



Identifying Partitions in Wireless Sensor Network

E. Anna Devi¹  · J. Martin Leo Manickam²

Received: 23 May 2018 / Accepted: 30 July 2018 / Published online: 7 August 2018
© Springer Science+Business Media, LLC, part of Springer Nature 2018

Abstract

Wireless sensor networks, consists of a group of tiny nodes, which are powered by small size batteries, which can get failed very easily. The failure of a single node may lead to the failure of an entire network in the following way. If a particular node is unaware of the failed situation of its neighbor node, it simply forwards the packet again and again as it will not receive any acknowledgment. This message passing merely wastes the precious energy of the node, which leads to its failure. This process continues which leads to the failure of the entire network. It is crucial, to identify the node which is failed, as and when it happens and to reconnect the network. In this paper, a hop count based cut detection algorithm is proposed to detect the failure of the nodes and, mobile nodes are used to reconnect the partitioned network. In HCCD at each hop, every node select the node with minimum hop count and maximum link cost, thereby the worst node can be identified as cut node and this cut can be identified before the network actually fails. Experiment results shows that hop count based cut detection outperforms the traditional distributed cut detection algorithm and reconnection using mobile nodes avoids data loss.

Keywords Wireless sensor networks · Cut detection · Hop count · Reconnection · Mobile nodes

1 Introduction

In the present world, wireless sensor networks play a dynamic part in observing human less or dangerous environments, including atmosphere observing, chemical course identifying, and natural calamity observing like earthquake, cyclone, Tsunami, and

✉ E. Anna Devi
annadevimit@gmail.com

J. Martin Leo Manickam
josephmartin_74@yahoo.co.in

¹ Faculty of Electronics, Sathyabama Institute of Science and Technology, Chennai, India

² Department of ECE, St. Joseph's College of Engineering, Chennai, India

war-field monitoring [12]. As these geographical areas are unrestrained and may be instable, the network and associated nodes will hurt injury, from dangers, direct outbreak or unintentional harm from nature and climate. They may also degrade because of exhaustion of battery power or any other hardware fault. In such cases the produced information will be neglected or the nodes may fail to gather the neighboring node information [10]. If one or two nodes fail in the network at different regions, then such situation can be managed by selecting an alternate path. But if more number of nodes fails, then managing the situation without data loss is a big challenge [3].

WSN is a collection of small sized nodes designed into a network such that each node have the skill to sense, pass and process the data. Each node has a radio-frequency transceiver, sensing element, storage element powered by a battery. Nowadays sensors are widely employed in various research fields since they can monitor any physical parameter and hence forecasting can be made easier [1–4]. In unmanned and dangerous area, the sensors are randomly deployed according to the applications for which they are being used [5]. In order to increase the life time of the network, the energy should be saved, as these sensors are powered with small sized batteries. In a network, each sensor nodes acts as relay nodes, performing hop by hop communication. There will be no direct connection between a sensor node and base station. So each and every node needs to communicate with all the other nodes, by establishing wireless links between them.

The wireless sensor networks has a number of small sized nodes, which may collapse because of various reasons like, perfunctory failure, electrical failure, environmental degradation, tampering of the node, battery depletion because of small sized battery. If a single node in the network gets failed because of the above said reasons, it may lead to the failure of the entire network in the following way. The nodes are inter-dependent; each node has to pass the data to the next node. If a node receives the data it sends acknowledgement back to the transmitted node. Likewise the transmission continues until the data reaches the destination. Consider a situation in which a node is failed and its neighbor node is not aware of this failed situation. The neighbor node simply forwards the data to the next node (failed node) again and again, as it does not receive the acknowledgement. The neighbor node thinks that the acknowledgement is lost and it will send the data again. This leads to the wastage of precious energy of the node, which ends in the failure of that node. The process will be cumulative which leads to the failure of the entire network.

In this paper a hop count based cut detection algorithm is proposed to detect the cuts quickly and the failed routes can be linked by mobile nodes. By calculating the residual energy, link quality and free queue size, the link cost between two nodes can be calculated. From AODV routing table each node calculates the minimum hop between each node and destination. By considering this link cost and hop count, the partition in the network can be identified. If this partitions are not identified it leads to loss of crucial data. Simulation results shows that the HCCD algorithm detects the cuts with less time when compared with traditional distributed cut detection algorithm and mobile nodes reconnect the partitioned network, to achieve 100% packet delivery ratio.

2 Related work

Alrikson et al. [1], proposes, an algorithm for repairing the partition that happened because of the failure of a group of nodes. This algorithm uses a fresh number which is broadcasted from the base station per epoch. If there is no response for that from the node then that node is declared as failed node. Then mobile nodes are initiated to repair the partition. It undergoes two phases, monitoring phase and verification phase. This paper discusses on the use of few mobile nodes. Here multi agent based routing is not employed.

El Alami and Najid [2], define a SEFP based algorithm for electing the cluster head. This paper gives a fuzzy based cluster head election for heterogeneous wireless networks, in order to improve the lifetime of the nodes and the entire network. Baroha [4] defines a distributed cut detection algorithm, in which cut is detected using two process called DOS and CCOS event. The DOS is a process in which the node itself declares that it is going to fail to its neighbor node. The CCOS event happens, if there is any change that happens in the environment, and then a neighbor node initiates a probe. By using these two events the cuts in a network can be detected very correctly. The drawback of this algorithm is, it generates many false alarm

Cerpa et al. [7], presents an algorithm called ASCENT, for providing a redundant connection in wireless sensor networks. In WSN nodes are densely deployed, so they will sense and relay the data. In this a set of nodes can be kept as redundant nodes, which can be used in situations where when critical nodes fail, these can be used. Chong and Kumar [9] proposes a technique to handle the challenge of few resources and large number of interworking nodes. This algorithm includes three methods like data centric approach, specific reaction on received information and local behavior using a policy control mechanism. Dini et al. [11] gives a partition detection algorithm for repairing the partitioned network. Here two types of partition are assumed. One is safe partition and the other is isolated partition. This algorithm uses a mobile node to link the partitioned network. Safe partition is one in which the nodes have contact with the base station and isolated partition is one in which the node lost the contact with the base station. The mobile node uses the safe partition to bridge the isolated partition. If number of partition is more the algorithm cannot be used.

Ou and He [8] proposes a path planning algorithm for localizing, to find the location of the nodes. This algorithm guarantees mobile anchor based localization is perfect and each and every node in the network knows their location. Calculating localization is a very tough process. In Distributed cut detection algorithm [6], each and every node detects the cuts using electrical analogy. Hauspie et al. [13] proposes, a partition detection mechanisms, one using a centralized approach, the other one utilizing the advantages of a distributed mechanism. The simulations show that both approaches detect partitioning reliably, with both having unique advantages. Jeyashri [14] proposes an algorithm to detect separation using two phase, one is Reliable neighbor discovery phase and the other is Distributed cut detection phase.

Kristalina et al. [15] proposes an Localization algorithm to find the best location of nodes in WSN. Ma et al. [16] proposes a new data-gathering mechanism for large-scale wireless sensor networks by introducing mobility into the network. A mobile data collector, for convenience called an M-collector, could be a mobile robot or a

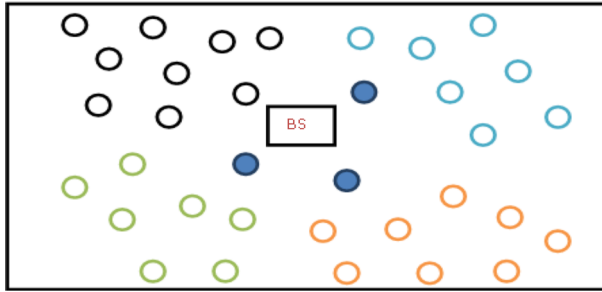


Fig. 1 Three isolated partitions of the network

vehicle equipped with a powerful transceiver and battery, working like a mobile base station and gathering data while moving through the field. Naskath et al. [17] uses a mobile nodes for coverage in WSN. Tewani et al. [18] proposes a fault tolerant algorithm which can withstand cuts in WSN. Pallotoino et al. [19] proposes a safe and secure traffic routing for mobile nodes. Ritter et al. [20] proposes an efficient partition detection system for mobile Ad hoc network.

Consider the situation shown in Fig. 1, in which due to the failure of three critical nodes (marked in blue) the network is divided into three isolated partition, which may lead to loss of data, and also the failure of other nodes. So these partitions should be identified quickly and it should be repaired efficiently. So in this paper a Hop count based cut detection algorithm is proposed to detect the cuts or partition and mobile nodes are used to reconnect the partition.

3 Hop Count Based Cut Detection Algorithm (HCCD)

3.1 AODV Routing

The routing protocol used here is Ad hoc on demand Distance Vector Routing (AODV Routing). The routes are maintained as long as they are required. In AODV the source node broadcast the route request (RREQ) to its entire neighbor nodes. The nodes forward this request to other nodes and make a record of this, until the request reaches the destination. Thus they create a temporary route. The destination node receives this request and sends a reply (RREP) to the source node through the temporary route to the source node. So each node in the route knows the entire route to the destination. From this each node calculates the hop count to the destination. Figure 2 explains the broadcasting of RREQ from source and destination and forwarding of RREP from destination to source.

3.2 Hop Count Based Cut Detection

In HCCD, every node exchange hello packet to its neighbor periodically. The hello packet as shown in Fig. 3 contains information like source id, from which node the

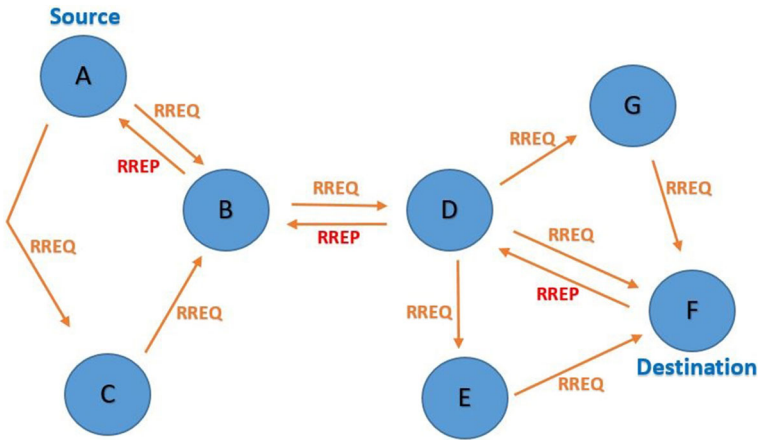


Fig. 2 Routing creation in AODV

Source ID	Packet ID	$E_{residual}$	Q_{free}	HOP _{minimum}
-----------	-----------	----------------	------------	------------------------

Fig. 3 Hello packet

packet is initiated, Packet ID is to avoid duplication of the packet. $E_{residual}$, the remaining energy of the node, Q_{free} the remaining Q size, and HOP min the minimum number of hops between that particular node and sink as shown in Fig. 3.

All the neighbor nodes receive this hello packet and update its neighbor table. Every node calculates two important factors the Hop count and link cost. Minimum hop count is the minimum number of hops between that node and sink and link cost is the cost of that link. The link cost is calculated based on residual energy and link quality.

The next hop node is also designated with the help of the link cost. Link cost value (C_{ij}) is intended with the help of free queue size of node i , residual energy of node i and the link quality within nodes i and j . The link cost is calculated by Eq. 1.

$$C_{ij} = \frac{E_{residual_j}}{E_{initial_j}} \times cc_E + \frac{Q_{free_j}}{Q_{total_j}} \times cc_Q + LQ_{ij} \times cc_L \tag{1}$$

where, CC_E , CC_Q and CC_L are signified as the constant coefficients, $E_{residual}$ and $E_{initial}$ signify the residual and initial energy of node j correspondingly, Q_{free} and Q_{total} represent the free queue size and maximum queue size of node j correspondingly and LQ_{ij} signifies the link quality between two nodes i and j . The node with maximum link cost value is designated as a next hop node. The condition is signified as

$$\max(C_{ij}), \forall j \in i \tag{2}$$

In the neighbor table the Hop count is arranged in ascending order and link cost value of the neighbor nodes are organized in descending order. The first element of the neighbor table (i.e.) the node with minimum hop count and maximum link cost is

designated as the first element to the next hop. By picking the subsequent hop node as next hop, the source node sends the packet to the cluster head and the cluster head diffuses the packet to the destination via the adjacent cluster heads.

From the neighbor table choose the node with minimum hop count and maximum link cost as the next hop node for routing. Now, the difference between the maximum cost and average cost is calculated using the below formula in Eq. (3). If the difference is less than the threshold then the node with minimum hop count is the cut node. So that particular node can be removed and the next node with next minimum hop count can be selected for routing.

$$diff(\%) = \frac{C_{High} - C_{Avg}}{C_{Avg}} \times 100\% \quad (3)$$

The flow chart in Fig. 4 gives a detailed explanation of hop count based cut detection algorithm. The first step is to calculate the hop count and link cost. The hop count is calculated from AODV routing protocol and the link cost is calculated using Eq. (2). The second step is to select the minimum hop count and maximum link cost node as the next node for transmission. The third step is to calculate the difference between the maximum link cost and average link cost using Eq. (3). If the difference is greater than the threshold then the transmission will continue or if the difference is less than threshold then that particular node with minimum hop count is considered as cut node.

3.3 Mobile Nodes

As soon as the HCCD algorithm detects the cut node, mobile nodes are initiated by the base station. The mobile node is assumed to be consisting of sensing element, Motor and MICA2 module. MICA2 was the main component helping computation and communication for the mobile nodes. The sensed information from the sensing element and control motors will be administered over a motor board. As such, sensors are skilled of responding to actions requiring interest, such as intrusions of fire, disaster management, which makes WSNs used in wide applications like surveillance and monitoring application.

The mobile nodes can be used in this situation, which has some form of movement. This mobile nodes or mobile robots, provides some form of physical intervention, including physical assistance, tracking and identification. This kind of mobile robots can also act as mobile gateways improving the efficiency of the network, e.g. in terms of carrying sensors which can replace failed sensors and also provide reliable connection. When mobile nodes and WSN are interoperated together, it can increase the lifetime of network. The mobile nodes are nothing but sensor nodes along with a moving device. So if there is a failed node, this can be replaced by a mobile node, which can perform the sensing and relaying function.

The number of failed nodes and the size of the network decide the number of mobile nodes needed to patch-up the partitioned network. Certain number of, mobile nodes will be allocated in each region to observe by the base station as shown in Fig. 5. The mobile nodes have to navigate through the network to the partitions, by the near -cut nodes (nodes which are going to fail). The mobile nodes are motorized with dominant

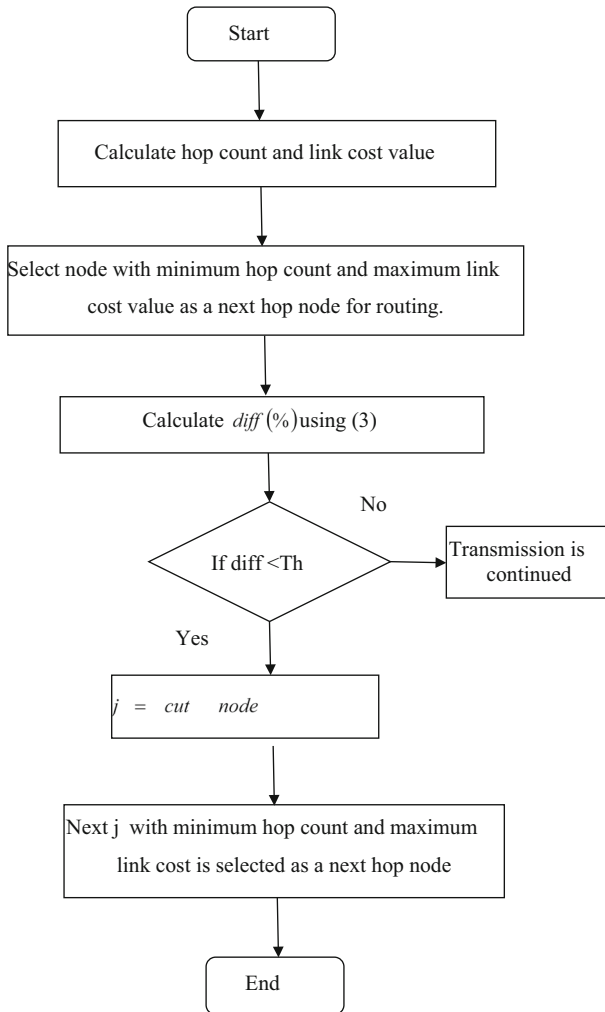


Fig. 4 Flow chart of HCCD

batteries that are thrice than the normal nodes. They cover more number of failed nodes. The nodes with low energy (near death nodes) are recorded in a queue in the base station, and mobile nodes serve them based on that queue. The first mobile node is assigned with nodes that are top in the queue.

$$\begin{aligned}
 & \text{Number of nodes to cover the cut} \\
 &= \text{Number of low energy sensor nodes} \\
 & / \text{Number of mobile nodes in a base station}
 \end{aligned}$$

Through the routing path, each sensor node can send packets, to request help when its energy is low. A navigation path will be created simultaneously by every low-energy

sensor nodes as it requests help. With the help of navigation algorithm, the mobile node identifies the path to throw energy nodes in the network, and reconnects the partition.

4 Performance Evaluation

Our proposed Hop-count based cut detection is simulated using network simulator NS2. In this work, we have considered 500 sensor nodes and the simulation parameters are shown in Table 1. These sensor nodes are performed in the 1000×1000 m region and are grouped as a cluster. Each node in a cluster has 250 m transmission range. This simulation uses IEEE802.11 wireless protocol. The node with minimum hop count is selected as a cluster head.

The transmitting power is 0.660 W and receiving power is 0.395 W. The energy of each node is initialized to be 40 J. The transmission range is 250 m. The constant bit rate is 500 kbps. Each time the node transmits, the power get reduced. The remaining energy is called as residual energy. The link cost is calculated, based on the residual energy of that node and link quality between two node and base station.

The nodes are randomly deployed and clusters are formed. In our simulation four clusters are formed, and each has its own cluster head. Then when the transmission starts, the nodes broadcast hello packet and the nodes which receive that packet, minimum hop count and link cost will be calculated, and the cut node will be detected. If needed, mobile node will be initiated. Figure 6 shows the formation of cluster and the detection of cuts using HCCD. The performance of HCCD is compared with traditional Distributed cut detection algorithm. In Distributed cut detection algorithm (DCD), cut is detected by using two events called Disconnected from the source (DOS) and Connected but cut occurred somewhere (CCOS) [6].

Table 2 gives a detail analysis of comparison of various parameters of HCCD and DCD with respect to number of nodes. Figure 7 give a detailed comparison of delay

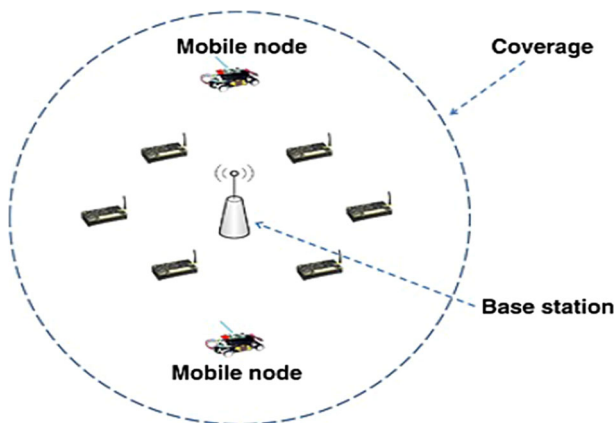


Fig. 5 Each base station has a set of mobile node

Table 1 Simulation parameters

Parameters	Value
Total no of sensor nodes	500
Wireless protocol	802.11
Area	1000 × 1000
Simulation time (s)	50
Packet size	500
Routing protocol	AODV
Transmitting power (W)	0.660
Receiving power (W)	0.395
Initial energy (J)	40
Transmission range (m)	250
Constant bit rate (kbps)	500

for HCCD and DCD. When the number of node is 100, the delay is 13 s for HCCD and it is 20S for DCD. As the number of node increases, the delay increases, but it is less than the value of DCD. Figure 8 gives the comparison of Number of nodes and delivery ratio. Packet delivery ratio is defined as the number of packets received by the destination to the number of packet generated by the source.

From Fig. 9, we understand that, the delivery ratio increases when compared to Distributed Cut Detection algorithm. Figure 10 gives the graph of number of nodes versus cut detection time. When the number of nodes is 100, the cut detection time is 15 s for HCCD and it is 18 s for DCD algorithm. As the number of node increases from

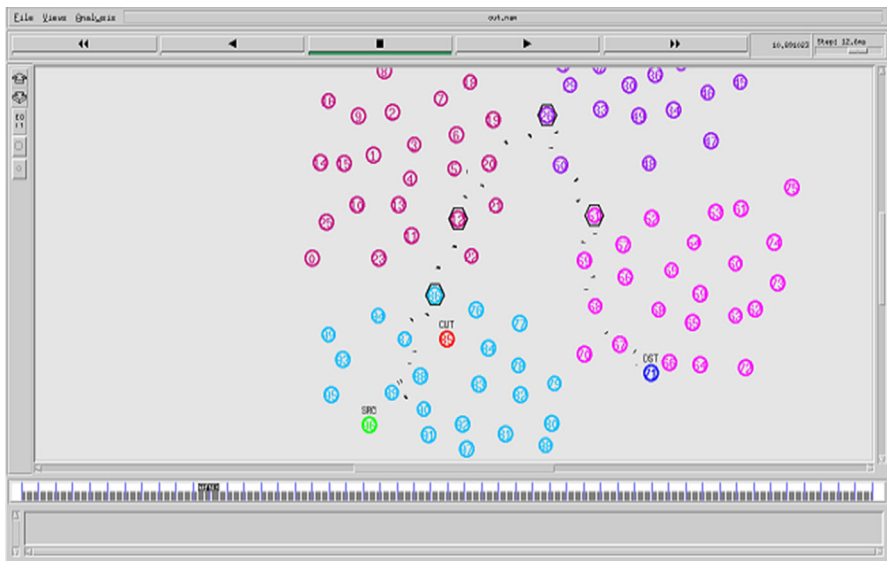
**Fig. 6** Detection of cut node

Table 2 Comparing various parameters of HCCD and DCD with No. of nodes

Nodes	Delay (s)		Delivery ratio		Cut detection time (s)	
	HCCD	DCD	HCCD	DCD	HCCD	DCD
100	13	20	982.98	474.71	15	18
200	20	27	540.535	280.295	21	25
300	28	34	435.78	306.95	26	32
400	36	43	373.04	174.21	34	41
500	45	49	296.982	132.394	43	47

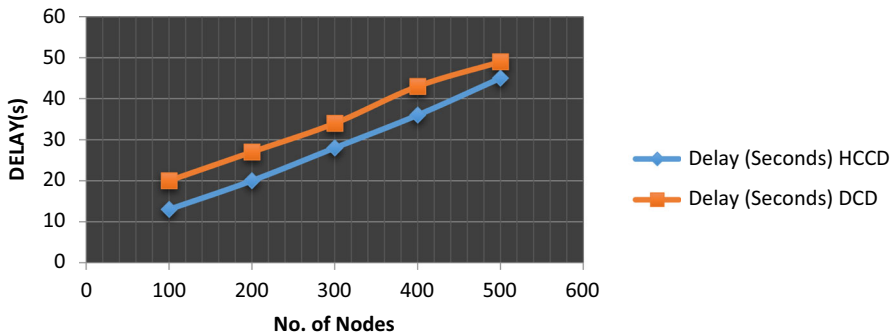


Fig. 7 No of nodes versus delay (s)

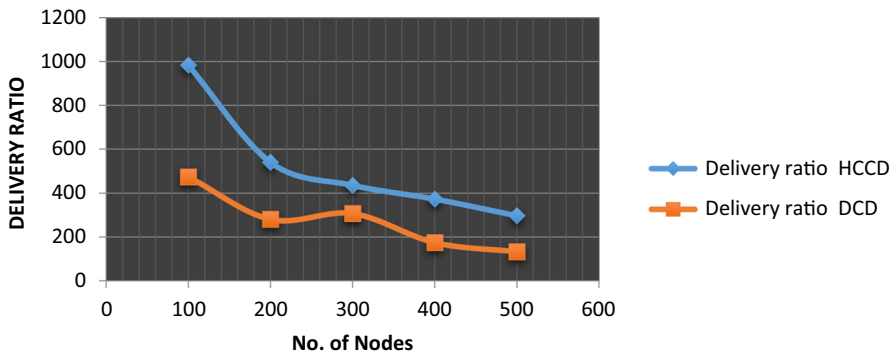


Fig. 8 No of nodes versus delivery ratio

200, 300, 400, 500, the cut detection time also increases, but it is less when compared to that of Distributed Cut Detection algorithm (DCD).

Table 3 gives a detailed comparison of Delay, Delivery Ratio and Throughput (kbps) with respect to Rate of transmission (kb). The simulation has been done with different rates like 100, 200, 300 kb, and various parameters were calculated and tabulated. Figure 10 gives the comparison of delay versus rates. It is clearly seen that the delay get reduced in Hop Count Based Cut Detection algorithm when compared to DCD algorithm. Figure 11 gives the comparison of Rate and Delivery ratio. The delivery

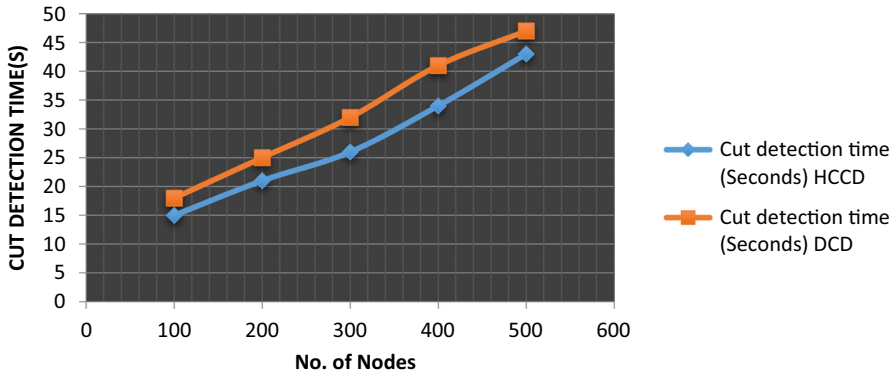


Fig. 9 No of nodes versus cut detection time (s)

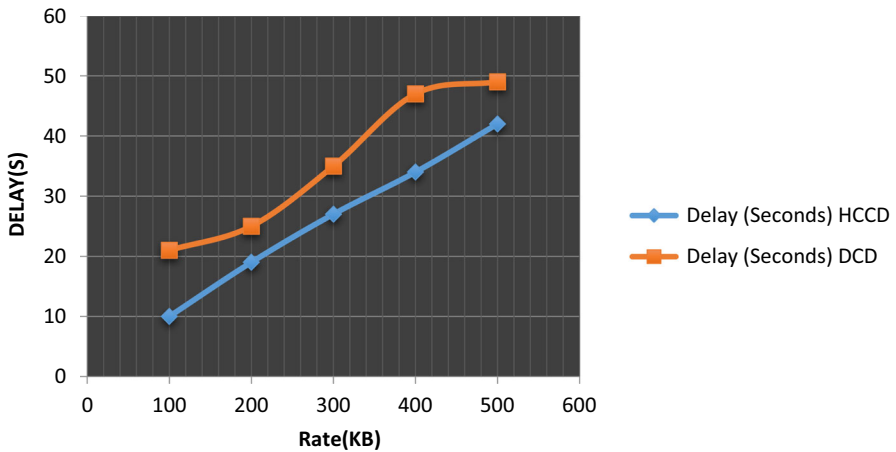


Fig. 10 Rate (kb) versus delay (s)

ratio increases in HCCD when compared to DCD algorithm. Figure 12 gives the graph of throughput versus Rate. Throughput is defined as the amount of packet received per unit time. The unit is kbps. Since the cut is detected quickly before it actually happens, therefore the throughput of HCCD increases as the rate increase when compared to DCD.

5 Conclusion

In this work to detect the partition in WSN, hop count based cut detection algorithm is used, and to reconnect the partitioned network, mobile nodes are initiated. In hop count based cut detection minimum hop count and link cost of each node to the sink node is calculated from the information which is got from the hello packet. Then the difference between maximum link cost and average link cost is calculated. If this

Table 3 Comparing various parameters of HCCD and DCD with respect to rates (kbps)

Rate (kb)	Delay (s)		Delivery ratio		Throughput (kbps)	
	HCCD	DCD	HCCD	DCD	HCCD	DCD
100	10	21	286.75	147.11	141	72
200	19	25	268.77	116.41	123	57
300	27	35	200.48	106.07	85	52
400	34	47	208.11	100.92	63	50
500	42	49	196.59	94.92	51	45

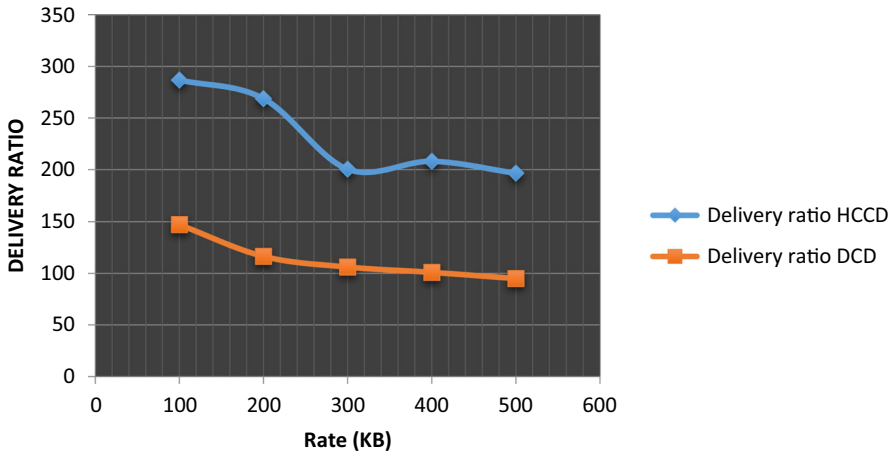


Fig. 11 Rate (kb) versus delivery ratio

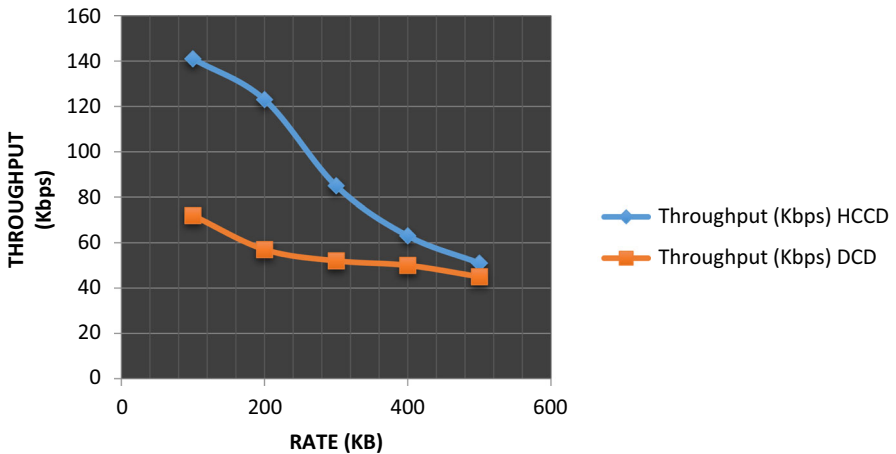


Fig. 12 Rate (kb) versus throughput (kbps)

value is less than the threshold, then that nodes is declared as cut node. Once a cut node is identified, the information will be given to base station. The base station will initiates mobile node to partition the cut, before the cut actually occur. The simulation has been performed, and it has been found that HCCD performs well when compared to traditional Distributed cut detection algorithm (DOS), in terms of packet delivery ratio, throughput and energy consumption. In future zone based recovery scheme can be used to cover the sensing regions of failed nodes.

References

1. Alriksson, P., Nordh, J., Arzen, K.H., Bicchi, A., Danesi, A., Schiavi, R., Pallottino, L.: Component-based approach to the design of networked control systems. In: ECC 2007. Proceedings of European Control Conference, Kos, Greece, July 2–5
2. El Alami, H., Najid, A.: SEFP: a new routing approach using fuzzy logic for clustered heterogeneous wireless sensor networks. *Int. J. Smart Sens. Intell. Syst.* **8**(4), 2286–2306 (2015)
3. Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. *IEEE Commun. Mag.* **40**(8), 1097–1102 (2002)
4. Barooah, P.: Distributed cut detection in sensor networks. In: Proceedings of 47th IEEE Conference on Decision and Control, pp. 1097–1102 (2008)
5. Yang, M.: Optimal cluster head number based on entry for data aggregation in wireless sensor networks. *International Journal of Smart Sensing and IntelligenceSystem* **8**(4), 1935–1955 (2015)
6. Barooah, P., Chenji, H., Stoleru, R., Kalmar-Nagy, T.: Cut detection in wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.* **23**, 483–490 (2009)
7. Cerpa, A., Estrin, D.: Ascent: adaptive self-configuring sensor networks topologies. *IEEE Trans. Mob. Comput.* **3**(3), 272–285 (2004)
8. Ou, C.-H., He, W.-L.: Path planning algorithm for mobile anchor-based localization in wireless sensor networks. *IEEE Sens. J.* **13**(2), 466–475 (2013)
9. Chong, C.-Y., Kumar, S.P.: Sensor networks: evolution, opportunities and challenges. In: Proceedings of the IEEE, vol. 91, pp. 1247–1256 (2003)
10. Ze, L.U.O., Lingzhi, Z.H.U., CHANG, Y., Qingyun, L.U.O., Guixiang, L.I., Weisheng, L.I.A.O.: False data filtering in wireless sensor networks. *Int. J. Smart Sens. Intell. Syst.* **9**(4), 1795–1821 (2016)
11. Dini, G., Pelagatti, M., Savino, I.M.: An algorithm for reconnecting wireless sensor network partitions. In: Proceedings of European Conference on Wireless Sensor Networks, pp. 253–267 (2008)
12. Gharghan, S.K., Nordina, R., Ismail, M.: Development and validation of a track bicycle instrument for torque measurement using the zigbee wireless sensor network. *Int. J. Smart Sens. Intell. Syst.* **10**(1), 124–145 (2017)
13. Hauspie, M., Carle, J., Simplot, D.: Partition detection in mobile ad-hoc networks. In: Proceedings of Second Mediterranean Workshop Ad-Hoc Networks, pp. 25–27 (2003)
14. Jayashree, R.: An algorithm to detect separation and reconnecting wireless sensor network partitions. *Int. J. Eng. Res. Technol.* **1**(10), 1–6 (2012)
15. Kristalina, P., Wirawan, Hendratoro, G.: Weighted hybrid localization scheme for improved node positioning in wireless sensor networks. *Int. J. Smart Sens. Intell. Syst.* **6**(5), 1986–2010 (2013)
16. Ma, M., Yang, Y., Zhao, M.: Tour planning for mobile data gathering mechanisms in wireless sensor networks. *IEEE Trans. Veh. Technol.* **62**(4), 1472–1483 (2013)
17. Naskath, J., Srinivasagan, K.G., Pratheema, S.: Coverage maintenance using mobile nodes in clustered wireless sensor networks. *Int. J. Comput. Appl.* **21**(4), 6–12 (2011)
18. Tewani, N.N., Ithapu, N., Rao, K.R., Sami, S.N., Pradeep, B.S., Deepak, V.K.: Distributed fault tolerant algorithm for identifying node failures in wireless sensor networks. *Int. J. Innovative Technol. Explor Eng (IJITEE)* **2**(5), 79–83 (2013). **ISSN: 2278-3075**
19. Pallottino, L., Savino, I.M., Schiavi, R., Dini, G.: A scalable platform for safe and secure decentralized traffic management of multi-agent mobile systems. In: REALWSN 2006. Proceedings of the ACM Workshop on Real-World Wireless Sensor Networks, Uppsala, Sweden (2006)

20. Ritter, H., Winter, R., Schiller, J.: A partition detection system for mobile ad-hoc networks. In: Proceedings of First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (IEEE SECON), pp. 489-497 (2004)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.