Cut Node Recovery Protocol for Wireless Sensor Network

E. Anna Devi
Assistant professor, Dept of ECE, Sathyabama Institute of Science and Technology OMR Road, Chennai, India annadevimit@gmail.com

J. Martin Leo Manickam
Professor, Dept of ECE, St.Joseph’s College of Engineering OMR Road, Chennai, India josephmartin74@yahoo.co.in

Abstract - In wireless sensor network (WSN), energy efficiency and network lifetime are the important attributes to be improved for enhancing the performance of the network. Due to the failure or cut of a node, performance of the network is affected. So cut node detection and recovery are presented in this paper. Initially, sensor nodes in the search area are clustered using the node weight of the sensor nodes. The node with maximum weight is selected as cluster head (CH) and this CH performs as a monitor node which monitors all non-CH members in a cluster. Backup nodes (BNs) are selected in each cluster using Particle Swarm Optimization (PSO) algorithm. Cut node in the network is detected based on its hop-count and link cost value. This cut node is recovered by the selected backup nodes and mobile nodes (MNs) in the network. Simulation results show that the performance of the proposed approach outperforms that of the existing approach in terms of overhead, energy efficiency, network lifetime and BN selection time.

Keywords: Wireless sensor network, Node weight, Backup Node (BN), Particle Swarm Optimization (PSO), Cut node recovery.

1. INTRODUCTION

WSNs are these days conveyed for a substantial assortment of uses are as yet picking up enthusiasm for a few fields [1-4]. The network comprises of battery-fueled wireless gadgets, with sensing and correspondence capacities, that gather information about the earth or around an occasion of intrigue (e.g. light, temperature, movement). The gathered information is then handed-off toward a “sink” in a multi-hop way [5]. Other than the requirements as far as preparing force and capacity limit, the most critical issue in WSNs is the constrained battery term of sensors and the resulting restricted network lifetime. As a result, WSN arrangements should adapt to the issue of sensor disconnection and multiple passing of hubs after some time.

In WSNs, connectivity and coverage of the network are critical all through the entire network task time for meeting the coveted application prerequisites. For example, if WSNs need to occasionally transmit the sensed information effectively to the sink node, at that point the network ought to be associated constantly. In like manner, for observing all spots of a district precisely, the network ought to give alluring coverage amid whole lifetime of the network. In nearness of successive disappointments, giving wanted nature of administration (QoS) is all the more difficult in remotely sent WSNs, where manual substitution of failed node isn't conceivable. Along these lines, keeping in mind the end goal to give self-recuperating capacity against such disappointments, mobile.Sensor nodes are broadly considered as a choice. In spite of the fact that, the utilization of mobile nodes in sensor networks expands the cost of the network, however is extremely helpful for connectivity rebuilding, supplanting failed nodes and dynamic adjustment of the network. Mobile nodes are significantly more adaptable than static sensor nodes as they can be sent in any situation and can adapt to quick topological changes. Rather, a blend of mobile and static sensor nodes has for the most part been favored.
Likewise, the utilization of mobile sensor nodes is very clear from some ongoing plans proposed for taking care of disappointments and topology administration where such nodes are broadly utilized [2-4]. Along these lines, to recover the failure node or cut node in the network, Backup Node (BN) based cut node recovery is exhibited.

2. PROBLEM STATEMENT

Krishna Pal Sharma and TeekParval Sharma [6] have presented failure recovery based on zone. In this approach, authors have selected a zone monitor using the approach based on consensus and agreement. The process of zone monitor was to monitor the activities of failure recovery. Based on the backup nodes and mobile nodes, failure nodes in the network have been recovered. The authors have used mathematical model to select the backup nodes from the available sensor nodes. Because of this selection method, backup node selection time has been increased and also energy efficiency of the network has been decreased. To overcome these issues, an optimization algorithm is presented in this paper for backup node selection.

Contributions of this paper are described below:

- Based on the parameters residual energy and average distance of the sensor nodes, cluster head (CH) is selected and cluster formation is done by this selected CH
- Backup Nodes (BNs) are selected using the Particle Swarm Optimization (PSO) algorithm in each cluster
- After establishing a path between source and destination using hop-count and link cost value, the source node detects whether the path has cut node using the percentage of difference
- Once a cut node is detected, the CH activates the BN if it is closer to the cut node to replace the cut node, otherwise nearby mobile node (MN) is activated replace the cut node

3. RELATED WORKS

In this section, some previous literatures are survived. These research works focused on the failure node or cut node recovery for improving the performance of the network. Krishna Pal Sharma and TeekParval Sharma [9] have presented failure node recovery method based on zone. Authors have selected a zone monitor using the consensus and agreement based scheme. The activities of failure node recovery were monitored by this zone monitor. This monitor node detected the suspicious node by receiving the heartbeat messages from the sensor nodes. The failure node was recovered using the backup nodes those were overlapped with the sensing range of cut node. Abdollahzadeh, Sanay and NimaJafariNavimipour. [2] have developed a failure detection and recovery method. In this method, authors have generated agent packet from the sink node and also they have formed a query path to the failure node using the agent. Based on these agent packets, the receiving node has detected the failure node. This failure node has been recovered using the Least-Disruptive Topology Repair. Mehajabin, Nusrat [3] have proposed an innovative solution for the issue of the connectivity due to the failure nodes. Authors have presented two steps to replace the failed nodes. Redundant nodes were identified using the selected cluster heads in the first step. Then the connectivity was restored in the second step. Cerulli, Raffaele, et al [4] have enhanced the network lifetime of WSN by presenting fault node recovery algorithm. This algorithm was performed based on the grade diffusion algorithm together with the genetic algorithm. Using these combined algorithms, authors have detected failure nodes.

4. BACKUP NODE SELECTION BASED CUT RECOVERY

4.1 Overview

Initially, cluster head (CH) is selected based on the node weights. Node weight is calculated using the parameters average distance and residual energy of the node. Then the selected CH selects the rest of the sensor nodes those are in its communication range and forms a cluster. CH of each cluster plays as a monitor node which monitors all cluster members in the cluster. In each cluster backup nodes (BNs) are selected by CH using particle swarm optimization algorithm. These selected BNs are in sleep mode initially. Before data transmission, the source node identifies the cut node based on the hop-count. This cut node is recovered using the selected
BN. If a cut node is detected, the CH turns on the BN which is in on hop distance to the cut node. The BN has sensing coverage of cut node so it replaces the cut node. If there is no BN in one hop distance to the cut node, a mobile node (MN) is turned-on active by the CH. Then MN moves to the location of cut node and replaces the function of cut node. Figure 1 shows the block diagram of this proposed approach.

![Block diagram of the proposed system](image)

**Figure 1:** Block diagram of the proposed system

4.2 Cluster formation

Sensor nodes in the network are grouped as number of clusters by selecting the cluster head. This cluster head is selected based on the node weight. Also it performs as a monitor node by monitoring the activeness of cluster members in each cluster. Weight of each node is calculated based on the parameters such as average distance and residual energy.

**Average distance:** It is defined as the average distance between each sensor node (SNi) to the base station (BS) in the network. Average distance of sensor node from the base station should be minimal.

\[ o_1(i) = \frac{1}{m} \frac{1}{d(SN_i, BS)} \]  \hspace{1cm} (1)

Where, \( m \) – denotes amount of sensor nodes in the sensing area

\[ \frac{1}{m} d(CH_i, BS) \] - defines the average distance between a base station and SNi

**Residual energy:**

It is defined as the difference between the initial energy and consumed energy of the sensor node (SNi).

\[ o_2(i) = E_{initial \_i} - E_{cons \_i} \]  \hspace{1cm} (2)
Where, \( E_{\text{initial}} \) and \( E_{\text{cons}} \) represent the initial energy and consumed energy of SNi.

Using equations (1) and (2), node weight (NW) is calculated as follows

\[
NW(i) = o_1(i) \times \mu + o_2(i) \times (1 - \mu), \quad 0 < \mu < 1
\]  

(3)

Where, \( \mu \) is the weight factor varying in the range of \([0, 1]\).

Node with maximum value of node weight is selected as optimal cluster head. Rest of the sensor nodes those are in the communication range of CH join to the corresponding CH. Finally, clusters are formed. The selected CH collects information about the all non-CH members in a cluster.

### 4.3 Backup node selection

To recover a node from the cut, backup nodes are selected in each cluster by corresponding CH using particle swarm optimization algorithm. Before the selection of backup nodes, k-coverage \([2x]\) areas are formed for set of sampling points which denoted as qm.

As shown in figure 2, qm points are considered in each cluster to cover sensor nodes based on their sensing range (rs). Euclidian distance \( e \) is calculated between each sensor node \( n_i \) and the point qm. This calculated distance is compared with the sensing range of the sensor node. If distance of \( k \)-non-CH members in each cluster is less than the sensing ranges from point qm, then k-coverage areas are formed for point qm i.e.,

\[
d_{n_i,q_m} < r_s, \quad i=1,2,...,N, \quad m=1,2,...,M
\]

(4)

Where, \( N \) denotes the number of sensor nodes or non-CH members, \( M \) denotes the number of points in a cluster.

Sensing ability of each sensor node is calculated based on its distance to the point qm i.e.,

\[
S(n_i,q_m) = \frac{\alpha}{[d(n_i,q_m)]^\beta}
\]

(5)

Where, the positive constants \( \alpha \) and \( \beta \) represent the sensor technology dependent parameters.

![Figure 2: k-coverage areas are formed in a cluster based on the sampling set points (q1, q2 and q3)](image-url)

After forming the k-coverage areas with respect to the sampling points, backup node (BN) is selected for each area in a cluster. These BNs are selected using the Particle Swarm Optimization (PSO) algorithm. Particle swarm optimization (PSO) is a nature inspired swarm intelligence based algorithm, modeled after observing the choreography of a flock of birds, i.e., how they can explore and exploit the multi-dimensional search space for food and shelter. Selection BNs using this algorithm is described as following,
Initialization: Particles or solutions of this approach are initialized. Backup nodes which are to be selected are the solutions. These solutions or backup nodes are initialized with its positions as follows,

\[ P_i = [Y_{i,1}(t), Y_{i,2}(t), \ldots, Y_{i,D}(t)] \]  

(6)

where \( Y_{i,d}(t) \) denotes the position of the \( i \)th particle in the \( d \)th dimension. This is also represented as

\[ Y_{i,d}(t) = Y_{i,1}(t), 1 \leq i \leq N, 1 \leq d \leq D \]  

(7)

Fitness: Fitness of each solution is calculated based on the sensing coverage of the sensor nodes. The closest sensor node to the sampling points or the node with maximum sensing ability is selected as the optimal node or BN. This fitness function is calculated using equation (8),

\[ Fitness = \text{Max} \left( S_{P,1:d} \right) \]  

(8)

Updating: Each particle or solution is initialized with a random position and velocity to move in the search space. In each generation, particles find their own best values also known as personal best value (Pbest) and also find the global best value (Gbest). Particle is updated using position (X) and velocity (V) equations until obtain the global best solution. Velocity is calculated using the Pbest and Gbest. New position of the particle is calculated using the velocity. Velocity and position of the particle are calculated as follows,

\[
\begin{align*}
V_{i,d}(t+1) &= V_{i,d}(t) + a_1 \gamma_1 (Y_{P,best,d} - Y_{i,d}(t)) + a_2 \gamma_2 (Y_{G,best,d} - Y_{i,d}(t)) \\
y_{i,d}(t+1) &= y_{i,d}(t) + v_{i,d}(t+1)
\end{align*}
\]  

(9)

(10)

Where, \( a_1 \) and \( a_2 \) represent the acceleration coefficients varying in the range \([0, 2]\), \( \gamma_1 \) and \( \gamma_2 \) represent the random values varying in the range \([0, 1]\), \( w \) denotes the inertia weight varying in the range \([0, 1]\).

After the particle is updated to the new position, fitness value of the particle \( P_i \) is compared with the fitness of previous Pbesti. If the fitness (Pi) is greater than the fitness (Pbesti), then the particle Pi is updated as the Pbesti. Similarly, if the fitness (Pi) is greater than the fitness (Gbest), then the particle Pi is updated as the Gbest.

Termination: The above phases are continued until find the optimal solution or BNs. Once the optimal solution is obtained the algorithm will be terminated.

![Figure 3: BNs are selected in each k-coverage area](image)

Once the BNs are selected, these nodes are in SLEEP mode initially i.e., the BN doesn’t sense data or transmit the data but it can listen to the remote medium to detect a packet. In sleep node, the BNs consume low power. These BNs are played as the alternative nodes to replace the cut node in the network. Figure 3 shows the BNs selection.
4.4 Cut node detection

Prior to the packet transmission, a source node transmits the information packet Route Request (RREQ) message to the destination by means of the assigned next hop nodes. After acknowledgment the RREQ message, the end transmits the Route Reply (RREP) message to the source node. At that point the source node passes on the packet by means of the assigned ensuing hop node. In this transmission, packet and neighbour table of every node are refreshed with the information of neighbour nodes like residual energy, hop tally, size of queue, link quality and link cost until achieve the destination.

As a result of the quantity of transmissions, couple of nodes may lose its energy level that is recognizes as cut nodes. Again the source node diffuse the RREQ to the destination, if the cut node available in the course, the source node doesn't acquire the RREP message inside the round trip time. So the source node retransmits the parcel and each hop node in the course checks link cost estimations of neighbour nodes. To identify the cut node in the course, contrast inside the normal link cost esteems and the most extreme link cost estimation of the neighbour nodes is outlined and the level of distinction is related with the threshold esteem. On the off chance that the threshold esteem (40%) is more prominent than the percentage of difference, at that point the node is to be viewed as that the node is to be cut. The percentage of difference is intended as

\[
diff (%) = \frac{C_{\text{High}} - C_{\text{Avg}}}{C_{\text{Avg}}} \times 100\%
\]  

(11)

Where, \( C_{\text{High}} \) and \( C_{\text{Avg}} \) are signified as maximum link cost value of a neighbour node with minimum hop count and average link cost value of all neighbour nodes except the node with minimum hop count correspondingly. This link cost value is calculated as

\[
C_{ij} = \frac{E_{\text{residual},ij}}{E_{\text{initial},ij}} \times C_{\text{EE}} \times Q_{\text{free},i} \times Q_{\text{total},i} \times LQ_{ij} \times C_{CL}
\]  

(12)

Where, \( C_{\text{EE}}, C_{\text{CC}} \) and \( C_{\text{CL}} \) are signified as the constant coefficients, \( E_{\text{residual},ij} \) and \( E_{\text{initial},ij} \) signify the residual and initial energy of node \( j \) correspondingly, \( Q_{\text{free},i} \) and \( Q_{\text{total},i} \) represent the free queue size and maximum queue size of node \( i \) correspondingly and \( LQ_{ij} \) signifies the link quality between two nodes \( i \) and \( j \).

Figure 4: Cut Node recovered by BN (34)

4.5 Cut node recovery

After detecting the cut node, the source node sends the message to the corresponding CH. This message includes the ID and status of the cut node. After receiving the message, the CH verifies the Neighbour
table of the cut node and decides whether the BN is in one hop distance to the cut node. If the BN places in one hop distance to the cut node, then the CH instructs the BN that turn on the BN as ACTIVE node itself. The coverage of this BN is overlapped with the sensing range of other neighbour nodes in the area of k-coverage so that BN replaces the cut node to continue the data transmission as shown in figure 4. New BN is to be selected when the BN replacing the cut node. If the cut node doesn’t has the BN in one hop distance, then the CH finds the nearby mobile node (MN) with the ACTIVE status and the MN moves to the received position of the cut node and replaces it.

5. RESULTS AND DISCUSSION

The proposed approach Backup Node selection for Cut node Recovery (BNCR) is implemented in the platform of Network Simulator (NS2). In this simulator, 500 sensor nodes are executed in the search space 1000m×1000m.

5.1.1 The performance metrics

Performance of this proposed approach is evaluated based on the following metrics. The performance metrics of the proposed Backup Node selection for Cut node Recovery (BNCR) is compared with that of the existing ZBFR [9]. The entire scenario is simulated in ns2 and various parameters are calculated.
Energy consumption: It is defined as the amount of energy consumed by individual node during the transmission. Also it is referred as the difference between the current energy and initial energy of a node. Unit of this parameter is Joule (J).

Energy consumption = Initial energy − current energy

Network lifetime: It is referred as the time of network disconnection due to the failure of one or more sensor nodes. Unit of this parameter is seconds (s).

BN selection time: It is defined as the time to select the number of backup nodes within the period of simulation time.

Coverage ratio: It is defined as the ratio of the covered area to the total area.

Message overhead: It is defined as the number of messages exchanged during the process of BN selection and cut node detection.

Recovery time: The time at which a cut node is to be recovered using the BNs or MNs in the network.

Delivery ratio: It is the ratio of the number of packets received successfully and the total amount of packets transmitted.

\[ \text{Delivery ratio} = \frac{\text{Number of packets received}}{\text{Amount of packets transmitted}} \]

Packet drop: It is the amount of packets dropped during the data transmission.

Packet delay: The delay of network describes how long the network takes to transmit a bit to the destination. Unit of this parameter is seconds (s).

6. CONCLUSION

In this paper, Particle Swarm Optimization algorithm based backup node selection for cut node recovery has been presented and this proposed approach has been simulated in the Network Simulator.
Cluster formation has been done based on the node weight. In each cluster, backup nodes were selected using the proposed particle Swarm Optimization algorithm. Then the cut node in the network has been detected based on the hop-count and link cost value of each sensor node. This cut node has been recovered by the selected backup nodes if it was in one hop distance to the backup node, otherwise it has been recovered by mobile nodes in the network. Simulation results showed that the performance of the proposed approach outperformed that of the existing approach in terms of overhead, energy efficiency, network lifetime and BN selection time.

REFERENCES:

[1] Gherbi, Chirihane, ZihoudaAliouat, and Mohamed Benmohammed. "An Adaptive Clustering Approach To Dynamic Load Balancing And Energy Efficiency In Wireless Sensor Networks". Energy 114 (2016): 647-662. Web.
[2] Abdollahzadeh, Sanay and NimaJafariNavimipour. "Deployment Strategies In The Wireless Sensor Network: A Comprehensive Review". Computer Communications 91-92 (2016): 1-16. Web.
[3] Mehajabin, Nusrat et al. "Energy-Sustainable Relay Node Deployment In Wireless Sensor Networks". Computer Networks 104 (2016): 108-121. Web.
[4] Cerulli, Raffaele, Monica Gentili, and Andrea Raiconi. "Maximizing Lifetime And Handling Reliability In Wireless Sensor Networks". Networks 64.4 (2014): 321-338. Web.
[5] Ojha, Tamoghna, SudipMisra, and Narendra Singh Raghuvanshi. "Wireless Sensor Networks For Agriculture: The State-Of-The-Art In Practice And Future Challenges". Computers and Electronics in Agriculture 118 (2015): 66-84.
[6] K. Sharma and T. Sharma, “ZBFR: zone based failure recovery in WSNs by utilizing mobility and coverage overlapping”, Wireless Networks, vol. 23, no. 7, pp. 2263-2280, 2016.
[7] S.Karthikeyan, S.Jayashri, “Energy Utilization Strategies using Novel Approach in Wireless Sensor Networks”, International Conference on Software Engineering and Mobile Application Modelling and Development, Dec 2012, pp: 1-5.
[8] S.Karthikeyan, S.Jayashri, “Energy Efficient System for Heterogeneous Wireless sensor Networks”, European Journal of Scientific Research, Volume 72,No 4 march 2012, PP:599-607. ISSN: 1450-216X.
[9] G. Savitha and S. Karthikeyan,"Architecture for diverse modulation techniques in SDR", concurrency computation practice and experience,John wiley & sons, Ltd, February 2019,PP: 1-10.