ISSN 1846-6168
UDK 681.596:004.021+531.5

DISTRIBUIRANI OBRAMBENI MEHANIZMI ZA CLONE NAPADE TEMELJENI NA ALGORITMU ZA ISTRAŽIVANJE GRAVITACIJE (GSA) U WSN

DISTRIBUTED DEFENSE MECHANISM FOR CLONE ATTACKS BASED ON GRAVITATIONAL SEARCH ALGORITHM (GSA) IN WSN

P. Uma Maheswari, P. Ganesh Kumar

Izvorni znanstveni članak

Sažetak: *Bežične senzorske mreže (WSN) često su raspoređene u neprijateljskom okruženju i ranjive su na napade zbog prirode senzora koji su tehnološki ograničeni. Clone napad u WSN jedan je od glavnih problema gdje se poruke prisluškuju, zarobljeni čvor se klonira te napadač proizvede višestruke čvorove istog identiteta. Kako bi nadvladali te probleme, ovaj rad predlaže* distribuirani obrambeni mehanizam *za clone napade temeljen na algoritmu za istraživanje gravitacije (GSA) u WSN. Kako bi se sumnjivi čvorovi efikasno detektirali, čvorovi u kanalu mogu se podijeliti u čvorove svjedoke i tražene čvorove. Čvorovi svjedoci odgovorni su za otkrivanje sumnjivih čvorova, dok traženi čvorovi trebaju za potrebe procesa detekcije navesti svoj identitet. Za izbor čvorova svjedoka, koristi se GSA kako bi se izabrala grupa čvorova koji su najprikladniji. Nakon izbora čvorova svjedoka, otkivanje clone napada vrši se promatranjem ponašanja susjednih čvorova. Otkrivanjem clone napada aktivira se proces opoziva kako bi se opozvao clone napad u čvorovima svjedocima. Prema rezultatima dobivenim iz simulacije može se zaključiti kako predloženi algoritam pruža bolju zaštitu od clone napada smanjivanjem odbacivanja paketa i povećavanjem omjera isporuke paketa.*

Ključne riječi: *GSA, DDM, WSN, clone napadi, otkrivanje, zakon gravitacije, zakon gibanja*

Original scientific paper

Abstract: *Wireless Sensor Networks (WSN) are often deployed in hostile environment and are vulnerable to attacks because of the resource constrained nature of the sensors. Clone attack in WSN is one of the major issues where the messages are eavesdropped, the captured node is cloned, and multiple nodes with same identity are produced by attacker. In order to overcome these issues, in this paper, a Distributed Defense Mechanism for Clone Attacks based on Gravitational Search Algorithm (GSA) in WSN is proposed. For efficiently detecting the suspect nodes, the nodes in the channel can be divided into witness node and the claimer node. The witness nodes are responsible for the suspect nodes detection, whereas the claimer nodes should provide their identities for the detection process. For the witness nodes selection, we utilize the GSA to pick out the best witness nodes set. After selecting the witness nodes, clone attack detection is performed by observing the behavior of the neighbor nodes. On detecting the clone attack, revocation procedure is triggered to revoke the clone attack in the witness nodes. By simulation results, it can be concluded that the proposed algorithm provides better protection to clone attacks by reducing the packet drop and increasing the packet delivery ratio.*

Keywords: *GSA, DDM, WSN, Clone attacks, Detection, law of gravity, law of motion*

1 INTRODUCTION

Wireless Sensor Networks (WSNs) is a network formed by a large number of distributed autonomous nodes with sensors, embedded processor, and low-power radio to enable wireless communication with each other and also with the base station [1]. Sensors are resource-constrained tiny devices with limited memory storage capacities, low computation capacities, and reduced energy supply, which can neither be charged nor be replaced [2, 3]. Sensor nodes can sense the environmental changes, that is, physical, mechanical or chemical changes, and transfer it to base station for user analysis. Signal can be processed, computed, and aggregated before forwarding data to base station in order to reduce both communication and energy costs. WSN can be used for collecting and monitoring data. It can be used in military applications, environmental applications, smart homes, health monitoring, and so forth [2, 4, 5]. Yet, sensor nodes are prone to various attacks because of limited computing resource and feeble wireless communication. Attacks in routing protocol make the network unstable. Limited processing power and resources may impose difficulty in utilizing defense mechanisms of wired networks in wireless networks [4].

Data security is devastated by threats, vulnerabilities, and attacks, which are the three crossly related entities that intend to bypass the security control of the system. In addition, there are active and passive attacks [1]. As sensors are unshielded devices, WSNs are prone to various types of novel attacks such as compromising attacks, intrusion attacks, deny-of-service (DoS) attacks, and nodes replication attacks [6, 7]. Several existing attacks in WSNs are physical attacks, sybil attacks, clone attacks, sinkhole attacks, and replication attacks [1, 3, 4, 6].

Clone attack is also called as replication attack in which the exchanged messages can be eavesdropped, and nodes can be captured by an attacker, thereby obtaining all information stored in devices. Then, the captured nodes can be cloned, and multiple nodes are generated by the attackers with same identity. The clones could be then deployed in network area to launch a variety of malicious activities [6, 7].

A self-healing efficient protocol, which is randomized and distributed, is presented for detecting node replication attacks [7]. Here, a random value is distributed among all the nodes using a centralized or distributed mechanism. A digital signature is then added by each node, and its claim ID and geographical location are broadcasted. The claim is not sent directly to the specific node instead transmitted to the node that is closest to the location. Moreover, the signature check will be carried out only in destination, but not in forwarding nodes. Here, the witness has to be captured faster within a window period so as to prevent clones in detecting the witness of clones.

However, this scheme focuses more on exhausting battery of nodes and verifies the received signature and the message freshness, but it failed to detect the attacks. Moreover, the witness selection procedure is not accurate and consumes more time. Most

of the existing mechanisms that detect clone attacks are not efficient and also, they did not provide any approaches to mitigate the attack.

In this paper, we propose a Distributed Defense Mechanism for Clone Attacks in WSN based on GSA where the witness nodes are used to detect the clone attacks. The witness nodes periodically broadcasts request messages with a time stamp to the claimer nodes. GSA is used to select the witness nodes in the network. Clone attack detection is performed by observing the behavior of the neighbor nodes. Then, revocation procedure is triggered by flooding the network with two incoherent response messages received by the witness node. This cancels the clone attacks in the witness nodes. This algorithm provides better protection to clone attacks with reducing packet drop and increased packet delivery ratio.

The paper is organized as follows. Section 2 presents the literature review of existing works on clone attacks in WSN. Section 3 presents the proposed methodology of distributed defense mechanism for clone attacks. The simulation results and analysis are presented in section 4 and the conclusion is given in section 5.

2 LITERATURE REVIEW

Mauro Conti et al [7] have proposed a new self-healing, Randomized, Efficient, and Distributed (RED) protocol for the detection of node replication attacks. This protocol is highly efficient in communication, memory, and computation; and is much more effective than competing solutions in the literature; also resistant to the new kind of attacks. RED is more resilient in its detection capabilities. However RED is more influenced by path lengths, since a malicious node can stop the protocol wherever it appears in the paths.

Mebi Sernaz and Anand Pavithran [8] have proposed risk aware mitigation scheme for the isolation of clone nodes and black hole nodes. It provides better response with adaptive isolation and the response cost calculation when IDS inform it with an alert confidence value. By combining Time Domain Detection and Space Domain Detection (TDD and SDD) with risk aware mitigation, a good detection and response mechanism can be developed for the detection of clone nodes in MANET.

H. Wen et al [9] have proposed a novel scheme to detect the node clone attack in WSN by channel identification characteristic in which the clone nodes are distinguished by the channel responses between nodes. The proposed scheme aims at achieving fast detection and minimizing the data transmission cost by taking advantage of temporal and spatial uniqueness in physical layer channel responses. The proposed approaches feature nearly-perfect resilience to node clone attack with low communication and computation costs, low memory requirements and high detection probability. However the detection rate falls with increase in the velocity.

Zhongming Zheng et al [10] have proposed a location-aware energy-efficient ring based clone detection protocol with clone detection protocol, which guarantees successful clone attack detection and has little negative impact on the network lifetime. Moreover, their proposed protocol can significantly improve the network lifetime, compared with the existing approach. The proposed protocol can approach 100% clone detection probability with trustful witnesses. However the energy consumption is high.

Richard Brooks et al [13] have proposed an algorithm to detect the presence of clones in the sensor network. Keys on the cloned nodes are detected by looking at how often they are utilized to authenticate the nodes in the network. The system can recover from a cloning attack by terminating connections with the help of cloned keys. Results show that this method can accurately detect the presence of clones in the system and support their removal. Moreover, the extent of false positives and false negatives in the clone detection process is qualified. However, the key encryption technique requires more computation and resource consumption.

Zhijun Li, and Guang Gong [14] have propose two node clone detection protocols with different tradeoffs on network conditions and performance. The first protocol depends on a distributed hash table (DHT) that is used to construct a fully decentralized, key-based caching and checking system. This protocol is capable of catching the cloned nodes effectively. As DHT-based protocol increases the communication cost, the second distributed detection protocol is proposed that provides good communication performance for dense sensor networks, by a probabilistic directed forwarding technique with random initial direction and border determination. These protocols can obtain better efficiency on communication overhead and satisfactory detection probability along with minimal storage consumption.

Yingpei Zeng et al [15] have proposed two new NDFD protocols, RAndom WaLk (RAWL) and Table-assisted RAndom WaLk (TRAWL) based on random walk. The random walk strategy performs weel as it distributes a core step, the witness selection, to every passed node of random walks. Hence, the adversary cannot easily determine the critical witness nodes. The number of walk steps needed for ensuring detection is theoretically analyzed. These protocols have the lowest overheads in witness selection, whereas the communication overheads of these protocols are higher but are affordable due to their security benefits.

Conti et al. [16] have proposes two distributed, efficient, and cooperative protocols to detect replicas. They are history information-exchange protocol (HIP) and HIP's optimized version (HOP). Both HIP and HOP influence local communications and node mobility. It differs for the amount of computation required. By considering two different mobility models, the behavior of these protocols are studied against the introduced types of attacker. Results show that the solutions provide high detection rate while experiencing limited overhead.

Soumya Sara Koshy and Sajitha et al [17] have proposed a zone-based node replica detection in WSN using trust in which the network is divided into a number of zones and the zone leaders are chosen before the node deployment. Hence, the zone is dynamically formed. Trust values are estimated for each node and replica nodes are detected by the zone leader based on these trust values. Before the zone formation, the trust values are checked and updated. Result shows that this technique can improve the packet delivery ratio and reduce the end to end delay.

Mayur R. Khandekar and U. K. Raut [18] have surveyed the node replication attack algorithms in WSN. Several replication attack detection algorithm are analyzed and these algorithms are classified into two schemes, namely, centralized and distributed schemes. Generally, distributed detection scheme is preferred over the centralized detection scheme, even if both the schemes have some disadvantages. The recent research has mainly focused on the communication costs and energy efficiency of the algorithm.

Wibhada Naruephiphat et al. [19] have proposed an area-based clustering detection (ABCD) method for WSN to achieve high successful detecting replica rate with less communication overhead than the line-selected multicast approach. Even though ABCD requires more memory capacity to store location claims in the central node, it can easily support more number of nodes in a network. When the number of sensor nodes in the network increases, ABCD can reduce the number of message stored in a node and preserve the network lifetime.

Lei Jin et al. [20] have proposed a detection framework to discover the suspicious identities and to validate them. For this, two approaches based on attribute similarity and similarity of friend networks are utilized. The first approach considers the scenario in which mutual friends in friend networks are taken into account, whereas the second approach captures the scenario in which similar friend identities are involved. The suspicious identities are finally validated. Results demonstrate the flexibility and effectiveness of these approaches.

Wazir Zada Khan et al. [21] have considered a node replication attack or clone node attack in which an adversary can create its own clone nodes and misinform the network to acknowledge them as legitimate nodes. The existing attack detection techniques are broadly categorized into distributed and centralized classes. The algorithms in both classes are capable of detecting and preventing clone attacks with some noteworthy shortcomings. This study analyzes the challenges and issues in clone detection schemes that need to be resolved to become more applicable to real-life situations.

Moirangthem Marjit Singh et al [22] have reviewed the existing node replication attack detection techniques for static WSNs. These techniques can be classified into two categories, namely, location aware protocol and location independent protocol. In location aware protocols, the nodes should know their geographic location. Hence, it requires GPS or relies on the base station to compute their location coordinates. In location independent protocol, the knowledge of the nodes' location is not required to detect node replication. The performance of these techniques depends on the density of the networks.

3 PROPOSED METHODOLOGY

3.1. Overview

In this paper, we propose a Distributed Defense Mechanism for Clone Attacks in WSN based on Gravitational search algorithm where the nodes are divided into two groups, namely, witness nodes and claimer nodes [9]. The witness nodes periodically broadcasts request messages with a time stamp to the claimer nodes. On receiving the messages, these claimer nodes send the response message with in a time interval to the witness node. The response message contains the node ID and the pilot. With this, the suspected nodes are detected, and they are stored in the suspected list. In order to confirm the clone attacks, the suspect list is broadcasted to the claimer nodes. Hence, after receiving the broadcasted message, the claimer nodes send the response message which contains node ID, sequence, and time stamp. If the two nodes have the same ID and different random sequence, then the nodes are considered to be attacked.

For selection of witness nodes in the network, GSA can be used that tends to find the global optimum faster than other algorithms with higher convergence rate. Gravitational constant adjusts the accuracy of the search, thereby decreasing with time. As GSA is a memory-less algorithm, it works efficiently like the other algorithms with memory. This shows the good convergence rate of the GSA [11]. After the detection of the clone attack, revocation procedure [7] can be used. Here, the revocation is triggered by flooding the network with two incoherent response messages received by the witness node. This cancels the clone attacks in the witness nodes.

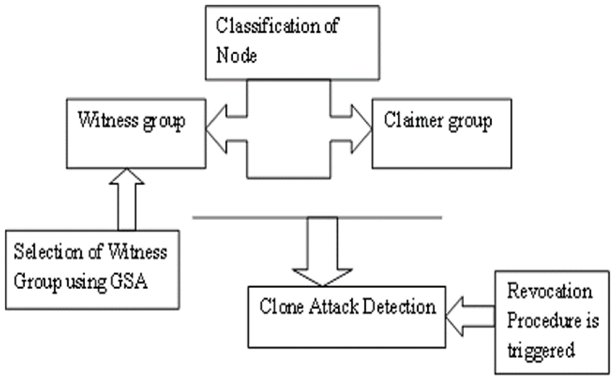

Figure 1. Block Diagram

3.2. Classification of Nodes

For effective identification scheme, all nodes are divided into two main groups: witness group and claimer group in which the nodes are represented as n_j^w and n_j^c, respectively. The nodes in the network channel periodically play these two roles.

Here, the witness node periodically transmits request messages M_w to claimer node with time stamp S_w. As a result, all neighboring claimer nodes, present within the signal range of the witness nodes, receive the previous request messages M_w . In return, the claimer node transmits the return response messages M_c to the witness nodes within time interval $\Delta d + \Delta \varepsilon$, where Δd represents response delay and $\Delta \varepsilon$ must be less than the network channel coherence time $\Delta \Gamma$.

The response message M_c can be represented in the following format:

$$M_c : \{Node\, ID, Pilot\} \tag{1}$$

Once witness node receives the messages M_c, then they extract the channel measure $\underset{-\,t_j}{\hat{B}}\ (n_j^c \rightarrow n_l^w)$, here j = 1, 2,……., H_c , $l = 1, 2, \ldots\ldots, H_w$, where H_c and H_w represents the number of nodes present in claimer group and witness group, respectively.

After that, they perform comparison, we normalize the likelihood ratio test (LRT) statistic as $\Theta_{j,k}^l$, which is given below.

$j \neq k \neq l$, j = 1, 2,……., G_c , $l = 1, 2, \ldots\ldots, G_w$.

$$\Theta_{j,k}^l = \frac{N_{co} \parallel \underset{-\,d_1}{\tilde{B}}\ (ol_j \rightarrow n_l) - \underset{-\,d_2}{\tilde{B}}\ (ol_k \rightarrow n_l) \parallel^2}{\left\| \underset{-\,d_1}{\tilde{B}}\ (ol_k \rightarrow n_l) \right\|^2} \begin{cases} < B_1 \\ > B_0 \end{cases} \gamma \tag{2}$$

where $\underset{-\,d}{\hat{B}}$ denotes the channel response with measurement errors.

N$_{co}$ represents the normalization factor and assume the threshold $\gamma \in [0,1]$.

3.3. Selection of Witness Node

In order to detect the clone attack, selection of the best and authenticated witness node is a major challenge. In the proposed scheme, the selection of the witness node is performed using Gravitational Search Algorithm (GSA), which is described in the following section.

3.3.1. Gravitational Search Algorithm (GSA)

This section describes about GSA optimization algorithm which is based on the law of gravity. Gravitational search algorithm (GSA) is a new optimization algorithm that depends on the law of gravity given by Newton: "Every particle in the universe attracts every other particle with a force that is directly proportional to the product of their masses and inversely proportional to the square of the distance between them." In GSA, agents are considered as objects, and their performance is estimated by their masses. All these objects attract each other by the gravitational force, and this force results in global movement of all objects towards the object which comprises heavier masses. Thus, masses assist using a direct form of communication based on gravitational force.

In GSA, each agent mainly has the following four specifications: position, inertial mass, active gravitational mass, and passive gravitational mass. The position of the mass gives the solution of the problem, and its gravitational and inertial massed are found with the help of fitness function. Or, in other words, each mass represent a solution and the algorithm is directed by properly regulating the gravitational and inertia masses. By elapse of time, it is assumed that masses be attracted by the heaviest mass and this mass is the optimal solution. The GSA need to be considered as an isolated system of masses.

In the proposed algorithm, heavy masses corresponds to good solution as it moves more slowly than lighter ones, assuring development step of the algorithm. Moreover, it mainly obeys Newtonian laws of gravitation and motion which is described as below:

Law of Gravity

Each particle attracts every other particle with a gravitational force, where gravitational force is directly proportional to product of their masses and inversely proportional to the distance between them. In GSA, distance is considered as D instead of D^2 to get better experimental result.

Law of Motion

The new current velocity of any mass is equivalent to the sum of the fraction of its previous velocity and variation in the velocity. Variation in the velocity called as acceleration of any mass can be defined by force acted on the system divided by mass of inertia.

Consider a system with P agents (masses). The position of the *i*th agent can be defined as below:

$$Y_i = (y_i^1, \ldots\ldots, y_i^e, \ldots\ldots, y_i^n \text{ , for i = 1,2,…,P} \tag{3}$$

where, y_i^e represents the position of the *i*th agent in *e*th dimension.

At a particular time 's', force acting on mass i from mass k is given as below:

$$F_{ik}^e(s) = G(t)\frac{N_{pi}(s) \times N_{ak}(s)}{D_{ik}(s) + \eta}(y_k^e(s) - y_k^e(s)) \tag{4}$$

where N_{ak} denotes the active gravitational mass related to agent k. N_{pi} represents the passive gravitational constant at time s. η represents small constant. D_{ik} (s) is the Euclidian distance between two agents (masses) i and k:

$$D_{ik}(s) = \| Y_i(s), Y_k(s) \|_2 \tag{5}$$

In order to give a stochastic characteristic to the GSA, it is assumed that the total that acts on agent i in a dimension e to be an arbitrarily weighted sum of eth component of the forces applied from other agents:

$$F_i^e(t) = \sum_{k=1, k\neq i}^{P} rand_k F_{ik}^e(s) \tag{6}$$

where $rand_k$ is a arbitrarily number in the interval [0,1]. Hence, according to the law of motion, the acceleration of the agent i at a particular instant time t, and in direction dth, $a_i^e(t)$ can be given as below:

$$a_i^e(t) = \frac{F_i^e(t)}{N_{ii}(t)} \tag{7}$$

where N_{ii} represents the inertial mass of the ith agent.

In addition, the next velocity of an agent is measured as a fraction of its current velocity additional to its acceleration:

Hence, its position and velocity can be computed as below:

$$v_i^e(s+1) = rand_i \times v_i^e(s) + a_i^e(s) \tag{8}$$

$$y_i^e(s+1) = y_i^e(s) + v_i^e(s+1) \tag{9}$$

where $rand_i$ denotes uniform random variable in the interval [0, 1]. This random variable is used to give a randomized characteristic to the search.

The gravitational constant, G, is initialized at the starting and is decreased with time to control the search accuracy. Otherwise, G is a function of the initial value G_0 and time (s):

$$G(s) = G(G_0, s) \tag{10}$$

Gravitational and inertia masses are simply computed by the fitness evaluation. A heavier mass represents more efficient agent, which means better agents have higher attraction level and walk more slowly. By assuming the equality of gravitational and inertia mass, the values of masses are computed using map of fitness. Hence, the gravitational and inertial mass is updated using the following equation:

$$N_{ai} = N_{pi} = N_{ii} = N_i \text{, i = 1, 2 ,…,P} \tag{11}$$

$$n_i(s) = \frac{fit_i(s) - worst(s)}{best(s) - worst(s)} \tag{12}$$

$$N_i(s) = \frac{n_i(s)}{\sum_{k=1}^{P} n_k(s)} \tag{13}$$

Where $fit_i(s)$ denotes the fitness value of the agent i at time s, and

$$best(s) = \begin{cases} \min\limits_{k\in\{1,\ldots\ldots P\}} fit_k(s) & \text{minimization problem} \\ \max\limits_{k\in\{1,\ldots\ldots P\}} fit_k(s) & \text{maximization problem} \end{cases} \tag{14}$$

$$worst(s) = \begin{cases} \max\limits_{k\in\{1,\ldots\ldots P\}} fit_k(s) & \text{minimization problem} \\ \min\limits_{k\in\{1,\ldots\ldots P\}} fit_k(s) & \text{maximization problem} \end{cases} \tag{15}$$

In order to perform a good comprise between exploration and exploitation, it is important to reduce the number of agents with elapse of time in equation (7). Here, only a set of agents with bigger mass apply their force to other. However, the policy needs to be used carefully as it may reduce the exploration power and enhance the exploitation abilities. In order to avoid the trapping in a local optimal, the algorithm use exploration at the commencement. By elapse of iteration, exploitation must fade in and exploration must fade out. To enhance the performance of GSA by controlling exploitation and exploration only the J_{best} agents attracts the others.

J_{best} is mainly a function of time, with initial value J_0 at the commencement and decreases with time . At the beginning, all agents apply the force; as time moves, J_{best} is decreased linearly and at the end, there will only one agent applying force to the others. Hence, equation (6) can be modified as:

$$F_i^d(s) = \sum_{k \in J_{best}, k \neq i} rand_k F_{ik}^d(s) \tag{18}$$

where J_{best} represents set of first J agents with the best fitness value and biggest mass. The different steps of the proposed algorithm can be described as below:

1. Search space identification
2. Randomized initialization
3. Fitness computation of agents
4. Update G(s), best(s), worst(s), and $N_i(s)$ for i = 1, 2… P
5. Computation of total force in different direction
6. Computation of acceleration and velocity
7. Update agents' exact position
8. Repeat Step 3 to 7 till stop criteria is reached
9. End

Hence, the agent with the best fitness value is selected as a witness node to detect the suspect node for clone attack.

3.4. Clone Attack Detection

Once, the witness node is selected, clone attack detection starts by observing the behavior of the neighbor node. Here, we have considered three scenarios for malicious node detection, as shown in Fig. 2.

Scenario 1: As in figure 2(a), two or more malicious node surrounds a witness node. Node oa_2 represents the captured node, and ol_2 represents the cloned node from oa_2 . The witness node here is, n_2^w . Lines shows the nodes surround n_2^w .

In case, $|d_j - d_k| \leq \Delta\varepsilon$ and $\Theta_{j,k}^l < \gamma$ for nodes oa_2 and ol_2, n₂ can then conclude that nodes oa_2 and ol_2 are suspects nodes. Hence, these suspects' nodes are saved into $susplist_{n_2^w}$. After that, n_2^w transmits $susplist_{n_2^w}$ to the network and ejects the clone node.

Scenario 2: As shown in figure 2(b), two or more malicious node belong to different witness nodes and here the witness nodes share part of their neighboring nodes.

Scenario 3: As shown in figure 2(c), two or more malicious nodes belong to the distinct witness nodes and the witness nodes are far from each other and hence they do not share any of their neighbor nodes. In both second and third situations, witness node is unable to find the suspect nodes based on performance comparison according to equation (2). Hence, each and every witness nodes transmits a list of its neighbor node to other witness nodes.

Once, a witness node n_l^w receives the list of their neighbor nodes from other witness nodes, it verifies all the list and save the witness ID that consist of at least one same notes ID with it into same node ID list $snIDlist_{n_l^w}$. Moreover, random sequence is transmitted to find the suspect nodes. Hence, all witness nodes present in $snIDlist_{n_l^w}$ transmits randomly selected sequences $sq_{n_l^w}$ to their neighboring node and requests all their neighbor nodes to reply back the received sequences along with their ID. The transmitted random sequences are different in different witness nodes. The response of the neighbor node can be

$$M_s : \{NodeID, sq_{n_l^w}\} \tag{19}$$

After that, every witness node transmits $report_{n_l^w}$ that consist of neighboring nodes' M_s to other witness nodes. Then, the witness nodes verifies the $report_{n_l^w}$ .In case, two nodes with same ID holds distinct random sequence, it indicate clone node attack occurred.

Consider, figure 2(a) where node n_2 denotes the witness node and the other seven nodes are the neighbor nodes, in which nodes oa_2 and ol_2 comprised of same ID. First, node n_2 transmits request message M_w . Once neighbor node received M_w , they reply back by transmitting response message to n_2, in which responses of node oa_2 and ol_2 are $M_o^{oa_2} = \{oa_2, pilot\}$ and $M_o^{ol_1} = \{ol_1, pilot\}$ within response time interval respectively. The node n_2 obtains channel response $\hat{\underset{-\ d_2}{B}}(oa_2 \rightarrow n_2)$ and $\hat{\underset{-\ d_2}{B}}(ol_2 \rightarrow n_2)$ from the pilot and computes the $\Theta_{oa_2, ol_2}^{n_2}$.

In case, $\Theta_{oa_2, ol_2}^{n_2}$ is larger than threshold γ , node n_2 can easily determine that oa_2 and ol_2 are suspects nodes. After that, oa_2 and ol_2 are saved into $susplist_{n_2^w}$. The node n_2 then transmits the $susplist_{n_2^w}$ and removes oa_2 and ol_2 .

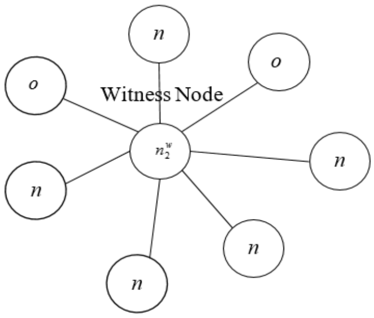

Figure 2(a). Scenario 1: Detection of Malicious Node

Now, we can consider the scenario in Figure.2b. Here, it is assumed that the witness nodes are randomly selected as n$_7$ and n$_{10}$. Their neighbor nodes (illustrated with lines and dashed lines in figure 2(b)) are listed here in Table 1. It is now easier to know that the clone nodes ol_2 and ol_4 belong to different witness node and the witness node is unable to find them in the situation as shown in Figure.2b. Hence, the detection scheme needs to depend on transmitting random sequence in order to find the suspect nodes. In this scenario, witness nodes n$_7$ and n$_{10}$ transmits the randomly selected sequences sq_7 and sq_{10} to their neighbor nodes, respectively and request their neighbor nodes to reply back the received sequences along with their ID:

$$M_s : NodeID, sq_{n_l^w}$$

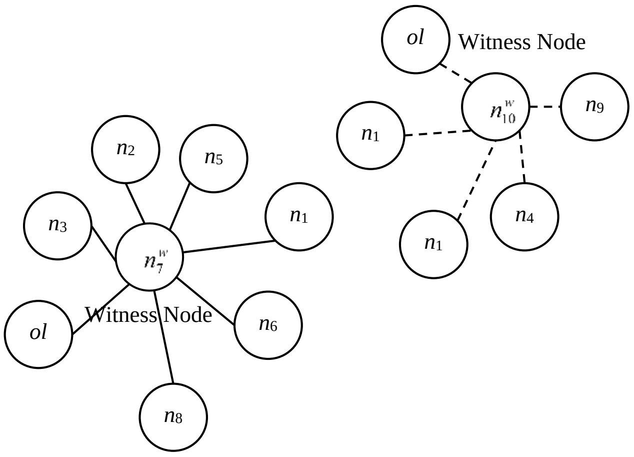

Figure 2(c). Scenario 3 for detection of Malicious Node

Table 1 Received Sequences and IDs of Neighbor Nodes

n$_7$	Received Sequences and IDs of Neighbor Nodes					
	ol$_2$	n$_2$	n$_5$	n$_3$	n$_6$	n$_{11}$
identity	ID_{oa_2}	ID_{n_2}	ID_{n_5}	ID_{n_3}	ID_{n_6}	ID_{n_8}
Received Sequences	sq_{n_7}	sq_{n_7}	sq_{n_7}	sq_{n_7}	sq_{n_7}	sq_{n_7} , $sq_{n_{10}}$
n$_{10}$	ol$_4$	n$_1$	n$_{11}$		n$_{13}$	n$_{14}$
identity	ID_{oa_2}	ID_{n_1}	$ID_{n_{11}}$	ID_{n_5}	$ID_{n_{14}}$	-
Received Sequences	$sq_{n_{10}}$	$sq_{n_{10}}$	sq_{n_7} , sq_{n10}	sq_{n10}	sq_{n10}	-

After that witness nodes n$_7$ and n$_{10}$ transmits $report_{n_7}$ and $report_{n_{10}}$. On receiving the $report_{n_7}$ and $report_{n_{10}}$, witness node verifies the $report_{n_7}$ and $report_{n_{10}}$. Nodes ol_2 and ol_4 are generated from the captured node oa_2 , hence their identity is the identity of node oa_2 . The witness node nodes n$_7$ and n$_{10}$ notices that node ol_2 and ol_4 have same identity but random sequences. Hence, it is concluded that clone attack occurred. Moreover, it is important to note that the node n$_5$ and node n$_{11}$ are not only the neighbor of node n$_7$ but the neighbor of node n$_{10}$, hence they can easily receive both sq_{n_7} and $sq_{n_{10}}$.

Hence, in this scenario, random sequence will fail to detect the clone node, in case there is a node which is the neighbor node of two or more witness node at the same time. To avoid the shortcomings of random sequence scheme, the witness node transmits the random sequence along with the timestamp S_w that cannot be modified by any node other than the sender witness node. Hence, the response of the neighbor node becomes:

$$M_s : \{NodeID, sq_{n_l^w}, S_w\} \tag{20}$$

If a node transmits the received sequence transferred from other nodes, timestamp S_w becomes $S_w + 1$.

3.4.1. Revocation Procedure

Once, the clone attack is detected, revocation scheme is triggered. It is performed by flooding the network with the two incoherent claims received by the witness node n_w . Here, every claim message of a node is signed with private key of the same node based on the response message received from the claimer node. This cancels the clone attack occurred in the witness node.

3.5. The Overall Algorithm

1. *Divide the nodes in to claimer node* $(n_j^c, j = 1,2,....P_c)$ *and witness node* $(n_j^w, j = 1,2,....P_w)$
2. *Witness node broadcast* $M_w : \{S_w = \Delta d + \Delta\varepsilon\}$
3. *Claimer node transmits the response message* $M_c : \{Node\,ID, Pilot\}$
4. *Extract channel response* $\underset{-\ t_j}{\hat{B}}\ (n_j^c \rightarrow n_l^w)$ *from pilot*
5. *Compute* $\Theta^l_{j,k}$
6. *If* $| d_j - d_k | \leq \Delta\varepsilon$ *and* $\Theta^l_{j,k} < \gamma$ *for nodes* oa_2 *and* ol_2 *, n$_2$ can then conclude that nodes* oa_2 *and* ol_2 *are suspects nodes*

```
7.  suspects' nodes are saved into $susplist_{n_l^w}$ .go to step 4.
8.  Broadcasts $susplist_{n_l^w}$ to other nodes.
9.  Broadcasts randomly chosen sequences
10. Response to the received sequences $M_s:\{NodeID, sq_{n_k^w}, S_w\}$.
11. If two node with same ID holds distinct random sequence,
12. Then clone node attack occurred
13. Trigger the revocation procedure
14. Flood the network with incoherent claims received by witness node.
15. // Selection of witness Node//
16. Apply GSA
17. // GSA Algorithm//
18. Generate initial population
19. Compute the fitness of each agent
20. Update the G, best and worst of population
21. Compute mass, and acceleration of each agent
22. Update each time velocity and position
23. If meeting the end of criteria
24. Then return best solution
25. Else go to step 19
```

4. SIMULATION RESULTS

4.1 Simulation Parameters

We use NS2 [12] to simulate our proposed Distributed Defense Mechanism for Clone Attacks based on Gravitational Search Algorithm (GSA). We use the IEEE 802.11 for wireless MAC layer protocol. It has the functionality to notify the network layer about link breakage. In our simulation, the number of attackers is varied as 1,2,3,4 and 5. The area size is 50 meter x 50 meter square region for 50 seconds simulation time. The simulated traffic is Constant Bit Rate (CBR).

Our simulation settings and parameters are summarized in Table 2.

Table 2. Simulation parameters

No. of Nodes	50,100,150 and 200
Area	500 X 500
MAC	802.11
Simulation Time	50 sec
Traffic Source	CBR
Rate	50Kb
Propagation	TwoRayGround
Antenna	OmniAntenna
Packet Size	512
No. of Attackers	1,2,3,4 and 5

4.2 Performance Metrics

We evaluate performance of the new protocol mainly according to the following parameters. The proposed DDMGSA is compared with the Distributed Detection of Clone Attacks (DDCA) [7] and ERCD: An Energy-efficient Clone Detection Protocol [10].

Average Packet Delivery Ratio: It is the ratio of the number of packets received successfully and the total number of packets transmitted.

Average end-to-end delay: The end-to-end-delay is averaged over all surviving data packets from the sources to the destinations.

Energy Consumption: It is the amount of energy consumed by the nodes to transfer the data to the receiver.

Packet Drop: It is the number of packets dropped during the data transmission.

Communication Overhead: It is measured as the ratio of number of controls packets exchanged to the total number of packets exchanged.**sults & Analysis**

A. Varying the Attackers

Among 100 nodes, the number of attackers is increased from 1 to 5 and the performance is evaluated for the above metrics.

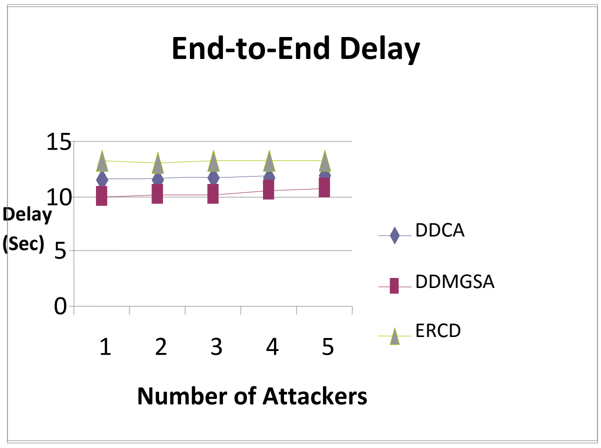

Figure 3. Attackers Vs Delay

Figure 3 shows the results of delay by varying the attackers from 1 to 5 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least delay among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 12% and ERCD by 22%.

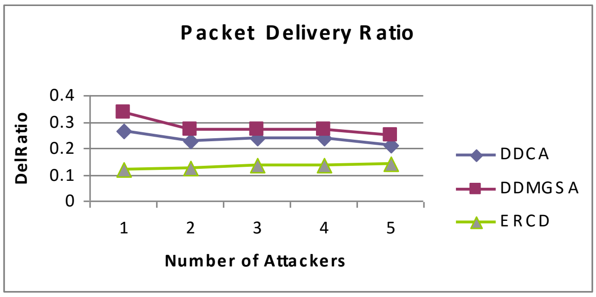

Figure 4. Attackers Vs Delivery Ratio

Figure 4 shows the results of delivery ratio by varying the attackers from 1 to 5 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the highest delivery ratio among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 15% and ERCD by 53%.

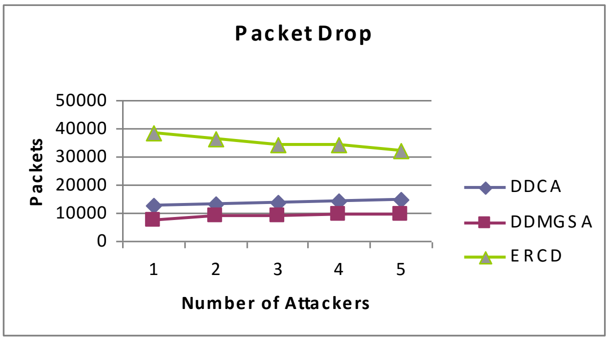

Figure 5. Attackers Vs Drop

Figures 5 shows the results of packet drop by varying the attackers from 1 to 5 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least packet drop among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 30% and ERCS by 75%.

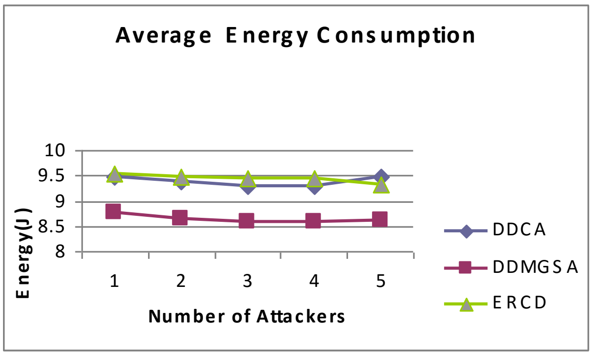

Figure 6. Attackers Vs Energy Consumption

Figure 6 shows the results of energy consumption by varying the attackers from 1 to 5 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least energy consumption among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 8% and ERCD by 9%.

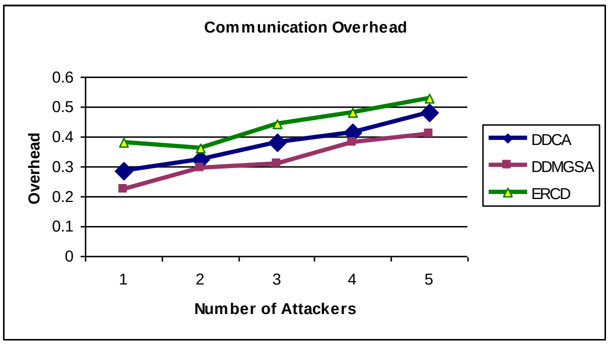

Figure 7. Attackers Vs Communication Overhead

Figure 7 shows the results of communication overhead by varying the attackers from 1 to 5 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least overhead among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 14% and ERCD by 26%.

B. Varying the Nodes

Keeping the number of attackers as 2**,** the number of nodes is increased from 50 to 200 and the performance is evaluated for the metrics.

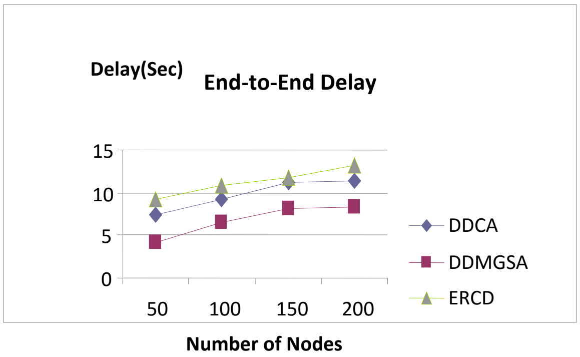

Figure 8. Nodes Vs Delay

Figure 8 shows the results of delay by varying the nodes from 50 to 100 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least delay among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 31% and ERCD by 40%.

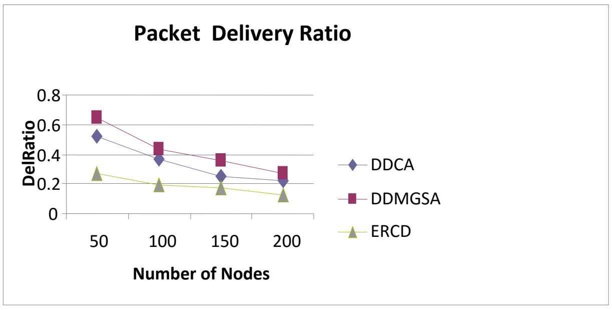

Figure 9. Nodes Vs Delivery Ratio

Figure 9 shows the results of delivery ratio by varying the nodes from 50 to 200 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the highest delivery ratio among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 19% and ERCD by 55%.

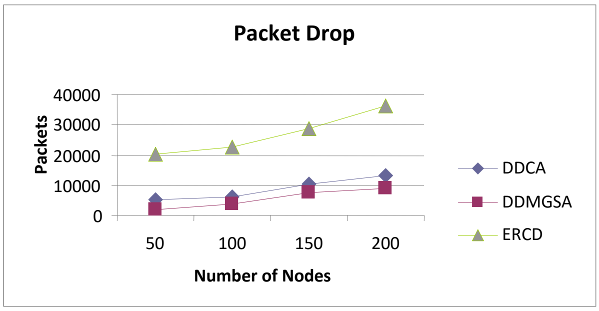

Figure 10. Nodes Vs Drop

Figure 10 shows the results of packet drop by varying the nodes from 50 to 100 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least packet drop among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 39% and ERCS by 80%.

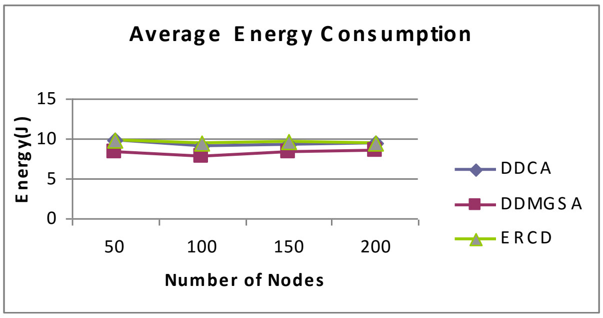

Figure 11. Nodes Vs Energy Consumption

Figure 11 shows the results of energy consumption by varying the nodes from 50 to 200 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least energy consumption among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 12% and ERCD by 13%.

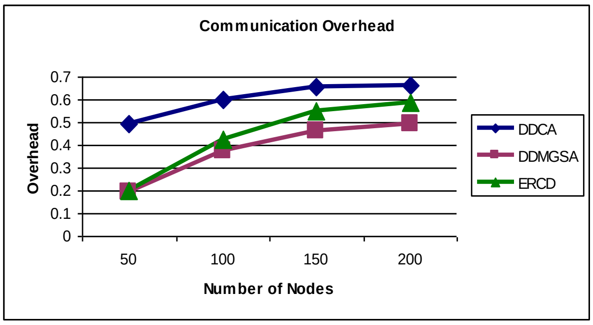

Figure 12. Nodes Vs Communication Overhead

Figure 12 shows the results of communication overhead by varying the nodes from 50 to 200 for all the 3 techniques. When comparing the performance of the techniques, we infer that DDMGSA has the least overhead among the 3 techniques followed by DBCA and ERCD. DDMGSA outperforms DBCA by 38% and ERCD by 11%.

5. CONCLUSION

In this paper, we have proposed a Distributed Defense Mechanism for Clone Attacks based on Gravitational Search Algorithm (GSA) in WS. For efficient detection of the suspect nodes, the nodes in the channel are divided into witness node and the claimer node. The witness node verifies the Node ID received from the claimer node along with the sequence and timestamp for efficient detection of clone attack. If a node contains same ID but different random sequences, it indicates that clone attack has occurred. In addition, to select the best witness node, GSA is used. GSA is actually a memory-less algorithm but works efficiently like algorithm with memory. It gives the one of the best optimal solution by following mass and change in velocity of the object. Once the clone attack is detected, revocation procedure is triggered to revoke the clone attack in the witness nodes. By simulation results, we have shown the proposed algorithm provides better protection to clone attacks by reducing the packet drop and increasing the packet delivery ratio. In future, we planned to extend the proposed work by analyzing and comparing several other related algorithms.

6 REFERENCES

[1] Yusnani Mohd Yussoff, Habibah Hashima, Roszainiza Roslib and Mohd Dani Babac, "A Review of Physical Attacks and Trusted Platforms in Wireless Sensor Networks", International Symposium on Robotics and Intelligent Sensors (IRIS), Procedia Engineering, 41, pp. 580 – 587, 2012.

[2] Meenakshi Tripathi and M.S.Gaur, V.Laxmi, "Comparing the Impact of Black Hole and Gray Hole Attack on LEACH in WSN", The 8th International Symposium on Intelligent Systems Techniques for Ad Hoc and Wireless Sensor Networks (IST-AWSN), 2013.

[3] Chakib Bekara and Maryline Laurent-Maknavicius, "A New Protocol for Securing Wireless Sensor Networks Against Nodes Replication Attacks", Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), 2007.

[4] Byung Goo Choi, Eung Jun Cho, Jin Ho Kim, Choong Seon Hong and Jin Hyoung Kim, "A Sinkhole Attack Detection Mechanism for LQI based Mesh Routing in WSN", International Conference on Information Networking, ICOIN 2009.

[5] Mukesh Tiwari, Karm Veer Arya, Rahul Choudhari, Kumar Sidharth Choudhary, "Designing Intrusion Detection to Detect Black hole and Selective Forwarding Attack in WSN based on local Information", Fourth International Conference on Computer Sciences and Convergence Information Technology, 2009.

[6] Mauro Conti, Roberto Di Pietro, Luigi V. Mancini, and Alessandro Mei, "Requirements and Open Issues in Distributed Detection of Node Identity Replicas in WSN", IEEE International Conference on Systems, Man, and Cybernetics, October 8-11, 2006, Taipei, Taiwan.

[7] Mauro Conti, Roberto Di Pietro, Luigi Vincenzo Mancini, and Alessandro Mei, "Distributed Detection of Clone Attacks in Wireless Sensor Networks", IEEE Transactions On Dependable And Secure Computing, Vol. 8, No. 5, 2011.

[8] Mebi Sernaz and Anand Pavithran, "Isolation of Clone Nodes and Black hole Nodes Using Risk Aware Mitigation", International Journal of Computer Trends and Technology (IJCTT) – volume 4 Issue 7– July 2013.

[9] H. Wen, J. Luo and L. Zho, "Lightweight and effective detection scheme for node clone attack in wireless

sensor networks", Published in IET Wireless Sensor Systems, 2011.

[10] Zhongming Zheng, Anfeng Liu, Lin X. Cai, Zhigang Chen and Xuemin (Sherman) Shen, "ERCD: An Energy-efficient Clone Detection Protocol in WSNs", INFOCOM, 2013 Proceedings IEEE. IEEE, 2013.

[11] Esmat Rashedi, Hossein Nezamabadi-Pour and Saeid Saryazdi: "GSA: Gravitational Search Algorithm"2009 ELSEVIER.

[12]NetworkSimulator: http:///www.isi.edu/nsnam/ns

[13] Richard Brooks, P. Y. Govindaraju, Matthew Pirretti, N. Vijaykrishnan, and Mahmut T. Kandemir, "On the Detection of Clones in Sensor Networks Using Random Key Pre-distribution", IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications And Reviews, Vol. 37, No. 6, 2007.

[14] Zhijun Li and Guang Gong, "On the Node Clone Detection in Wireless Sensor Networks", IEEE/ACM Transactions on Networking, IEEE 2013.

[15] Yingpei Zeng, Jiannong Cao, Shigeng Zhang, Shanqing Guo and Li Xie, "Random-Walk Based Approach to Detect Clone Attacks in Wireless Sensor Networks", IEEE Journal on Selected Areas in Communications, Vol. 28, No. 5, 2010.

[16] M. Conti, R. Di Pietrob, A. Spognardic, "Clone wars: Distributed detection of clone attacks in mobile WSNs", Journal of Computer and System Sciences, Vol. 80, 2014, pp. 654–669.

[17] Soumya Sara Koshy and Sajitha M, "Zone Based Node Replica Detection in Wireless Sensor Network Using Trust", International Journal of Computer Trends and Technology (IJCTT), Vol. 4, No. 7, 2013.

[18] Mayur R. Khandekar and U. K. Raut, "Node Replication Attack Detection Algorithms in Wireless Sensor Networks: A Survey", IOSR Journal of Computer Engineering (IOSR-JCE), Vol. 16, No. 5, pp. 65-70, 2014.

[19] Wibhada Naruephiphat, Yusheng Ji and Chalermpol Charnsripinyo, "An Area-Based Approach for Node Replica Detection in Wireless Sensor Networks", 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, 2012.

[20] Lei Jin, Hassan Takabi, James B.D. Joshi, "Towards Active Detection of Identity Clone Attacks on Online Social Networks", CODASPY'11, February 21–23, 2011, San Antonio, Texas, USA.

[21] Wazir Zada Khan, Mohammed Y. Aalsalem, Mohammed Naufal Bin Mohammed Saad, and Yang Xiang, "Detection and Mitigation of Node Replication Attacks in Wireless Sensor Networks: A Survey", International Journal of Distributed Sensor Networks, 2013.

[22] Moirangthem Marjit Singh, Ankita Singh and Jyotsna Kumar Mandal, "Towards Techniques of Detecting Node Replication Attack in Static Wireless Sensor Networks", International Journal of Information and Computation Technology, Vol. 4, No. 2, pp. 153-164, 2014.

Authors' contact:

P. Uma Maheswari
Assistant Professor/MCA, Regional Centre of Anna University,
Madurai, Tamilnadu, INDIA
E-mail: dharshukiran@gmail.com

P. Ganesh Kumar
Professor/IT, KLN College of Engineering, Sivagangai,
Tamilnadu, INDIA
E-mail: ganesh_me@yahoo.com