www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 6 Issue 5 May 2017, Page No. 21212-21221
Index Copernicus value (2015): 58.10 DOI: 10.18535/ijecs/v6i5.05

Cut Detection in Wireless Sensor Networks

M Manasa Ravali, Vemula K Harika, M Chaithanya Kishore Reddy
Department Of Computer Science And Engineering

Abstract

A wireless sensor network (WSN's) can be separated into multiple connected components due to the failure of some of its nodes, which is called a "cut." In this paper, here the considered problem of detecting cuts by the remaining nodes of a wireless sensor network. By the proposal an algorithm that allows

1) Each and every node to detect when the connectivity to a specially designated node has been lost.

2) One or more nodes (that are connected to the special node after the cut) to detect the occurrence of the cut.

The algorithm is distributed and asynchronous: Every node needs to communicate with only nodes which are within its communication range. The algorithm is based on the iterative computation of a fictitious "**Electrical Potential**" of the nodes. The convergence rate of the underlying iterative scheme is independent of the size and structure of the network. We demonstrate the effectiveness of the proposed algorithm through simulations and a real hardware implementation.

Keywords-Wireless networks; cut; Sensors; energy efficiency; Nodes.

I. Introduction

Wireless sensor networks (WSNs) are the technology used for bugs monitoring in the real world applications as large regions at high spatial and temporal resolution. In fact, disconnection and node failure is expected to be quite common due to the usage and typically limited energy budget of the nodes that are powered by small batteries. Failure of a set of nodes will reduce the number of multi-hop paths in the network which causes many of the intrusion to attack. 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.

Here considered the problem, as detecting cuts by the nodes of a wireless network. Some one can assume that there is a specially designated node in the network, which we call the source node, it may be a base station that serves as an interface between the network and its users. as 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. When a node M in usage is disconnected from its source, there occurred a DOS (Disconnected from Source) event. Apart from DOS the node M gets a bug, we defines it self as CCOS (Connected, but a Cut Occurred Somewhere) event. By the detection method algorithm we used it processes the below two methodes (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. By "approximate location" of a cut we mean the location of one or more active nodes that lie at the boundary of the cut and that are connected to the source. Nodes that detect the occurrence and approximate locations of the cuts can then alert the

robust to temporary communication through which the nodes compute their (fictitious) electrical potentials. The convergence rate of the computation is.

Problem Defination

When source has to send a data to the node, by which connection is not supporting . Without the knowledge of the network's disconnected state, the rest node transfers it simply to the next node in the routing tree, which is and endless process unless and until the data gets to the destination. However, this message passing wastes precious energy of the nodes, and takes time. Due to the cut the data gets lost, reaching the destination. Therefore, if a node is does't able to detect the occurrence of a cut, it could simply wait for the network to be repaired and eventually reconnected, which saves energy of multiple nodes and prolongs their lives. On the other hand, the ability of the source node to detect the occurrence and location of a cut will allo to undertake repairs. Thus, the ability to detect cuts by both the disconnected nodes and the source node will lead to the increase in the operational lifetime of the network as a whole.

Ii. Existing System

Wireless Multimedia Sensor Networks (WMSNs) has many challenges such as nature of wireless media and multimedia information transfer. Consequently traditional mechanisms of network layers are no longer acceptable, applicable or used for these networks. Wireless sensor network can get separated into multiple connected components due to the failure of some of its nodes, which is called a "cut".

1. E-linear cut detection: Cut detection in wireless networks have proposed, an algorithm that is worked by the help of a base station to detect an e-linear cut in a network. An e-linear cut is a separation of the network across a straight line so that at least end of the nodes (n is the total number of nodes in the network) are separated from the base station. The base station detects cuts when they occur, based on, when it is able to receive messages from specially placed source nodes.

3. Critical node: A critical node is the longest path node, on whose removal, renders the network disconnected.

4. Single path Routing approach:it is used when the source node needs apath to send the information.it take a single path only.

5. Unsuitable for dynamic network reconfiguration: while the process reconfiguration this network is not used as it is not suitable for creating network of increasing or decreasing no. of sensor nodes.

Disadvantages

- Nonsuitable for dynamic network reconfiguration.
- Single path routing usage.
- Algorithm proposed only for detecting linear cuts in the network.
- In flooding based technique, routes from the nodes to the base station and vice versa have to be recomputed when node failure occurs.

III. Proposed System

Wireless sensor networks (WSNs) are a promising technology for monitoring large regions at high spatial and temporal resolution. Failure 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 detecting cuts by the nodes of a wireless network. We assume that there is a specially designated node in the network, which we call the source node.

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. 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.

1. DCD algorithm is applicable even when the network gets separated into multiple components of arbitrary shapes, and not limited to straight line cuts.

2. DCD algorithm enables not just a base station to detect cuts, but also every node to detect if it is disconnected from the base station.

3. CCOS event detection part of the algorithm is designed for networks deployed in 2D regions, the DOS event detection part is applicable to networks deployed in arbitrary spaces.

ADVANTAGES

- Comes with provable characterization on the DOS detection accuracy.
- CCOS events detection can be identified.
- DCD algorithm enables base station and also every node to detect if it is connected from the base station.

IV.Distributed Cut Detection (DCD) :

DCD is the algorithm proposed to identify and detect the cuts in networks.

The algorithm allows each node to detect DOS events and a subset of nodes to detect CCOS events. The algorithm we propose is distributed and asynchronous: it involves only local

DOI: 10.18535/ijecs/v6i5.05

communication between neighboring nodes, and is robust to temporary communication failure between node pairs. A key component of the DCD algorithm is a distributed iterative computational step through which the nodes compute their (fictitious) electrical potentials. The convergence rate of the computation is independent of the size and structure of the network.

Cut:

A CUT is a disturbance created in connected nodes at a wireless sensor network. In fact, node failure is common due to the typically limited energy budget of the nodes that are powered by small batteries while the process. Failure 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 be disconnected from the rest, resulting in a “cut”. Two nodes are said to be disconnected if there is no path between them.

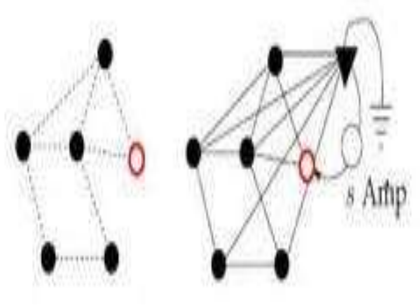

Figure: 1 nodes connected to electrical potentials

SOURCE NODE:

Here, as considered the problem of detecting cuts by the nodes of a wireless network. Someone can assume, there is a specially designated node in the network, which is the source node. The source node may be a base station that serves as an interface and a path 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.

CCOS And DOS:

When a node M gets disconnected from the source, ie when the electrical potentials shows 0amp at the disconnected node, we say a DOS (Disconnected from Source) event have occurred for M. When a cut occurs in the network that does not separate the node M from the source node, we can assume that CCOS (Connected, but a Cut Occurred Somewhere) event have occurred for M.

Network Seperation:

Failure of a set of nodes will reduce the number of multi-hop paths in the networking system. Such failures can cause a subset of nodes – that have not failed – to get disconnected with the rest, resulting in a “cut”. Because of the cut, some nodes may get separated from the network, that results the separated nodes cannot receive the data from the source node.

System Architecture

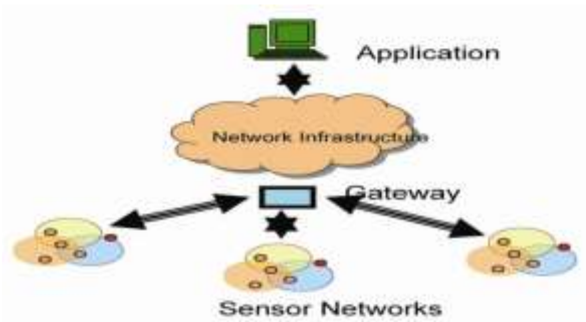

Figure: 2 General Architecture

DOI: 10.18535/ijecs/v6i5.05

V. Algorithm

1: if vi = vinit then

2: Cut id ← id of initiator.

3: δ: a rectangle with width δ.

4: if ∀p ∈ F2 ∪ {pi}, p is in δ then

5: F2 ← F2 ∪ {pi}.

6: forward.

7: else

8: F2 ← ∅.

9: F1 ← F1 ∪ {the last element in F2}.

10: forward.

11: end if

12: else

13: P ← F1.

14: broadcast P.

15: end if

Data Flow Diagram

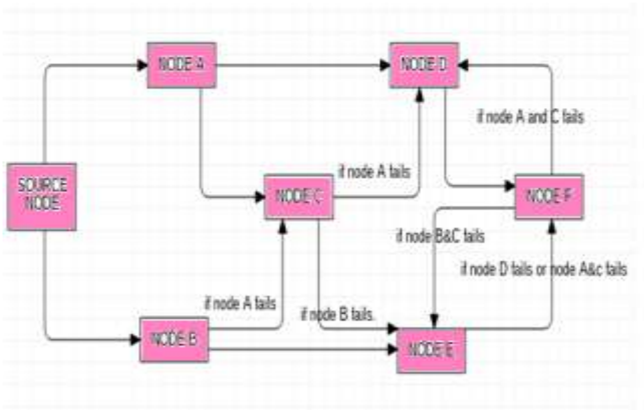

Figure: 3 Data Flow Diagram of Cut Detection.

MODULE: 1

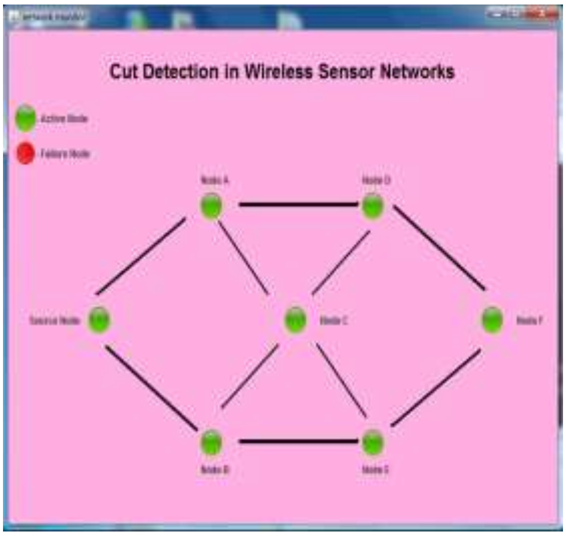

Figure: 4 Connection of different nodes with the source.

Figure: 5 Main Source Node.

Modules Of Implementation

DOI: 10.18535/ijecs/v6i5.05

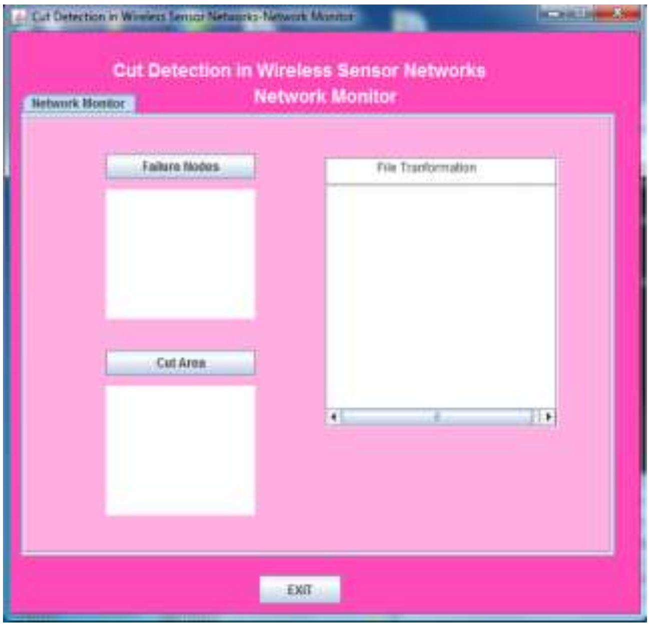

Figure: 6 Network Monitor

Figure: 8 Node B Template.

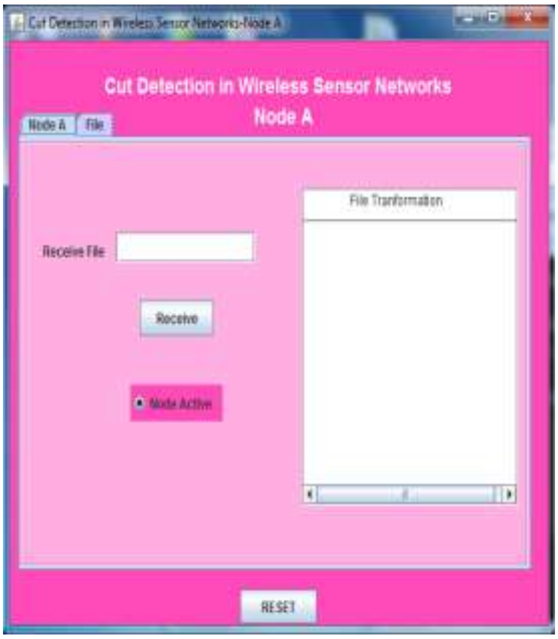

Figure: 7 Node A Template.

Figure:9 Node C Template.

DOI: 10.18535/ijecs/v6i5.05

Figure: 10 Node D Template.

Figure:12 Node F Template.

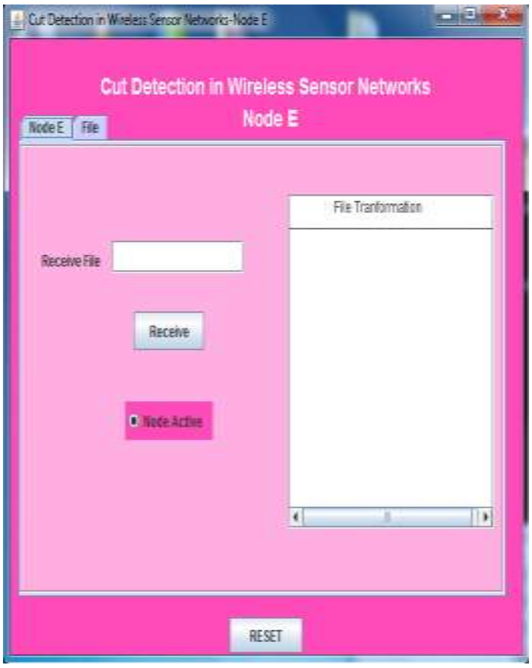

Figure:11 Node E Template.

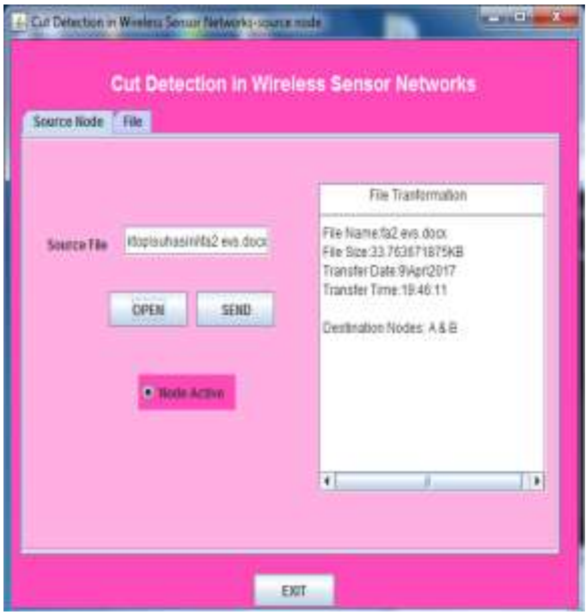

Figure:13 Uploading a file in the Source Node.

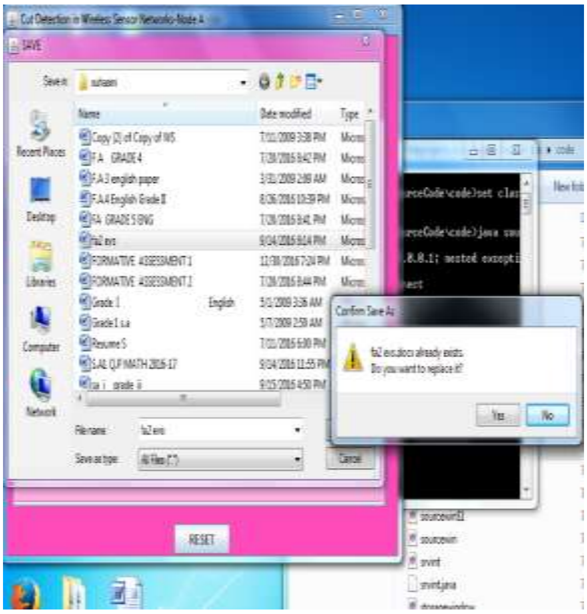

Figure:14 File doesn't able to transfer because of the CUT occurrence.

MODULE: 2

Figure:15 Destination Router Template

Figure: 16 Source Router Template.

Figure: 17 Receiver Template

DOI: 10.18535/ijecs/v6i5.05

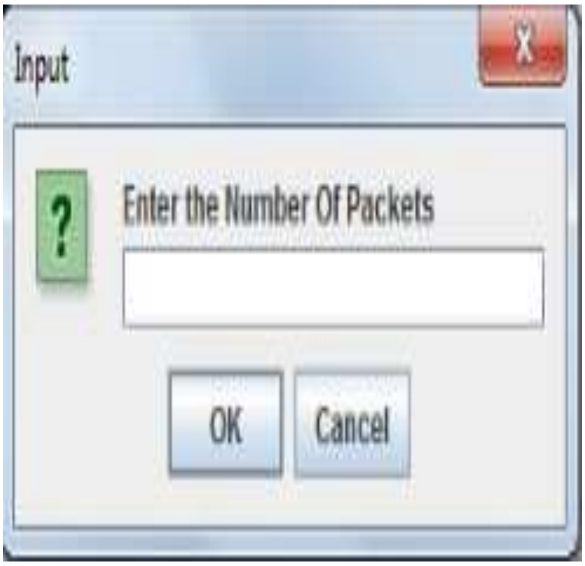

Figure: 18 Sender Template.

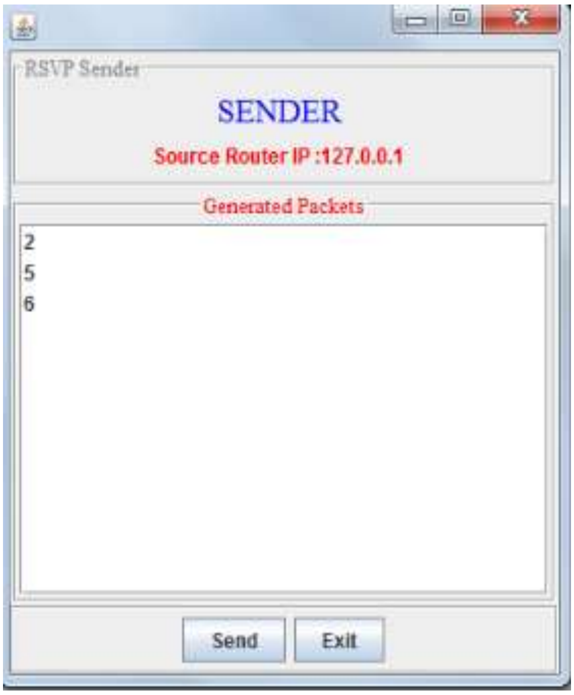

Figure: 19 Generated Packets in the Sender Template.

Figure: 20 Packets received by the receiver and stored in the destination and source routers.

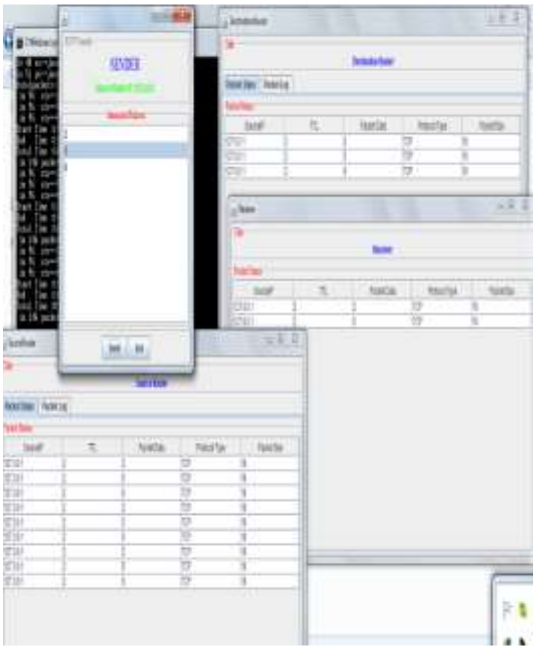

Figure:21 Packets status is loading in the source router.

IX. System Testing

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product.

It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product

Types Of Tests

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Conclusion

The DCD algorithm proposed here enables each and every node of a wireless sensor network to detect DOS (Disconnected from Source) events if they occur and it enables a subset of nodes that experience CCOS (Connected, but Cut Occurred Somewhere) events to detect them and estimate the approximate location of the cut in the form of a list of active nodes that lie at the boundary of the cut/hole. For certain scenarios, the algorithm is assured to detect connection and disconnection to the source node without error. A key strength of the DCD algorithm is that the convergence rate of the underlying iterative scheme is quite fast and independent of the size and structure of the network, which makes detection using this algorithm quite fast. Application of the DCD algorithm to detect node separation and reconnection to the source in mobile networks is a topic of ongoing research.

Reference

References Made From:

[1] 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. 253–267.

[2] N. Shrivastava, S. Suri, and C. D. T´oth, “Detecting cuts in sensor networks,” *ACM Trans. Sen. Netw.*, vol. 4, no. 2, pp. 1–25, 2008.

[3] H. Ritter, R. Winter, and J. Schiller, “A partition detection system for mobile ad-hoc networks,” in *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (IEEE SECON 2004)*, Oct. 2004, pp. 489–497.

[4] M. Hauspie, J. Carle, and D. Simplot, “Partition detection in mobile ad-hoc networks,” in *2nd Mediterranean Workshop on Ad-Hoc Networks*, 2003, pp. 25–27.

[5] P. Barooah, “Distributed cut detection in sensor networks,” in *47th IEEE Conference on Decision and Control*, December 2008, pp. 1097–1102.

Sites Referred:

http://java.sun.com

http://www.sourcefordgde.com

http://www.networkcomputing.com/

http://www.roseindia.com/