International Journal of Pure and Applied Mathematics
Volume 116 No. 10 2017, 425-433
ISSN: 1311-8080 (printed version); ISSN: 1314-3395 (on-line version)
url: http://www.ijpam.eu
Special Issue
ijpam.eu

A Light Weight Hop Count Based Link Failure Detection Protocol in Wireless Sensor Networks

E. Anna Devi[1], J. Martin Leo Manickam[2]

[1]Research Scholar, Satyabhama University, Chennai, India, annadevimit@gmail.com, 9884711667
[2]Professor, St. Joseph's College of Engineering, Chennai, India, josephmartin74@yahoo.co.in, 9445209197

Abstract:* Wireless sensor network are group of sensor nodes, that converts the physical phenomenon e.g. heat, light, motion, vibration, and sound into electrical signals. These nodes can fail because of various reasons like mechanical failure, environmental degradation, battery failure, since nodes are powered with small size battery. This failure should be detected and corrected then and there, because the failure of two or more nodes can lead to the failure of an entire link. In this paper, the link failure can be detected by Hop count based cut detection algorithm. The difference between the average link cost value and maximum link cost value is calculated and it is compared with the threshold value. If the difference value is less than the threshold value, then the node with minimum hop count is assigned as a cut node, and the link is said to be failed. The algorithm chooses another link through which the data can be routed, so that it reaches the destination without any data loss. Simulation results show that cut detection time and the energy conservation of our proposed work is reduced than that of Distributed Cut Detection algorithm.

Key Words: Wireless Sensor Network, link failure, link cost, hop count, cut Detection, node failure

I . INTRODUCTION:

Due to the advances in micro-electro-mechanical systems (MEMS) technology, digital electronics, wireless communications, have enabled the growth of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate unchained in short distances. These tiny sensor nodes, which consist of sensing, data gathering and compiling, and Trans receiver components, influence the idea of sensor networks based on cooperative effort of a large number of nodes. Sensor networks characterize a significant enhancement over traditional sensors [6]. Sensors can be located far from the actual location, i.e., something known by sense perception.. The sensor nodes communicate by hop by hop communication, since the coverage range of sensor node are small .Sensors are small in size powered with small size batteries with less energy.

Recently scientists have found that the noteworthy energy saving can be realized by elegant managing the duty cycle of nodes in WSN with high node density and it can prolong the network lifetime. In this method, some nodes are scheduled to sleep (or enter a power saving mode) while the remaining active nodes keep working. However, if more number of nodes are in sleep mode it will lead to a WSN to be disconnected, i.e. the set of working nodes will be isolated[7]. The over-loading of working sensor nodes will cause easily exhausted and failed. In addition to possible hardware or software faults, sensors may fail because of severe climate conditions or other hash physical atmosphere in the sensor filed. It is therefore vital to construct a fault-tolerant WSN that will uninterruptedly provide needed services despite sensor failures. This is the fault-tolerance requirement. In this paper a fault detection protocol is proposed , which is based on hop count based mechanism, which can detect the faulty node and which routes the data packet through other path. Thereby the data reaches the destination without any data loss , at the same time the failure of the nodes can be identified. This algorithm consumes less power, so it is also called as light weight protocol.

II .Node Failure in WSN:

Nodes in WSN are interdependent, because they use hop by hop communication. A node may fail because of various reason. If a node fails it leads to the failure of the entire network in the following way. In WSN each node is not aware of its neighbor node status, so the failure of a node is not known by its neighbor. Without knowing its failed status the neighbor node simply forwards the data, again and again, which leads to the failure of that particular node by depletion of its energy.

If the same situation prolongs, it leads to the failure of the entire networks or at least the partition of the network. So it is mandatory to check the link failure and transmit the data through the other path, so that the data loss can be avoided.

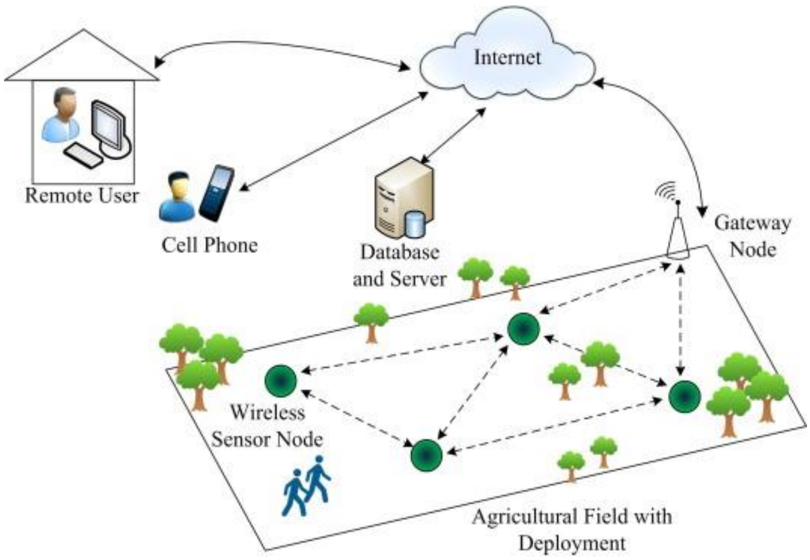

Figure 1: Architecture of Wireless sensor network

III. DISTRIBUTED CUT DETECTION (DCD)

The algorithm is asynchronous and distributed: it involves communication between neighboring nodes, and is strong to short-term communication breakdown between node pairs. A key component of the DCD algorithm is a circulated iterative computational step through which the nodes from sensor network compute their electrical potentials.

3.1 CUT: Wireless sensor networks are a hopeful technology for taking care of large regions at high spatial and temporal clarity. In fact, node failure is expected to be quite common due to the typically limited energy account of the nodes. Breakdown of a set of nodes will reduce the number of multi-hop paths in the network. Such failures can cause a subset of nodes – that have not failed – to become disconnected from the rest, resulting in a "cut". Two nodes are said to be disconnected if there is no path between them. We consider the problem of identifying cuts by the nodes of a wireless sensor network. There is a specially designed in the network which is called the source node. The source node may be a base station that serves as an interface between the network and its users. Since a cut may or may not separate a node from the source node, we distinguish between two distinct outcomes of a cut for a particular node.[1]

3.2 CCOS AND DOS:

When a node u is disconnected from the source, we say that a DOS (Disconnected from Source) event has occurred for u. When a cut occurs in the network that does not separate a node u from the source node, we say that CCOS (Connected, but a Cut Occurred Somewhere) event has occurred for u. By cut detection we mean (i) detection by each node of a DOS event when it occurs, and (ii) detection of CCOS events by the nodes close to a cut, and the approximate location of the cut.[1]

IV. HOP COUNT BASED CUT DETECTION (HCCD)

A cut node in the network is identified based on hop count. Initially, hop count and link cost value of each node are calculated. Link cost value of each node is calculated using residual energy, queue size, and link quality. Using these values data structure and neighbor table of each node is updated. For routing, the node with minimum hop count and maximum link cost value is selected as a next hop node from the neighbor table of a node. Before transmitting the data packet, the source node sends route request (RREQ) to the destination and the destination node sends route reply (RREP) to the source node after receiving the message. If the source node doesn't receive the message from the destination, it retransmits the message by comparing the difference of link cost value with the threshold value of each node.

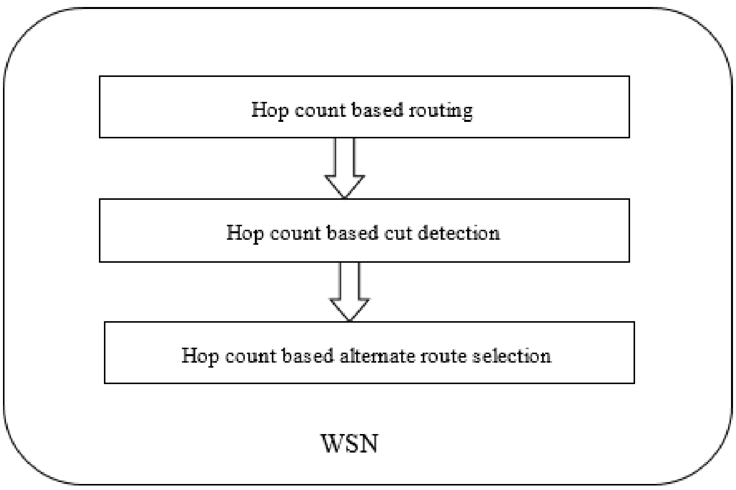

Figure 2: Block diagram of HCCD

If the difference is less than the threshold value, then the node is assigned as a cut node. So another node with minimum hop count and maximum link cost value is selected as an alternate hop node. Figure 2 shows the block diagram of our proposed approach.

V. LINK FAILURE DETECTION

Before the packet transmission, the source node transmits the data packet Route Request (RREQ) message to the destination through the selected next hop nodes. After receiving the RREQ message, the destination transmits the Route Reply (RREP) message to the source node. Then the source node transmits the packet to the selected next hop node. In this transmission, packet and neighbor table are updated with the information of neighbor nodes such as residual energy, hop count, the size of the queue, link quality and link cost until reaching the destination. This process is continued for each transmission as shown in figure 3. Table 1 shows the example data structure of the source node.

Due to the number of transmissions, few nodes may lose its energy level that is known as cut nodes. Again the source node transmits the RREQ to the destination, if the cut node available in the route, the source node doesn't receive the RREP message within the round trip time. So the source node retransmits the packet and each hop node in the route checks link cost values of neighbor nodes. To find the cut node in the route, the difference between the average link cost values and the maximum link cost value is calculated and the percentage of difference is compared to the threshold value. If the threshold value is greater than the percentage of difference, then the node with minimum hop count and maximum link cost value is decided as a cut node. The percentage of difference is calculated as

$$diff(\%) = \frac{C_{High} - C_{Avg}}{C_{Avg}} \times 100\%$$

Where, C_{High} *and* C_{Avg} are represented as maximum link cost value of a node with a minimum hop count and average link cost value except node with minimum hop count respectively.

As shown in figure 4, if the node (N_{12}) is found as a cut node the source node selects another node (N_{11}) which has minimum hop count and maximum link cost value as an alternate next-hop node from the neighbor table. Similarly, the cluster head (CH_3) selects another node (N_{37}) as a next hop node from the neighbor table of CH_3. After receiving the packet through the alternate path or route, the destination node compares the node ID and its minimum hop count of the previous packet with the node ID and that of current packet as shown in figure 5. By comparing those, the destination node can identify the cut nodes and it alerts the source node with IDs of cut nodes.

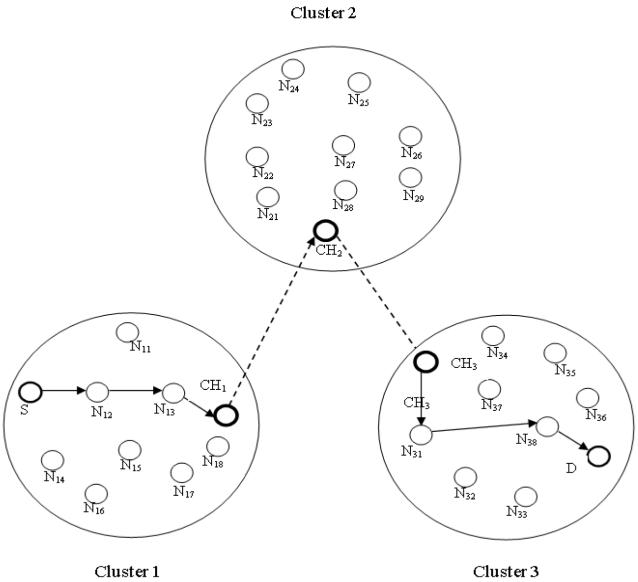

Figure3: Cluster Based routing

Table 1: An example Neighbor Table of source node.

Neighbor ID	Hop count	Cost
N_{12}	**7**	**7**
N_{11}	7	6
N_{14}	8	6.5
N_{15}	8	6

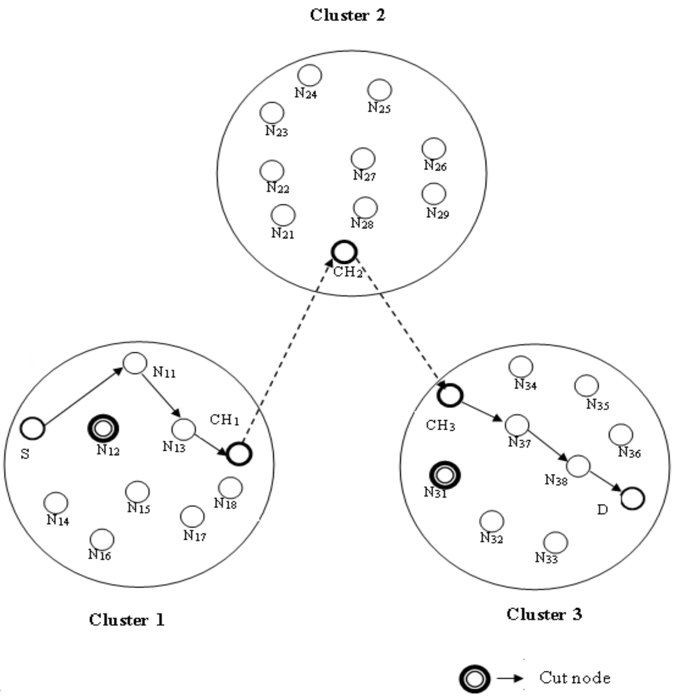

Figure 4: Hop count based cut detection

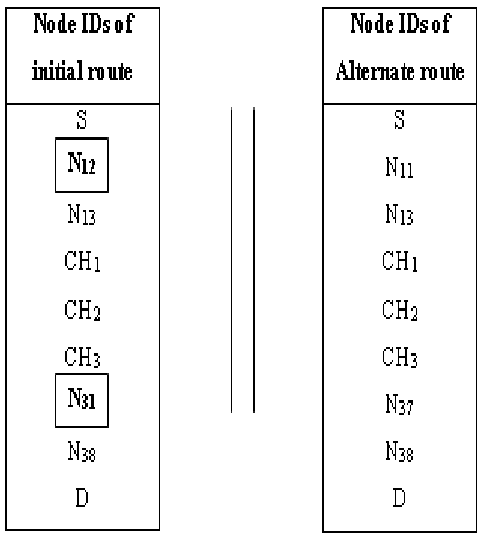

Figure 5: Destination node confirms the cut node by comparing the route IDs of initial route and alternate route.

VI. RESULTS AND DISCUSSIONS

Table2: Simulation parameters

Parameters	Value
Number of nodes	500
Wireless protocol	802.11
Area	1000×1000
Simulation time	50s
Packet size	500
Routing protocol	AODV
Transmit power	0.660W
Receiving power	0.395W
Initial energy	40J
Transmission range	250m
Constant bit rate	500kbps

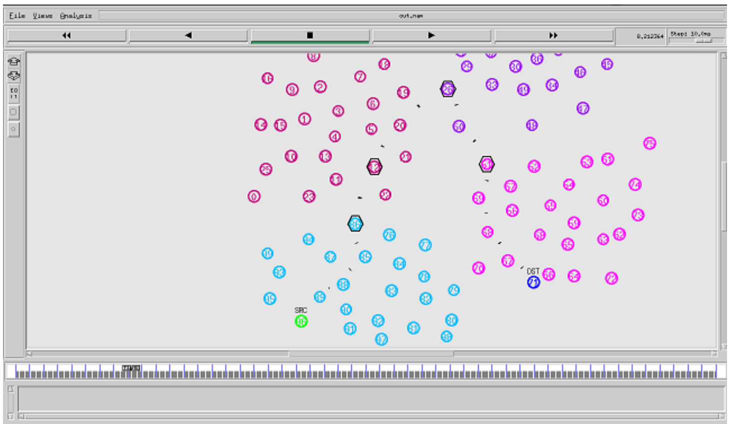

Figure 6: Cluster formation

Figure 6 shows the cluster formation of our proposed approach. From the figure, four set of clusters is formed for 100 nodes. Node 96 and node 71 are considered as a source and destination respectively. After the cluster formation, the hop count is calculated for each node in a cluster. Then the node with minimum hop count is selected as a cluster head. Node 86, node 12, node 26, node 51 are selected as a cluster head for each cluster as shown in the figure. Figure 7 shows the cut detection of our proposed approach. As shown in the figure, node 85 is detected as a cut node so the node 88 selects the node 87 as an alternate node. Then the destination sends ACK about the cut node to the source node.

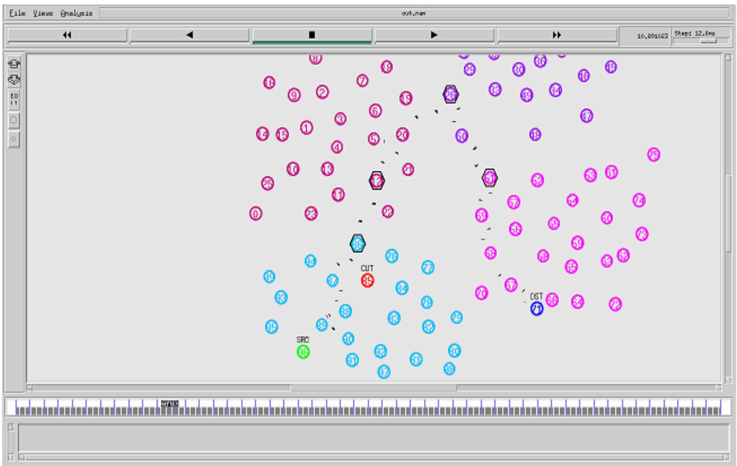

Figure 7: Cut detection

Performance based on Rates

In this section, performance metrics of our proposed approach is estimated for varying rates 100, 200, 300, 400 and 500kbps. Figure 8-11 show that packet delay, delivery ratio, energy consumption and throughput of our proposed approach for varying rates. Packet delay of our proposed approach is shown in figure 8 and it is reduced to 23% than that of the DCD algorithm. Figure 9shows the delivery ratio of our proposed approach. Compared to the existing work, delivery ratio of our proposed approach is increased to 88%. Figure 10 and 11 show the energy consumption and throughput of our proposed approach respectively. Compared to the existing work, those of our proposed reduced to 23% and increased to 63% respectively

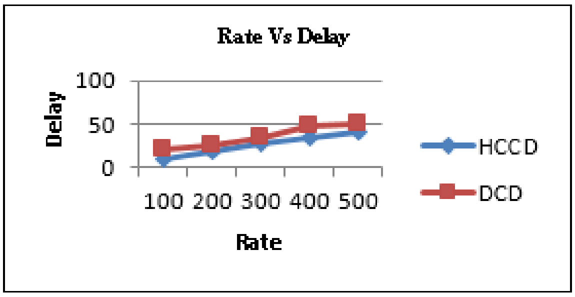

Figure 8: Rate Vs Delay

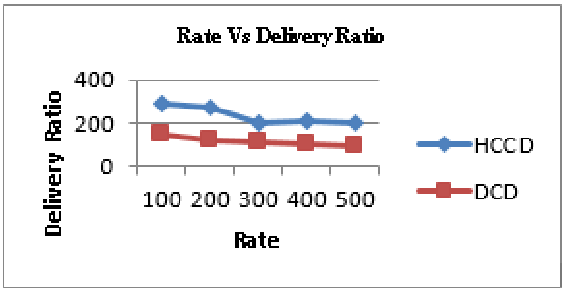

Figure 9: Rate Vs Delivery Ratio

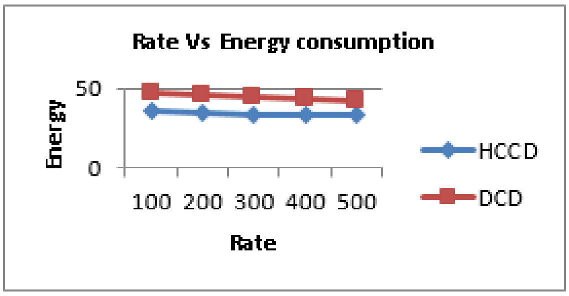

Figure 10: Rate Vs Energy Consumption

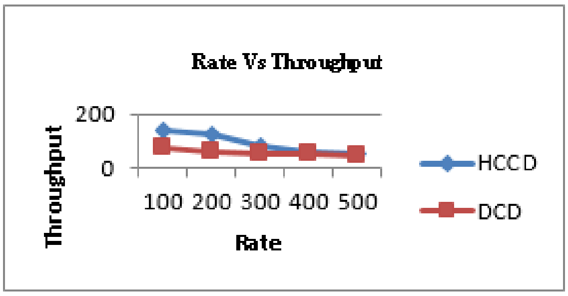

Figure 11: Rate Vs Throughput

VII. CONCLUSION

In this paper, Hop-count based cut detection have been proposed and we have simulated our work using the network simulator NS2. In our work, The difference between the average link cost value and maximum link cost value is calculated and it is compared with the threshold value. If the difference value is less than the threshold value, then the node with minimum hop count is assigned as a cut node, and the link is said to be failed. The algorithm chooses another link through which the data can be routed, so that it reaches the destination without any data loss. Then the node with a second minimum hop count and maximum cost value has been selected as an alternate node. Simulation results showed that cut detection time of our proposed approach has been reduced than that of DCD algorithm.

REFERENCES

[1]Gherbi, Chirihane, Zibouda Aliouat, 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 Nima Jafari Navimipour. "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, Sudip Misra, and Narendra Singh Raghuwanshi. "Wireless Sensor Networks For Agriculture: The State-Of-The-Art In Practice And Future Challenges". Computers and Electronics in Agriculture 118 (2015): 66-84. Web

[6]C. Intanagonwiwat, R. Govindan, D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," Proceedings of the ACM MobiCom'00, Boston, MA, 2000, pp. 56–67

[7]S. Yang, F. Dai, M. Cardei, J. Wu, F. Patterson, "On Connected Multiple Point Coverage in Wireless Sensor Networks" International Journal of Wireless Information Networks, 4(13), 2006 , pp. 289-301