International Journal of Engineering & Technology, 7 (3.33) (2018) 309-313

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

A Clique based Identification of Wake-Up Nodes against Reactive Jamming Attacks in Wireless Sensor Networks

Incheol Shin[1]*

[1] *Department of Information Security*
Mokpo National University
Muan, Jeon-Nam, Korea
**Corresponding author E-mail:ishin@mokpo.ac.kr*

Abstract

Although there are many countermeasures and mitigation techniques against jamming attacks proposed in literature, those methods still require excessive computational capabilities to wireless devices. Especially, for the most two well-known approaches, frequency hopping and channel surfing techniques, they necessitate excessive computational resources to overcome the attacks. That is, long-term historical countermeasures against the jamming attacks would cause serious side effects in wireless sensor networks (WSNs). In this paper, we propose novel countermeasure against the reactive jamming attacks, one type of wireless DoS(Denial-of-Service) attacks, by the identification of the wake-up nodes whose wireless signal transmission triggers the activation of the jammers. For the efficient identification of wake-up nodes, we exploit the group testing techniques and clique-based methods. Based on this identification, we further reduce the overall computational overhead in WSNs. Theoretical analysis and simulation result show that our solution can efficiently identify the wake-up nodes, which provides an efficient protective mechanism against reactive jamming attacks.

Keywords: *Wireless Denial-of-Service, Jamming Attacks, Wireless Sensor Network Security, Group Testing.*

1. Introduction

The broadcasting characteristic of the wireless communication bears the vulnerability in the physical layer and link layer. Especially, in the physical layer, the several types of jamming attacks can be easily launched from adversaries and jam wireless communication by producing excessive level of interference signal to either transmitter or receiver of messages without following any legitimate protocols. This interference drastically decreases the probability of successful broadcasting in wireless communication. The jamming attacks are categorized in physical layer attacks because of the signal jamming nature in the layer, so that efficient physical layer solutions could be the best way to prevent for other layers from the modification against the adversarial jamming activities. Existing solutions from lower layers would provide good countermeasures against jamming attacks in the wireless network, however, those are not suitable for resource-limited network environment like sensor network since those methods necessitate lot of computational capabilities all the time. In this paper, an efficient security method against this reactive jamming attack will be presented.

Frequency-hopped spread spectrum communication in [8] and [9] is one of the most well-known techniques against various jamming attacks in the physical layer since the spread spectrum communication is highly resistance to narrowband interferences and eavesdrops without pseudorandom sequences for channel switching. In other words, this method is resistant to the attacks by utilizing rapid shifting a carrier among the multiple frequency channels, and the carrier switching is performed based on pseudorandom sequences. An adversarial jammer has to sniff out the current using channel to achieve its goal to the wireless networks, but the unpredictable frequency hopping makes it hard to follow the frequencies in use.

However, frequency hopping is not suitable for wireless sensor network since it requires an excessive amount of computational resources and messages. Due to the fact that the channel shifting sequence has key role in this mechanism and only secure channel synchronization between transmitters and receivers would complete the defense, there have been abundant researches on how to select secure channel, how to synchronize the channel shifting among sending party and receiving party and how to distribute the sequences. That is, even though frequency-hopping spread spectrum communication has been exploited in many areas already such as cellular network, wireless ad-hoc network and etc, frequency hopping mechanism result a burden on resource-limited sensor nodes in WSN. In addition, it also has limitation against repeater jamming attacks so far, and there have been intensive researches in progress.

Channel surfing in [7] motivated by frequency hopping is also well-known approach for victim nodes to escape from the interference signal. As discussed, frequency hopping is a common physical layer modulation technique against the jamming attacks and affected the link layer evasion mechanism. One more noticeable difference, except the link layer operative defense, is the fact that channel surfing is on demand, which means that channel shifting takes place by requests from the communicative parties when the communication is collapsed. However, because of the similarity of the mechanisms between the channel surfing and frequency hopping, channel surfing has also synchronization problem and in the case of coordinated channel shifting, additional latency problem

among the nodes with old channel and other nodes with new one. For the defense against reactive jamming attack, even though not all the nodes in a jammed area require to perform the channel surfing, existing technique force for every single node in the jammed area to perform the channel surfing. Our approach will show better countermeasure against the reactive jamming attacks over these straightforward methods by identifying wake-up nodes.

Recently, a few of new approaches in [12] have been introduced and JAM (Jammed Area Mapping) is the one of the new approaches. JAM mechanism tried to isolate the possible jammed regions, which consists of a series of jammed nodes, and de-route all the messages around the jammed region. However, JAM approach could quarantine unnecessarily large jammed regions against the jamming attacks and in worst case, isolate networks in parts. The overhead of messages is relatively higher as mapping process then our defensive model. By restricting the wake-up nodes to be only receivers after identifying out of victim nodes, our novel approach can achieve the minimum size of jammed area to be constructed for broadcasting messages from the base station.

In this paper, we develop an efficient countermeasure against the reactive jamming attack in WSNs. As has been mentioned, reactive jamming attack is one of the most difficult attack types to defense due to the sporadic interference signal. However, our protocol can find out all the wake-up nodes among the victim nodes and build new de-routing paths in order to try to avoid the activation of jammers with minimum latency and message overhead by utilizing the group testing theory.

- Existing countermeasures against jamming attack necessitate excessive resources of sensors in WSN, but our defense mechanism reduces the energy consumption of itself and computational overhead among sensors by identifying the wake-up nodes among victim nodes and building de-tour path.
- By applying group testing theory into network area, more surgical defense mechanism can be studied against furious and blast various type of attacks in many network security areas with this beginning.
- Since our work leverages the identification wake-up nodes by using GT, the defensive latencies would be minimized, and the network would not waste their resources, including internal information, like how to assign new channel or channel ranges. This will make harder for the jammer to attack further.

The organization of this paper follows here. In second section, we briefly show the definition of the jamming problem, network model as well as some assumptions and notation. Section III provides preliminary of group testing theory. The improved identification of wakeup nodes is introduced in Section IV, V and VI with algorithms. The simulation results and simulation environment are discussed detail in the section VII. The discussion about our paper takes place in section VIII.

2. Network Model and Problem Definition

Our approach against reactive jamming attacks to cull out the wake-up nodes from the network by utilizing group testing theory and build new virtual detour paths by converting the wake-up nodes into receivers only. Our novel countermeasure remedies only infected wake-up nodes by limiting their transmission functionality and keeps other normal nodes in healthy. Due to the resource-limited sensor nodes in the WSN, it is not appropriate for all the nodes to perform anti-jamming historical defending scheme, such as frequency hopping or channel surfing, constantly. That is, our protocol would provide desirable and robust network framework with any legacy defending mechanism.

The WSN defined from the jamming problem consists of a base station *BS* with a wireless signal transmission range $\rho = \beta r$ where $\beta > 1$ and the number of sensors n with the same transmission range r. The jammer nodes, $J << N$, whose wireless transmission signals travel uniformly distance $R = \alpha r$ where $\alpha > 1$, are distributed in the wireless network. However, positions of those jammers are not known in beforehand with the exception of the fact that all jammers are assumed to be located sparsely to disrupt or jam as large region as possible (Here let's say the overlapping areas have the maximum distance of band *R'* as shown in Figure 1.) and the jammers' locations will not change during our detection. Wireless sensor or jamming device is geared by m radios and k wireless channels ($m < k$).

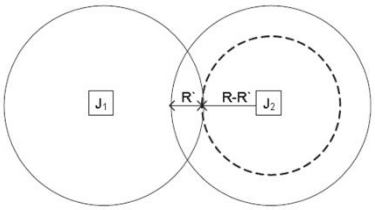

Fig. 1: Overlapped Area

We project the considered network into connected graph $G = (V, E)$ (where $E = \{(u, v) \mid \delta(u, v) \leq r, u, v \in V\}$) with $|V| = n$ nodes, where graph G is a unit disk graph. Any sensors that broadcasting would activate the jamming signal from the jammers are called wake-up nodes, while sensors whose communications are disrupted by the jamming signal are defined as victim nodes. Consequently, the node v is classified into a victim node set if $\delta(J, v) \leq R$ while J means a jamming device. Note that wake-up nodes are also defined as victims when the transmission range of the jammers R is greater than the transmission range of a sensor node, where a victim w is a wake-up sensor if $\delta(J, \omega) \leq r$.

The goal to achieve from the solution against this problem is to identify all the wakeup sensors by the minimum latency and message complexity. Then, a new virtual backbone path would be constructed to mitigate the any activation of the reactive jammers by utilizing Connected Dominating Set (CDS) algorithm.

The explanation on some notations throughout this article is located in the table below.

Symbol	Meaning
r	The transmission range of sensors
R	The noise range of jammers
ρ	The transmission range of basestation
k	The number of wireless channels in WSN
m	The number of wireless radios in WSN
J	The maximum number of jammers in WSN
V	The set of sensor nodes in WSN, ($\|V\| = n$)
W	The set of victim nodes in WSN
Wi	The set of victim nodes in group i
T	The number of Wi groups
U	The set of wake-up nodes in WSN
Ui	The set of wake-up nodes in group i
ωij	The set of victims in subgroup i of group j
ti	The number of ωij groups
d	The number of wake-up nodes
di	The number of wake-up nodes in group i
Di	The upper bound of di
$\delta(u,v)$	The distance between two nodes u and v
$D(u, r)$	The disk with center node u and radius r
$G = (V, E, r)$	The unit disk graph with the radius r
τ	The total testing rounds

3. Preliminaries

This section explains about the *Non-Adaptive Group Testing* to locate the jammers and *Maximum Clique Problem* to bound the

performance, especially how to test and identify nodes in synchronized parallel to minimize the time complexity.

3.1. Clique Problem

The Problem of Maximum Clique is depicted as follows. Where *G(V, E)* is an undirected arbitrary graph, a clique is the sub-graph *G`(V`, E`) (V` ∈ V)* if all its vertices of nodes *v` ∈ V`* are connected pairwise adjacent. Then the maximum clique is one of the cliques with max |V`|. In accordance with M. Bomze and et. al in [1], this clique problem is one of the first one proved to be NP-complete [1].
Boppana and Halldorsson in [1] designed an approximation algorithm with the one of the best polynomial-time performances for this clique problem so far . Furthermore, there have been some more research regarding these issues to develop approximation algorithms on special or arbitrary graphs as well.
Throughout this paper, the algorithms on this maximum clique problem is employed to bound the upper number of wake-up sensors in accordance with the size of reactive jamming nodes. Due to the fact that the jamming signal would be only triggered by the sensors located within an explicit distance range, we define a unit disk graph by all wireless sensors with the radius twice the distance to investigate the bound, the upper number of wake-up nodes.

3.2. Non-Adaptive Group Testing

The identification methods from [3] for the non-adaptive Group Testing (GT) in [2] are designed to reduce the number of testing rounds by sophisticated grouping and testing the suspicious defective items in the pools at the same time, rather than performing individual testing to them. The method to group the items is according to the binary 0-1 matrix $M_{t\times n}$ where the rows indicate the groups to test simultaneously and an item is represented by each column in the matrix. For instance, the implication of *M[i, j]* = 1 is that the *jth* item belongs to the testing group of the *ith* set, and the number of rows is the size of testing to perform the identification. The testing outcome from each group is to be recorded to the testing result vector with the matrix row size *t* - a negative result 0 and positive result of 1. The *M*, *d*-disjunct matrix, plays a key role for minimum testing latency for the non-adaptive GT [3], where any single column is contained by the union of any other *d* columns. Consequently, *O(tn)* decoding and *O(1)* tests are required for the identification.

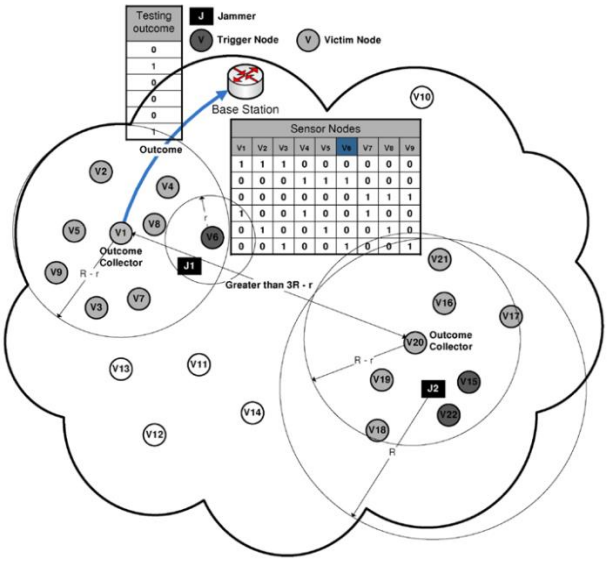

Fig. 2: Non-Adaptive Group Testing

4. Identification of Wake-Up Node Procedure

In order to find out all wake-up nodes and avoid activating the jammers by evading these wake-up nodes in transmission, we leverage the classification of the wireless sensor nodes by using the non-adaptive GT technique, along with clique-based clustering. Assume that there are *n* victims, a jammer *J* and wake-up sensors in the range of disruptive noise from the jammer. Despite *d* is always referred as the number of malice, we use it for denote the number of wake-up nodes. The goal of our approach is to identify all the sensors *d* which activate the jamming signal among victim sensors *n* by the minimum latency of testing. That is, with a given pool of victim nodes, any of which wakes up the reactive jammer by transmitting wake-up testing messages refers to be positive, otherwise negative. Since our model is based on the time synchronized system, each test will be conducted in synchronized sequence without any confliction of testing signals among the testing groups. Figure 3. Briefly illustrates the GT scheme for the identification of the wake-up nodes. The small circles represent the transmission range of victim nodes and *J* is a jammer. In the *d*-disjunct matrix, the first row of the matrix is a testing group composed of three nodes, node 1, node 2 and node 3, and let node 1 be the outcome collector for these victim nodes since node 1 can be sure to sense the interference signal from the jammer. Each collector node will know the matrix and the time slot for each of them to transmit the testing messages. At the first time slot, the nodes in the first group will transmit the signal and node 1 will listen to the noises from the activated jammers. After hearing the noise, node 1 will mark outcome as positive. By decoding the matrix and the final outcome vector, we can determine which nodes are the wake-up nodes.

5. The IWN Approach

In this section, we will propose a novel approach - Identification of Wake-up Nodes (IWN) - against reactive jamming attacks. By identifying all wake-up sensors in the wireless networks, the jamming attacks can be mitigated when these wake-up sensors keep off their transmission of messages.

5.1. Algorithm Description

The whole idea is as follows. To detect all wake-up nodes, firstly this solution investigates all victim nodes *W* out of all sensors in the networks by a single and efficient broadcasting method as described in the DVN-BFS algorithm — Detection of Victim Nodes based on adapted Breadth-First Search tree. Next the testing will be divided into two steps: (1) For the victim nodes *W* in overlapped jamming areas, by using the GVN-MCE algorithm — Group Victim Nodes Based on Maximal Clique Enumeration, we classify all victim nodes into the groups $\{W_1, W_2, \ldots, W_J\}$, where the nodes in different groups do not interfere each other (Any two nodes u and v are called interference-free nodes if any jammers waken up by either node u or v cannot jam the other node). This allows all the groups can be tested simultaneously without interfering the testing result of each other. Then for each group in *Wi* of victim nodes, we develop the DWN-NCGT algorithm — *Detection of Wake-up Nodes based on Non-Adaptive Combinatorial Group* Testing to utilize the Group Testing technique to efficiently identify all wake-up nodes in a short period of time. (2) For the victim nodes in overlapped jamming areas, we just use individual testing since the number of these leftover nodes is very limited. The whole algorithm can be described as the following algorithm below.

Algorithm 1 The IWN Algorithm

1: Input: WSN $G(V, E)$
2: Output: The set of wake-up nodes U
3: $W \leftarrow \{\}$
4: $U \leftarrow \{\}$
5: {Detect Victim Nodes}
6: W ← Victims nodes based on the DVN-BFS algorithm
7: {Group Victim Nodes}
8: Wi ← Groups of victim nodes based on the GVN-MCE algorithm
9: {Identify Wake-up Nodes for each group at the same time}
10: Ui ← Wake-up detected in each group Gi based on the DWN-NCGT algorithm
11: {Test the leftover nodes by individual testing}
12: $U \leftarrow \cup \{Ui \cup UI\}$ {UI is the set of wake-up nodes in leftover victim nodes}
13: return U

In the remainder of this section, the algorithm DVN-BFS, GVN-MCE, DWN-NCGT will be introduced in next section respectively.

1) The DVN-BFS Algorithm (Algorithm 2): The a*dapted Breadth-First Search tree* based DVN-BFS algorithm would be able to effectively identify all $|W|$ victim sensors among the sensors n in the given sensor network by limited ACK messages. This works as follows: The n-dimension broadcast message mg is represented where each sensor $v \in V$ has an entry in the message, where $|V| = n$. The mg message is broadcasted through the network from the base station BS to all sensors n along with a BFS tree out of the network. Once a wireless device in the network receives the message, it sets 1 on the its corresponding entry in mg if the node listens any one of the channels is interfered by jammers. Then hop to another secure communication channel to send out the broadcast message to the next sensors. After all, the BS obtains a set of messages of MG from all other leaf sensors in the network. In this case, the size of ACK messages is only the number of leaves in the BFS tree. Finally, the BS classify all the victims W by investigating the union MG of all broadcast message mg (i.e., the entry i is 1 iff there is a message $mg \in MG$ with the entry i equal to 1, where $1 \le i \le n$.). $MG \times M$ is the set of victim nodes, by constructing a diagonal matrix $M = diag(1, 2, \dots, n)$,

Algorithm 2 The DVN-BFS Algorithm

1: Input: WSN $G(V, E)$
2: Output: All victim nodes W
3: $mg \leftarrow \{\}$
4: Calculate the BFS tree on G
5: Construct $\{mg1, \dots, mgk\}$ for each path on BFS {The number of leaf sensors on BFS is k}
6: {Calculate W}
7: $mg \leftarrow \cup(i=1\dots k)\ mgi$
8: Calculate a diagonal matrix $M \leftarrow diag(1, 2, \dots, n)$
9: $W \leftarrow mg \times M$
10: return W

2) The GVN-MCE Algorithm (Algorithm 3): After identifying all victim nodes, we devise the GVN-MCE algorithm based on *Maximal Clique Enumeration* to divide victim nodes into groups $\{W_1, W_2, \dots, W_J\}$. The testing of nodes in different groups do not interfere with each other so that all these groups can be tested simultaneously. The algorithm works as follows. We construct a unit disk graph $G = (W, E, 2(R-R`))$, where $W$ are all victim nodes and $E$ are all links between victim nodes if the distance between two nodes are less than or equal to $2(R - R`)$. In this case, all victim nodes in the same noise range of each jammer will construct a clique in graph G and each clique is one group W_i of victim nodes. We then employ Gupta's MCE algorithm [4] on unit disk graph G to find out all Maximal Cliques.

Algorithm 3 The GVN-MCE Algorithm

1: Input: All victim nodes W
2: Output: The set of wake-up nodes U
3: The groups $\{W1, W2, \dots, WJ\}$ of victim nodes W
3: Construct a graph $G = (W, E)$, $E = \{(u, v) | \delta(u, v) \le 2(R - R`)\}$
4: Find all maximal cliques by using Gupta's MCE algorithm [4]
5: Divide W into groups based on the cliques
6: return $\{W1, W2, \dots, WJ\}$

3) The DWN-NCGT Algorithm (Algorithm 4): For divided victims in each group W_i, we devise the DWN-NCGT algorithm by Non-Adaptive Combinatorial Group Testing scheme to identify all wake-up sensors. In this algorithm, we construct D_i-disjunct matrix to further divide victim nodes into groups $\{W_{i1}, W_{i2}, \dots, W_{i|t_i|}\}$ in each group W_i (where $|t_i|$ is the number of groups divided for victim nodes in group W_i) based on the estimated upper bound D_i of wake-up nodes d_i. To estimate the upper bound D_i of wake-up nodes d_i in each group W_i, we consider constructing a unit disk graph $G_i = (W_i, E_i, 2r)$ for each group W_i (where $E_i = \{(u, v) | \delta(u,v) \ 2r, u, v \in W_i\}$. Then similarly we find out all maximal cliques on G_i.

Algorithm 4 The DWN-NCGT Algorithm on Group Wi

1: Input: The set of victim nodes in group Wi, R, r
2: Output: Wake-up nodes Ui in this group ($|Ui| = di$)
3: {Find the upper bound Di of di}
4: Construct $Gi(Wi, Ei)$, $Ei = \{(u, v) | \delta(u, v) \le 2r, u, v \in Wi\}$
5: Find all maximal cliques by using Gupta's MCE algorithm [4]
6: {Test by using Non-Adaptive Group Testing}
7: Construct a Di-*disjunct Matrix* Mi
8: For each row, group the nodes with the entity 1
9: Test these groups on different channels
10: Decode the testing result to find out all wake-up nodes Ui
11: return Ui

6. The Improved IWN Approach

6.1. The Improved GVN-MCE Algorithm

Recall the GVN-MCE algorithm in section VI-A2, we construct a unit disk graph $G = (W, E, 2(R - R`))$ and simply employ Gupta's MCE algorithm [4] to find all maximal cliques in it. To reduce the running time of the GVN-MCE algorithm, we notice that the graph G is composed of disjoint cliques. Therefore, instead of iterating all edges to find out all maximal cliques in Gupta's MCE algorithm [4], we just need to iterate one edge in one clique. By using this improved Gupta's MCE algorithm for disjoint cliques on G, the running time can be decreased.

6.2. The Improved GVN-MCE Algorithm

Recall the DWN-NCGT algorithm in the previous section, we construct D_i-disjunct matrix to test the nodes in group W_i, where D_i is the upper bound of wake-up d_i in W_i. The estimate D_i is simply equal to the sum of the 5 largest size of maximal cliques. To tighten the upper bound D_i of d_i, the idea is as follows. After finding out all maximal cliques $\{c_k\}$ on G_i, we use a simple greedy algorithm — the DMCE algorithm to find out the maximum number of disjoint maximal cliques as described in Algorithm 5. By testing these disjoint cliques to see how many of them have positive results (contain wake-up nodes in it), let's say there are l cliques with positive results, the upper bound D_i of wake-up nodes d_i will be only the sum of the first l size of maximal cliques.

Algorithm 5 The Greedy DMCE Algorithm

1: Input: Graph G
2: Output: The set of the maximum number of disjoint maximal cliques $\{dci\}$
3: Find out the set of all maximal cliques C by using Gupta's MCE algorithm [4]
4: while $\exists A$ maximal clique in C do
5: Choose dci the clique ck with the maximum size in C
6: Remove all nodes and incident edge in ck from graph G
7: end while
8: return $\{dci\}$

7. Simulation Results

In these tests, there are sensors N = 1000 with wireless radio channel m = 3, on the size of 1500 × 1500 network with jammers by J=5, R`=30 and α=3 as default parameters where each graph has a varied range of parameters from one of these values.

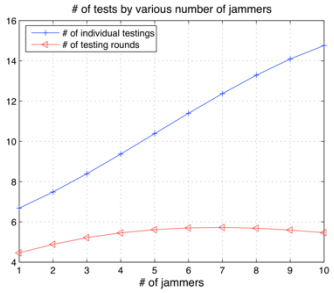

Fig. 3: # of Testing Rounds by Jammers

1) Performance by the number of jammers J.: Our work minimizes the defensive computational overhead by maximizing the parallel testings as Algorithm 3 and Algorithm 4 have mentioned. Even in the case that multiple jammers are deployed into the network, those algorithms divide victim nodes into groups efficiently to perform the parallel identification of wake-up nodes. For the graph of Figure 3, it shows stable trend of the testing rounds, where $J \in$ [1, 10]. This security method explores a sophisticated scheme to conduct as many testings as possible simultaneously as shown, consequently the size of tests, T, has a steady curve in the duration of increment in the number of jammers J and victim nodes n in accordance with Figure 3. Due to the sporadic adversarial behaviour of the reactive jammers, existing countermeasures have not shown

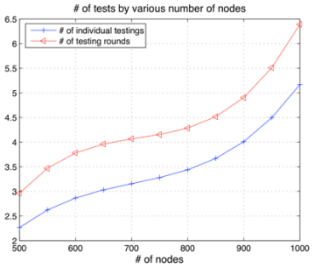

Fig 4: # of Testing Rounds by Nodes

effective defending mechanisms against multiple reactive jammers. However, our approach can promptly act against the multiple reactive jamming attacks.

2) Performance by the number of nodes n.: The size of sensors in the WSN is one of the important issues to support the scalability of the network. In the case of channel surfing, it suffers from the large number of messages between the sensors or between sensors and the base stations since they need to synchronize their safe channels by exchanging messages. However, our approach would keep stable number of message and have less burden from the message overhead even in increasing number of nodes, $n \in$ [500, 1000] , from Figure 4. Constant number of messages only during testing mode implies the fact that our approach is more suitable for WSN than other approaches since messages for synchronization in channel surfing or distribution hopping sequences in frequency hopping technique are additional continuous communicational overhead which needs to be reduced for the longer network lifetime. After the identification procedure, no additional message overhead is required in our approach, which economizes the communicational complexity further.

8. Conclusion

Throughout this work, we introduce an efficient security countermeasure against the reactive jamming attack to wireless sensor networks. By defining and identifying the wake-up sensors whose transmission activates the disruptive signal from jammers, we would be able to support the resilience and robustness of the network against the lethal wireless DoS attack.

Acknowledgement

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning(2017 R1C1B5018413)

References

[1] M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo. The maximum clique problem. *In Handbook of Combinatorial Optimization*, pages 1–74. Kluwer Academic Publishers, 1999.
[2] D.-Z. Du and F. Hwang. Combinatorial Group Testing and its Applications (2nd ed.). *World Scientific*, Singapore, 1999.
[3] D.-Z. Du and F. Hwang. Pooling Designs: Group Testing in Molecular Biology. *World Scientific*, Singapore, 2006.
[4] O. G. Rajarshi Gupta, Jean Walrand. Maximal cliques in unit disk graphs: Polynomial approximation. *http://walrandpc.eecs.berkeley.edu/Papers/RG CliqueUDG.pdf.*
[5] M. Thai and D.-Z. Du. Connected dominating sets in disk graphs with bidirectional links. *Communications Letters*, IEEE, 10(3):138–140, Mar 2006.
[6] S. S. J. Bellardo. 802.11 Denial-of-Service attacks: real vulnerabilities and practical solutions. In Proceedings of the 12th *conference on USENIX Security Symposium*, 2003.
[7] Q. Ling, J. Ren, and T. Li. Message-driven frequency hopping design and analysis. In WASA '08: *Proceedings of the Third International Conference on Wireless Algorithms, Systems, and Applications*, pages 373–384, Berlin, Heidelberg, 2008. Springer-Verlag.
[8] O. Sidek and A. Yahya. Reed solomon coding for frequency hopping spread spectrum in jamming environment. *American Journal of Applied Sciences*, 5(10):1281–1284.
[9] M. Strasser, C. P¨opper, S. Capkun, and M. Cagalj. Jamming resistant key establishment using uncoordinated frequency hopping. *IEEE Symposium on Security and Privacy*, May 2008.
[10] J. Y.-C. H. Chiang. Dynamic jamming mitigation for wireless broadcast networks. *INFOCOM*, 2008.
[11] Y. Desmedt, R. Safavi-Naini, H. Wang, C. Charnes, and J. Pieprzyk. Broadcast anti-jamming systems. Networks, 1999. (ICON '99) *Proceedings. IEEE International Conference on Networks*, pages 349–355, Sept.-1 Oct. 1999.
[12] S. S. A. D. Wood, J.A. Stankovic. A jammed-area mapping service for sensor networks. *Proceeding. 24th IEEE Intl. Real-Time System Symposium*, pages 286–297, 2003.