electronics

MDPI

Article

Algorithm for Increasing Network Lifetime in Wireless Sensor Networks Using Jumping and Mobile Sensor Nodes

Muhammad Amir Khan [1,*], Jawad Khan [1,*], Khalid Mahmood [2], Inam Bari [3], Haider Ali [4], Naveed Jan [4] and Rania M. Ghoniem [5]

1 Department of Computer Science, COMSATS University Islamabad, Abbottabad Campus, Abbottabad 22060, Pakistan
2 Electrical Engineering Department, Abasyn University, Peshawar 25000, Pakistan
3 Systems Engineering Department, Military Technological College, Muscat 111, Oman
4 Department of Electrical and Electronics Engineering Technology, University of Technology, Nowshera 24100, Pakistan
5 Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh 11671, Saudi Arabia
* Correspondence: amirkhan@cuiatd.edu.pk (M.A.K.); jawadzjadoon@gmail.com (J.K.)

Citation: Khan, M.A.; Khan, J.; Mahmood, K.; Bari, I.; Ali, H.; Jan, N.; Ghoniem, R.M. Algorithm for Increasing Network Lifetime in Wireless Sensor Networks Using Jumping and Mobile Sensor Nodes. *Electronics* **2022**, *11*, 2913. https://doi.org/10.3390/electronics11182913

Academic Editor: Cheng-Chi Lee

Received: 1 August 2022
Accepted: 8 September 2022
Published: 14 September 2022

Abstract: Sensor networks' network connectivity must be restored as part of any solution. This strategy's goal was to come up with a concept. Many approaches to restoring connections after a network outage can be implemented by relying on these factors: low mobility, minimal field coverage drop and a reduction in the overall number of messages sent. All of the following objectives can be met with this solution. Based on detailed simulations and a comparison with the PACR and SNR methods, it can be concluded that the proposed methodology is effective. The sensor nodes' batteries slowly depleted over time due to power restrictions. Network nodes fail as a result; data transmission stops, and the network's lifespan is shortened because of it. As a result, one of the most difficult challenges in wireless sensor networks is to minimize energy consumption while also maximizing the network's lifespan. In this study, the network lifetime of a wireless sensor network is extended through the use of special jumping nodes. Instead of using wheels or other means of transportation, these nodes leap into the network, and they are used to recharge other nodes in the network that are dying upon request. Results show that the proposed technique works more efficiently with figures of 83.76%, 84.84% and 87.3% for SNR, PACR and the proposed technique, respectively, with 250 nodes. This significantly increases the network's lifetime. The simulation results suggest that the proposed technique outperforms other strategies that have been used in the literature.

Keywords: wireless sensor networks; network lifetime; hopping nodes; failure recovery; mobile sensors; network connectivity

1. Introduction

WSNs (Wireless Sensor Networks) are commonly regarded being among the most significant technologies in the 21st century [1]. Small, inexpensive, and intelligent sensors placed in a physical space and connected via wireless networks and the Internet, recently developed microelectronic mechanical systems (MEMS) with wireless communication technologies, open up previously unimaginable possibilities for a range of civilian and military uses, such as monitoring the environment, keeping an eye on the battlefield, and controlling industrial processes [2]. WSNs differ as comparison to conventional wireless communication networks such as Cellular networks and ad hoc mobile networks (MANET) in that they possess particular qualities, such as a higher density of node deployment, higher sensor node significant energy, processing, and storage limits, as well as unreliability [3], all of which pose new difficulties in development and application of WSNs. WSNs have gotten a lot of interest from worldwide academic and industrial communities in the last decade. There has been a significant amount of research done to investigate and solve numerous designing and implementing difficulties, and significant progress has been achieved in the development and implementation of WSNs. WSNs are expected to be widely deployed in numerous military and civilian fields in the near future, revolutionizing the way people live, work, and engage in physical activities [4]. WSNs are comprised of significant number of cheap, low-power, and multi-purpose sensor nodes that are placed in a certain area. These sensor nodes are compact, yet they contain radio transceivers, embedded microprocessors, and sensors, allowing them to perform not only data processing but also sensing communication. They communicate across a short distance using a wireless medium and work together on a same goal, such as environmental monitoring, battlefield surveillance, or industrial process control.

Mobile entities are typically used as sensor nodes in mobile sensor network systems to carry out network deployment and monitoring functions. Nevertheless, several mobile nodes are required for mobile sensor networks. The cost rises in direct proportion to the number of mobile sensor nodes. Therefore, combining a lot of inexpensive static sensor nodes with some mobile sensor nodes is a practical option.

In order to complete their objectives, jumping nodes equipped with sensors and wireless communication can enter hostile and dangerous settings. Cooperatively deploying static sensor nodes and mobile nodes is efficient. The mobile nodes are able to expand network coverage and improve network connectivity. In WSN systems, jumping nodes are platforms carrying wireless communication modules. According to the application needs, the communication modules can be selected from short- to long-range communication modes. This research presents a comprehensive approach for determining predefined thresholds and demanding that jumping nodes be recharged by such robots. The contributions of the paper are listed below.

When a sensor node's battery is ready to run out, the suggested algorithm provides a detailed plan for recharging it by jumping node. Because nodes closer to base stations are more likely to fail due to battery loss, a configuration with three zones around the base station is recommended. As a result, the closer nodes are put in zone 1, where they act as relays, transmitting data from the sensor nodes to the base station in zones 2 and 3. Zone 2 nodes relay to zone 3 nodes, whereas zone 3 nodes relay to zone 2 nodes. Zone 3 nodes use the least amount of energy because base station receive data from sensor nodes.

Last but not least, simulations reveal that the suggested technique outperforms other standard techniques. Parameters like total distance travelled by nodes, number of messages sent and received among all nodes, average nodes moved, and the proportion of field coverage drop are utilized to improve service quality.

The remainder of the paper is organized in the following manner. The proposed algorithm's discrepancy with earlier work on the subject is discussed in Section 2. Section 3 contains the details of the suggested algorithm as well as the pseudo code. Section 4 discusses the findings validation, and Section 5 concludes the essay.

2. Related Work

Using a small number of mobile nodes to assure connectivity while maintaining overall coverage while working within resource constraints is a difficult task. A critical topic that has been examined in recent years in the literature is that in the event of battery drain, harsh weather or any other fault with a node, the system will fail. This field has seen a slew of new algorithms come to light. The system will fail if the battery runs out, the weather is bad, or a node fails. Many new algorithms have been developed in this area. Sensor nodes are spread at random using an aerial approach in most cases [5]. This type of deployment may result in a non-uniform distribution of mobile nodes over the AOI, with higher densities in certain areas and lower densities in others. As a result, in this scenario, relocation is required to assure connectivity between mobile nodes and end users while also increasing coverage area.

The literature has addressed the issue of connectedness, and numerous solutions have been presented [6–8]. To solve the connectivity problem, coverage gaps in certain places, must be avoided. As a result, it's crucial to strike a balance between connectivity and connectedness. Several methods have been tested in recent years. The concept of a two-connected network was proposed by the authors of [8]. According to this technique, each set of nodes in the network must have at least two linked pathways. This approach uses a two-degree connection and requires each node to hold two hopes. A cascading movement of nodes is recommended in [9] as a technique to restore connection when a node fails. This strategy calls for the failing node to be replaced by a neighboring node. At a redundant node, the cascade relocation mechanism begins and ends.

For the analysis of mobile ad hoc networks, a technique is proposed that suggests using a probabilistic, energy-conscious broadcast calculus. In order to make the deterministic decision between the probability distributions across target states, the semantics of our model are represented in terms of Segala's probabilistic automata that are driven by schedulers [10].

With the help of a formal specification in AWN (Algebra for Wireless Networks), a process algebra that has been especially designed for the modelling of Mobile Ad Hoc Networks and Wireless Mesh Network protocols, here a thorough analysis of the Ad hoc On-Demand Distance Vector (AODV) routing protocol is presented. The explanation of how the logic and proofs may be quite simply modified to protocol variants is one of the paper's significant contributions [11].

Mobile ad hoc and sensor networks are crucial in a number of application areas. The use of wireless links and the mobility of the nodes makes networks vulnerable to security attacks; among them, jamming attacks are sneaky and involve one or more nodes sending out false packets repeatedly to keep some wireless links busy. With regard to the Hybrid Wireless Mesh Protocol (HWMP), it assesses a network's susceptibility to jamming and unintentional interception using two different routing algorithms.

C3R [12] proposes a method for restoring coverage-aware connections in mobile sensor networks by replacing failed nodes with adjacent nodes, moving back and forth between the failed node's location and their initial position. The node that arrives first at the failed node creates the timeline for changing the turns of these nodes. A similar approach, known as RIM, is described in [13]. (Recovery by invert motion). In RIM, the failing node was recovered by its neighbouring nodes via inversion migration to the failed node's location. The basicidea is to make use of the surroundings. If the HEART BEAT message is not received, they will be the first to know of a node failure. As a result of their inward march towards the malfunctioning node, the network link is restored. In DARA [14], an approach for detecting cut vertices was devised by selecting a suitable neighbour node for relocation. The number of communication links is utilized to determine which neighbour is the best. Ref. [15] proposes VCR (Volunteer-initiated Connectivity Restoration). Restoring network connectivity is the responsibility of the failed node's near neighbours. The surrounding nodes are chosen based on their proximity to the failing node. Cascaded motions towards the failing node are limited with this method. In [16] introduces PADRA (Partition Detection and Recovery Algorithm). This approach is used to determine the cut-vertex nodes in advance. This may cause the network to divide, resulting in multiple disconnected networks.

Each cut-vertex node's failure state is handled by selecting an appropriate neighbour. The failing node is replaced by the selected neighbour, which prevents the network from breaking up as a result. Replacements lead to a chain reaction of events. It has been proposed by [17] that the CRAFT technique (Connectivity Restoration with Assured Fault Tolerance) be used. The discontinuous area is encircled by a BP (Backbone Polygon) using this technique's approach. Two non-overlapping restoration paths are established between each outer partition, and the BP by deploying low-cost Relay Nodes (RNs). By recommending the novel hybrid algorithm PACR, researchers attempt to close these disparities (Position-Aware protocol for Connectivity Restoration). The idea behind PACR is similar to that of someone who makes a will on their deathbed before passing away. Similar to this, a recovery coordinator is formed when the sensor energy falls below the threshold, and a recovery plan is produced. Cutting down on the time that is essential for failure identification quickens the recovery process. The neighbouring nodes do not move to the exact location of the failing node during the recovery process [18].

In ref. [19] also suggests the HRSRT (hybrid recovery strategy on random terrain) method. The land in the area of concern is divided into equal-sized cells. The weight of a pathway is determined by summing the weights of cells that line the path. Connectivity can be restored using the RTPP (Random Terrain Path Planning) technique. Many different strategies are available for deploying relay nodes because the number of mobile data collectors varies. Reduced data collection and acquisition costs and resources are the primary goals of these algorithms. Survivability-Aware Connectivity Restoration for Wireless Sensor Networks with Mobile Nodes was proposed in [20] by the authors of the paper. One-of-a-kind technology connects data load levels between disconnected sections. Re-connectivity is restored after the discovery of these parts. To connect the AOI to the base stations, the authors of ref. [21] propose a robot-controlled approach. Mobile robots need to find the shortest route possible while still staying connected to the network, and this strategy is designed to accomplish that. The proposed method makes use of two different algorithms. The first algorithm assigns the closest robot to the event area. Finally, a non-connected robot is asked to move to a connected portion with an allotted robot by the programme. A search algorithm is looking for unconnected robots until the network is fully connected. As a result of this algorithm, there are fewer hops between base stations and event zones.

The cut vertexes have been established, according to [22]. This method employs two distinct approaches to determining whether a node is critical or not. Nodes critical and noncritical can be identified using two-hop local and connected dominant set (CDS) information. In order for the first algorithm to work, the second one is crucial. Unknown network portions can be found without scanning the entire network using a constrained distributed depth-first search strategy. This approach uses long test bed trials to learn the status of all nodes, and simulation results show that this algorithm can detect all essential nodes with relatively little energy if CDS information is available.

Mobile sensor networks are becoming conceivable and practicable as the capabilities and affordability of mobile sensors improve. Although WSN deployments were never intended to be completely static, mobility was previously seen as posing a number of issues, including connectivity, coverage, and energy consumption, to name a few. Recent research, on the other hand, has portrayed mobility in a more positive light [23]. There are two basic techniques to achieving this goal: “recharging” or “replacing” the sensor nodes that have run out of energy. Solutions that use mobile robots to carry out the above-mentioned duties to automatically and independently maintain the WSN, reducing human intervention, are of special interest. The advancement of wireless power transmission techniques has recently boosted research activities in the area of battery recharge, with great hopes for its applicability to WSNs [24].

There are a number of solutions that have been discussed in the literature, and Table 1 summarizes the pros and cons.

Table 1. Algorithms for related work.

Algorithm	Year	Advantages	Shortcomings
RIM	2008	The inversion movement of neighbour nodes is used to recover a failed node, restoring both connectivity and coverage.	RIM is hampered by a large number of nodes. As a result of its propensity to cause a large number of nearby nodes to be moved. This increases the network's total overhead distance.
PADRA	2008	By locating the cut-vertex ahead of time, less network disjointedness can be avoided.	As a result of the movement of the cascade. During the connection restoration phase, all participating nodes expend more energy.
C3R	2010	Node failure will not affect connectivity or coverage any further because the nearby nodes of the failed node will perform back-and-forth moves from their original positions to those of failed node.	The main disadvantage of this method is that it involves many node transfers, which consumes additional power and acts as a stand-in for the failing node in the meanwhile.
VCR	2010	VCR solves the connection problem by choosing nearby nodes that are closer to the cut-vertex node.	VCR does not guarantee that cascaded movement will be reduced in a dynamic context.
Movement assisted sensor deployment	2012	The cascade motions of all surrounding nodes of the failing nodes provide a mechanism for restoring connectivity in this technique.	This method uses an excessive amount of energy due to the high number of cascading movements it employs.
CRAFT	2015	In this approach, a BP (Backbone Polygon) is built around the disjoint network's core. There are no partitions within that network.	This algorithm is extremely complex and time-consuming for mission-critical applications, as it must explore the entire network for BP formation.
GSR	2016	A geometrical skeletal backbone is constructed, consisting of a group of nodes with the greatest number of connections to other nodes.	GSR assumes that all nodes are aware of the field dimensions (where the nodes are distributed).
HRSRT	2017	The main goal of this algorithm is to reduce the amount of energy and money spent on data compilation and acquisition by changing (c) and (d) (P).	Because it must calculate a large number of weights, this technique is quite difficult and takes a long time for mission-critical applications.
SACR	2017	The data load levels of disjoint segments are taken into account in this technique. The connection of the partitioned segments is examined in this technique.	Because the approach must detect disconnected segments, which takes a long time, it is not suited for mission-critical applications.
CDS	2019	Two algorithms are used in this strategy. The first algorithm is used to determine which nodes are critical and which are not.	There is no recovery strategy in place if a node fails because this method only finds cut vertexes.

Table 1. *Cont.*

Algorithm	Year	Advantages	Shortcomings
PACR	2020	PACR is like writing a will on a deathbed. When a sensor's energy is below the threshold, it is termed a dying node. It will become a rehabilitation coordinator and create a plan. It speeds recovery by minimising failure diagnosis time, establishing a recovery coordinator, and generating a recovery plan.	Method combines localised and distributed algorithms. Small networks have little messaging overhead using the PACR method.
SNR	2021	Smart node relocation (SNR) detects and restores single or multiple node failures. SNR reduces control packets to save energy. It aims to relocate only important nodes while restoring connectivity to achieve coverage-awareness [25].	It avoids cascaded relocation by moving the fewest nodes necessary to restore connectivity.

3. Proposed Technique

The technique illustrated above is intended to extend the life of wireless sensor networks. Figure 1 depicts the overall network for which we extended longevity. A 150 m radius is provided by the nodes that surround the base station. Some environmental data can be sensed by nodes around the base station and routed to the base via a precise routing mechanism. In addition to the fixed sensor nodes, mobile entities are deployed. For example, if a sensor node runs out of battery power, a mobile entity can recharge it. Some jumping or jumping nodes are planted alongside these static sensor nodes and mobile entities. The jumping nodes' job is to convey data from sensor nodes that are in demand.

Static nodes are set up in accordance with the requirements of the application, but the movement of mobile entities is not at random. To capture robust movement, we have used the freeway mobility model [23]. In our plan, the base station is surrounded by three distinct zones. One zone is from the base station to 50 m, and two zones are from fifty meters to 100 m. The third zone is between 100 m and 150 m long. Zone 1 is more likely to run out of energy than zone 2; this is the primary reason for creating zones around the base station. Nearby sensor nodes, such as zone 2 and zone 3, are prone to failure because they not only receive and transmit their own data, but they also act as relays for data from sensor nodes further away from the base station. There are fewer chances for a zone 3 sensor node to fail because the zone 2 sensor nodes relay data only from zone 3 sensor nodes, whereas the zone 3 sensor nodes only communicate their own data.

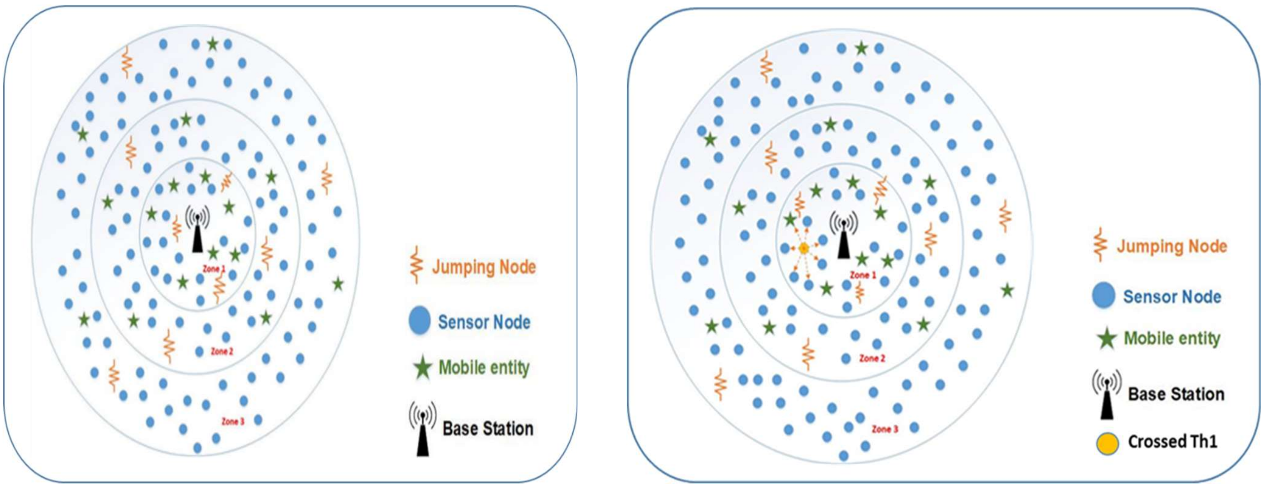
Zone 1 nodes are more likely to fail than zone 2 and zone 3 sensor nodes as a result of forming the previously specified zone surrounding the base station, hence we can expect a greater demand for mobile entities in zone 1. Zone 1 will have 12 mobile entities, zone 2 8 mobile entities, and zone 3 only 5 mobile entities. In the same way, zone 1 has a greater number of leaping nodes than zones 2 or 3.

Th1 and Th2 are the energy thresholds used by our sensor nodes. Nodes will send a message to their 1-hop neighbours whenever they run out of energy and hit threshold 1 (th1). They will then look for a mobile entity in their 1-hop neighbour. Sensor nodes respond when a mobile entity is detected and notify the mobile entity, which then proceeds to move towards the sensor node to charge it. No mobile entity was found at threshold 1, so the sensor node will wait until threshold 2 (th2), at which point the node will notify its neighbours, who will then search for the mobile entity in their neighbours. The sensor node will be notified, and the mobile entity will move toward the sensor node to begin charging it if it is found in the 2-hop neighbours.

As long as there are not any mobile entities within two hops, the sensor node will ask for help by sending a message to the base station. When a sensor node's location is known, the base station can locate the nearest mobile entity and alert it. If a sensor node fails, the base station will send an alert to the nearest mobile entity to let it know where the nearest sensor node is so that the mobile entity can go to the sensor node and begin charging it. Until it receives a new message from another sensor node or base station, the mobile entity will remain in its current location. As a result, it will only use energy if it has to move.

Sensor nodes will send a message to the base station if they are nearing the energy threshold, and there is no mobile entity to charge them. To find a free mobile entity, the base station looks for the jumping node, which is used to transmit data from the sensor nodes. It then locates the nearest jumping node, notifies the node of its location, and the node moves toward the failing node. It is like having a sensor node on a leash, except that when it gets to its destination, it turns into one.

The jumping node will continue to function in the same manner as the failing sensor node until the base station locates a mobile creature that is capable of charging the sensor node. The base station will communicate with a mobile entity to instruct it to charge the sensor node once it has arrived. After the sensor node has been fixed, all of the sensors will begin to function normally, and the jumping node will be available for use whenever it is necessary for travel to other nodes.



(**a**) (**b**)

Figure 1. *Cont.*

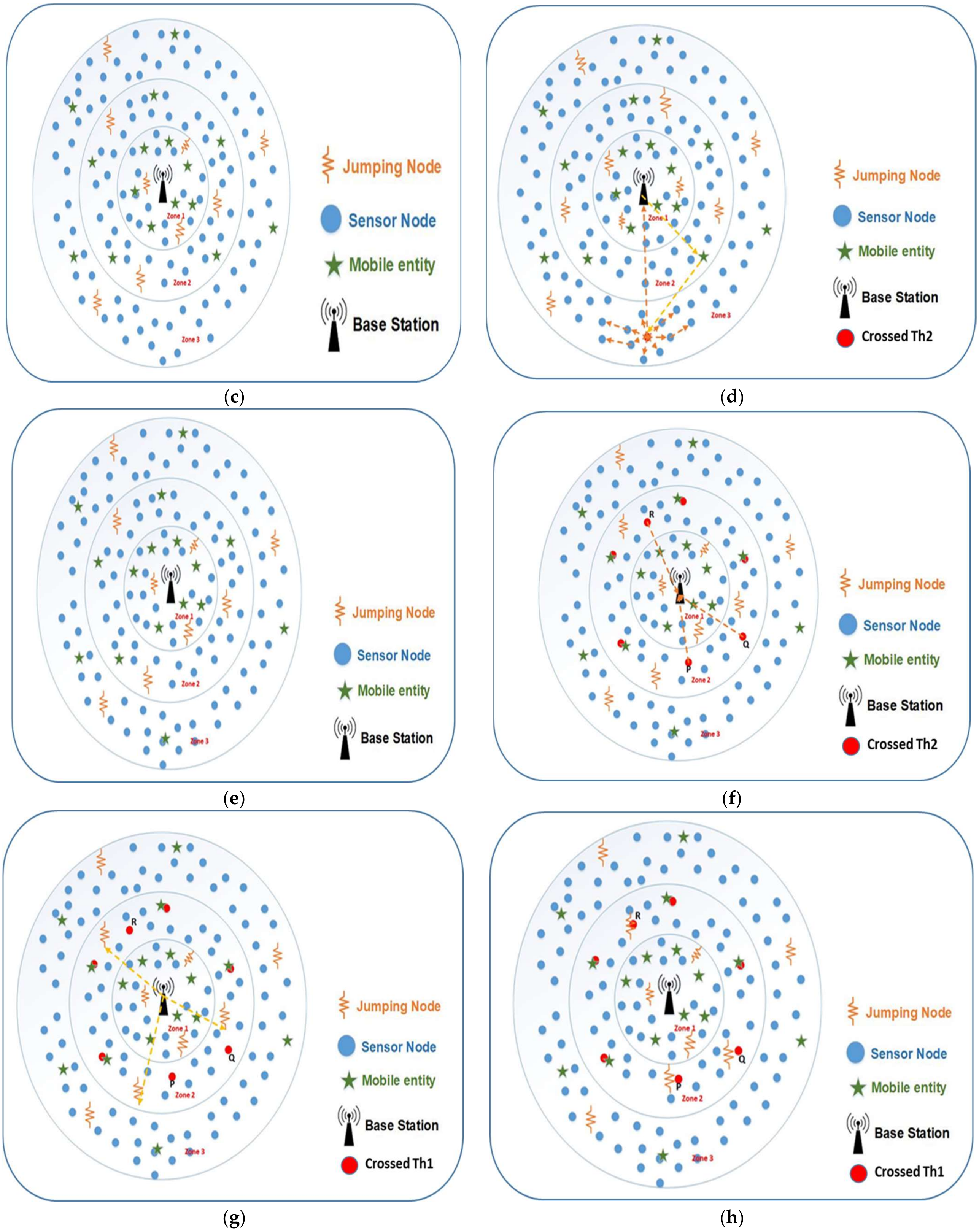


Figure 1. An example that illustrates overall operation of proposed methodology: (**a**) proposed system; (**b**) ME in neighbouring nodes; (**c**) ME awaited arrival at station charge; (**d**) sensor node send message to base station; (**e**) sensor node being charge; (**f**) failure of many nodes; (**g**) jumping nodes being alerted; (**h**) backup received.

As shown in Figure 1, Figure 1a explains overall proposed system. Figure 1b depicts that ME in neighbouring nodes is being checked by the node. Here it should also be mentioned that the yellow sensor node scouring its 1-hop neighbour for the mobile object seen is as shown in Figure 1a. Figure 1c shows ME awaited arrival at the charging station. Further, its shows that a mobile entity arrives and charges the sensor node before moving on to another sensor node or base station. A sensor node sends a message to the base station as shown in Figure 1d. When no ME could be found in Figure 1d, the sensor node contacted the base station, which was able to detect and route the nearest ME to SN. The sensor node was charged as it got closer to ME as shown in Figure 1e. It shows the ME arriving at the failing sensor node and charging it. Figure 1f depicts the failure of many nodes. Multiple nodes failed at the same time as shown in Figure 1f, and there was no ME available; thus, the sensor node contacted the base station. The jumping node has been alerted as depicted in Figure 1g. It also shows how base stations find jumping nodes and guide them towards failing nodes. Figure 1h shows that backup has been received. Here we can see how jumping nodes arrive at failing nodes and replace them, allowing the failing nodes to resume operation.

Algorithm 1 explains the overall recovery process of the proposed technique.

Algorithm 1 Propose Technique

Step 1: Initialization:
Sensor node: SN, Mobile entity: ME, Hopping node: JN, Base station: BS, Zone: ZN
Step 2: Input:
3 zones over 150 m, 120 SN deployed, 25 MEs, 15 JNs
2.1: Zone 1:
from BS Up to 50 m, 40 SN, 12 MEs, 10 JN
2.2: Zone 2:
from 50m to 100 m, 40 SN, 8 MEs, 5 JN
2.3: Zone 3:
from 100m to 150 m, 40 SN, 5 MEs

Main Working Algorithm

```
Step 3: If (E_SN <= TH1)
            Msg_1hop_N()
Step 4: Else If (ME == true)
            ME charge SN
Step 5: Else
              Wait till TH2
Step 6: End
Step 7: End
Step 8: If (E_SN <= TH2)
            Msg_2hop_N
Step 9: Else If (ME == true)
            ME charge SN
Step 10: Else
               MsgtoBS()
               BS notifies nearer ME
               ME charges SN
Step 11: End
Step 12: If (ME not available)
              BS locate near JN
              JN transmit SN data
              Till nearer ME is free to charge SN
Step 13: End
Step 14: End
Step 15: Output: Life time increased
```

3.1. Algorithm

Step 1 defines initialization and step 2 shows that we have taken three zones, i.e., zone1, zone 2 and zone 3 along with the different parameters defining each. Steps 3-11 explain that our sensor nodes use Th1 and Th2 energy thresholds. Nodes send a message to their 1-hop neighbours when they reach threshold 1 (th1). They look for a mobile neighbour in 1-hop. Sensor nodes respond when a mobile entity is spotted, notifying it to charge the node. No mobile entity was discovered at threshold 1, and hence, the sensor node waits until threshold 2 (th2) before notifying its neighbours to hunt for it. If the sensor node is within 2-hops, the mobile entity will move toward it to begin charging.

The sensor node will deliver a message to the base station if no mobile entities are within two hops. The base station can find the nearest mobile entity based on a sensor node's location. If a sensor node fails, the base station alerts the next mobile entity so it can charge it. The mobile entity stays put until it receives a communication from another sensor node or base station. It only uses energy when moving. This node's primary responsibility is to locate an alternative path for the data or to buffer it until the neighbouring node returns. If sensor nodes are nearing their energy threshold and there is no mobile entity to charge them, they will send a message to the base station as explained in steps 12–15. The base station looks for a free mobile entity, the jumping node, to transmit sensor node data. It then finds the nearest jumping node, notifies it, and it moves toward the failing node. When it reaches its target, it becomes a sensor node. Until the base station finds a mobile creature that can charge the jumping node, it acts as the failed node. The base station instructs a mobile entity to charge the sensor node. Once the sensor node is repaired, all sensors are operational, and the jumping node can be used for travel.

3.2. Model for Energy

A diagram illustrating the authors' energy model can be found in [24]. Equations (1) and (2) show the connection between the energy needed to send and receive a B-bit data packet (2). The receiver circuitry consumes a fixed amount of energy per bit (3).

$$E_{Tx}(B,d) = \begin{cases} (E_{elec} + \varepsilon_{fs} d^2)B & d < d_0 \\ (E_{elec} + \varepsilon_{mp} d^4)B & d \geq d_0 \end{cases} \tag{1}$$

$$E_{Rx}(B) = E_{Rx-elec}B \tag{2}$$

$$E_{reng}(n) = E_{max} - E_{Tx}(B,d) - E_{Rx}(B) \tag{3}$$

Several terms used in this model are defined in Table 2.

Table 2. Definition of terms used in the energy model.

Term	Meaning
E_{Tx}	Transmission node energy usage
E_{Rx}	A node's energy usage when receiving data
E_{reng}	Remaining energy
E_{elec}	Energy disbursed per bit by the transmitter circuitry
E_{max}	Sensor node's initial energy
$E_{Rx-elec}$	Energy spent per bit of a receiver
ε_{fs}	Energy required by radio frequency (RF) amplifier in free space
ε_{mp}	Energy required by radio frequency (RF) amplifier in multi path
d_0	Threshold distance

4. Assessment of Performance

Simulations validate and support the effectiveness of the proposed algorithm. The simulation setup, measurements of performance, and simulation results are all covered in this section. The OMNeT++ platform is used to verify the results. Table 3 summarises all of the simulation variables. For the simulation, source and destination are chosen by the user. The initial energy allocation for each node is the same. We identify each node's neighbours based on the battery life and transmission range. We determine the shortest path by using the neighbour and distance. The mathematical model is used to compute the transmission energy. As a result, we determine the network's remaining energy after routing.

The simulation's output shows that the initial deployment of node topology is taken into account by deploying nodes on a uniform path, and that each node's energy consumption and the network's overall energy consumption are taken into account and compared with a random placement pattern.

Table 3. Parameters for simulation.

Simulation Parameters	Values
Area of Simulation	800×800 m^2
No. of nodes	150 to 300
$\boldsymbol{R_c}$	30 to 170 m

4.1. Aggregate Relocation Distance via Movement

The results were all within 10% of the sample mean, with a 90% confidence interval. The analysis is carried out using the results of all three proposed approaches, including PACR [15] and SNR [25]. We have got the results right here. The entire distance travelled for relocation vs. the number of nodes added is depicted in Figure 2. The proposed strategy surpasses existing baseline techniques, according to the results, because only recovery nodes are shifted during operation. The number of cascaded relocations is reduced as a result. As a result, the suggested methodology has a shorter average distance travelled than the other methods investigated. The proposed method consistently outperforms previous strategies as the network's node count grows. The main reason for this is that the proposed method restricts node motions, which causes more network fragmentation. In terms of performance, PACR outperforms other algorithms. The key reason for the increase in performance is PACR working approach. As the number of nodes in the network rises, the network's backbone becomes stronger, allowing the recovery mechanism to become more competent and trustworthy, resulting in improved performance. Aside from that, PACR assumptions are unrealistic, and in the event of a topology change, they would result in a significant increase in packet exchange costs. When compared to PACR, Figure 3 indicates that our technique still performs admirably.

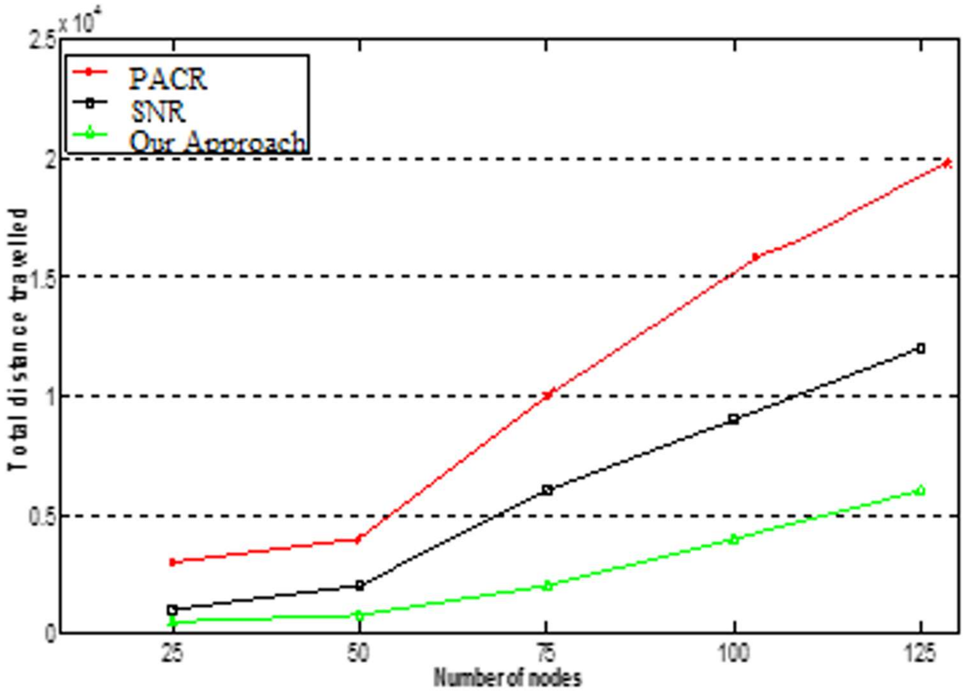


Figure 2. Number of nodes vs. distance travelled.

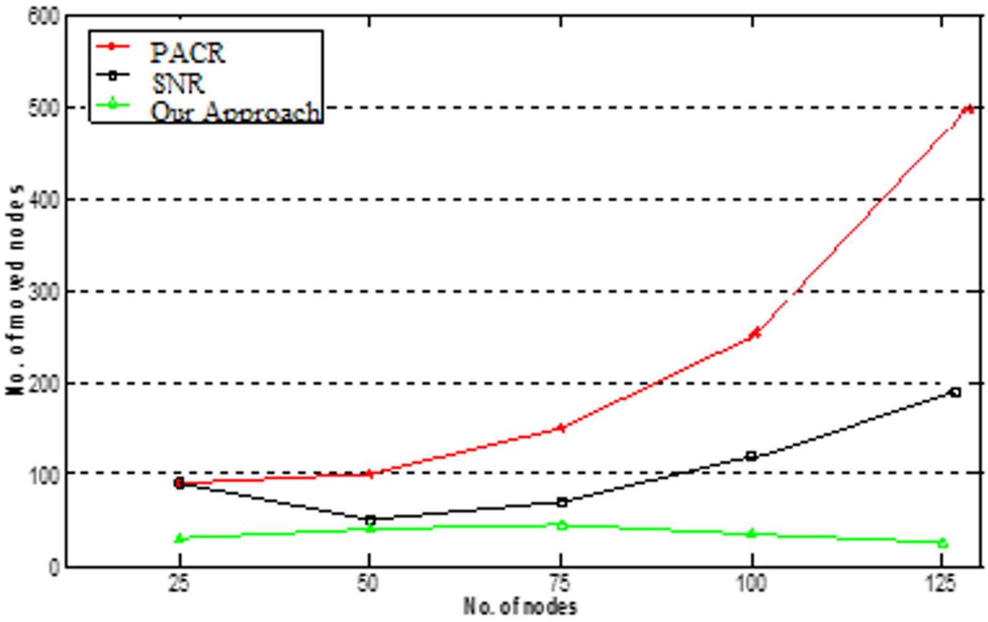


Figure 3. The average number of nodes moved throughout the recovery process.

4.2. Field Coverage Drop

The throughput produced by each of the three methods is compared in Figure 4, which shows the comparison in regards to the total number of nodes. The throughput is measured in terms of the number of bits that are transmitted in a certain amount of time (second). If all of the transmitters and receivers follow the same channels and are not required to change their radio between channels, we believe that we have achieved 100% of the theoretical throughput over the whole network. It takes a node two hundred microseconds to change its radio from one channel to another, which reduces the period of time that can be devoted to transmission and, as a result, the amount of data that can be sent in one second. It also indicates that our proposed methodology is upgraded relative to the baseline techniques. The total number of nodes increases, which results in an increase in the throughput overall.

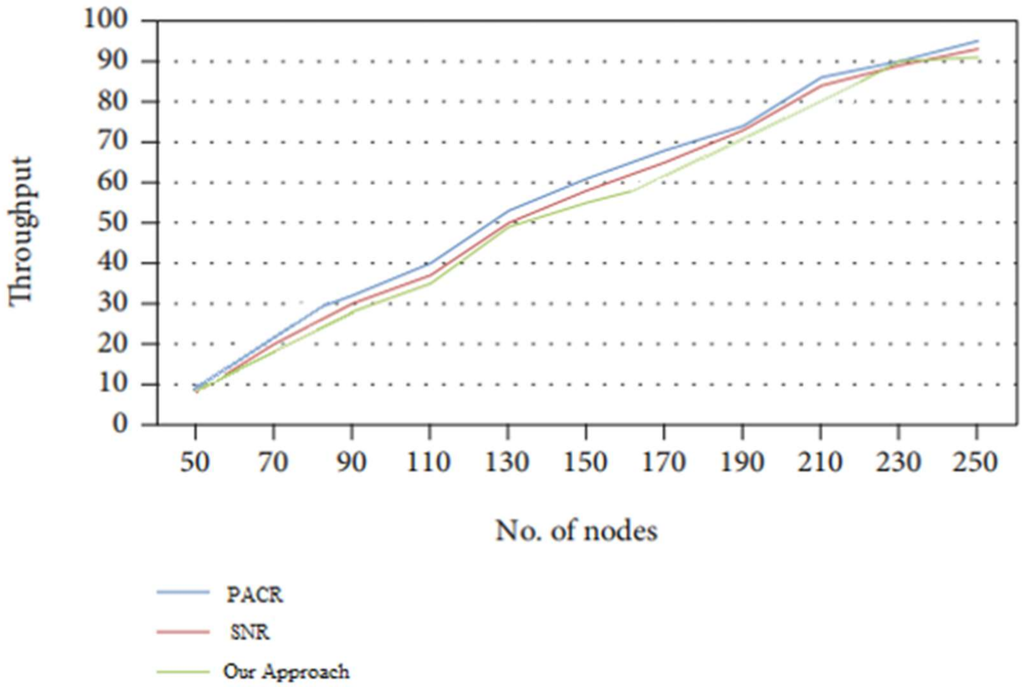


Figure 4. Number of nodes vs. throughput.

Figure 5 illustrates the effect that the overhead has on the lifetime average of sensor nodes as a function of the size of the network. We compared the node lifetime when PACR, SNR and proposed techniques were utilised, with particular attention paid to the average lifetime when no recovery mechanism was implemented. It has come to our attention that the lifespan average drops when the number of nodes in the network is increased. For instance, while using 50 nodes, the two strategies provided lifetime averages that were extremely similar to the case when there was no recovery. On the other hand, in sparse networks, PACR performs somewhat better than proposed technique. When applied to dense networks, the proposed technique shows improved performance, with efficiencies of 83.76%, 84.84% and 87.3% for SNR, PACR and proposed technique, respectively, with 250 nodes. When the proposed technique is utilised, there is a significantly lower occurrence of interferences and retransmissions, which is directly responsible for this result. On the other hand, the network lifetime is increased as a result of the failure of an articulation node, which causes WSN to become non-functional. The implementation of a recovery strategy will, thus, extend the lifetime of the network even though it will produce some additional overheads. Regarding the benefits that can be gained from the various recovery strategies, we believe that the reduction in the average node lifetime is still quite negligible. Therefore, compared to the other baseline techniques, the suggested methodology performs better in extending the network's lifespan.

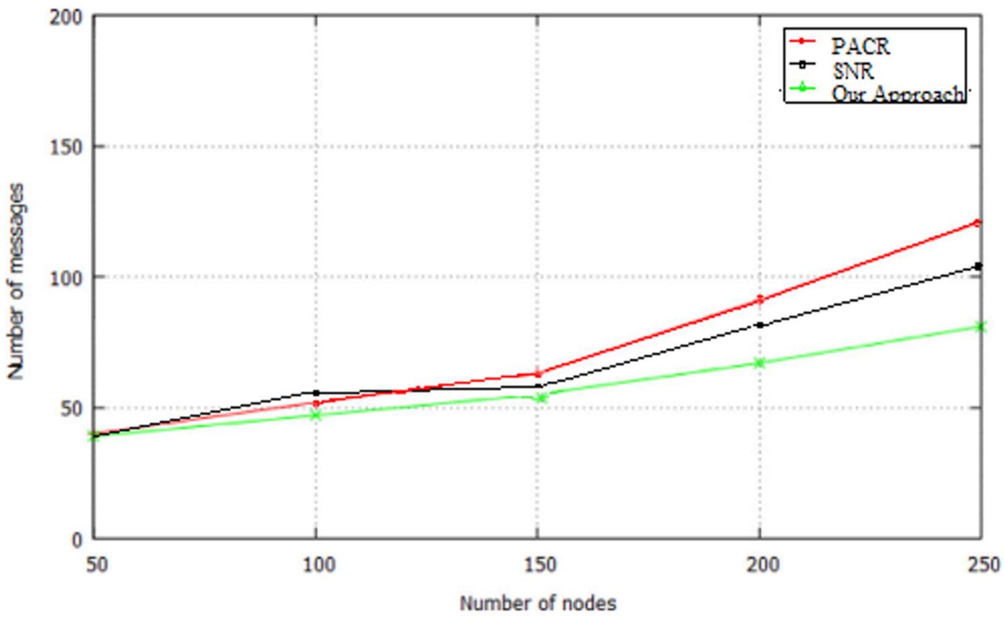


Figure 5. The number of nodes messages overhead.

5. Conclusions and Future Work

The importance of connectivity, coverage and energy efficiency in wireless sensor networks cannot be overstated. Sensor node failure can result in major issues such as network partitioning and coverage loss. As a result, connection, coverage and energy efficiency must all be addressed appropriately, and nodes must be capable of resolving these issues on their own. A revolutionary approach suggested in this paper provides a solution to these problems. By utilizing jumping nodes, it is possible to provide coverage as well as connection restoration in a seamless manner. The proposed algorithm's methodology makes it an energy-efficient solution. Simulations have confirmed and endorsed the proposed algorithm as an energy efficient option. The results of simulations reveal that, when compared to SNR and PACR algorithms, the suggested technique has a shorter average distance between nodes. Furthermore, these results suggest that the proposed technique enhances the field coverage percentage reduction when compared to the other two algorithms. Finally, the results reveal that the suggested technique requires less messages to operate than the other two techniques evaluated, SNR and PACR. Because jumping nodes are part of the network, the cost of these advantages is that a larger number of nodes are required to cover a given area of interest. Even so, for mission-critical applications, it is cost-effective. The in-depth investigation will be finished in the future.

Author Contributions: Conceptualization, R.M.G., M.A.K. and J.K.; methodology, K.M. and I.B.; software, H.A. and N.J; validation; J.K. and K.M.; formal analysis, M.A.K. and I.B.; investigation, I.B.; resources, H.A. and N.J.; data curation, R.M.G. and H.A.; writing—original draft preparation, K.M. and I.B.; writing—review and editing, K.M.; and I.B.; visualization, H.A.; supervision, J.K., M.A.K. and R.M.G.; project administration, H.A.; funding acquisition, J.K. and M.A.K. All authors have read and agreed to the published version of the manuscript.

Funding: Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R138), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Data Availability Statement: No data were used to support this study. We have conducted the simulations to evaluate the performance of proposed protocol. However, any query about the research conducted for this paper is highly appreciated and can be requested from the principal author (Muhammad Amir Khan (amirkhan@cuiatd.edu.pk)) upon request.

Acknowledgments: The authors sincerely appreciate the support from Princess Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R138), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Conflicts of Interest: The authors declare that there are no conflicts of interest.

References

1. Khan, J.; Khan, M.A.; Jhanjhi, N.Z.; Humayun, M.; Alourani, A. Smart-City-based Data Fusion Algorithm for Internet of Things. *Comput. Mater. Contin.* **2022**, *73*, 2407–2421. [CrossRef]
2. Zhu, C.; Zheng, C.; Shu, L.; Han, G. A survey on coverage and connectivity issues in wireless sensor networks. *J. Netw. Comput. Appl.* **2012**, *35*, 619–632. [CrossRef]
3. Nawaz, R.N.; Zhou, W.; Khan, I.A.; Khan, M.A.; Jadoon, W. EEHRT: Energy efficient technique for handling redundant traffic in zone-based routing for wireless sensor networks. *Wirel. Commun. Mob. Comput.* **2019**, *2019*, 7502140.
4. Bhuiyan, M.Z.A.; Wang, G.; Cao, J.; Wu, J. Deploying wireless sensor networks with fault-tolerance for structural health monitoring. *IEEE Trans. Comput.* **2013**, *64*, 382–395. [CrossRef]
5. Zeng, Y.; Sreenan, C.J.; Xiong, N.; Yang, L.T.; Park, J.H. Connectivity and coverage maintenance in wireless sensor networks. *J. Supercomput.* **2010**, *52*, 23–46. [CrossRef]
6. Younis, M.; Lee, S.; Abbasi, H.A. A localized algorithm for restoring internode connectivity in networks of moveable sensors. *IEEE Trans. Comput.* **2010**, *59*, 1669–1682. [CrossRef]
7. Jadoon, R.N.; Zhou, W.; Jadoon, W.; Khan, I.A. RARZ: Ring-zone based routing protocol for wireless sensor networks. *Appl. Sci.* **2018**, *8*, 1023. [CrossRef]
8. Akkaya, K.; Younis, M. Coverage and latency aware actor placement mechanisms in WSANs. *Int. J. Sens. Netw.* **2008**, *3*, 152–164. [CrossRef]
9. Basu, P.; Redi, J. Movement control algorithms for realization of fault-tolerant ad hoc robot networks. *IEEE Netw.* **2004**, *18*, 36–44. [CrossRef]
10. Gallina, L.; Hamadou, S.; Marin, A.; Rossi, S. A Probabilistic Energy-Aware Model for Mobile Ad-Hoc Networks. In Proceedings of the International Conference on Analytical and Stochastic Modeling Techniques and Applications, Venice, Italy, 20–22 June 2011; Springer: Berlin/Heidelberg, Germany, 2011; pp. 316–330.
11. Höfner, P.; van Glabbeek, R.J.; Tan, W.L.; Portmann, M.; McIver, A.; Fehnker, A. A Rigorous Analysis of AODV and its Variants. In Proceedings of the 15th ACM international Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, Paphos, Cyprus, 21–25 October 2012; pp. 203–212.
12. Wang, G.; Cao, G.; Porta, T.F. Movement-assisted sensor deployment. *IEEE Trans. Mob. Comput.* **2006**, *5*, 640–652. [CrossRef]
13. Tamboli, N.; Younis, M. Coverage-aware connectivity restoration in mobile sensor networks. *J. Netw. Comput. Appl.* **2010**, *33*, 363–374. [CrossRef]
14. Younis, M.; Lee, S.; Gupta, S.; Fisher, K. A localized self-healing algorithm for networks of moveable sensor nodes. In Proceedings of the IEEE GLOBECOM 2008—2008 IEEE Global Telecommunications Conference, New Orleans, LA, USA, 30 November–4 December 2008; pp. 1–5.
15. Imran, M.; Younis, M.; Said, A.M.; Hasbullah, H. Volunteer-instigated connectivity restoration algorithm for wireless sensor and actor networks. In Proceedings of the IEEE International Conference on Wireless Communications, Networking and Information Security, Beijing, China, 25–27 June 2010; pp. 679–683.
16. Akkaya, K.; Aravind, T.; Fatih, S.; Suleyman, U. 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, 31 March–3 April 2008; pp. 2480–2485.
17. Wang, X.; Li, X.; Shuming, Z.; Wei, W. Hybrid recovery strategy based on random terrain in wireless sensor networks. *Sci. Program.* **2017**, *2017*, 5807289. [CrossRef]
18. Nawaz, J.R.; Awan, A.A.; Khan, M.A.; Zhou, W.; Malik, A.N. PACR: Position-aware protocol for connectivity restoration in mobile sensor networks. *Wirel. Commun. Mob. Comput.* **2020**, *2020*, 8859256. [CrossRef]
19. Liu, X. Survivability-aware connectivity restoration for partitioned wireless sensor networks. *IEEE Commun. Lett.* **2017**, *21*, 2444–2447. [CrossRef]
20. Wichmann, A.; Korkmaz, T.; Tosun, A.S. Robot control strategies for task allocation with connectivity constraints in wireless sensor and robot networks. *IEEE Trans. Mob. Comput.* **2017**, *17*, 1429–1441. [CrossRef]
21. Chanak, P.; Banerjee, I.; Sherratt, R.S. Energy-aware distributed routing algorithm to tolerate network failure in wireless sensor networks. *Ad Hoc Netw.* **2017**, *56*, 158–172. [CrossRef]
22. Yatish, J.K.; Younis, M. Autonomous recovery from multi-node failure in Wireless Sensor Network. In Proceedings of the 2012 IEEE Global Communications Conference (GLOBECOM), Anaheim, CA, USA, 3–7 December 2012; pp. 652–657.
23. Rezazadeh, J.; Moradi, M.; Ismail, A.S. Mobile wireless sensor networks overview. *Int. J. Comput. Commun. Netw.* **2012**, *2*, 17–22.
24. Militano, L.; Erdelj, M.; Molinaro, A.; Mitton, N.; Iera, A. Recharging versus replacing sensor nodes using mobile robots for network maintenance. *Telecommun. Syst.* **2016**, *63*, 625–642. [CrossRef]
25. Khalid, M.; Saeed, M.K.; Ali, S.; Zaman, S.; al Awady, A.; Saqib, M. Smart node relocation (SNR) and connectivity restoration mechanism for wireless sensor networks. *EURASIP J. Wirel. Commun. Netw.* **2021**, *2021*, 180.