

Performance Appraisal of Cut Vertex Detection in Wireless Sensor Networks

E. Anna Devi ¹, J. Martin Leo Manikam ²

¹Research scholar, Department of Electronics and Communication Engineering
Sathyabama University, Chennai, Tamilnadu, India

²Professor, Department of Electronics and Communication Engineering
St. Joseph's College of Engineering, Chennai, Tamilnadu, India

Abstract: Many researches have been made in the area of cut vertex detection. Since cuts can lead to the breakdown of the entire system, it is vital to implement efficient methods to detect the cuts and rectify it for the proper restoration of the broken network. In this paper, we present two different approaches for detection of cut vertex with a difference that one is based on a small network and the other is an extension of the first with a difference in the method of determining cut vertex. We also compare the parameters with the help of simulation in NS2. Thus performance evaluation is done in this article which is based on the time delay/iterations as well as the average residual energy.

Keywords – cut vertex, detect, restoration, simulation, performance evaluation

I. INTRODUCTION

A wireless sensor network is a network of sensors deployed in large numbers such as to monitor physical or environment conditions such as temperature, pressure, humidity etc. Each node performs the functions of sensing, processing and communication. A typical wireless sensor network is created by these nodes on combining with routers and gateways. These distributed nodes communicate wirelessly to a central gateway which is connected to the wired world from which we can collect process and analyse the data. The routers are generally used to extend the network so as to provide additional connectivity between the distant nodes and the gateway

Since wireless networks are randomly deployed, they often suffer from disrupted connectivity due to various reasons such as limited battery power and tampering. These disrupted connectivity is often referred to as cut which may lead to packet loss, energy wastage etc. A number of methods have been proposed which may differ in the type of routing.

There are mostly two different types of routing: static and dynamic routing. Static routing is not regarded as a routing protocol since it is the process of manually determining the routing paths or tables via configuration files that is loaded when the routing device start up. Since these routes mostly do not change after it is configured, they are known as static routes. Dynamic routing is done during the execution of the routing depending on the routes which are used in the network. Since they change during the execution, they are known as

dynamic routes. Most of the methods used nowadays employ dynamic routing styles.

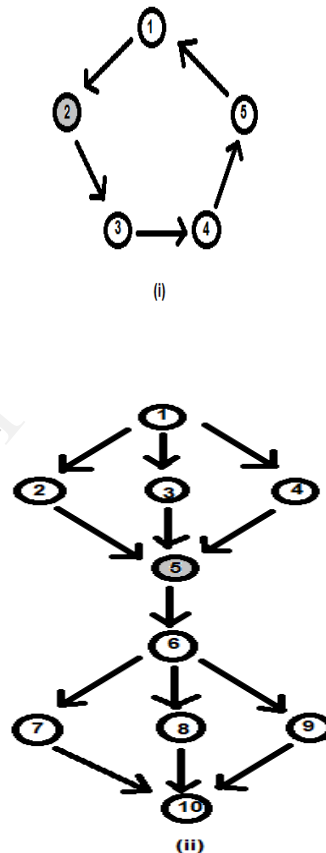
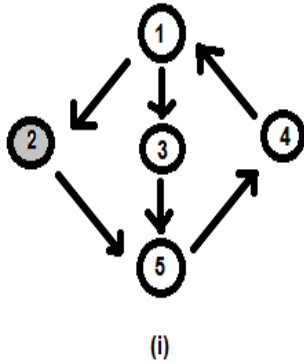


Fig.1 Critical cut in (i) a ring network (ii) a tree network

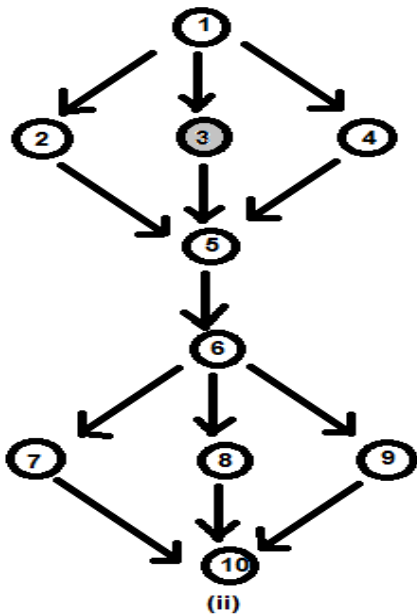
Cuts can be critical or noncritical. Critical cuts can be defined as the cut occurring to critical node. Noncritical cuts are those occurring to noncritical nodes. A critical node is referred to as cut vertex whose removal can break the connectivity of network.

Here in the fig 1, the removal of the marked nodes will lead to the network connectivity. Hence they are termed as critical cuts. The same network

can be modified such that the critical cut can be changed to noncritical cut



(i)



(ii)

Fig. 2 Noncritical cut in (i) a ring network (ii) a tree network

A slight modification in fig. 1 is given in fig. 2. Cuts in these networks are noncritical since alternate paths can accomplish the same. Always routing is chosen via the shortest possible path in order to avoid time delay.

Critical cuts can be explained with the help of base station concept. There can be two partitions after a cut: safe partition and isolated partition. A safe partition is the part having the base station and hence will be always connected with the base. But the isolated partition separated from the base station will be far away or disconnected from the base station and hence will be isolated from the base station.

Base station without knowing about the cut will be continuously sending information to its nodes. Because of the cut, information will be lost in the middle. As a result, information is wasted. At the same time, each node will be continuously waiting to receive information. Hence considerable amount of energy as well as time will be wasted in this way. Wireless sensor networks are limited with energy since they are powered up by batteries. Nodes are randomly deployed in the field and once they are deployed, it is difficult to manually monitor each and every node for failure because of limited battery power. In order to prolong the life of wireless sensor network nodes, it is desirable to conserve energy which is the limited resource s concerned with sensor nodes.

So on employing methods to detect cuts in wireless sensor networks, measures should be taken to reduce energy consumption and thus increasing the residual energy which could be used efficiently for data transmission. The proposed methods must consider both small scale as well as large scale networks depending upon the need.

II.RELATED WORK

In DCD [2], algorithm allows each node to detect DOS events and subset of nodes to detect CCOS event. The involved only local communication between neighbouring nodes and is robust to temporary communication failure between node pairs. So here the detection of cuts takes place. But when the cut detected is of cut vertex, and then it can lead to the reconstruction of the damaged network by informing it to the source.

A distributed algorithm CVD [1] scans the nodes of WSN parallel and edges are coloured on the interval coded spanning tree for cut vertex detection. Here only the cut vertices are detected. It can be modified to include network reconstruction by informing source about the failure.

A BFS based algorithm for cut edge detection is proposed in [5] but the cut edge detection is different from cut vertex detection. It should be noted that if a node is a cut vertex, then none of the edges incident on it is a cut edge and reversely if an edge incident on a node is a cut edge, that node is not a cut vertex.

DDFS [6] is a tree based approach in which each time the message visits a node, a counter is incremented. Each leaf node sends it to parent node and parent node collects all indices received from its children. If the indices received by parent node are smaller than that of children node, then that parent is called cut vertex. Time delay is much larger since DDFS has to traverse the edges serially. DDFS is sensitive to link/node fails due to serial nature.

Previously an algorithm [4] was developed to repair network partitions. They have employed mobile nodes to replace the position of the failed nodes to establish network connectivity. They have considered wireless sensor network partition into two types-safe partition and isolated partition. Safe partition will have base station whereas isolated does not. So a mobile node will take a proper position in between safe partition and isolated partition to establish connectivity between them. After detecting the network partitions, base station sends a fresh number called epoch. Safe partition which is in contact with base station will receive the new epoch whereas isolated partition will be holding the epoch. Mobile node will adjust itself to a position to establish the connection between safe partition and isolated partition.

Network was considered as a unidirectional graph in [3]. Each node sends probe message and a node was detected as cut vertex based on the reply message. Detection, computation, and neutralization are the steps involved in this approach. Cut vertex candidate sends message to different connection. If two such messages meet, then an arrival message is sent back to the issuer. At computation stage, CAM graph was constructed including the connections/nodes. If a node receives two arrival messages of different components, then an edge is added to the graph and a node was decided as cut vertex based on the graph. In neutralization stage, detected cut vertex was made as a non cut vertex. Disconnected components are made connected thus reducing cut vertices.

III. METHODOLOGY

A. Method 1: Cut Vertex Detection Based On Remove And Check

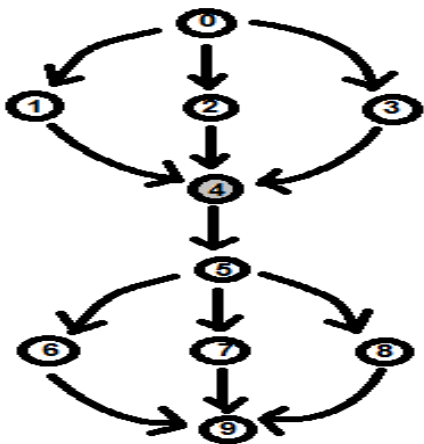


Fig 3 Cut vertex Detection of a small network

This method involves cut vertex detection based on remove and check. The initial stages are the same for both the methods.

In this method, cut vertex is detected by removing each node and then sending the data from

source to the destination. So each of the nodes is removed and checked in this way. On removal of which node the network communication get disrupted, that node is termed as cut vertex. When the cut vertex gets cut, this can lead to a critical damage of the system. So every node is checked for cut vertex starting from node 1 to node 9 in figure3 and finally node 4 and node 6 are detected as cut vertex because the cut of that node makes the entire network to be failed

B. Method 2: Cut Vertex Detection Based On Routing

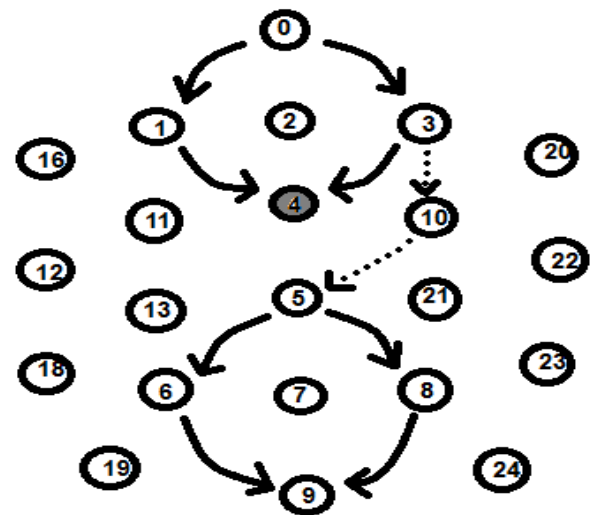


Fig 4 Routing table based cut vertex detection and alternate routing

The previous method based on few nodes has been extended to include some more nodes to allow the option for alternate routing. The initial stages of this method are similar to the previous method with modification only in the cut vertex detection and alternate routing.

IV. IMPLEMENTATION

How to identify the cut vertex and how to detect the failure or cut of cut vertex is explained using the following three modules

(a).Module 1 Algorithm

It includes three stages as explained below

i) Node Initialization

It includes initialization of real time parameters associated with each node such as q length, communication, initial energy etc.

ii) Node deployment

Nodes are generated and aligned in specific locations according to the requirement. After fixing each node to a specified, a wake up message will be sent by each node indicating a change in their position.

iii) Broadcasting

Within the communication range of each node, neighbour node detection and updating of each node with its nearest neighbour will be done. Broadcast by choosing different sources and destination in order to form the routing table and choose routing in the shortest possible path.

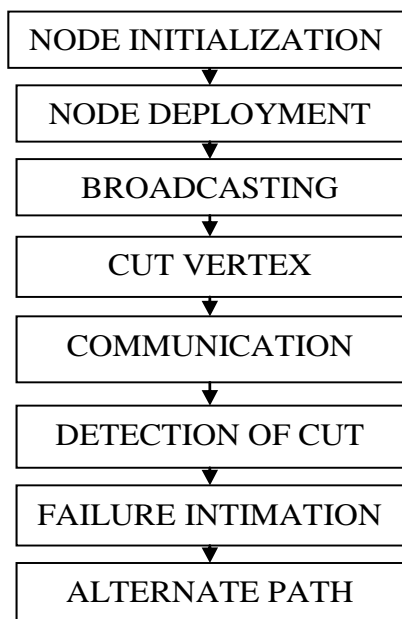


Fig. 5 Flowchart for Remove and check method

(b).Module 2 Algorithm

It includes the following two stages:

(i)Cut Vertex Detection Phase

Based on the information about which node is found in more number of routing tables, cut vertex is detected. In this way, three cut vertices have been determined.

(ii)Communication Phase

After finding the cut vertex, source continuously communicate with the destination with the available path

(c).Module 3 Algorithm

It includes the following three stages:

(i) Detection of Cut in the Cut Vertex

After finding out the cut vertices, a single path has been selected with two cut vertices. After reaching a threshold energy level (here 91J) compared with the full energy level (100J), a message about node failure has been sent from the cut vertex to the neighbouring node. The neighbouring node then passes this message of node failure to source and destination.

(ii) Failure Intimation to Source and Destination

After obtaining the message from the cut vertex about failure, the neighbouring node informs to source and destination via the shortest possible path.

(iii) Employment of Alternate Path

After the source receives the information about the occurrence of a cut in the cut vertex, it then makes provision for path regeneration by employing an alternate path replacing the failed cut vertex path with a new one.

V. SIMULATION AND EXPERIMENTAL RESULTS

NS2 simulator was used for simulation. Here the average energy, throughput and time delay of the above mentioned two methods are compared. It has been seen that all the parameters could be improved by using the second methodology since it is independent of network size.

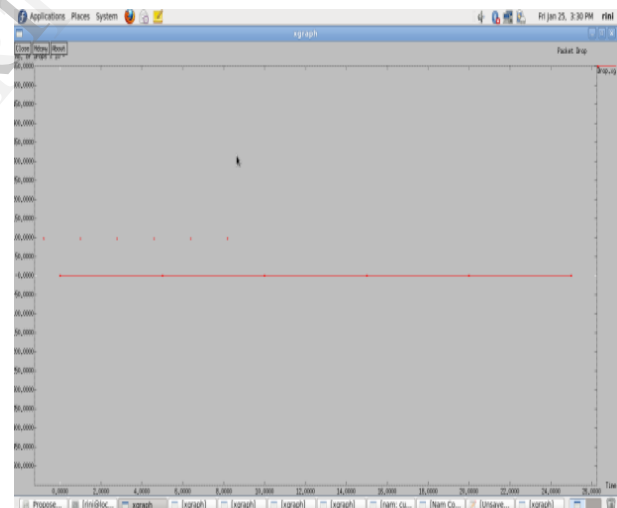


Fig.6 The graph showing the packet drop with time to indicate efficiency

The graph shows that packet drop is negligible since the failure is intimated before network breakdown. This graph is same for both the methods

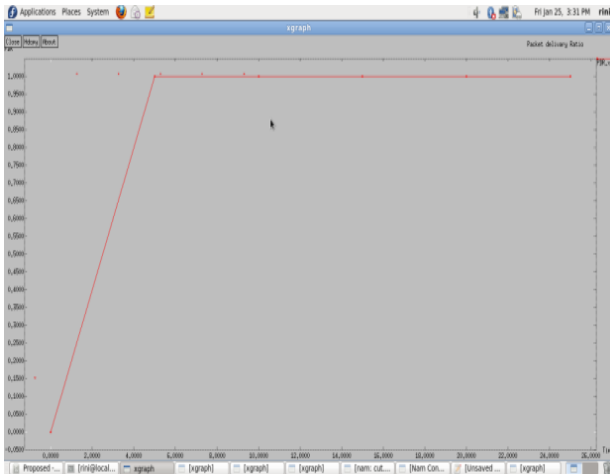


Fig.7 The graph showing the packet delivery ratio of the network with time

The above graph shows that the packet delivery ratio is more, since packet drop is minimum leading to high throughput.

This graph is same for both the methods

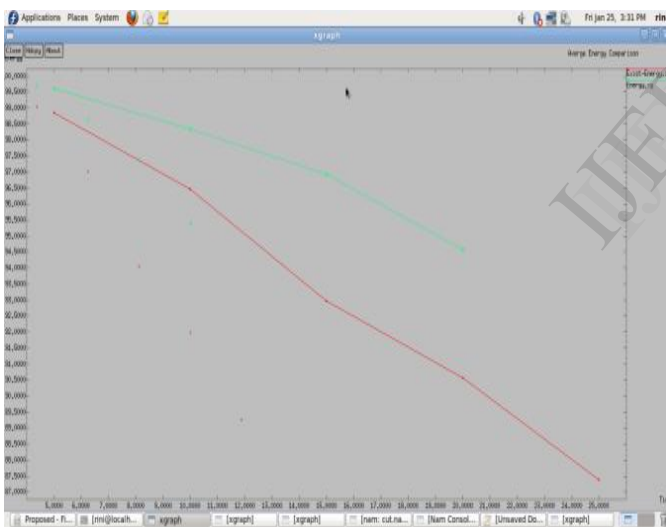


Fig.8 The graph shows the comparison of the average residual energy of the previous method with the proposed method.

The above graph shows that the average residual energy is high for both the methods initially but it decreases more with the previous method than with the proposed method. So the proposed method is highly efficient in terms of energy, throughput. The time required to detect the cut in the cut vertex is also less compared with previous method in which

the iterations or the time delay increases with the network size.

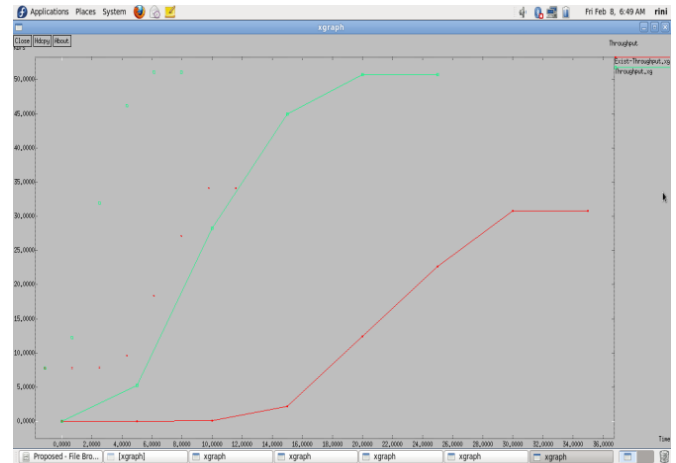


Fig.9 The graph shows the comparison of throughput using the proposed method and the previous method

It has been seen that previous method has less throughput compared with the existing method since residual energy for the previous method is less than that for the proposed method

VI CONCLUSION

Thus the two approaches have been compared based on the simulation results. The two parameters average energy and hence throughput is high in the later method and hence seems to be an efficient method among two. The time required to detect the cut in the second method is also less as compared to the first method. It is notable that the first method based on remove and check is network dependent since the time required to detect the cut vertex will increase if the network size increases but in the second method based on routing table, information from the routing table is used to determine the cut vertex and hence is network independent. The network is based on immobile nodes. In the future, it can be extended for mobile nodes as well

REFERENCES

- [1] Shuguang Xiong and Jianzhong Li, "An Efficient Algorithm for Cut Vertex Detection in Wireless Sensor Networks", in 2010 International Conference on Distributed Computing Systems, pp368-377.
- [2] Prabir Barooah, Harshavardha Chenji, Radu Stoleru, and Tamal Kalmar-Nagy, "Cut Detection in Wireless Sensor Networks", IEEE Transaction on parallel and distributed system.
- [3] X.Liu, L.Xiao, A.Kreling and Y.Liu, "Optimizing Overlay Topology by Reducing Cut Vertices", in ACM Workshop on Network and Operating System support for Digital Audio and Video (NOSSDAV), 2006.
- [4] G.Dini, M.Pelagatti, and I.M.Savino, "An algorithm for reconnecting wireless sensor network partitions", in European Conference on Wireless Sensor Networks, 2008, pp.253-257.
- [5] B. Milic and M. Malek, "Adaptation of the Breadth First Search Algorithm for Cut-edge detection in Wireless Multihop Networks", in ACM
- [6] Ritter.H, Winter.R, and Schiller.J, "A Partition Detection System for Mobile Ad-hoc Networks," Proc. First Ann. IEEE Comm. Soc. Conf.and Ad Hoc Comm. and Networks (IEEE SECON '04), Oct. 2004, pp. 489-497.
- [7] Shrivastava.N, Suri.S, and Toth.C.D, "Detecting Cuts in sensor Networks," ACM Trans. Sensor Networks, vol. 4, no. 2, 2008, pp. 1-25.

IJERT