# Cut Detection and Recovery in Wireless Sensor Networks

Nivedha.M [1]
M.Tech Scholar, Department of Electronics and Communication Engineering, Manakula Vinayagar Institute of Technology, Pondicherry, Pondicherry-605107,India.

Velmurugan.S [2]
M.Tech Scholar, Department of Electronics and Communication Engineering, Manakula Vinayagar Institute of Technology, Pondicherry, Pondicherry-605107,India.

Sukanya.S [3]
M.Tech Scholar, Department of Electronics and Communication Engineering, Manakula Vinayagar Institute of Technology, Pondicherry, Pondicherry-605107,India.

Balaji.S [3]
M.Tech Scholar, Department of Electronics and Communication Engineering, Manakula Vinayagar Institute of Technology, Pondicherry, Pondicherry-605107,India

***Abstract*—**In WSN(wireless sensor network) will get partition into multiple number connected elements due to the failure of a number of its nodes, that is termed has cut. We tend to contemplate the drop back of detective work cut by the remaining nodes of a wireless sensor network. We tend to propose associate degree algorithmic rule that enables. Each node to notice once the property to a specially selected node has been lost. One or a lot of nodes that are connected to the special node when the cut.
The rule of algorithmic is distributed and not synchronized: Each node has to communicate with solely those nodes that are among its communication limit. The algorithmic rule relies on the repetitious computation of a untrue electrical pulse of nodes. The meeting time of the underlying repetitious theme is freelance of the dimensions and structure of the network. We tend to demonstrate the effectiveness of the planned algorithmic rule through simulations and real implementation.

***Keywords***- *Ad-Hoc On-Demand Distance Vector, Disconnected from Source, Mobile Ad-hoc networks, Wireless Sensor Network , CCHS, DCD*

*****

## I. INTRODUCTION

To observe the profit of cut detection capacity, picture a sensor that wishes to drive data in the direction of the source node which have been disjointed on or after the source node. Lacking the information of the systems detached condition, it might just transfer the information on the way to subsequent node of the direction-finding tree, which forces to carry out the same to its subsequent node, as a result. On the other hand, the information sent simply destroys valuable power of the nodes; the cut avoid the information to attain the target.

Alternatively, the capability of the start node to sense the incidence and position of a cut allows it on the way to take on system patch up. As a result, the aptitude to become aware of cuts through both detached nodes and start node resolve to the raise in the equipped life span of the system . The process of restoring a disjointed system with the use of movable nodes had been projected in [1]. Algorithms used for sensing cuts, while the one planned at this point, could be able to hand out as helpful gear used for such system renovating process. As a evaluation of previous effort resting on cut detection in sensor networks, e.g., [2], [3], [4].

Let us think about the difficulty of detecting cuts by means of the nodes in a wireless set of connections. The consideration made is to facilitate a specially chosen node in the system, called as the start node. The start node might exist as a base station so as to hand round as an boundary among the system and their clients(u); the cause of this specific name is that the electrical correlation commenced given that a cut might or might not detach a node from the start node, we make a distinction among two separate results of a cut for a specific node. Once a node u is disjointed from the source, let us say a DOS incident had taken place for u. As soon as a cut happens in the system which will not take apart a node u commencing the start node, let us say that Connected, but a Cut happened someplace (CCHS) incident had taken place for u. By cut detection we mean 1) finding by each
node of a DOS incident while it arises, and 2) finding of CCHS incident with the help of the nodes nearer to a cut, and the exact position of the cut.

## II. RELATED WORK

### CUT DETECTION IN WIRELESS SENSOR NETWORK

Though the impact of network disconnection has been recognized in several papers, such as in [1], [2], [3], the problem of monitoring disconnection has attracted only limited attention. A noteworthy work on cut detection in wireless networks is the paper by Shrivastava *et al.* [4], which proposes an algorithm to be engaged by means of a base station to sense an ϵn linear cut in a system. An ϵn linear cut is a partition of the system directly such that smallest amount ϵn of the nodes (n is the total number of nodes in the network) are partitioned from the base station. The base station notices cuts while they arise depending on the ability to obtain information's from specially placed guard nodes. The algorithm in [4] is guaranteed to report every ϵn linear cut and also provides certain bounds on false positives. Due to its use of line point duality in the plane, the algorithm in [4] is limited to networks that can be modeled as 2D geometric graphs.

In contrast, the projected DCD algorithm is appropriate once the set of connections acquires portioning of several components of random forms, and not restricted to directly lined up cuts. The DCD algorithm facilitates not only the base station (source node) to sense the cuts, however all node to sense suppose it is detached as of the base station. Although the CCHS incident finding branch of the algorithm can be planned for systems positioned on 2D area, the DOS incident discovery division is relevant to systems positioned on random places.

## ALGORITHM FOR RECONNECTING WIRELESS SENSOR NETWORK PARTITIONS

Networked Embedded Systems play an increasingly important role and affect many aspects of our lives. New applications are being developed in areas such as health-care, industrial automation, smart building and rescue operations.

The European Integrated Project "Reconfigurable Ubiquitous Networked Embedded Systems" (RUNES) [1] brought together 21 industrial and academic partners with the aim of enabling the creation of large scale, distributed, heterogeneous networked embedded systems that inter-operate and adapt to their environments.

A set of nodes with wireless communication capabilities are deployed inside the tunnel for monitoring purposes. As soon as an emergency situation occurs, for example an accident involving many cars, the nodes need to transmit data regarding the tunnel conditions to a base station responsible for tunnel control. In such scenario, accurate and comprehensive information must be provided to the base station so that correct counter measures can be taken. It is of fundamental importance that the network would maintain connectivity, so that the flow of critical data to the base station is guaranteed. However, the network could be partitioned because of a malfunction of the nodes, caused by a fire, or because the presence of obstacles that deteriorates or even nullifies metrics of the Quality of Service.

The problem of network partitioning in WSN is not entirely new even though so far has received limited attention. Chong and Kumar raise the problem of partitions with a security focus [7]. So do Wood and Stankovic with respect to denial of service [7]. In [6], Cerpa and Estrin propose methods to self-configuring WSNs topologies. Although they mention the problem of network partitions as an important one, however, they leave such methods to future work. Finally, Shrivastava *et al.* propose a low overhead scheme to detect network partitioning, "cuts" in their parlance, but they do not propose any method to repair them [6].

They presented a method for repairing network partitions based on mobile nodes. The paper has the following merits. First of all, it treats an important problem that, so far, has received limited attention. Furthermore, the paper suggests a method that is based on a few mobile nodes that move through the network reducing the communication overhead. The paper presents the main factors influencing the algorithm behavior and performance and discusses their selection criteria. By simulation, the paper shows that the proposed method is effective in terms of disconnection probability and efficient in terms of communication overhead. Future steps consist in deploying an early prototype on the multi-agent platform we have been developing.

## PARTITION DETECTION IN MOBILE AD-HOC NETWORKS

A classical difficulty caused through nodes progress in an adhoc set of connections gets separated. Calculating those separation
would exist as a extremely valuable characteristic which can be offered for the applications inside a mobile ad-hoc set of connections surroundings. Certainly, being conscious of a upcoming detachment in the set of connections know how to guarantee a enhanced excellence of service by means of adjusting the application activities. An algorithm previously survives to perform the above. Although they require location message to be offered via a arrangement method. The paper suggest an unique connection strength assessment process stands in the view of displaced paths that permit well-ordered separation discovery lacking of some type arrangement system.

In such purposes, mobile computers, or nodes, resolve to converse through direction-finding information by the system by multi-hopping set of rules [7, 8, 5]. Such set of connections is termed as MANET for Mobile Ad-hoc networks [1].

A likely difficulty inside these type of set of connections is separating. Here the link splits are true breakdown in a wired surroundings, in ad-hoc networks, the assignment could not be like the same for the reason that it may be able to happen following a ordinary set of connections activities (as a node have been in motion , a client put his system off...). This particular action of the node movement is given in (Figure 1), which could exist helpful for calculating separation and reports the variety of uses in the absence of a location system that is a lot pricey and large [4].

This manuscript, suggest several thoughts depending on dislodged trail place an make an effort to illustrate through simulation with the intention of use of this message to encompass proficient separation discovery.

We suggest two parameters intended for linkage costing in a mobile ad-hoc network surroundings. By means of the costing of our first parameter, one know how to calculate a objective detachment among the nodes if this linkage is due to a set of connections topology alteration. The testing demonstrates that this type of calculation is likely lacking by utilizing costly equipments similar to location spotting methods if we have a scattered algorithm lending a set of dislodged trail among two nodes. Thus, it is currently applicable to work on this type of algorithm. Furthermore, a set of dislodged trail could be used on QoS multi trail direction-finding procedures.

## III. WIRELESS SENSOR NETWORK

The WSN is collection of "nodes" as of a small number to a number of hundreds or even thousands, in which every node is linked to single (or at times numerous) sensor. All such sensor network node have characteristically numerous components: a radio transmitter and receiver having an interior aerial or linkage in the direction of an exterior aerial, a microcontroller, an electronic system for combining through the sensors and an power resource, generally a battery or an embedded type of power harvesting. A sensor node may differ in dimension as of that of a shoebox losing to the dimension of a particle of dirt, while operation "motes" of real tiny size have to be produced. The price of sensor nodes is likewise uneven, varying from a little to hundreds of dollars, based on the complication of the each sensor nodes. Dimension and price limitations of sensor nodes ends in related limitations of resources such as power, memory, calculation speed and communication band of spectrum. The configuration of the WSNs varies as of a easier star set of connections to an highly developed multi-hop mesh set of connections. The transmission method among the hops of the system can be direction-finding or flooding.

Wireless sensor networks consist of autonomous sensor nodes placed in an operational place jointly in order to supervise diverse ecological and physical state of affairs such as motion, temperature, pressure, vibration sound or pollutants. The major cause for the development of wireless sensor network is military applications in battleground earlier but at the present the application region is unlimited to further domains such as industrial monitoring, controlling of traffic and health monitoring [6].
Important limitations such as dimension and price fallout in limitations of power, bandwidth, memory and calculation speed of sensor nodes.

A wireless sensor node in a network consists of the following components:

- Microcontroller.
- Radio transceiver.
- Power source (battery).

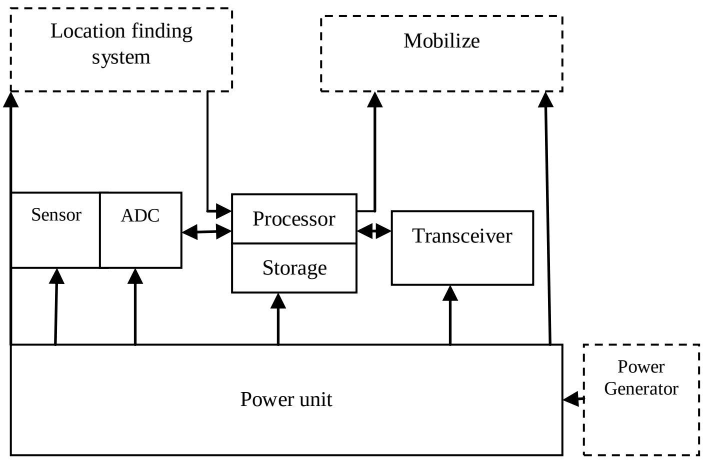


*Fig. 1 Components of wireless sensor network (WSN)*

## IV. AODV ROUTING PROTOCOL

AODV stands for Ad-Hoc On-Demand Distance Vector [4] and is, as the name already says, a reactive protocol, Although it utilizes features of a proactive protocol. AODV seizes the remarkable elements of DSR and DSDV, in the logic that it utilizes the conception of 8 path detection and path preservation of DSR and the notion of series numbers and forwarding of cyclic hello messages from DSDV.

Presently there are two category of direction-finding set of rules which are reactive and proactive. Inside reactive direction-finding set of rules the path is produced merely while the start node wishes to transfer information to end node while proactive direction-finding set of rules is table determined. Staying as s reactive direction-finding set of rules

AODV employs customary direction-finding tables, single access for each end node and series numbers are helpful to decide whether direction-finding information is new in order to avoid direction-finding loops. Let preservation of time-dependent status be an essential characteristic of AODV that refers to a direction-finding entrance that is not just utilized gets terminated. The neighbors are informed if any path split occurs. Detection of path from start node to end node depends on question and answer rotation and in-between nodes stock up the path information in the type of path table admission beside the path.

Inside AODV, the system is quiet unless a link is required. By this note the system node which requires a link to transmit a demand for a link. Remaining AODV nodes transmit this message, and trace the node which they listened to , generating an blast of provisional paths reverse to the poor node. After a node obtains such a information and previously have a path to the preferred node, it forwards information reversely by a provisional path to the demanding node. In the case of poor node begins with the help of the trail which have the less count of jump to further nodes. Idle entrance in the direction-finding tables are reused subsequent to a time.

As soon as a connection breaks, a direction-finding error is conceded back to a forwarding node, and the procedure reiterates. Most of complication of the procedure is to reduce the count of information to preserve the aptitude of the system. For illustration, every demand for a path have a series number. Nodes utilizes this series number as a result they do not replicate path demands which they had previously conceded . One more such characteristic is that the path demands have a "time to be alive" number that restricts the number of retransmission. An additional such trait is that suppose a path demand gets failure, one more path demand might not be forwarded in anticipation of two times as much time have conceded as the timeout of the preceding path demand.

The benefit of AODV is it produces no additional traffic intended for contact beside the present connections. Furthermore, distance vector routing is easy, and requires no huge memory or computation. Though AODV needs much time to set up a link, and the preliminary contact to set up a path is weighted more than various loom.

## DISTRIBUTED CUT DETECTION (DCD) ALGORITHM

In this manuscript, we suggest a scattered algorithm to sense cuts, called the Distributed Cut Detection (DCD) algorithm. The algorithm permit every node to sense DOS actions and a group of nodes to sense CCOS actions. The algorithm suggested is scattered and uneven: it engages just confined contact among adjacent nodes, and is healthy to provisional contact breakdown among node pairs. An important part of the DCD algorithm is a distributed repetitive calculative pace during which the nodes calculate

ISSN: 2321-8169
3331 – 3336

their (untrue) electrical pulses. The junction speed of the calculation is free from the dimension and formation of the system.

## DOS DETECTION

The technique at this time is to develop the detail that if the status is nearer to zero in that case the node gets detached from the start node, or else not (this is completed particularly in Theorem 1 of Section 2.4). To decrease sensitivity of the algorithm to difference in system dimension and formation, we utilize a standardized condition. DOS discovery branch contains of stable-status discovery, regularized condition calculation, and linkage/partition discovery. Each node i retain a dual variable $DOS_i(k)$, which is located to 1 if the node considers it is disjointed from the source and 0 if not. This variable is known the DOS condition, is started to 1 as there is no cause to trust a node is linked to the start node at first.

A node maintains path of the optimistic stable condition viewed in the earlier period by means of the subsequent technique. Each node i calculate the standardized condition diversity $\delta x_i(k)$ as follows:

$$\delta x_i(k)=\begin{cases}\frac{xi(k)-xi(k-1)}{xi(k-1)}, & \text{if } xi(k-1) > \epsilon zero\\ \infty, & \text{otherwise}\end{cases}$$

where $\epsilon_{zero}$ is a undersized positive number. A node i carries a Boolean variable Positive Steady State Reached (PSSR) and updates PSSR(k) 1 if $|\delta x_i(k)|<\epsilon\Delta x$ for $k=k-T_{guard},k-T_{guard}+1,\ldots.k$ (i.e., for $T_{guard}$ consecutive iterations), where $\epsilon\Delta x$ is a small positive number and $T_{guard}$ is a small integer. The first 0 value of the condition is not assigned to a stable situation, so PSSR(k)=0 for k=0,1,…, $T_{guard}$.

Every node keep an estimate of the most recent "stable status" observed. This approximation is reorganized at each time k according to the subsequent law if PSSR(k)=1, then, or else $xi^{\wedge 88}(k) \leftarrow xi^{\wedge 88}(k-1)$. It is started as, each node i also holds a roll of stable condition observed in the precedent, one assessment intended for every unpunctuated period of time through which the condition was sensed to be stable. This information is placed in a vector, which is stated to be unfilled and is rationalized as follows: If PSSR(k)=1 but PSSR(k-1)=0, the $x^{\wedge}{}_{88}(k)$ is add on to as a fresh entrance. If stable condition arrived was sensed in both k and k-1 (i.e., PSSR(k)=PSSR(k-1)=1), next the final admission of it is rationalized to. For example, the node ν in the system and, $X_v{}^{\wedge 88}(3)=\Phi$(empty), $X_v{}^{\wedge 88}(60)=[0.0019]$ and $X_v{}^{\wedge 88}(150)=[0.019,0.012]$. For upcoming exercise, we moreover describe an unstable period for a node i, which is a group of two confined time counters $[k_i^{(1)},k_i^{(2)}]$ to facilitate the condition $x_i(k_i^{(1)}-1)$ is a stable-condition (i.e., $PSSR(k_i^{(1)}-1)=1$) but $x_i(k_i^{(1)})$ is not, and $x_i(k_i^{(2)})$ is not stable except $x_i(k_i^{(2)}+1)$ is. The final unstable period for node ν on time 150.

Every node calculates a standardized condition $x_i^{norm}(k)$ as

$$x_i^{norm}(k)=\begin{cases}\frac{xi(k)}{xi^{\wedge 88}(k)}, & \text{if } xi^{\wedge 88}(k) > 0\\ \infty, & \text{otherwise}\end{cases}$$

where, the final stable condition viewed through i at k, i.e., the final admission of the vector. If the standardized condition of i is a smaller than $\epsilon_{DOS}$, where $\epsilon_{DOS}$ is a undersized positive number, after that the node confirms a cut had occurred: $DOS_i$. Suppose the standardized condition is ∞, denotes no stable condition was viewed in anticipation of k, then $DOS_i(k)$ is put to 0 if the condition is positive (i.e.,$x_i(k) > \epsilon_{DOS}$) and 1 or else.

## CCOS DETECTION

The algorithm meant for sensing CCHS procedures falls on sensing a undersized route about a hole, if this survives, and is incompletely stimulated by means of the congestion discovery algorithm projected in [6]. The technique make use of node condition to allocate the job of hole-discovery to the majority suitable nodes. As soon as a node sense a huge alter in its confined condition and also breakdown of one or many of its likelihood, and together these actions take place in a (predestined) tiny time period, the node starts a PROBE message. The pseudo code for the algorithm so as to make a decision at what time to begin a probe is integrated .

Every PROBE message consists of the subsequent information:

- a unique probe ID,
- probe centric Cp
- end node,
- route passed through (in sequential array), and
- the angle pass through the probe in the region of the centroid.

The probe is transferred in a way with the purpose of the probe is activated as a result of the formation of a tiny hole or cut (through boundary less than 'max), the probe passed through a route in the region of the hole in a counter-clockwise (CCW) way and arrive at the node that started the probe. During this event, the overall angle passed through the probe is 360 degree. Further, if the probe be started as a result of the incidence of a border cut, still but the probe finally attains its node of beginning, the total angle passed through the probe is 0. Nodes sends a probe merely if the space passed through the probe (the count of jump) is lesser than a threshold assessment 'max. Consequently, if a probe is started because of a huge interior cut/hole, after that it would exist engrossed through a node (i.e., not sent since it goes beyond the space threshold limitation), and the engrossing node confirms to a CCHS happening had taken place. Particularly as soon as the start node is warned regarding the incidence of a cut in the system.

The message necessary to calculate and renew these probe variables demands the subsequent consideration intended for CCOS detection:

The position message required by the nodes may not exist accurate, as it is merely employed to calculate end node of probe messages. The consideration of the system being 2D is desirable to be proficient to describe CW or CCW way unmistakably, that is employed in sending

 ISSN: 2321-8169
3331 – 3336

probes. At the start of replication, each node begins with a file of probes to progress. The file of probes is the combination of the probes it obtained from its neighbors and the probe it determined to start, if some. The behavior in which the message in all of the probes in its file is rationalized by a node .

## V. PROPOSED SYSTEM

Here in the proposed system we suggest algorithms for identifying the cut which is recognized as Distributed Cut Detection (DCD) algorithm and it offers electrical likeness. Proposed algorithm permits every node to discover DOS happenings and a group of nodes to identify CCHS happenings. A probe message ability is established in this DCT algorithm. As a result the proposed System could converse with the destination not including any drop.

The proposed system focuses on randomized re-routing where the trail node themselves transfers the route to end node.

Randomized re-routing works as follows:

- Identify the cut through the route node itself
- The route node prior to the cut transfers RREQ message to its neighboring nodes.
- Transfer a RERR message to the unique source.
- Select the next small route to the end node by-traversing the cut.

Scatter the energy of the special node to former nodes or maintain the energy as a endorsement

## VI. SIMULATION OUTPUT

The simulation result illustrates the working of cut detection mechanism by the comparison of various performance parameters.

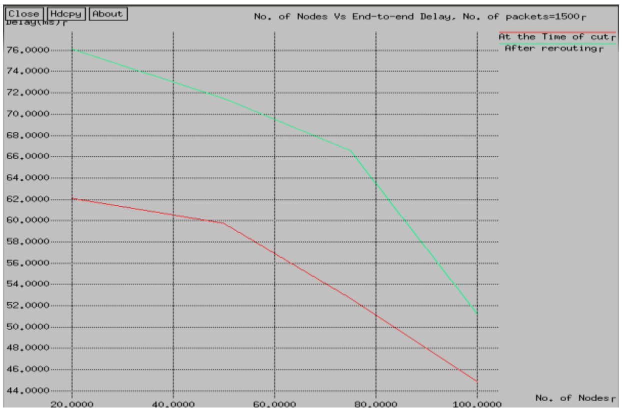


*Fig. 2 NO.OF NODES VS END-TO-DELAY*

The above graph illustrates the comparison of No.of nodes vs end-to-end delay on the time of cut and after performing randomized re-routing.

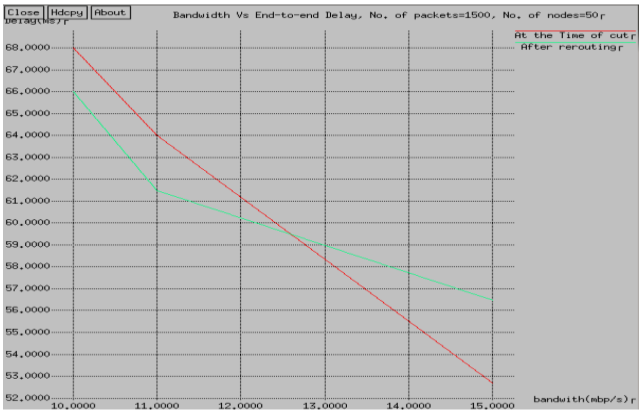


***Fig. 3 BANDWIDTH VS END-TO-END DELAY***

The above graph illustrates the comparison of bandwidth vs end-to-end delay on the time of cut and after performing randomized re-routing.

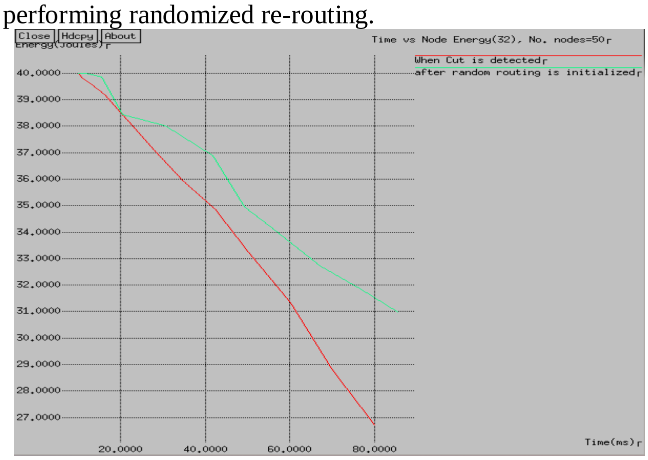


***Fig. 4 TIME VS NODE ENERGY***

The above graph illustrates the comparison of time vs node energy on the time of cut and after performing randomized re-routing.

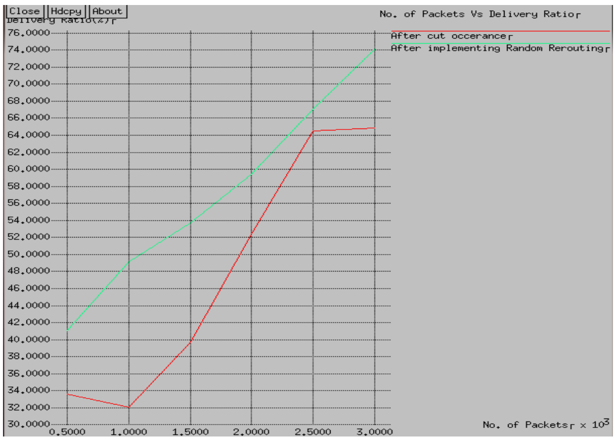


***Fig. 5 NO OF PACKETS VS DELIVERY RATIO***

The above graph illustrates the comparison of No.of packets vs packet delivery ratio on the time of cut and after performing randomized re-routing.

## VII. CONCLUSION

The DCD algorithm we suggest allows each node of a wireless sensor network to identify Disconnected from Source actions if they take place. Next, it allows a group of nodes to practice CCHS events and to sense them and compute the exact position of the cut in the type of a directory of lively nodes that falls at the border of the cut/hole. The DOS and CCHS actions are described by means of a specifically nominated start node. The algorithm depends on thoughts commencing electrical set of connections and corresponding repetitive key of linear equations. Numerical simulations and experimental assessment on a actual WSN network consist of mica Z notes, demonstrate to facilitate the algorithm workings efficiently by huge module of graphs of various dimension and configuration, not necessitates alters in specification. For some circumstances, the algorithm is guaranteed to identify link and detachment to the start node devoid of error. The power of the DCD algorithm is that the junction speed of the fundamental repetitive method is fairly quick and free of the dimension and configuration of the system, which build discovery by means of this algorithm somewhat speedy. Application of the DCD algorithm to identify node partition and reconnection to the source in mobile networks is a theme of ongoing study.

### REFERENCES

[1] Prabir Barooah, Harshavardhan Chenji, Radu Stoleru, Tama´s Kalma´r-Nagy "Cut Detection in Wireless Sensor Networks" *IEEE Transactions on Parallel And Distributed Systems*, Vol. 23, No. 3, MARCH 2012.

[2] N. Shrivastava, S. Suri, and C.D. To´ the, "Detecting Cuts in Sensor Networks," *ACM Trans. Sensor Networks*, vol. 4, no. 2, pp. 1-25, 2008.

[3] G. Dini, M. Pelagatti, and I.M. Savino, "An Algorithm for Reconnecting Wireless Sensor Network Partitions," *Proc. European Conf. Wireless Sensor Networks*, pp. 253-267, 2008.

[4] M. Hauspie, J. Carle, and D. Simplot, "Partition Detection in Mobile Ad-Hoc Networks," *Proc. Second Mediterranean Workshop Ad-Hoc Networks,* pp. 25-27, 2003.

[5] P. Barooah, "Distributed Cut Detection in Sensor Networks," *Proc. 47th IEEE Conf. Decision and Control*, pp. 1097-1102, Dec. 2008.

[6] H. Ritter, R. Winter, and J. Schiller, "A Partition Detection System for Mobile Ad-hoc Networks," *Proc. First Ann. IEEE Comm. Soc. Conf. Sensor and Ad Hoc Comm. and Networks (IEEE SECON '04),* pp. 489-497, Oct. 2004.

[7] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister, "System Architecture Directions for Networked Sensors," Proc. Int'l Conf. Architectural Support for Programming Languagesand Operating Systems(ASPLOS),2000.

[8] A.D. Wood, J.A. Stankovic, and S.H. Son, "Jam: A Jammed-Area Mapping Service for Sensor Networks," *Proc. IEEE Real Time Systems Symp.*, 2003.