Available online www.ejaet.com

European Journal of Advances in Engineering and Technology, 2015, 2(6): 103-106

Research Article ISSN: 2394 - 658X

RCDPWSN: Reliable approach for Cut Detection and Prevention in Wireless Sensor Networks

Priti Subramanium and Vijay J Chaudhari

Department of Computer Science and Engineering,
Shri Sant Gadge Baba College of Engineering and Technology, Bhusawal, Maharashtra, India
viju.ch14@gmail.com

ABSTRACT

In a dynamic Wireless Sensor Network fault occurring in sensor nodes are common due low-cost sensors used in WSNs, deployed in large quantities and prone to failure. A wireless sensor network can get separated into multiple network connected components due to the failure of some of its component nodes that is called a "cut". Cuts are group of failure nodes. An algorithm is proposed for detection and prevention that allows (i) every node to detect when the connectivity to a source node has been lost, and (ii) one or more nodes (that are connected to the neighbour node after the failure nodes) to detect the occurrence of the cut using nearest neighbours. (iii) Path selection using neighbour nodes. In this paper we have implemented NNS algorithm for detecting failure nodes by neighbours and avoiding failure paths for transmission in dynamic wireless sensor network. The RCDPWSN system can very easily prevent from data lost using rerouting.

Key words: Cut detection, Re-routing, Wireless sensor network

---

INTRODUCTION

WSNs are self-organized and usually deployed in dynamic environments. The underlying network topology constantly changes and no fixed routing path can be expected for each node. Wireless sensor networks combines sensing, computation, and communication into a single tiny device. System Architecture Directions for Networked Sensors proposed in [3]. Through advanced mesh networking protocols, these devices form a sea of connectivity that extends the reach of cyberspace out into the physical world. The mesh networking connectivity will seek out and exploit any possible communication path by hopping data from node to node in search of its destination. While the capabilities of any single device are minimal, the composition of fifty to hundreds of devices cooperatively passes their sensory information through the wireless network to a main location. Today such dynamic networks are used in many industrial and consumer applications, such as industrial process monitoring, ship monitoring, animal migration monitoring, medical monitoring, vehicle monitoring and, health monitoring, and so on. The method of cut detection in wireless sensor network is proposed in [7], [1].

We consider the problem of detecting partitions proposed in [2], data management for security and avoiding failure paths by the nodes of a wireless network. First we apply DCD algorithm for consider that a cut may or may not separate a node from the source node, we differentiate between two outcomes of a cut for a particular node. When a node u in the network is disconnected from the source, we say that a Disconnected from Source (DOS) event has arises for u. When node connected to source but failure occurs somewhere detection by nodes close to failure nodes. NNS algorithm able to failure detection via neighbours coordination and create new path for transmission for dynamic network. In this paper, we consider permanent faults for dynamic wireless sensor networks only, means faults occur due to battery depletion, which when not noticed would cause loss in connectivity and coverage. In this paper a cluster based fault management schemes which identify and rectifies the problems that occur due to energy depletion in sensor nodes.

DISTRIBUTED CUT DETECTION

Our problem is divided into three parts. First, we want to enable every node to detect if it is disconnected from the source (i.e., if a DOS event has occurred). Second, we want to enable nodes that are close to the failure path but are still connected to the source. Third, detecting failure nodes alert the source node and select another path for

transmission. There is an algorithm-independent limit to how accurately cuts can be detected by nodes still connected to the source, which are related to holes. Fig. 1 represented in [7] shows example of cuts, holes and failure nodes appeared in the network.

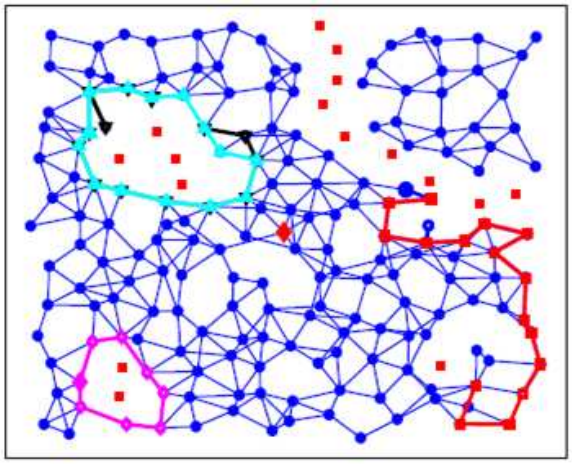

Fig.1 Example of cuts and holes in random network [7]

DOS DETECTION USING SPRAYING

The approach here is to exploit the fact that if the state is close to 0 then the node is disconnected from the source, otherwise not. In order to reduce sensitivity of the algorithm to variations in network size and structure, use a normalized state. DOS detection part consists of steady-state detection, normalized state computation, and connection or separation detection proposed in [5-6]. Every node i maintain a binary variable $\widehat{DOSi}$ (K), which is set to 1 if the node believes it is disconnected from the source and 0 otherwise. This variable, which is called the DOS event status, is initialized to 1 since there is no reason to believe a node is connected to the source initially. A node maintains track of the positive steady states seen in the past using the following method. Each node i computes the normalized state difference $\partial x_i(k)$ as shown in Eq. (1) as follows:

$$\partial \mathrm{xi(k)} = \begin{cases} \frac{x_i(k)-x_i(k-1)}{x_i(k-1)}, & if\, x_i(k-1) > \varepsilon_{zero} \\ \infty, otherwise \end{cases} \quad (1)$$

Where ε_{zero} is a small positive number. A node i maintains a Boolean variable PSSR (Positive Steady State Reached) and updates PSSR (k) ← 1 if $|\partial \mathrm{xi(k)}| \varepsilon_{\Delta x}<$ for k = k $-\tau_{\mathrm{guard}}$, k $-\tau_{\mathrm{guard}}$ + 1, . . . , k (i.e., for τ_{guard}consecutive iterations), where$\varepsilon_{\Delta x}$ is a small positive number and τ_{guard} is a small integer. The starting 0 value of the state is not considered a steady state, so PSSR (k) = 0 for k = 0, 1, ... τ_{guard}. Each node keeps an estimate of the most recent "steady state" observed, which is denoted by $\hat{x}_i^{ss}(k)$. This estimate is updated at every time k according to the following rule: if PSSR (k) = 1, then $\hat{x}_i^{ss}(k) \leftarrow x_i(k)$, other-wise $\hat{x}_i^{ss}(k) \leftarrow \hat{x}_i^{ss}(k-1)$. It is initialized as $\hat{x}_i^{ss}(0) = \infty$. Every node i also keep a list of steady states seen in the past, one value for each unpunctuated interval of time during which the state was detected to be steady. This information is kept in a vector $\hat{x}_i^{ss}(k)$, which is initialized to be empty and isupdated as follows: If PSSR (k) =1 but PSSR (k-1) = 0, then $\hat{x}_i^{ss}(k)$ is appended to $\hat{x}_i^{ss}(k)$ as a new entry. If steady statereached was detected in both k and k -1 (i.e.,PSSR(k) = PSSR (k-1)=1), then the last entry of $\hat{x}_i^{ss}(k)$ isupdated to $\hat{x}_i^{ss}(k)$. For instance, for the node *v* in the network,$\hat{x}_v^{ss}(3) = \emptyset$(empty),$\hat{x}_v^{ss}(60)$=(0:0019) and $\hat{x}_v^{ss}$ (150)=(0:019; 0:012). For future use, we also define an unsteady interval for a node *i*, which is a set of two local time counters $(k_i^{(1)}, k_i^{(2)})$ such that the state $x^i(k_i^{(1)} - 1)$is a steady-state (i.e.PSSR $(k_i^{(1)} - 1)$) but$x^i k_i^{(1)}$is not, and $x^i k_i^{(2)}$is not steady but $x^i(k_i^{(2)} = 1)$is.

Each node computes a normalized state $\hat{x}_i^{norm}(k)$ as:

$$x_i^{norm}(k) := \begin{cases} \frac{x_i(k)}{\hat{x}_i^{ss}(k)}, if\ \hat{x}_i^{ss}(k) > 0 \\ \infty,\ otherwise \end{cases} \quad (2)$$

Where$\hat{x}_i^{ss}(k)$ is the last steady state seen by *i* at *k*, i.e., the last entry of the vector$\hat{x}_i^{ss}(k)$. If the normalized state of *i* is less than εDOS, where εDOS is a small positive number, then the node declares a cut has taken place:$\widehat{DOS} \leftarrow 1$. If thenormalized state is ∞, meaning no steady state was seenuntil *k*, then $\widehat{DOSi}$ (K) is set to 0 if the state is positive (i.e.,$x_i k > \varepsilon_{zero}$) and 1 otherwise.

PREVENTION

One of the main goals of sensor networks is to provide accurate information about a sensing field for an extended period of time. Because sensor networks may interact with sensitive data and/or operate in hostile unattended

environments, it is imperative that these security concerns be addressed from the beginning of the system design. An algorithms provides a sufficient levels of security, it simply converts original packet into binary format, only destination nodes able to convert data into original form.

NNS ALGORITHM

Detection by NN

The algorithm for detecting nodes connected but cut occurs somewhere in the network events relies on finding a short path around a hole, if it present. The algorithm is based on Single-Link Failure Detection proposed in [8], restricted for single link. We simply issue a broadcast message in the network, and each node is expected to receive multiple probe messages through different paths.

There is following information in each PROBE message p:

- Node ID
- Counter that contains information hops from sink to current node.
- Timestamp
- Destination
- Packet Size

Each probe message contains a counter that records the number of hops from sink to current node. As cuts or failure nodes may impact this process, there will be mismatches between the received hop counters in sensor nodes.

Re-routing

Many algorithms are proposed in [4] for disturbance or failure in wireless sensor network. The path selection for data transmission is done based on the availability of the nodes in the region using the Nearest Neighbour Search Algorithm (NNS) algorithm. The routes are created on the basis of cost of edge between the nodes. In order to facilitate determination of the freshness of routing information NNS maintains the time since when an entry has been last utilized. A routing table entry is "expired" after a certain predetermined threshold of time. Consider all the nodes to be in the position. Now the shortest path is to be determined by implementing the NNS in the wireless simulation environment for periodically sending the messages to the neighbour's and the shortest path.

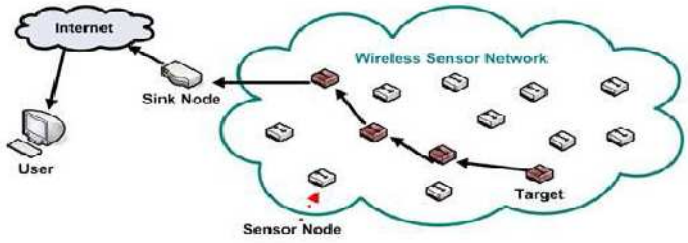

Fig. 2 Routing in Wireless Sensor Network

RESULTS AND DISCUSSIONS

In this paper, we will show and discuss all of the results obtained by the computer simulation program DOT NET. Simulations are conducted on 200 nodes 2D dynamic network. In this paper, cut detection and rerouting methods for dynamic wireless sensor network are studied. Table 1 shows the comparative chart for existing fault detection technique and proposed system for 2D random Wireless Sensor Networks. The algorithm correctness is evaluated by probability of two type's errors. The probability of DOS0/1 error at time k is the ratio between the number of nodes that incur a DOS0/1 error (who believe they are connected but are not) at that time to the number of nodes that are disconnected from the source at that time. Probability of DOS1/0 error at k is the ratio between the number of nodes that incur a DOS1/0 error (who believe they are disconnected from the source but are in fact connected) to the number of nodes that are connected to the source at that time. The mean and standard deviation of DOS detection delay for a network are computed by averaging over the nodes that detected DOS events.

The mean and standard deviation of DOS detection delay for a network are computed by averaging over the nodes that detected DOS events. In Table 1 that the algorithm is able to successfully detect initial connectivity to the source and Table shows the delay mean for existing system as well as proposed system in dynamic WSN.

Table -1 Comparative Chart for DOS Detection Performance

Comparison	Existing System	Proposed System
Prob(DOS0/1 Error)	0	0
Prob (DOS1/0 Error)	0	0
DOS Delay (mean)	35	32
DOS Delay (Std. dev.)	3.9	3.3

Table -2 Comparative Chart for Detection by NN Performance

Comparison	Existing System	Proposed System
Prob(DOS0/1 Error)	0.33	0
Prob (DOS1/0 Error)	0	0
Delay (mean)	40	37

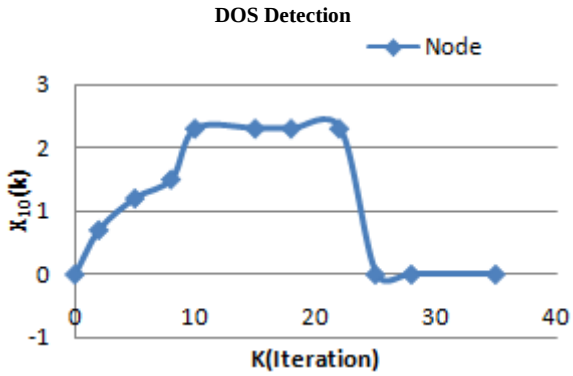

Fig. 3 State of node 10 disconnected from source

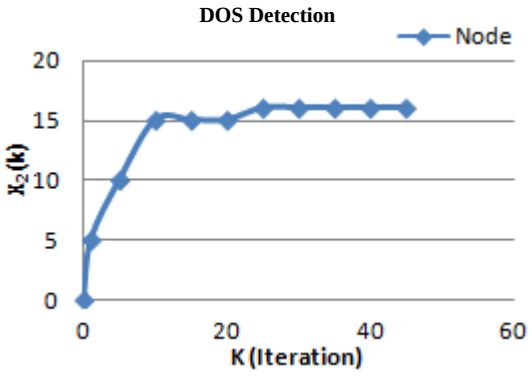

Fig. 4 State of node 2 connected to source

Simulation results for two sensors in network of 25 nodes shown in Fig. 3 and Fig. 4.In Fig. 3 node 10 is disconnected from source node. We get state of the failure node 10 after 25 iterations in the simulation study. The state of Node 2 is connected to the source represented in Fig.4.

CONCLUSION

In this paper we proposed RCDPWSN, a Reliable approach for Cut Detection and Prevention in Wireless Sensor Networks. RCDPWSN didn't use any communication model. The DCD algorithm detects Disconnected from Source if occurred. NNS (Nearest Neighbour Search) algorithm detects nodes connected to the source but cut occurred anywhere and failure path avoidance using rerouting for transmission of data successfully. So RCDPWSN is very sophisticated technique for detecting and preventing cuts in WSN. Through Simulation we will try that avoid failure path due to cuts and detect failure node from source at minimum time period. This method gives good performance; it only loses part of its effectiveness under situations large holes created in dynamic network. Further research will be done to develop a method for security and routing protocol for cut network.

REFERENCES

[1] G Dini, M Pelagatti and IM Savino, An Algorithm for Reconnecting Wireless Sensor Network Partitions, *Proc European Conf Wireless Sensor Networks*, **2008,** 253-267.
[2] 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 Communication and Networks* (IEEE SECON '04), **2004**, 489-497.
[3] J Hill, R Szewczyk, A Woo, S Hollar, D Culler and K Pister, System Architecture Directions for Networked Sensors, *Proc International Conf Architectural Support for Programming Languages and Operating Systems* (ASPLOS), **2000.**
[4] R Klempous, J Nikodem, L Radosz Raus and N Byzantine, Algorithms in Wireless Sensors Network, *IEEE International Conference on Information and Automation*, **2006**, 319-324.
[5] M Hauspie, J Carle and D Simplot, Partition Detection in Mobile Ad-Hoc Networks, *Proc Second Mediterranean Workshop Ad-Hoc Networks*, **2003,** 25-27.
[6] P Barooah, Distributed Cut Detection in Sensor Networks, *Proc 47th IEEE Conf Decision and Control*, **2008,** 1097-1102,
[7] Prabir Barooah, Harshavardhan Chenji, RaduStoleru and Tamas Kalmar-Nagy, Cut Detection in Wireless Sensor Networks, *IEEE Transactions on Parallel and Distributed Systems*, **2012,** 23, 1-8.
[8] SS Ahuja, S Ramasubramanian and MM Krunz, Single-Link Failure Detection in All-Optical Networks using Monitoring Cycles and Paths, *IEEE ACM Transactions on Networking*, **2009,**17(4),1080–1093.