number of HEARTBEAT messages from the neighbouring sensor node, let us say F, then, consider F as the failed node. After node failure detection, a sensor node will send a message about its movement towards the failed node to each of its neighbour nodes. Each heartbeat contains node's ID and its location information. Each node sends HEARTBEAT messages after  $\tau$  seconds. On missing HEARTBEAT message, each node waits for some allowed number of heartbeats. As there is a chance, a HEARTBEAT message can be missed due to packet loss due to some unavoidable reasons. After the completion of countdown time, still if a neighbour's heartbeat is not received, then that neighbour is declared as failed node. These thresholds of  $\tau$  and countdown must be chosen wisely that it must not be short enough to declare an alive node failed, which would trigger unnecessary recovery process, and these thresholds must not be long enough that the recovery process gets delayed. We have not focused on optimization of these thresholds in this research article, as it is itself a research problem. Our focus is on the recovery process.

The recovery process is initiated by node A as soon as it comes to know that the neighbouring node, i.e., F has failed. It is notable that, in the literature, there are two approaches discussed when some node fails in the network. The first approach determines if the failure of node F will divide the network into disjoint partitions, and the response will be established only if the failed node is a cut-vertex node. This technique avoids the overhead of communicating all the nodes in the network. The drawback of this technique is that it requires 2-hop information to find cut vertices in the network; hence, messaging overhead is increased. The second approach proposes the restoration process when connectivity and coverage are vanished irrespective of the failed node is a cut vertex or not. The proposed technique uses the simplest technique used in PCR (partitioning detection and connectivity restoration) [9]. In which, each sensor node

TABLE 1: Protocol summary.

Reference	Main objective	Secondary goal	Approach type	State of network	Type of movement
[7]	Connectivity restoration	Minimize the distance and coverage aware	Centralized	Global	Controlled
[8]	Connectivity restoration	Minimize the maximum and total distance travelled	Centralized	Global	Controlled
[10]	Connectivity	Minimize the maximum and total distance travelled	Centralized	Global	Controlled
[12]	Connectivity	Minimize the recovery scope and total distance	Distributed	1-hop	Cascaded
[13]	Connectivity	Minimize the message cost and distance travelled	Distributed	2-hop	Cascaded
[14]	Connectivity	Minimize the total distance travelled	Distributed	1-hop	Block
[15]	Connectivity	Minimize the relays populated number and create an efficient topology	Distributed	None	Cascaded
[17]	Connectivity	Minimize the maximum and total distance travelled	Distributed	2-hop	Cascaded
[20]	Connectivity	Minimize the total distance	Distributed	2-hop	Cascaded
[21]	Connectivity	Minimize the total distance	Distributed	2-hop	Cascaded
[22]	Connectivity	Minimize the coverage loss and total distance travelled	Distributed	1-hop	Cascaded
[23]	Connectivity	Minimize the total distance travelled	Centralized	Global	Cascaded
[24]	Connectivity	Minimize the coverage loss	Distributed	1-hop	Controlled
[25]	Connectivity	Minimize the coverage loss	Distributed	1-hop	Controlled
[26]	Connectivity	Minimize the coverage loss	Distributed	1-hop	Controlled

determines itself as a critical or noncritical node by calculating the distance between their neighbours by GPS coordinates by the famous haversine formula (details of the formula are given in Section 3.5 of this research article). If the distance is less than the communication range ( $R_c$ ), then it declares itself as the noncritical node as its neighbours will stay connected on its failure. Otherwise, it declared it as the critical node. This technique will avoid all the recovery processes triggered for leaf nodes, and critical nodes will be determined in advance.

*3.3. Neighbours' Node Management.* As mentioned earlier that as momentarily as a node perceives that its neighbouring sensor node has failed, it quickly starts the process of restoration of prefailure connectivity and coverage. The first thing that the sensor node must need to know is whether there are any other neighbouring sensor nodes of the failed node which can take part in the recovery process. It is possible if this node and each of the other neighbouring nodes ought to travel to the location of the failed node till they come in the communication range of each other. These neighbouring nodes of the failed node must travel the distance of  $R_c/2$  towards the failed node. If their communication range is  $R_c$ , then they can connect with each other. As there is no solution provided in  $C^3R$  for neighbouring nodes to detect the node failure and react to such situation, when a recovery process is already going on. Therefore, some synchronisation of neighbouring nodes is needed for this situation. Besides this concern, neighbouring nodes could travel to the position of the latest failed node to synchronise. The travelling distance will be increased as there will be some common neighbouring nodes participating in the previous node fail recovery process for which the recovery process is going on. Such common nodes must travel to a previous node failure location for its recovery process and to travel to

the latest node failure location. This will result in an increase in overhead, and their battery will be drained off very soon. Our technique gives the solution that such common nodes must decide to restrict them either with a previous failed recovery node process or with the latest failure node process based on the minimum distance and the maximum coverage area from the failed nodes. The calculations of distance and coverage area and their relationship with each other are described in detail in the next section.

*3.4. Distance between Nodes and Coverage Area.* The sensor node which reaches to the location of the failed node becomes the coordinator of the recovery process. If two nodes reach the failed location at the same time, then the node with lesser ID will become a coordinator. The role of the coordinator is to develop a recovery plan and distribute the turns of each node participating in the recovery process. The criteria for selecting the nodes taking part in the recovery procedure are constructed on the coverage overlap area of the nodes, the distance between the failed node, and the remaining energy they have. The sensor nodes must reckon all these parametric values as they are criteria for selection in the recovery process. Then, these values are shared with the coordinator of the recovery process. The coverage overlap of two sensor nodes is the intersectional area of their communication range (Figure 1). The calculation of coverage overlap becomes very simple when the disk coverage model is taken in to account. There is an inverse proportional relationship between the distance between sensor nodes and the coverage overlap. When the distance between nodes is smaller, then there will be a larger intersection between two circles around the sensor nodes, thereby increasing the coverage overlap. Two neighbour nodes will have coverage overlapped if the distance ( $d$ ) between them fulfils the condition  $d \leq 2 \times R_c$ . For the distance " $d$ " between two

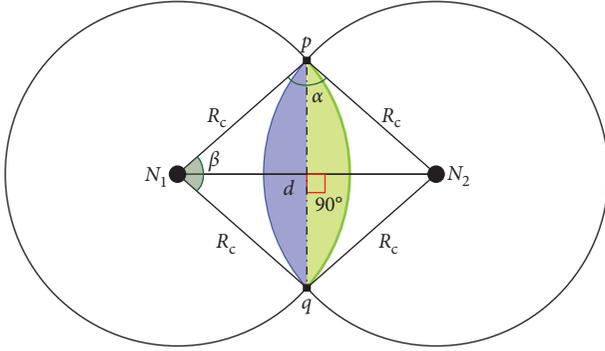


FIGURE 1: Calculation of overlapped coverage area of nodes N1 and N2.

nodes, the following equations can be derived by using GPS coordinates of a sensor node 1 and its neighbour sensor node 2 by the famous haversine formula:

$$\begin{aligned}
 d\text{long} &= \text{long}_2 - \text{long}_1, \\
 d\text{lati} &= \text{lati}_2 - \text{lati}_1, \\
 a &= (\sin(d\text{lati}/2))^2 + \cos(\text{lati}_1) * \cos(\text{lati}_2) \\
 &\quad * (\sin(d\text{long}/2))^2, \\
 c &= 2 * a \tan 2(\sqrt{a}, \sqrt{1-a}), \\
 d &= R * c,
 \end{aligned} \tag{1}$$

where  $\text{lati}_1$  is the latitude of node 1,  $\text{lati}_2$  is the latitude of node 2,  $\text{long}_1$  is the longitude of node 1,  $\text{long}_2$  is the longitude of node 2,  $d\text{lati}$  is the difference of  $\text{lati}_1$  and  $\text{lati}_2$ ,  $d\text{long}$  is the difference of  $\text{long}_1$  and  $\text{long}_2$ , and  $R$  is the earth's radius (mean radius = 6,371 km); note that angles need to be in radians.

From Figure 1, the overlap of two nodes N1 and N2 can be calculated, if the area of the chord ( $pq$ ) have angle  $\beta$  also called as area ( $\beta$ ). The distance between nodes N1 and N2 is found at the start of nodes after deployment by GPS coordinates. By using the law of cosines angle,  $\alpha$  can be found as

$$\angle \alpha = 2 \sin^{-1} \frac{d}{2R_c}. \tag{2}$$

The area ( $\beta$ ), the area of the sector " $pq$ ," depicted by blue shade in Figure 2 can be found by using the right-angled triangle and using  $\beta/2$ :

$$\text{area}(\beta) = \frac{\beta}{\pi} R_c^2 - \frac{d}{2} R_c \times \sin\left(\frac{\beta}{2}\right). \tag{3}$$

The overall coverage area can be found by multiplying twice the area of sector depicted by blue shade as the sector area in the blue shade is equal to the sector area in a light green shade.

$$\text{Overlap} = \left( 2 \times \frac{\text{area}(\beta)}{\pi R_c^2} \right). \tag{4}$$

**3.5. Proposed Algorithm Implementation.** The implementation of the proposed algorithm can be best understood by the scenario given in Figure 2, by considering the scenario depicted in Figure 2(a) where node n7 has failed. The steps for restoration of connectivity and coverage of the proposed algorithm are explained below.

- (i) Initially, the neighbouring nodes of failed node n7, which are n2, n3, n4, n6, n8, and n10, will detect that n7 has failed due to missing HEARTBEAT message from node n7 and become recovery nodes.
- (ii) Then, these recovery nodes will calculate the distance with n7, its coverage overlap, and remaining energy before they begin to move towards the position of n7.
- (iii) Each recovery node sends a message of temporary relocation to their immediate neighbouring nodes and asks their neighbours to find alternating route or buffer their data until such nodes return to their original position.
- (iv) All the recovery nodes move towards n7 to get connected with each other. The recovery node which reaches the location first will become a recovery coordinator. If two recovery nodes reach at the same time, then the node with the lowest ID will become the recovery coordinator.
- (v) The list of recovery nodes according to their priority with regard to the parameters like the overlap of coverage, distance travelled during relocation, and amount of energy in remaining is maintained by the coordinator node.
- (vi) After this recovery, the node on the top of the list, suppose n2, will stay at the location of n7 and rest of the neighbours will return to their respective positions and stay there until their turn.
- (vii) After some time, the recovery node on the top of the list returns to its position and the second on the list will resume the position of n7. On the returning to the original position, a recovery node informs its neighbour by broadcasting message to them and begins to establish the transmission of the buffered data again. The same node will repeat a like preset process after that.
- (viii) Once the energy of a recovery coordinator node reaches below the specified threshold level, the coordinator will transmit a request to other recovery nodes. The recovery node presently situated in the locality of n7 will accept the request. The request-receiving node will become a new recovery coordinator, travel to previous coordinator position, and generate the new schedule.
- (ix) There might be another situation such that if a neighbouring node of some recovery node fails during the recovering process of some other nodes, e.g., in Figure 3(a), recovery process of n7 is going on and n10 fails. Then, the neighbours of n10

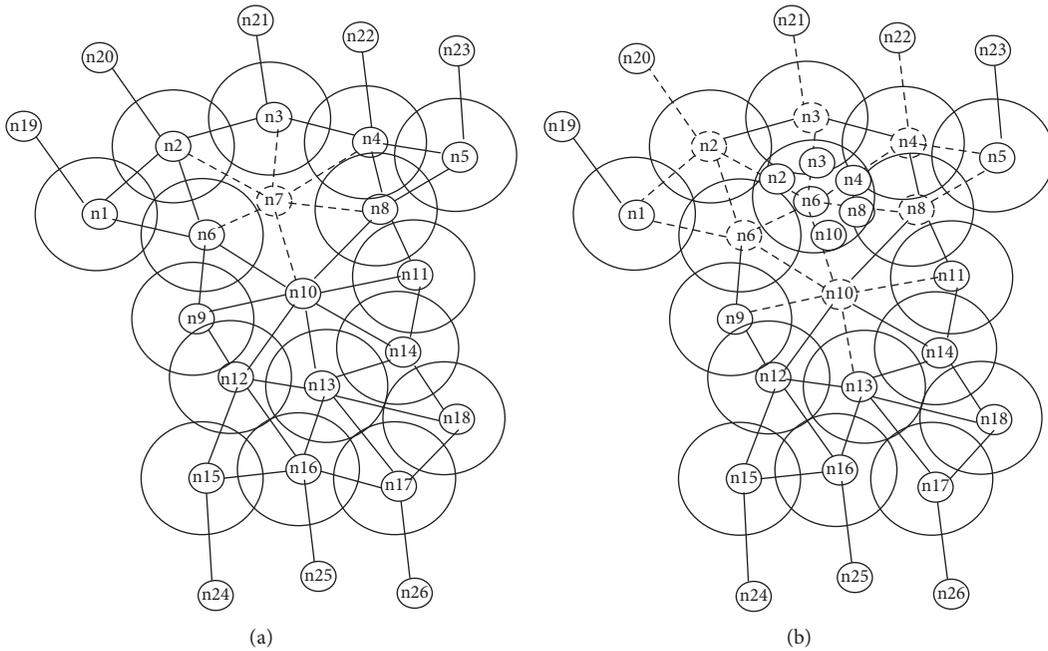


FIGURE 2: (a) First node failure. (b) Recovery process after the first node failed.

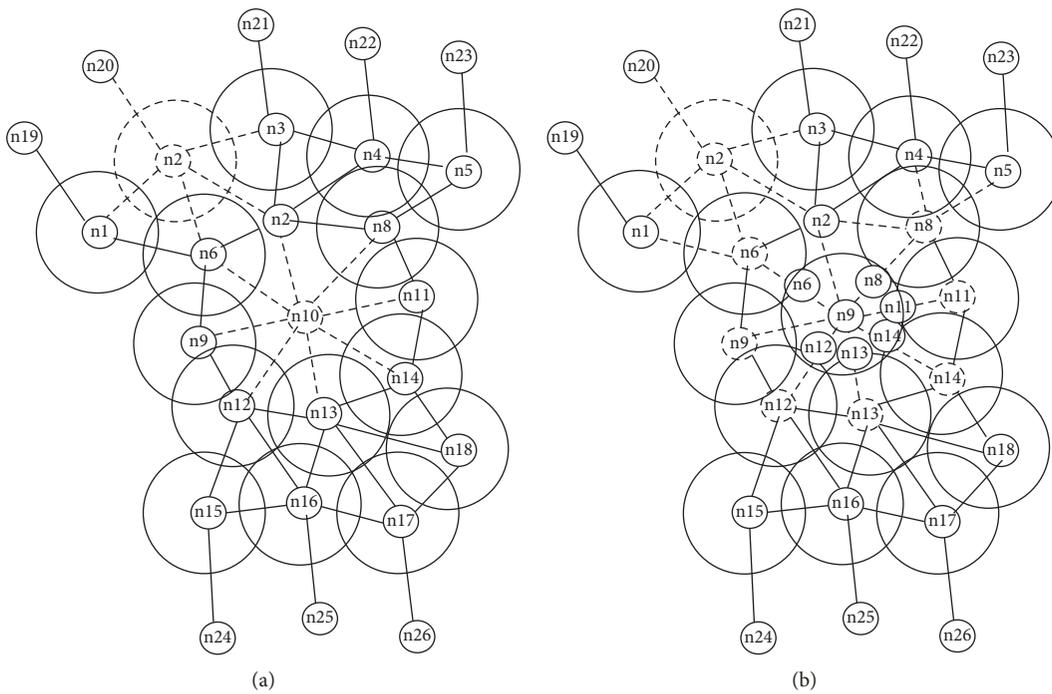


FIGURE 3: (a) During the recovery process, another node failure happened. (b) Recovery process after the failure of the latest node.

which are n2, n6, n8, n9, n10, n11, n12, n13, and n14 will have to take part in the recovery process. However, n2, n6, and n8 must decide whether they remain to restrict with the n7 recovery process or take part in the n10 recovery process.

(x) For the decision purpose, the one having lesser distance amongst n6, n8, and n9 from n10 will take

part in the recovery process of n10. Besides this, the node which is already at the place of the previous failure node, i.e., n2 in the example cannot take part in the latest node failure recovery process. The selected node will take part in the latest node failure and will detach itself from the recovery process of the previous node failure.

- (xi) The recovery coordinator of the previous node failure will update the recovery list by excluding the entry of a selected node, which has detached itself from such a recovery process.
- (xii) If the recovery coordinator of the previous node failure is self-selected as the recovery node for latest node failure, then it will inform other recovery nodes taking part in the previous node failure recovery, so they can nominate any node as their new coordinator.
- (xiii) Figure 3(b) shows that n2 will remain at the failed location of n7, whereas n6 and n8 will compete for the recovery node of n10 and the node with lesser distance will win the competition.
- (xiv) If both n6 and n8 have same distance with failed node n10, then higher ID will become the recovery node for latest node failure and lesser ID node remains as the recovery node for the previous node failure (Algorithm 1).

#### 4. Energy Model

An energy model which we have used in this paper is presented in [28]. The energy consumption of a node to transmit and receive a “B-bit data packet” over distance “ $d$ ” is shown in equation (5). The energy expended per bit by the receiver circuit is given by (6). Whereas the residual energy can be calculated by equation (7).

$$E_{Tx}(B, d) = \begin{cases} (E_{elec} + \epsilon_{fs}d^2)B, & d < d_0, \\ (E_{elec} + \epsilon_{mp}d^4)B, & d \geq d_0, \end{cases} \quad (5)$$

$$E_{Rx}(B) = E_{Rx-elec} B, \quad (6)$$

$$E_{reng}(n) = E_{max} - E_{Tx}(B, d) - E_{Rx}(B). \quad (7)$$

Different terms used in this model are explained in Table 2.

#### 5. Simulations and Results

All the simulations were carried out using the platform of OMNet++ for the performance appraisal of the proposed algorithm. A comparison is made between the proposed algorithm with and increased robustness against recurrent failure (RIR) of damaged WSN topologies in the event of multiple node failures [25] and autonomous repair (AuR) [26] protocols. This section presents niceties of the simulation setup and the discussion of the results.

*5.1. Simulation Parameters.* The accomplishment of the proposed algorithm is authenticated through simultaneous simulations. This section discusses the simulation setup, performance metrics, and results. For the validation of the results, the OMNet++ platform is used. All the simulation parameters are given in Table 3.

All the results are subjected to 90% confidence analysis interval and stay within 10% of a simple mean.

*5.2. Results and Discussion.* For the analysis and performance comparison of the parameters such as the number of messages exchanged, distance travelled by nodes, number of nodes relocated, and reduction in the percentage of field coverage and recovery time consumption of the proposed algorithm, RIR and AuR were employed.

*5.2.1. Distance Travelled.* The total distance that the nodes travelled while performing the recovery process is presented in Figure 4. The sensing and communication ranges are taken as equal in all these series of experiments. How far the node should travel depended on nearness of one node to another node. This vicinity was at most the  $R_c$  communication range. Therefore, an increase in  $R_c$ , there was an increase in how far a sensor node travelled. This was easy in cases of AuR and RIR as the distance travelled increases rapidly. Unlike AuR and RIR, with the proposed technique, the sensor node participation was limited in the recovery process to the neighbours of the sensor node that failed. It does not use cascaded relocation that is initiated in AuR and RIR. It is very significant to designate that when communication range ( $R_c$ ) is too high, the proposed procedure shows the growth in connectivity of the WSN very well.

*5.2.2. Number of Packets Exchanged.* The number of packets that were exchanged, i.e., received and delivered during the restoration of connectivity using each of the three methods is presented in Figure 5. Every broadcast message was a single message. Using the proposed technique, messaging overhead was negligible. However, using AuR provided the maximum number of the packets exchanged. The reason behind this is that, in the proposed technique, only neighbour nodes of the failed node were involved in the restoration process. On the contrary, with AuR and RIR, the message exchange had to coordinate their achievement with a total number of nodes that were relocated. However, there was an increase in network connectivity for large  $R_c$ . This is because of the interaction which rarely takes place between other involved sensor nodes and recovery coordinators. The proposed technique can be scaled up for the several rounds. Also, it would offer coverage and connectivity reestablishment at the sensible price. Comparative performance has shown improvement in the proposed solution. Hence, by Figure 5, it is confirmed that a minute messaging overhead is added by the proposed procedure.

*5.2.3. Number of Relocated Nodes.* Figure 6 shows the recovery process of the total number of nodes relocated in the algorithms used for the comparison purpose with the proposed algorithm. For this purpose, the total distance travelled will increase with a greater number of travelling nodes. From Figure 6, the result shows that the node movement of RIR is greater than AuR. In any circumstances in the proposed technique, there are a smaller amount of node movements because the proposed algorithm restricts the time of the recovery process to the neighbour nodes of the failed node.

```

Input: distances of neighbouring nodes
(1) IF (node,  $A$ , detects the neighbour node,  $F$ , had been failed)
(2) Routing table updated (information about failed node will be deleted)
(3) IF  $F$  = leaf node
(4) Do not move
(5) Else
(6) On-board energy level check
(7) IF (energy is sufficient)
(8) Calculate the area of the overlapped coverage
(9) Broadcast "Temporary Relocation" MSG to neighbours
(10) move to ( $F$ )
(11) END IF
(12) ELSE IF (the node,  $A$ , receives "Temporary Relocation")
(13) Search (new route)
(14) IF new route available
(15) Transmit data to a new path
(16) IF (new route undiscovered)
(17) Buffer the Data
(18) END IF
(19) ELSE IF node  $A$  receives ("Back to original position")
(20) IF (data is buffered)
(21) Send data using "reallocated back node"
(22) END IF
(23) END IF
    Temporarily Relocation to ( $F$ )
(24) Step towards  $F$ 
(25) IF (reached  $F$ )
(26) Broadcast "Will manage recovery" message
(27) Coordinator = 1
(28) END IF
(29) IF (received "Will manage recovery" from node  $j$ )
(30) IF ( $ID \geq j$ ) Coordinator = 0
(31) Stop moving
(32) Transmit relevant data to the coordinator
(33) Receive recovery schedule
(34) IF (first node not on schedule)
(35) Relocate back to origin ()
(36) END IF
(37) IF Coordinator = 1
(38) From all concerned nodes collect significant information
(39) Form ranked list and recovery schedule
(40) Broadcast recovery schedule
(41) IF (not the first node on schedule)
(42) Relocate back to self ()
(43) END IF
(44) ELSE IF
(45) Relocate to ( $F$ )
(46) END IF
    Another neighbour node failed during the recovery process
(47) IF (node,  $A$ , detects the neighbour node,  $J$ , had been failed)
(48) Routing table updated
(49) On-board energy level check
(50) IF (energy is sufficient)
(51) Compare (distance  $F$  with distance  $J$ )//distance  $F$  = distance b/w  $A$  and  $F$ , distance  $J$  = distance b/w  $A$  and  $J$ 
(52) IF distance  $F <$  distance  $J$ 
(53) Do not move to  $J$ 
(54) IF distance  $F >$  distance  $J$ 
(55) Broadcast "Going to save  $J$ " MSG to recovery nodes
(56) move to ( $J$ )
(57) Go to (step 5)//replace  $F$  with  $J$ 
(58) IF receive (Going to save  $J$ )
(59) Update list
(60) END IF

```

ALGORITHM 1: Proposed algorithm's pseudocode.

TABLE 2: Meaning of different terms in the energy model.

Term	Meaning
$E_{Tx}$	Energy consumption of node for transmission
$E_{Rx}$	Energy consumption to receive data by a node
$E_{reng}$	Residual energy
$E_{elec}$	Energy consumed per bit by the transmitter circuitry
$E_{max}$	The initial energy of sensor nodes
$E_{Rx-elec}$	Energy consumed per bit of a receiver
$\epsilon_{fs}$	Energy required for a radio frequency (RF) amplifier in free space
$\epsilon_{mp}$	Energy required for a radio frequency (RF) amplifier in multipath
$d_0$	Threshold distance

TABLE 3: Simulation parameters.

Simulation parameters	Values
Simulation area	$900 \times 900 m^2$
Number of nodes	25–150
Data packet size	800 bits
$E_{elec}$	50 nano-Joule/bit
$E_{max}$	20 J
$\epsilon_{fs}$	10 pico-Joule/bit/m <sup>2</sup>
$\epsilon_{mp}$	0.0013 pico-Joule/bit/m <sup>2</sup>
$d_0$	75 m
Rc	25–150 m
Simulation tool	OMNeT++

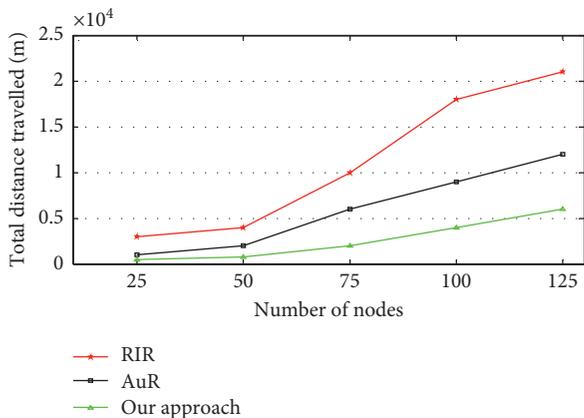
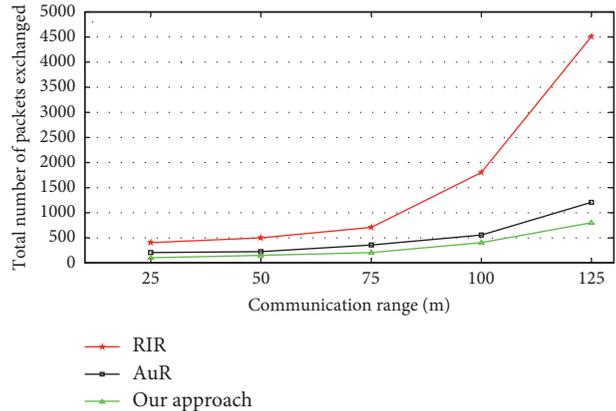
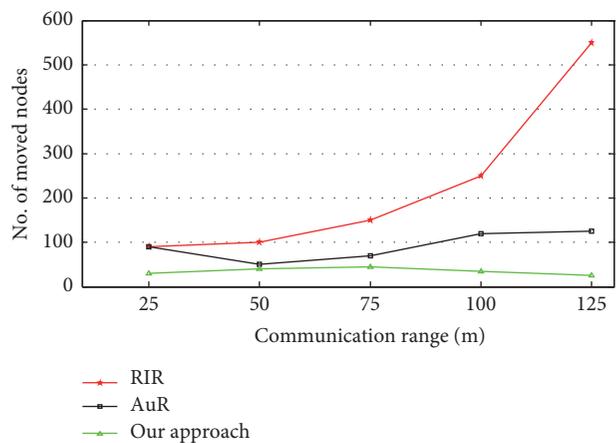
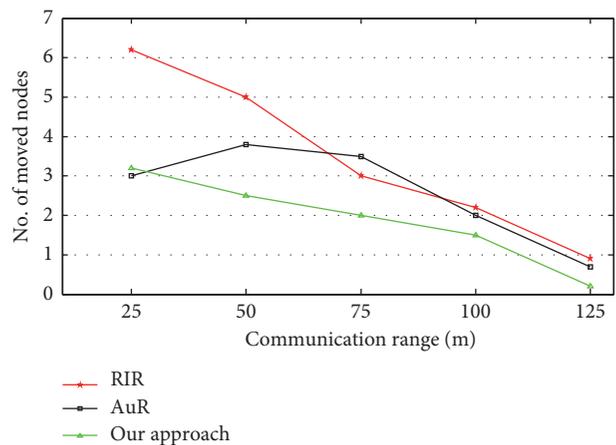


FIGURE 4: Total distance travelled vs. number of nodes.

5.2.4. *Percentage Field Coverage Reduction.* The impact of the restoration process on coverage is illustrated in Figures 7 and 8. It can be observed from the figures that the percentage reduction of the field coverage is considered by using the level of coverage before the failure and the level after the failure. The overall loss in coverage is limited reasonably by our proposed algorithm. Furthermore, for networks where nodes are not dense and evenly distributed, the overlapping coverage is at the minimal level under the proposed algorithm and the field coverage is reduced equitably as compared to RIR. A greater field coverage reduction was observed in RIR, when the overlapping of the coverage of the nodes began to rise, which are in cases where the nodes were deployed in a dense manner. The pre-failure field coverage level was restored by applying the proposed algorithm. This

FIGURE 5: No. of packets exchanged vs.  $R_c$  in meters.FIGURE 6: The total number of relocated nodes vs.  $R_c$  in meters.FIGURE 7: Percentage field reduction vs.  $R_c$  in meters.

restoration was done by the fact that the auxiliary nodes only move a little distance or do not move because of the growth in the coverage overlap. Besides this, there were numerous nodes available for the relocation process. However, networks with sparse node positioning did not have adequate nodes that could be substituted by the failure node. Furthermore, a greater area was left unattended with the

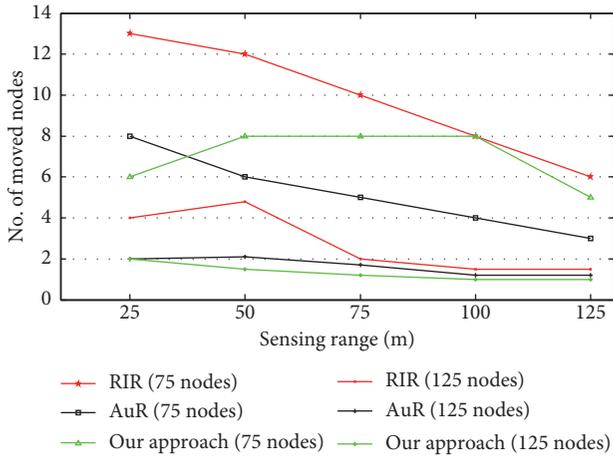


FIGURE 8: Percentage field reduction for the proposed technique vs.  $R_s$  in meters.

relocation of nodes, resulting in creation of a gap in the coverage of the network. The percentage reduction of the field coverage for several communication and sensing ranges is revealed in Figure 9. Fair coverage reduction was perceived when  $R_c$  (communication range) dominated  $R_s$  (Sensing range). This was because longer distances needed to be travelled by the nodes between their home area and the position of the node they were replacing. Even so, the reduction was limited to 10% with the proposed algorithm even if  $R_s$  is taken as six times the  $R_c$ , i.e.  $R_c = 6R_s$ .

**5.2.5. Recovery Time Consumption.** Figure 10. shows the time consumption of recovery process by the sensor nodes when the number of recovery nodes increases due to the failure of other sensor nodes in WSNs. Therefore, for calculating recovery time of our approach and other baseline approaches,  $\tau$  is taken as 5 secs. Whereas there are three allowed number of missed heartbeats. So, overall, a node waits for 25 seconds for its neighbour to respond. It is obvious that with the increase in recovery nodes, time consumption will be increased as recovery nodes must cover more distance. But, it is also a fact and it is clear from Figure 10 that our proposed approach takes less recovery time as compared to RIR and AuR. The main reason is that as it is discussed earlier that our approach does not have cascade movements and the recovery process is only restricted to the neighbouring nodes. Then, it is obvious that the proposed approach will take less time to recover because now recovery nodes do not cover large distances as in the case of cascade movements. Besides this, distance is the prime criteria for the recovery nodes in case of multiple node failure, i.e., recovery nodes restrict themselves to the recovery process of the failure nodes having lesser distance to them. That is why the recovery time of our approach is lesser than the other baseline approaches.

**5.2.6. Results' Summary.** The OMNeT++ platform is used to evaluate the performance of the proposed protocol to compare alongside the existing baseline algorithm. It is

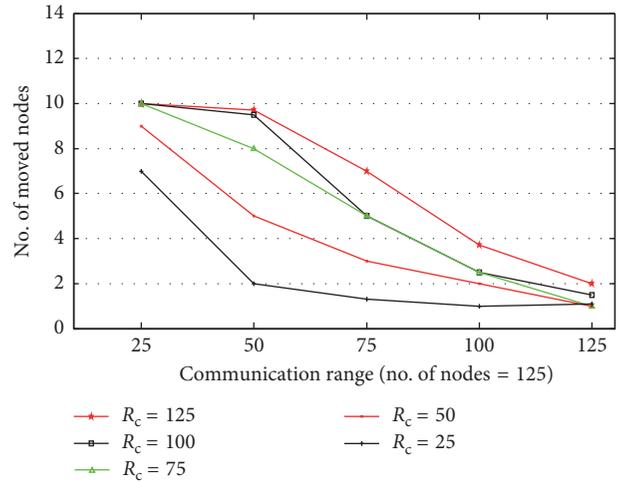


FIGURE 9: Percentage reduction in field coverage for the proposed technique vs.  $R_c$  for different communication ranges in meters.

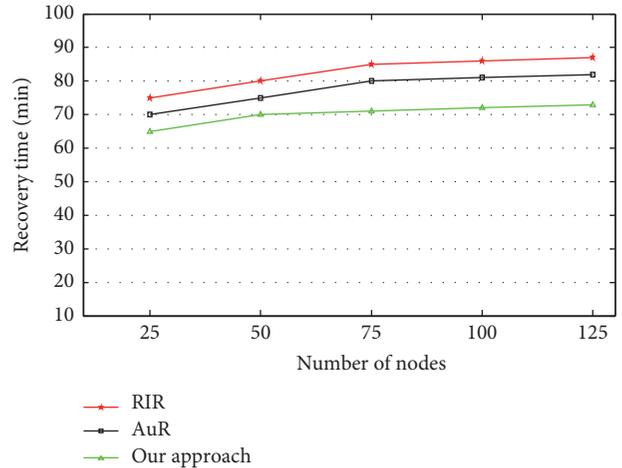


FIGURE 10: Number of nodes vs. their recovery time consumption.

revealed from the results of simulations that the proposed algorithm accomplished substantial energy-saving and improved the network lifetime. It can easily be concluded from results that the proposed approach is operative in refining parameters of quality of service like the number of exchanged messages, average number of nodes relocated, and reduction of the percentage of field coverage reduction, when compared with RTR and AuR techniques. Table 3 reviews the conclusions from these results (Table 4).

## 6. Conclusion

The maintenance of connectivity and coverage is very important in mobile sensor networks. Node failure may partition the network which causes the operations of application to be malfunctioned. The proposed procedure deals with the connectivity-loss issue as well as the issue of coverage by not relocating the nodes in a permanent manner. The neighbouring nodes are responsible for the node failure recovery. The neighbouring nodes of the failed

TABLE 4: Results' summary.

Comparison of protocols for 125 nodes	Average distance travelled (m)	No. of messages exchanged (average)	No. of nodes relocated (average)	% age reduction in field coverage (average)
Proposed technique	2600	400	50	4
RIR	14000	2000	200	13
AuR	19000	4500	700	19

node organize between themselves to fix roles of each node in the process of the recovery. For the restoration of connectivity, all the neighbouring nodes contribute in the recovery process to transfer the nodes to the location of the failed node, to bring pre-failure coverage in the location of the failed node. After expenditure, an allocated expanse of time at the place of the failed node, each recovery node returns to its original location. Stability in connectivity and coverage provided by the proposed algorithm enhanced the lifetime of the network. The proposed method is a hybrid form of localized and distributed algorithms. Overall messaging overhead is minor, and for trivial networks, the proposed algorithm can be evaluated. The endorsement, effectiveness, and validity of the proposed scheme are accomplished by extensive simulations.

### Data Availability

No data were used to support this study. We have conducted the simulations to evaluate the performance of the proposed protocol. However, any query about the research conducted in this paper is highly appreciated and can be asked from the principal author (Adnan Anwar Awan) upon request.

### Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

### Authors' Contributions

Adnan Anwar Awan, Muhammad Amir Khan, and Aqdas Naveed Malik conceived and designed the experiments; Adnan Anwar Awan, Muhammad Amir Khan, and Syed Ayaz Ali Shah performed the experiments; Muhammad Amir Khan, Aamir Shahzad, and Naveed Shahzad analyzed the data; Adnan Anwar Awan, Muhammad Amir Khan, and Aqdas Naveed Malik wrote the paper; and Babar Nazir, Iftikhar Ahmed Khan, and Waqas Jadoon technically reviewed the paper. Rab Nawaz Jadoon contributed in technical revisions and final technical proofreading.

### Acknowledgments

The authors are thankful to Isra University, Islamabad, and COMSATS University Islamabad for fully supporting by providing all key resources during the implementation and all afterward phases of this project. The authors would also like to personally thank Dr. Muhammad Amir Khan and Dr.

Aqdas Naveed Malik for their continuous encouragement and massive support both academically and socially during this project. This project is partially funded by Wireless Sensor Network Lab.

### References

- [1] R. N. Jadoon, W. Zhou, I. A. Khan, M. A. Khan, and W. Jadoon, "EEHRT: energy efficient technique for handling redundant traffic in zone-based routing for wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 7502140, 12 pages, 2019.
- [2] R. N. Jadoon, W. Zhou, W. Jadoon, and I. A. Khan, "RARZ: ring-zone based routing protocol for wireless sensor networks," *Applied Sciences*, vol. 8, no. 7, p. 1023, 2018.
- [3] M. A. Khan, H. Hasbullah, B. Nazir, and I. A. Khan, "An energy efficient simultaneous-node repositioning algorithm for mobile sensor networks," *The Scientific World Journal*, vol. 2014, Article ID 785305, 14 pages, 2014.
- [4] A. Khelil, F. K. Shaikh, A. Ali, N. Suri, and C. Reinl, "Delay-tolerant monitoring of mobility-assisted WSN," in *Delay Tolerant Networks: Protocols and Applications*, p. 189, Auerbach Publications, Boca Raton, FL, USA, 2016.
- [5] Y. K. Joshi and M. Younis, "Restoring connectivity in a resource constrained WSN," *Journal of Network and Computer Applications*, vol. 66, pp. 151–165, 2016.
- [6] M. Ilyas and I. Mahgoub, *Smart Dust: Sensor Network Applications, Architecture, and Design*, CRC Press, Boca Raton, FL, USA, 2016.
- [7] A. Alfadhly, U. Baroudi, and M. Younis, "Optimal node repositioning for tolerating node failure in wireless sensor actor network," in *Proceedings of the 2010 25th Biennial Symposium on Communications*, IEEE, Kingston, Canada, May 2010.
- [8] M. Y. Sir, I. F. Senturk, E. Sisikoglu, and K. Akkaya, "An optimization-based approach for connecting partitioned mobile sensor/actuator networks," in *Proceedings of the IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, Shanghai, China, April 2011.
- [9] M. Imran, M. Younis, A. M. Said, and H. Hasbullah, "Partitioning detection and connectivity restoration algorithm for wireless sensor and actor networks," in *Proceedings of the 2010 IEEE/IFIP 8th International Conference on Embedded and Ubiquitous Computing (EUC)*, Hong Kong, China, December 2010.
- [10] I. F. Senturk, K. Akkaya, and S. Fatih, "An effective and scalable connectivity restoration heuristic for mobile sensor/actor networks," in *Proceedings of the Global Communications Conference (GLOBECOM)*, IEEE, Anaheim, CA, USA, December 2012.
- [11] X. Bai, S. Kuma, D. Xua, Z. Yun, and T. H. La, "Deploying wireless sensors to achieve both coverage and connectivity," in *Proceedings of the 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ACM, Florence, Italy, March 2006.
- [12] M. Imran, M. Younis, A. Md Said, and H. Hasbullah, "Localized motion-based connectivity restoration algorithms for wireless sensor and actor networks," *Journal of Network and Computer Applications*, vol. 35, no. 2, pp. 844–856, 2012.
- [13] 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, no. 2, pp. 258–271, 2010.
- [14] M. Younis, I. F. Senturk, K. Akkaya, S. Lee, and F. Senel, "Topology management techniques for tolerating node

- failures in wireless sensor networks: a survey,” *Computer Networks*, vol. 58, pp. 254–283, 2014.
- [15] S. Lee and M. Younis, “Recovery from multiple simultaneous failures in wireless sensor networks using minimum Steiner tree,” *Journal of Parallel and Distributed Computing*, vol. 70, no. 5, pp. 525–536, 2010.
- [16] G. Robins and A. Zelikovsky, “Tighter bounds for graph Steiner tree approximation,” *SIAM Journal on Discrete Mathematics*, vol. 19, no. 1, pp. 122–134, 2005.
- [17] S. Vemulapalli and K. Akkaya, “Mobility-based self route recovery from multiple node failures in mobile sensor networks,” in *Proceedings of the 2010 IEEE 35th Conference on Local Computer Networks (LCN)*, Denver, CO, USA, October 2010.
- [18] K. Akkaya, A. Thimmapuram, F. Senel, and S. Uludag, “Distributed recovery of actor failures in wireless sensor and actor networks,” in *Proceedings of the IEEE Wireless Communications and Networking Conference*, Las Vegas, NV, USA, March 2008.
- [19] K. Akkaya, I. F. Senturk, and S. Vemulapalli, “Handling large-scale node failures in mobile sensor/robot networks,” *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 195–210, 2013.
- [20] Y. K. Joshi and M. Younis, “Exploiting skeletonization to restore connectivity in a wireless sensor network,” *Computer Communications*, vol. 75, no. 1, pp. 97–107, 2016.
- [21] X. Wang, L. Xu, S. Zhou, and W. Wu, “Hybrid recovery strategy based on random terrain in wireless sensor networks,” *Scientific Programming*, vol. 2017, Article ID 5807289, 19 pages, 2017.
- [22] X. Liu, “Survivability-aware connectivity restoration for partitioned wireless sensor networks,” *IEEE Communications Letters*, vol. 21, no. 11, pp. 2444–2447, 2017.
- [23] A. Wichmann, T. Korkmaz, and A. S. Tosun, “Robot control strategies for task allocation with connectivity constraints in wireless sensor and robot networks,” *IEEE Transactions on Mobile Computing*, vol. 17, no. 6, pp. 1429–1441, 2017.
- [24] M. Cobanlar, V. KhalilpourAkram, D. Orhan, and B. Tavli, “Analysis of the tradeoff between network lifetime and connectivity in WSNs,” in *Proceedings of the 2018 26th Telecommunications Forum (TELFOR)*, pp. 1–4, Belgrade, Serbia, November 2018.
- [25] P. Chanak, I. Banerjee, and R. S. Sherratt, “Energy-aware distributed routing algorithm to tolerate network failure in wireless sensor networks,” *Ad Hoc Networks*, vol. 56, pp. 158–172, 2017.
- [26] Y. K. Joshi and Y. Mohamed, “Autonomous recovery from a multi-node failure in wireless sensor network,” in *Proceedings of the IEEE Global Communications Conference (GLOBECOM)*, Anaheim, CA, USA, December 2012.
- [27] N. Tamboli and M. Younis, “Coverage-aware connectivity restoration in mobile sensor networks,” *Journal of Network and Computer Applications*, vol. 33, no. 4, pp. 363–374, 2010.
- [28] H. Essam, M. Younis, and E. Shaaban, “Minimum cost flow solution for tolerating multiple node failures in wireless sensor networks,” in *Proceedings of the 2015 IEEE International Conference on Communications (ICC)*, pp. 6475–6480, London, UK, June 2015.

