A Jamming Defending Data-Forwarding Scheme for Delay Sensitive Applications in WSN

Amrita Ghosal, Subir Halder
Dept. of Computer Science and Engineering
Dr. B. C. Roy Engineering College
Durgapur, India
{amrita.ghosal, subir_ece}@rediffmail.com

Md. Mobashir, Rajesh K. Saraogi, Sipra DasBit
Dept. of Computer Science and Technology
Bengal Engineering and Science University, Shibpur
Howrah, India
siprad@hotmail.com

Abstract— **Wireless sensor network (WSN) has emerged as an important application area where nodes are generally placed in an unattended environment and therefore are vulnerable to attack by adversaries. Moreover in real time application domains delay even by fraction of a second may deceive the purpose of the application. Therefore, designing an attack-defending scheme for WSN without involving any additional delay in data-forwarding is an important challenge in this domain. The present work proposes a data-forwarding scheme having no additional delay to defend jamming attack in WSN. The scheme considers a multi-layer architecture where the layers are made up of hexagonal cells each containing sensor nodes in the form of clusters. The nodes are randomly deployed throughout the network and clusters are formed so that every cell hosts two sets of clusters operating in two different predefined frequencies. We have provided design guideline to determine cell size in terms of network parameters. This guideline ensures that for a cluster head there is at least another cluster head at one-hop distance towards the sink and thereby ensures data-forwarding in shortest path. During data-forwarding, if a frequency is jammed, the cluster operating in that frequency becomes inoperative and the other cluster acts as back-up. We claim that the scheme defends jamming attack without incurring any additional delay and the claim is substantiated through simulation.**

Keywords- Wireless sensor network, Jamming attack, Hop distance, Connectivity, Coverage, Delay

I. INTRODUCTION

A Wireless sensor network (WSN) [1] consists of several hundreds of resource-constrained sensor nodes which have limited battery power, memory capacity and processing ability. The sensor nodes collect data from its sorroundings and send it to their neighbouring nodes located in single-hop which in turn send the data to their neighbouring nodes located in single-hop. The data is finally transmitted to the sink node in multi-hop that exfiltrates data to the outside world.

In WSN the major mode of communication is broadcast in nature and in most of the application domains resource-constrained nodes are placed in an unattended environment. Moreover, in real time application domains e.g. tracking of a moving objet, delay even in fraction of a second may deceive the purpose of the application. Therefore, considering these issues coupled with the need of delay-sensitive data-forwarding scheme for real time applications, defending the WSN from attacks is a real challenge. In this environment DoS (Denial of Service) is a common type of attack, which affects the performance of the network by disrupting the normal functioning of the network. Several types of DoS [2] attacks are prevalent in sensor networks and one of them is the jamming attack. In jamming attack, the intention of the attacker is to block the communication frequency used by the sensor nodes so that they are compelled to stop their communication and wait for communication to resume till the jammer releases the frequency in use. So jamming attack can result in widespread disturbance leading to disruption in data transmission.

There are many applications in sensor networks that are delay-sensitive in nature. These applications include all sorts of real time applications e.g. tracking of a moving object [3], fire alarm systems etc. A wireless sensor network that supports these applications must therefore guarantee the timely delivery of alarms even in presence of jamming attacks.

A number of works towards defending WSN from jamming attack have been reported so far. In one such work [4], authors have proposed strategies for jamming detection and mitigation in physical as well as MAC layers. Physical layer detection of jamming attack has been done measuring the differences in noise levels of the signals during normal operation and in presence of jamming attack. MAC layer detection has been done via a threshold mechanism based on the channel sensing time by the nodes. Jamming mitigation has been done using two strategies- channel surfing and spatial retreats. But the spatial retreat based evading technique has a serious drawback as it involves use of nodes capable of mobility adding to the overhead factor of the network.

Authors in [4] have further extended [5] their work by introducing certain measurement parameters such as, signal strength, carrier sensing time and packet delivery ratio (PDR) to detect jamming. If a node is jammed all these parameter values would change compared to the normal condition. To confirm that the changes are due to jamming, more than one parameter's values are considered together. The authors have proposed two enhanced protocols capable of detecting jamming. The first protocol uses a combined consistent check of signal strength and PDR whereas the second one uses combined consistency checks of location of a node and PDR. But the location consistency checking protocol requires the support of additional structure such as GPS or other localization techniques thereby increasing the cost of the

978-1-4577-0787-2/11/$26.00 ©2011 IEEE

network set up. Nonetheless, this work has not considered the issue of jamming mitigation.

The extended version of the works [4], [5] appears in [6] where jamming evading technique known as the channel surfing scheme has been used. Channel surfing refers to channel shifting on detection of jamming in the current channel being used for communication. This scheme has a drawback that whenever a region is jammed, some of the nodes have to take the responsibility of rebuilding the network in addition to their basic tasks of sensing and relaying. This results in quick drainage of battery of those nodes affecting energy balance, network lifetime and coverage of the network to a great extent.

In [7], the authors have proposed a jamming detection strategy where the nodes sense jamming with the help of received signal strength and bit-error-rates. Once the nodes in a region observe that the values of these two parameters fall below a pre-determined threshold, they consider that the changes are due to jamming and inform their neighbouring nodes about the jamming of that region. The nodes demarcate the jammed region by exchanging messages among themselves. The rest of the network isolates itself from the demarcated jammed region. But this operation leads to forgoing the activity of a portion of the network thereby breaking coverage of the network.

Authors in [8] have proposed a protocol which is able to extort data from the jammed area using multiple communication channels simultaneously. When a node detects that it is jammed, it switches from the normal mode of operation to the exfiltration mode. The exfiltration mode involves several complicated operations such as creating of interference matrix, exfiltration matrix and scheduling algorithm. Implementation of all these complicated procedures involves a lot of energy consumption of sensor nodes and therefore, this is a major hindrance for sensor networks as nodes are equipped with very limited battery power.

The present work proposes a scheme that is able to continue data-forwarding even in the presence of multiple jamming attacks without incurring any additional delay. The proposed scheme can defend the attack irrespective of the locations of the attacked nodes within the network. The rest of the paper is organized as follows. In section II, objective and the WSN architecture considered here are described. The proposed scheme and implementing algorithm are presented in section III. Qualitative analysis is performed in section IV. In section V, performance of the scheme is evaluated. Concluding remarks and future scope have been stated in section VI.

II. Background and Objective

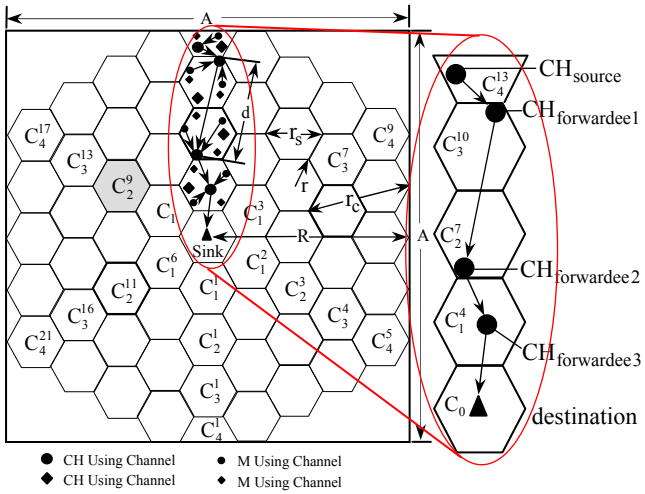
The following sub-sections describe objective of the proposed work after defining the attack and the WSN architecture considered in the work.

A. Objective

An attack [2] is defined as an attempt to gain unauthorized access to a service, resource, or information, or the attempt to compromise integrity, availability, or confidentiality. One such attack in WSN is known as Denial of Service (DoS) where a node under attack is denied access. Jamming is one type of DoS attack where deliberate interference is made by blocking the communication channel used by the attacked node. Once a jammer attacks a region, it jams all the nodes that are operating in a specific channel and located within its range. The objective of the present work is to develop a jamming attack-defending data-forwarding scheme incurring no additional delay.

B. Multi-layer Architecture

We consider multi-layer architecture, similar to regular hexagonal cell architecture RHC [9], where the network coverage area is divided into layers containing regular hexagonal cells as shown in Figure 1. In an n-layered architecture, the number of cells in each layer is 6y, where y=1,2,…,n. Here y=1 indicates the layer nearest to and y=n indicates the layer farthest from the sink. The centre cell where the sink is located is considered as layer 0 having one cell called as sink cell. A cell is identified by C_y^x where for a given y, $x = 1, 2, \ldots, (6y)$. For example, the cell C_2^9 identifies the 9th cell in layer 2.



(a) Network area (b) Data forwarding

Figure 1. Multi-layer architecture

1) Deployment: The layers are made up of hexagonal cells within which the nodes are placed randomly. Once the nodes are deployed in the entire network, clusters are formed in such a manner that there are two sets of clusters where each set of clusters operate in a predefined frequency. A cluster has its own cluster head and member nodes. Moreover each cell hosts two clusters with their respective cluster heads where one cluster operates in the predefined frequency say channel 1 and the other cluster operate in say channel 2. We call this feature of hosting two clusters with two different operating frequencies in a cell area is pair-wise deployment.

We denote the member nodes in C_y^x operating in channel 1 and channel 2 by $M_{y_1}^x$ and $M_{y_2}^x$ respectively. Similarly, we denote the cluster heads in C_y^x operating in channel 1 and channel 2 by $CH_{y_1}^x$ and $CH_{y_2}^x$ respectively.

The member nodes are responsible for sensing the environment and passing on the data to their corresponding cluster heads in one time slot. The task of cluster head is to aggregate the data received from its member nodes and transmit the same to the neighbouring cluster head (to be defined) in the next time slot.

2) Design guideline:

Neighbouring cluster head: We define a cluster head as a neighbouring cluster head ($CH_{neighbour}$) of another cluster head (CH) where $CH_{neighbour}$ maintains the following:

- located in the adjacent layer towards sink
- at one-hop distance from the CH

- operates in the same frequency as in the CH

Coverage: A unit area is said to be covered if every point in the area is within the sensing range of an active node [9].

Connectivity: A network is connected if any active node can communicate with any other active node either in single hop or in multiple hops [9].

The following relevant notations are used to describe the architecture:

- r– radius of a cell
- r_s – sensing range of a sensor node
- r_c – communication range of a sensor node
- d– distance between a cluster head and its neighbouring cluster head

The relationship between cell radius r and node's sensing range r_s must satisfy the condition- $r \le \frac{r_s}{2}$ [9] to cover the cell area and thereby ensures the coverage of the network. On the other hand, for ensuring connectivity the relationship between r and communication range r_c must be $r \le \frac{r_c}{\sqrt{13}}$ [9]. Moreover, as d is at one-hop distance, $d \le r_c$.

3) Energy Model: We have considered the first order radio model [10] as our energy model where the energy consumption of a node is dominated by its wireless transmissions and receptions; so we have neglected the other energy consumption factors such as for sensing and processing.

Lemma 1: For a given network area A×A, in order to maintain connectivity of the network, the number of layers (n) stands in relation with r_c as $n \ge \sqrt{\frac{13}{3}}\frac{R}{r_c}$ where $R = \frac{1}{2} \times A$.

Proof: If the radius of each cell of the hexagonal layer architecture is r, then the distance between the centre of the sink cell and the farthest edge of a cell of any other layer is given by

$$\sqrt{3}\, r\, y + \frac{\sqrt{3}}{2} r$$

where y is the layer number.

If the distance between the centre of sink cell to the farthest point in the network area is R, then replacing y by n, we get

$$\sqrt{3}\, r\, n + \frac{\sqrt{3}}{2} r \ge R \quad or, \quad n \ge \frac{R - \frac{\sqrt{3}}{2} r}{\sqrt{3}\, r}$$

replacing, $r \le \frac{r_c}{\sqrt{13}}$ in above equation, we have $n \ge \sqrt{\frac{13}{3}}\frac{R}{r_c}$.

Corollary 1: For a given network area A×A, in order to maintain network coverage, the number of layers (n) stands in relation with r_s as $n \ge \frac{2}{\sqrt{3}}\frac{R}{r_s}$.

Proof: From lemma 1, the relationship between R and n is evaluated as,

$$\sqrt{3}\, r\, n + \frac{\sqrt{3}}{2} r \ge R$$

replacing, $r \le \frac{r_s}{2}$ in the above equation, we have $n \ge \frac{2}{\sqrt{3}}\frac{R}{r_s}$.

III. THE SCHEME

In this section the proposed data-forwarding scheme and the algorithm implementing the scheme is presented.

A. Data-forwarding scheme

Considering n-layer architecture, initially the member nodes of all the n layers sense data and transmit the same to their respective cluster heads in the first time slot. In the next time slot, all the cluster heads send the aggregated data to their neighbouring cluster heads. In the subsequent time slot, the neighbouring cluster head in turn forwards data to its neighbouring cluster head. In this way, data reaches the sink in multi-hop.

Once a jammer attacks a region, it jams all the nodes operating in a particular channel. For example, if the nodes in a region operating, say in channel 1 are jammed, the nodes no more sense data unless the action of the jammer ceases. However, as the nodes are deployed maintaining pair-wise feature (section II.B), the cluster operating in the other channel say channel 2 continues to perform the task of sensing and forwards the same through neighbouring nodes.

Let us consider a WSN of 4-layer architecture as shown in Figure 1(a). The sink is denoted by- ▲ and cluster heads operating in channel 1 & channel 2 are represented by ● and ◆ respectively. Similarly member nodes operating in channel 1 & channel 2 are represented by ● & ◆ respectively. When there is no jamming, clusters operating both in channel 1 & channel 2 are active and environmental data is collected by both the clusters. For example, referring Figure 1(b), a cluster in the cell C_4^{13} operating in channel 1 senses the data and sends it in time-slot 1 say to the CH_{source} which aggregates the data and forwards it to its neighbouring node $CH_{forwardee1}$ in time-slot 2 say. In this way the data reaches to the sink in 4 time-slots.

B. Algorithm

The algorithm describing the scheme has been presented in this section.

Begin
1: m:=0
2: time_slot:= 2m+1
3: **do** while .T.
4: **for** y = n; y≥1; y - -
5: **for** x =1; x≥6y; x + +
6: **if** time_slot = 2m+1
/* Member nodes collect data and send them to their CHs*/
7: **if** no jamming **then**
8: $M_{y_1}^{x} \xrightarrow[\text{Collected Data}]{\text{Send}} CH_{y_1}^{x}$
9: $M_{y_2}^{x} \xrightarrow[\text{Collected Data}]{\text{Send}} CH_{y_2}^{x}$
10: **else** /*there is jamming */
11: **if** channel 1 is jammed **then**
12: $M_{y_2}^{x} \xrightarrow[\text{Collected Data}]{\text{Send}} CH_{y_2}^{x}$
13: **else** /* channel 2 is jammed */
14: $M_{y_1}^{x} \xrightarrow[\text{Collected Data}]{\text{Send}} CH_{y_1}^{x}$
15: **end if**
16: **end if**
17: **else** /* time_slot=2m */

978-1-4577-0787-2/11/$26.00 ©2011 IEEE

/* CH sends data to the neighbouring CH */

if no jamming **then**

$CH^{x}_{y_1} \xrightarrow[\text{data}]{\text{forward}} \left(CH^{x}_{y_1}\right)_{\text{neighbour}}$

$CH^{x}_{y_2} \xrightarrow[\text{data}]{\text{forward}} \left(CH^{x}_{y_2}\right)_{\text{neighbour}}$

else /*there is jamming */

if channel 1 is jammed **then**

$CH^{x}_{y_2} \xrightarrow[\text{data}]{\text{forward}} \left(CH^{x}_{y_2}\right)_{\text{neighbour}}$

else /*channel 2 is jammed */

$CH^{x}_{y_1} \xrightarrow[\text{data}]{\text{forward}} \left(CH^{x}_{y_1}\right)_{\text{neighbour}}$

end if

end if

end if

m:= m+1

time_slot:= time_slot+1

end for

end for

end do

End

IV. ANALYSIS ON DATA FORWARDING

In this section the analysis is performed with the help of a sample sensor network containing 4 layers as shown in Figure 1. We represent data-forwarding at every time cycle with the help of a timing diagram represented in Figure 2. The first column denotes the time cycles. The rest of every two columns represent the layer number where one column indicates member nodes (M) and the other column indicates the cluster head nodes (CH) of the corresponding layer. The timing diagram shows data communication from a member node to its cluster head node of the same layer and also communication from cluster head node to another cluster head node of adjacent layers toward sink. Further, channel 1 is indicated by ——● and channel 2 is indicated by ----●. The last column represents the sink. The entries in the last column represent data $(D_{11}, D_{21}, D_{12}, \ldots, D_{13})$ forwarded to the sink where D_{ij} represents the data sensed at layer j at sequence i.

If there is no jamming, at time t_1, member nodes send data generated at sequence 1 to their cluster heads. During t_2 cluster head of layer-1 transmits data D_{11} to the sink. Also at t_2 cluster heads of layer-2, layer-3 and layer-4 forward data D_{12}, D_{13}, D_{14} to their respective neighbouring cluster heads at layer-1, layer-2 and layer-3.

At t_3, member nodes again transmit data generated at sequence 2 to their cluster heads. So at the end of t_3, buffer of cluster head of layer-1 contains- D_{21}, D_{12} buffer of cluster head of layer 2 contains- D_{22}, D_{13} buffer of cluster head of layer-3 contains- D_{23}, D_{14} and buffer of cluster head of layer-4 contains- D_{24}. At t_4, two sets of data reach the sink- D_{21} and D_{12}.

In presence of jamming, if channel 1 (say) is jammed in an area, as the nodes are deployed in pair-wise (section II.B), the member nodes and CHs operating in channel 2 are operative and the data gets forwarded by these back-up nodes. So even in presence of jamming data gets transmitted with no additional delay.

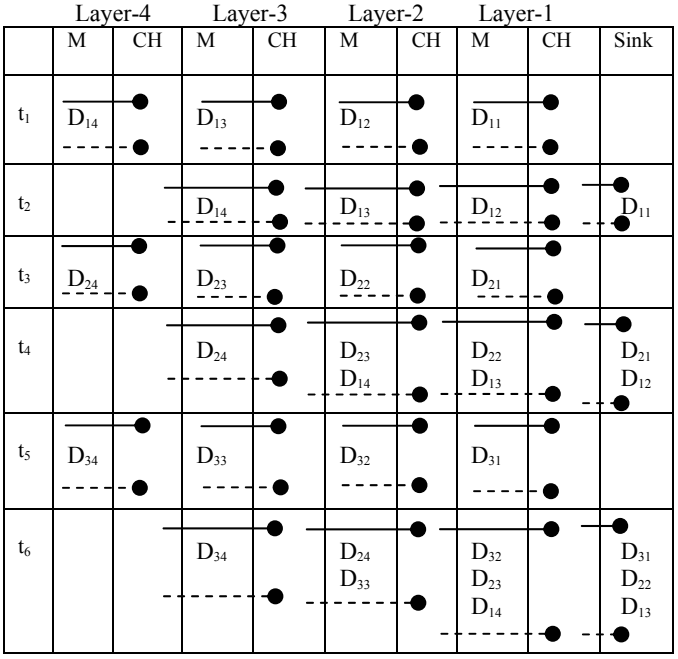


Figure 2. Timing diagram for no jamming

A. Qualitative Analysis

Lemma 2: Numbers of hops required for data to reach from the member node to the sink are (y+1), where y is the layer number in the network.

Proof: Initially member nodes of all layers collect data and send them to their cluster heads in one hop. Now the cluster heads send data to the sink through cluster heads of neighbour cells. Suppose a member node M^{x}_{y}, belonging to cell-x of layer-y, collect data and sends it to its cluster head CH^{x}_{y} in one hop. After that CH^{x}_{y} sends data to the cluster head of neighbour cell i.e. CH^{x}_{y-1} also in one hop. So for every increase in hop number, the layer number decreases by 1 and the communication continues likewise till data reaches sink. So before reaching the sink, the data passes through y layers. From CH^{x}_{y} to sink, the data travels through y hops. Another 1 hop is required for M^{x}_{y} to CH^{x}_{y} communication. So total (y+1) numbers of hops are required for data to reach the sink and this is also the requisite minimum number of hops.

Lemma 3: The number of clock cycles required for the data sensed at layer y to be forwarded to the sink node is 2y.

Proof: Suppose t_k is the clock cycle at which operation starts at layer-n. At t^{th}_{k} clock cycle M^{x}_{y} send data to CH^{x}_{y} using channel 1 and channel 2. In the next clock cycle i.e. $(t_k+1)^{th}$, CH^{x}_{y} send data to CH^{x}_{y-1} using both channels 1 and 2. This operation continues till data reach the sink.

Let us consider at t^{th}_{k} clock cycle in layer-y, M^{x}_{y} sends data to CH^{x}_{y} using both channels 1 and 2. At $(t_k+1)^{th}$, CH^{x}_{y} sends data to CH^{x}_{y-1}. This operation continues till data reaches the sink using both channels 1 and 2. So the time taken for the data to reach the sink from M^{x}_{1} is $[\{(t_k+1)\}-\{t_k-1\}]=2$ clock cycles.

978-1-4577-0787-2/11/$26.00 ©2011 IEEE

Similarly at $(t_k+2)^{th}$ clock cycle in layer-y, again M_y^x sends data to CH_y^x using both channels 1 and 2. In the next clock cycle i.e. $(t_k+3)^{th}$, CH_y^x sends data to CH_{y-1}^x and so on till data reaches the sink using channels 1 and 2. So the time taken for the data to reach the sink from M_1^x is $[\{(t_k+3)\}-\{t_k+1\}]=2=2\times1$ clock cycles. So from the start of operation, total time taken for data to reach sink from M_2^x is $[\{(t_k+3)\}-\{t_k-1\}]=4=2\times2$ clock cycle. Similarly from the start of operation time required for the data to reach the sink from M_3^x is $[\{(t_k+5)\}-\{t_k-1\}]=6=2\times3$ clock cycles. Hence the time required for the data to reach the sink is 2y clock cycles.

V. Performance Evaluation

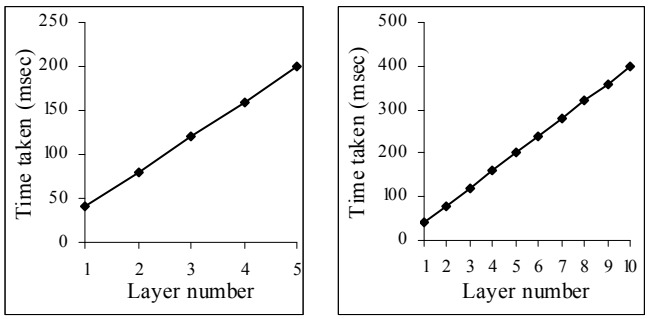
The performance of the proposed data-forwarding scheme has been evaluated through simulation under no jamming and with jamming conditions.

A. Simulation Environment

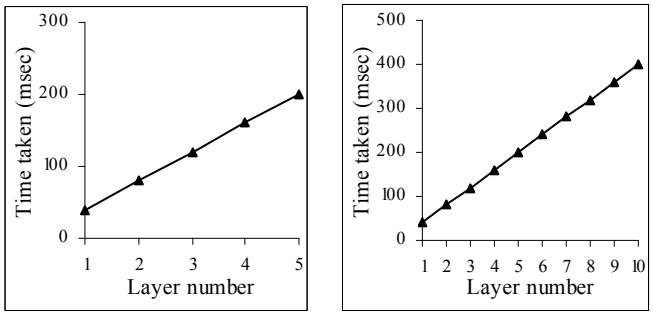
The simulation is performed using MATLAB (version 7.1). Th network size is taken as input parameter for simulation. The other relevant parameters considered are- sensing range (r_s)- 80m, communication range (r_c) of – 160m, no. of frequencies used for communication-2 and clock cycle time- 20msec.

B. Simulation Results

This section presents the results of two sets of experiments carried out under 'no jamming' and 'with jamming' conditions. The performance of the scheme is measured in terms of time taken for a data to reach the sink.



(a). no jamming



(b). with jamming

Figure 3. Time taken for data to reach sink

Figure 3(a) plots the time required for the data genearated at different layers to reach the sink for two different network sizes under 'no jamming' condition. We observe from both the plots that irrespective of network size, time to reach sink inreases steadily with layer numbers. For example, for network with 10 layers, data generated at layer 3 needs 120 msec whereas data at layer 4 needs 160 msec to reach the sink. As the clock cycle is considered as 20 msecs, for the former case it requires 6 (120/20) clock cycles and for the later it requires 8 (160/20) clock cycles. Therefore, this results conform with the qualitative analysis (Lemma 3) where it is proved that number of clock cycles required is (2×layer number). Similar to the plots in Figure 3(a), results under 'with jamming' condition are ploted in the Figure 3(b). Both the plots 3(a) & 3(b) show same results which implies that there is no additional time required for a data to reach to the sink in presence of jamming.

VI. Conclusion

Nodes in wireless sensor network are susceptible to various types of attacks including jamming attack. This work develops a data-forwarding scheme to defend one of such attacks irrespective of the locations of the attacked nodes within the network and ensures no additional delay in data-forwarding even in the presence of jamming. This delay-sensitive, attack-tolerant data-forwarding scheme is proposed to cater the need of real time applications in WSN. The cost that has to be paid for ensuring no delay in data propagation is that deploying redundant number of nodes, a realistic and common solution generally employed in most of the application domains.

As a future extension, further analysis may be performed to find out the optimal number of nodes required to implement the scheme. Moreover, the scheme may be extended to make it applicable for various other architectures.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless Sensor Network: Survey," IEEE Computer Networks, vol 38, no 4, pp 393-422, 2002.

[2] A. D. Wood, and J. A. Stankovic, "Denial of Service in Sensor Networks", IEEE Computer, vol 35, no 10, pp 54-62, 2002.

[3] T. He, P. Vicaire, T. Yan, L. Luo, L. Gu, G. Zhou, R. Stoleru, Q. Cao, J. A.Stankovic, and T. Abdelzaher, "Achieving Real-Time Target Tracking Using Wireless Sensor Networks", Proceedings of the 12th IEEE Real-Time and Embedded Technology and Applications Symposium, pp 37 – 48, 2006.

[4] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "Channel Surfing and Spatial Retreats: Defenses Against Wireless Denial of Service," Proceedings of ACM Workshop on Wireless Security, pp 80–89, 2004.

[5] W. Xu, W. Trappe, Y. Zhang, and T. Wood, "The Feasibility of Launching and Detecting Jamming Attacks in Wireless Networks," Proceedings of MobiHoc, pp 46-57, 2005.

[6] W. Xu, W. Trappe, and Y. Zhang, "Channel Surfing: Defending Wireless Sensor Networks from Jamming and Interference," Proceedings of ACM Conference on Embedded Networked Sensor Systems (SenSys), pp 499-508, 2006.

[7] A. D. Wood, J. A. Stankovic, and S. H. Son, "JAM: A Jammed-Area Mapping Service for Sensor Networks," Proceedings of the 24th IEEE International Real-Time System Symposium," pp 286–297, 2003.

[8] Ghada Alnifie, and Robert Simon, "A Multi-channel Defense Against Jamming Attacks in Wireless Sensor Networks," Proceedings of the 3rd ACM workshop on QoS and Security for Wireless and Mobile Networks, pp 95-104, 2007.

[9] S.Halder, A. Ghosal, S. Sur, A. Dan, and S. DasBit, "A Lifetime Enhancing Node Deployment Strategy in WSN," Proceedings of Future Generation Information Technology, LNCS-5899, pp 296-308, 2009.

[10] D. Wang, B. Xie, and D. P. Agrawal, "Coverage and Lifetime Optimization of Wireless Sensor Networks with Gaussian Distribution," IEEE Transactions on Mobile Computing, vol 7, no 12, pp 1444-1458, 2008.

978-1-4577-0787-2/11/$26.00 ©2011 IEEE