Volume 4, Issue 9, September 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Distributed Cut Detection Using Back Tracking Algorithm and Prevention of Cut Using Cryptography Puzzle Hiding Schema in WSN

Prof. Prashant P. Rewagad
Head, Department of CSE ,G.H.R.I.E.M ,
Jalgaon, N.M.U. Jalgaon, Maharashtra, India

Mr.Harish Prakash Patil
Department of CSE ,G.H.R.I.E.M ,
Jalgaon, N.M.U. Jalgaon, Maharashtra, India

Abstract— Wireless Multimedia Sensor Networks (WMSNs) has many challenges such as nature of wireless media and multimedia information transmission. Consequently traditional mechanisms for network layers are no longer acceptable or applicable 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". Existing cut detection system deployed only for wired networks. A wireless sensor network can get separated into multiple connected components. Consider the problem of detecting cuts from the remaining nodes of a wireless sensor network. Propose an algorithm that allows (i) every node to detect when the connectivity to a specially designated node has been lost, and (ii) 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 those nodes that are within its communication range. The algorithm is based on the iterative computation of a fictitious "electrical potential" of the nodes. The algorithm for detect cut using back tracking method and prevent the cut using some cryptography puzzle hiding schema. The convergence rate of the underlying iterative scheme is independent of the size and structure of the network.

Index Terms—Epidemic Routing, Any path Routing Opportunistic networks, Probabilistic Routing.

I. INTRODUCTION

Wireless sensor network is composed of a powerful base station and a set of low-end sensor nodes. Base station and sensor nodes have wireless capabilities and communicate through a wireless, multi-hop, ad-hoc network.[3]Wireless sensor networks (WSN) have emerged as an important new technology for instrumenting and observing the physical world. WIRELESS sensor networks (WSNs) are a capable scenario for sensing large areas at high spatial and positive resolution. However, the tiny size and low cost of the processing machines that makes them attractive for large deployment also causes the loss of low operational reliability[1].Wireless sensor networks (WSN) have emerged as an important new technology for instrumenting and observing the physical world. The basic building block of these networks is a tiny microprocessor integrated with one or more MEMS (micro-electromechanical system) sensors, actuators, and a wireless transceiver.[2] A WSN is usually collection of hundreds or thousands of sensor nodes. These sensor nodes are often densely deployed in a sensor field and have the ability to gather data and route data back to a base station (BS). A sensor has four basic parts: a sensing unit, a processing unit, a transceiver unit, and a power unit [5]. Most of the sensor network routing techniques and sensing tasks require knowledge of location, which is provided by a location finding system. Wireless sensor network contains large number of nodes and each node may be very close to each neighbor. Since WSN should use multihop techniques because it consume less power than single hop techniques.

Multihop techniques can also effectively overcome some of the signal propagation outcomes experienced in long-distance wireless communication [6]. WSN may also have additional application dependent components such as a location finding, system, power generator, and mobilizer (Fig. 1). Sensing units are usually composed of two sub units: sensors and analog-to-digital converters (ADCs). The ADCs convert the analog signals produced by the sensors to digital signals based on the observed phenomenon. The processing unit, which is generally related with a small storage unit, controls the procedures that make the sensor node collaborate with the other nodes. A transceiver unit connects the node to the network. One of the most important units is the power unit. A power unit may be finite (e.g., a single battery) or may be supported by power scavenging devices (e.g., solar cells).

CHARACTERSTICS OF WSN

The important characteristics of a WSN include

- Limited Power consumption for nodes using batteries or energy harvesting
- Communication failures
- Mobility of nodes

- Dynamic network topology
- Ability to run with node failures
- Heterogeneity of nodes
- Scalability to large scale of exploitation
- Easy to use
- Unattended operation
- capacity to survive hard environmental conditions
- Power consumption

As WSNs are lots of similar to traditional wireless ad hoc networks, important differences exist which greatly influence how security is achieved [4]. In [8], I. F. Akyildiz et al proposed the differences between sensor networks and ad hoc networks are:

1. The number of sensor nodes in a sensor network can be several orders of magnitude higher than the nodes in an ad hoc network.
2. Sensor nodes are densely deployed.
3. Sensor nodes are lying face down to failures due to harsh environments and energy constraints.
4. The topology of a sensor network changes very frequently due to failures or mobility.
5. Sensor nodes are limited in computation, memory, and power resources.
6. Sensor nodes may not have global identification.

II. PROBLEM STATEMENT

Consider a sensor network modeled as a undirected graph G=(V,E), whose node set V represents the sensor nodes and the edge set E consists of pair of nodes (u, v) such that nodes u and v can exchange messages between each other. Note that we assume inter-node communication is symmetric. An edge (u, v) is said to be incident on both the u and v. The nodes that share an edge with a particular node u are called the neighbors of u. A cut is the failure of a set of nodes Vcut from G results in G being divided into multiple connected components. Recall that an undirected graph is said to be connected if there is a way to go from every node to every other node by traversing the edges, and that a component Gc of a graph G is a maximal connected sub graph of G. We are interested in devising a way to detect if a subset of the nodes has been disconnected from a distinguished node, which we call the source node, due to the occurrence of a cut.

PROBLEMS DUE TO CUTS

As mentioned above if any node breaks down then the network is separated into different parts so the topology of the network changes but still network works. But because partition affects reliability, data loss, QOS of the network, efficiency, data processing speed. Because if any data passes unfortunately in a wrong route so data loss occurs this also shows unreliability of the network.

CUT DETECTION IN WSN

We consider the problem of detecting cuts between nodes of a wireless sensor network. We assume that there is a special designated node in the network, which we call the *source station or node*. Suppose source station may be a base station that serves as a mediator 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. When a node u got disconnected from the source, we can say that a disconnection from Source station so can say that event has happened for u. When a cut happens in the network that does not separate a node u from the source node, we can say that connected, but a Cut Occurred Somewhere so can say another event has occurred for u. [1]. 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 source node or the base station.

DISTRIBUTED CUT DETECTION ALGORITHM

```
Procedure: DCD(C)onsider S=Source node; Neighbors of node S are A,B.
ack=active;
dack=inactive
  if the node A is active i.e. ack state then
    Wait for 500ms.
    Send _le to node A.
  else if the node A is deactive nodefailed i.e. dack state then
    file sending to A failed.
  if the node B is active i.e ack state then
    Wait for 500ms.
    Send file to node B.
  else if the node B is deactive nodefailed i.e dack state then
    file sending to B failed.
```

Algorithm Description

Here we briefly describe the proposed DCD algorithm. One of the nodes of the network is a specially designed node which is always active called as source node". Let G = (V,E) denote the undirected sensor network that consists of all the nodes and edges of G that are active at time k, where k = 0, 1, 2 . . . is an iteration (repetitive) counter. Every node p of node set V maintains a scalar state (xp) (k) that is iteratively updated.

1. If no cut occurs or else no node fails then state of every node converges to a positive number.
2. If a cut occurs at a time T _ 0 which separates the graph G into N connected components (Gs) ,.. . . , GN, where the component (Gs) ((Vs)., (Es)) contains the source node, then
   (a) the state of every node disconnected from the source node converges to 0 (deactive) and
   (b) the state of every node in (Vs) converges to a positive number.

Hence by monitoring the states of the nodes one can know about the status of the network connection. For effectiveness we proposed a prototype model by taking small number of nodes and their corresponding edges in the graph G. Hence the nodes can effectively detect first if there is any cut occurred and second they are still connected to source. We modified this algorithm by adding additional parameters to reduce redundant information at destination. We designed it in such a way that once the file is sent from a node, it is sent to its respective neighbors so that each and every node has the information. If there is any node failure from where information cannot be forwarded and a cut is detected, the information at the nodes is combined and then sent to the destination without the occurrence of redundancy.

III. EVALUATION OF ALGORITHM

In this algorithm we are having two phases. One is state update law, it works very efficient to calculate the node potentials in electrical network (Gelec,1) when s Ampere current is injected to the source node and extracted to the nodes Vfict, with all nodes in V . The other phase of the algorithm consists of monitoring the state of a node, it is used to detect the cut occurred. Now we describe about the each phase.

A. State update law for WSN

The nodes use the computed potentials to detect if DOS events have occurred (ie., if they are disconnected from the source node).

To detect CCOS events, the algorithm uses the fact that the potentials of the nodes that are connected to the source node also change after the cut. CCOS detection proceeds by using probe messages that are initiated by certain nodes that encounter failed neighbors, if a short path exists around a "hole" created by node failures, the message will reach the initiating node.

Every node keeps a scalar variable, which is called a state Let G(k) = (V(k),E(k)) represent the sensor network that consists of all the nodes and edges of G that are still active at time k, where k=| 0,1,2,…. Is an iteration counter. We index the source node as 1. Every node u maintains a scalar state xu(k) that is updated. At every iteration k, nodes broadcast their current states. Let Nu(k) = {v|(u,v) € E(k)} denote the set of neighbors of u in the graph G(k). Every node in V except the source node updates the following state law. Where strength is design parameter:

1
Xi(k+1) = ------------ Σ xj(k) + s1{1} (i) (1)
Di(k) + 1 j€Ni(k)

Where di(k) := | Ni(k) | is the degree of node i at time k, and 1A(i) is the indicator function of the set A. That is, 1{1} (i) = 1 if i=1, and 1{1}(i) = 0 if i=1 and. After that, i

can update its neighbor list Ni(k) as follows: if no messages have been received from a neighboring node for the past Tdrop iterations, node i drops that node from list of neighbors. The integer parameter Tdrop is a design choice.

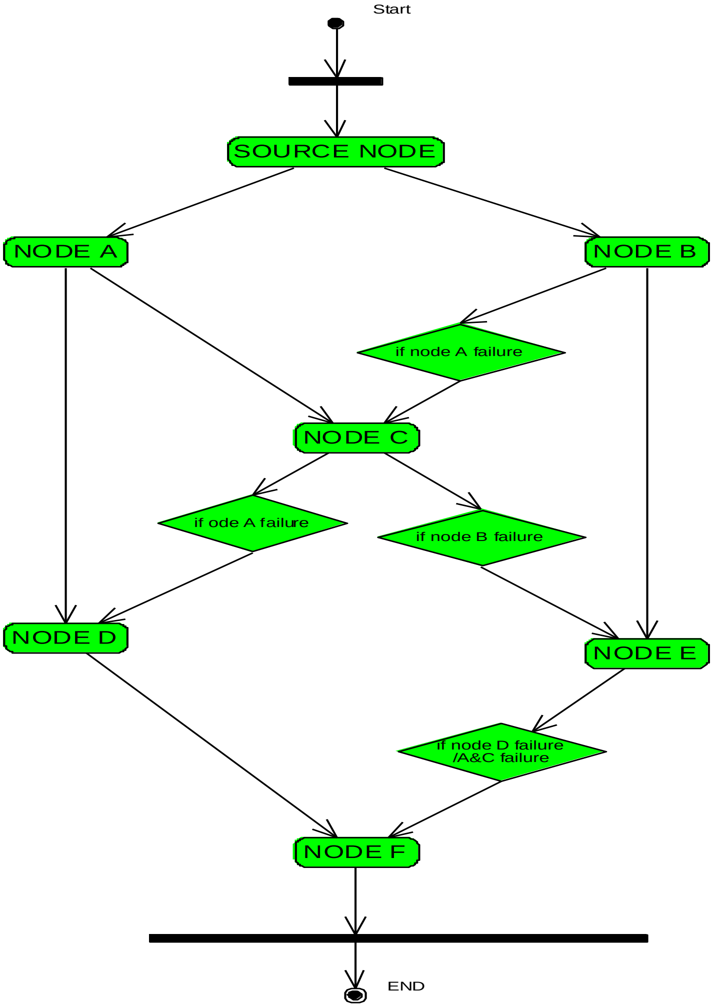
When the network is connected, the state of a node converges to its potential in the electrical network (Gelec ,1), which is a positive number. The potential of a node that is disconnected from the source is 0; this is the value converges to . If the reconnection occurs after a cut, the states of reconnected nodes again converge to positive values. Especially with wireless communication an asynchronous update is preferable

B. State monitoring for cut detection

Theorem 1 shows how the occurrence of a cut in the Network is manifested in the states of the nodes. By analyzing their own states, nodes can detect if a cut has occurred.

Suppose a cut occurs at some time $\tau > 0$ which separates the network into n components Gsource, G2, . . . , Gn, the component Gsource containing the source node. Since there is no source (and therefore no current injection) in each of the components G2, . . . , Gn disconnected from the source,it follows from Theorem 1 that the state of every node in each of these components will converge to zero. When the potential at a particular node drops below a particular threshold value, the node can declare itself cut from the source node. In fact, there may be additional node failures (and even increase in the number of components) after the cut appears. Since the state of a node converges to 0 if there is no path to the source, additional time variation in the network will not affect cut detection. If additional failures do not occur after the cut occurs, it follows from Theorem 1 that the states of the nodes that are in the component Gsource (which contains the source) will converge to new steady state values. So, if a node detects that its state has converged to a steady state, then changed, and then again converged to a new steady state value that is different from the initially seen steady state, it concludes that there has been a cut somewhere in the network. A node detects when steady state is reached by comparing the derivative of its state (with respect to time) with a small number ϙ that is provided a-priori. The parameters s and ϙ are design variables. It updates its state from that neighbor, in the asynchronous setting every node keeps a local iteration counter that may differ from those of other nodes by arbitrary amount.

The source node is at the center. The nodes 3b fail at k = 100, and thereafter they do not participate in communication or computation. Figs. The state of a node u decays to 0 after reaching a positive value, where the state of node v says positive.



IV. CUT PREVENTION

Cut Prevention using Cryptographic Puzzle Hiding Scheme

A sender S has a packet m for transmission. The sender selects a random key k , of a desired length. S generates a puzzle (key, time), where puzzle() denotes the puzzle generator function, and tp denotes the time required for the solution of the puzzle. Parameter is measured in units of time, and it is directly dependent on the assumed computational capability of the adversary, denoted by N and measured in computational operations per second. After generating the puzzle P, the sender broadcasts (C, P). At the receiver side, any receiver R solves the received puzzle to recover key and then computes.

PERFORMANCE EVALUATION

1. SIMULATION SETUP

We have performed simulations using the tool ns-2 to evaluate our proposed ideas. The simulation parameters that we have chosen are summarized in table I.

TABLE I. SIMULATION PARAMETERS

Parameters	**Values**
Simulation Time	200ms
Monitoring Area	1500 X 500
Number of Nodes	30
Communication Range	30m
Length of Packets	100bytes

2. SIMULATION RESULTS

a. Cut Detection

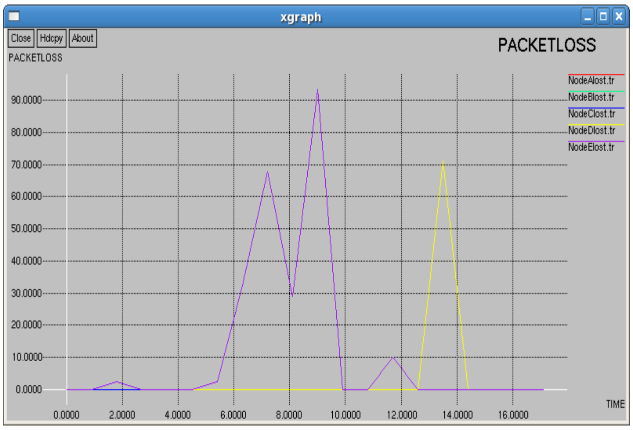


Fig. Packet Loss Ratio across the simulation time

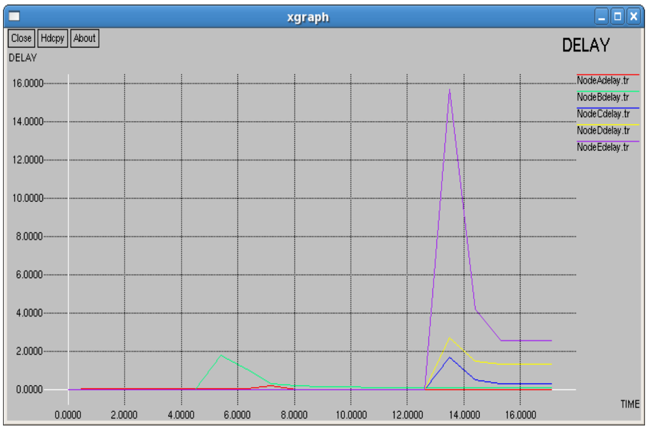


Fig. Packet Delay across the simulation time

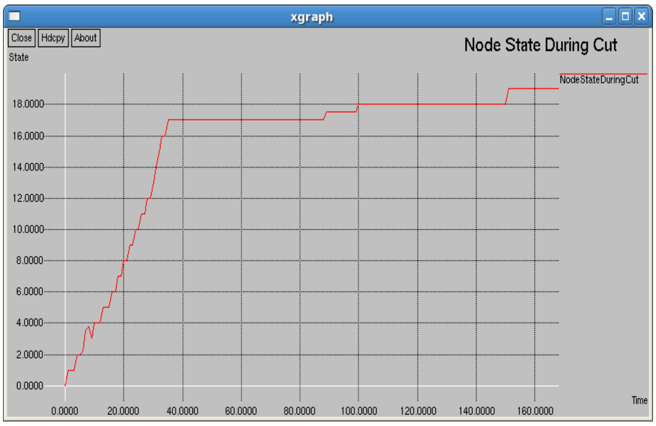


Fig. Node State during Cut across the simulation time

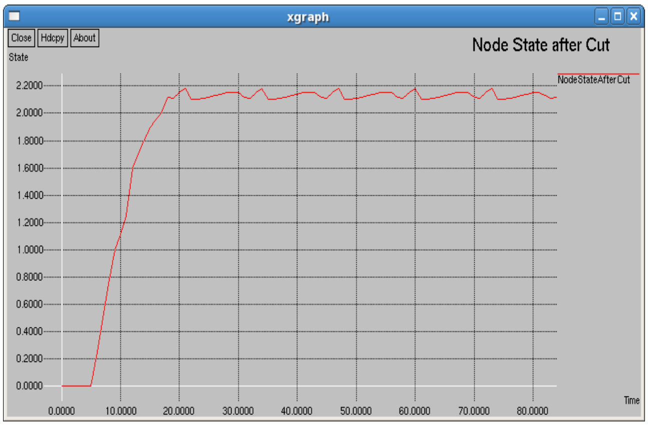


Fig. Node State after Cut across the simulation time

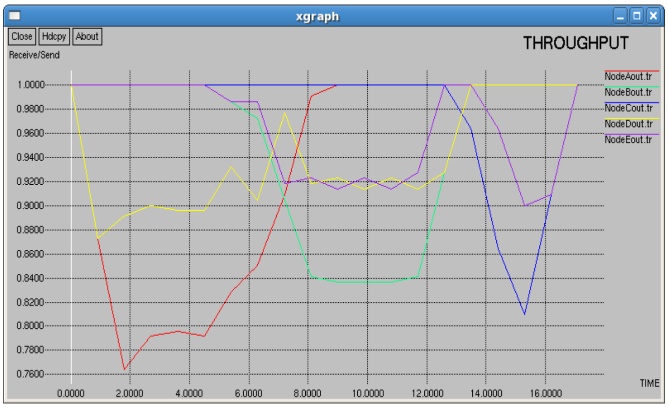


Fig. Throughput across the simulation time

CUT PREVENTION USING PUZZLE HIDING SCHEME:

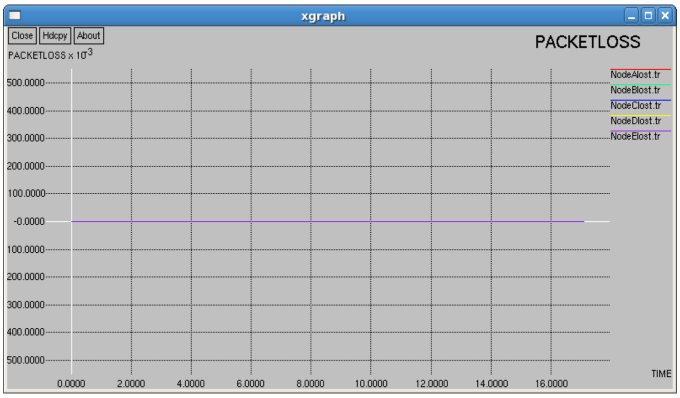


Fig. Packet Loss Ratio across the simulation time

V. CONCLUSION

The proposes algorithm enables every node of a wireless sensor network to detect DOS (Disconnected from Source) events if they occur. A key strength of the 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. System has applied a security scheme of puzzle hiding for preventing the cut detection in a network. Proposed system helps to increase delivery ratio and reduce the loss rate in the network. Proposed system shows the efficient cut detection in a wireless sensor network and also preventive measures to prevent occurring of cuts in network. So results have proved system works better than existing system and cuts can be prevented using the proposed scheme for increasing overall throughput of network.

REFERENCES

[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. N. Halgamuge, M. Zukerman, K. Ramamohanarao, and H. L. Vu, "An estimation of sensor energy consumption, "Progress In Electromagnetics Research B, Vol. 12, 259-295, 2009.

[5] Sankarasubramaniam, Y., I. F. Akyildiz, and S. W. McLaughlin, "Energy efficiency based packet size optimization in wireless sensor networks," Proc. IEEE Int. Sensor Network Protocols and Applications Conf., Vol. 1, No. 8, 2003.

[6] Zou, Y. and K. Chakrabarty, "Target localization based on energy considerations in distributed sensor networks," Proc. IEEE Int. Sensor Network Protocols and Applications Conf., Vol. 51, No. 58, 2003.

[7] Cut Detection in Wireless Sensor Networks : Prabir Barooah, Harshavardhan Chenji, Radu Stoleru, and Tamas Kalm´ ar-Nagy

[8] Detecting Cuts in Sensor Networks: Nisheeth Shrivastava Subhash Suri Csaba D. T´oth

[9] A Partition Detection System for Mobile Ad-Hoc Networks: Hartmut Ritter, Rolf Winter, Jochen Schiller

[10] Attacks in Wireless Sensor Networks :Rishav Dubey, Vikram Jain, Rohit Singh Thakur, Siddharth Dutt Choubey

[11] A Survey on Sensor Setworks,‖ : I. F. Akyildiz et al IEEE Commun. Mag., vol. 40, no. 8, Aug. 2002, pp. 102–112.

[12] C. Intanagonwiwat, R. Govindan, D. Estrin, Directed diffusion: a scalable and robust communication paradigm for sensor networks, Proceedings of the ACM Mobi- Com’00, Boston, MA, 2000, pp. 56–67.

[13] A Destination-based Approach for Cut Detection in Wireless Sensor Networks : Myounggyu Won, Student Member, IEEE, and Radu Stoleru, Member, IEEE

[14] Akyildiz, I.F., Su, W., ankarasubramaniam, Y., Cayirci, E.: Wireless Sensor Networks: A Survey. Computer Networks Journal (Elsevier), Vol. 38, No.4 (2002)pp. 393-422.

[15] 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.

[16] Detecting Cuts in Sensor Networks: Nisheeth Shrivastava Subhash Suri Department of Computer Science, University of California, Santa Barbara, CA 93106, USA.