Using Metaheuristic Algorithms of Genetic, Particle Swarm Optimization and Glowworm in The Intrusion Detection System

Maryam Athari, Keivan Borna
Information Technology department, Islamic Azad University, Science and Research Branch, Kerman, Iran
Department of Computer Science, Faculty of Mathematics and Computer Science, Kharazmi University, Tehran, Iran

Abstract:
Intrusion detection in wireless mobile networks without infrastructure is of great importance because of the dynamic structure, non-central locations and limited resources of nodes. Due to the characteristics of wireless sensor network, similarities between them and the natural communities can be found. Natural communities working together can do something much bigger than each can handle. For this reason, it seems that the natural communities can be used as a model in wireless sensor networks. These systems have a mechanism that can be matched with a wireless sensor network. With the help of particle swarm algorithm in intrusion detection system problems like falling into local optimality trap and slowness of convergence rate can be solved. Given the important elements that affect the network security, we have investigated a model in this study which evaluates energy and throughput of metaheuristic algorithms such as particle swarm and genetic and glowworm that occur in natural communities. Since the sensor nodes in the network are critical, since numerous attacks can put networks safety in danger which in terms of activity are active and passive, in this study also we have used active attacks with the help of NS2 simulator and MATLAB software. Also the impact of these metaheuristic algorithms in wireless sensor networks was considered to assess lifetime of nodes in wireless sensor network.

Keywords:
Intrusion detection system, Wireless mobile networks without infrastructure, glowworm algorithm, Genetic algorithm, Particle swarm algorithm

1. Introduction

During the past few decades, the use of the network has been growing and today, almost all organizations use computer networks to store and transfer their data. These networks play an important role in enhancing the precision of doing things, but when they are attacked, they can cause irreparable damage into organizations and individuals. With the spread of new threats and attacks against computer networks, in case of non-compliance with security infrastructure by an organization, all organization's data and information place at risk. Using intrusion detection system is one of the ways to protect networks against attacks. Intrusion detection system continuously collects network traffic including data since the implementation of the system operation and in case of existing suspicious traffic, it warns network manager after analyzing data and aware him of existence of attacks [1]. The construction of an efficient intrusion detection model, is a challenging task. Although there are several types of intrusion detection systems, all of these systems have a common problem and it is the influx of a variety of attacks and lifetime of each algorithm in this intrusion detection system is different.

2. Intrusion Detection System

Intrusion detection system, monitors network traffic and system activities to detect attacks or acts contrary to the system policy and warns as soon as attack detection. Intrusion detection systems work automatically so they can be used as the primary means of establishing security in networks. In other words, in network attacks or denial of service, a remote attacker could exploit the defect within the network, causing disabling the server, router and the network. In root attacks that occur in multi-user operating system a user uses the other users' privileges and do not consider controls that produce access limitation. Intrusion that is considered as a successful attack, has the ability to obtain unauthorized access to files and programs and the control of a computer system.

- Intrusion detection system architecture includ es three classification.

2.1 Network Intrusion Detection Systems

These systems which are called NIDS for short are existed in network platform and analyze all the different layers of the network to detect intrusion attempts and attacks. Network intrusion detection system is responsible to detect unauthorized intrusions before they reach critical systems. Types of attacks that we are facing at the network level, are DOS attacks and port scanning attacks. It uses multiple sensors in different parts of the network to get network traffic. Central database analysis uses the traffic features that has been sent to them and use different methods of intrusion detection to detect intrusion activities [2].

2.2 Host Intrusion Detection Systems

These systems are called HIDS in short. HIDSs protect hosts that are deployed on them and detect unauthorized

Manuscript received October 5, 2016
Manuscript revised October 20, 2016

activities on the host computers. HIDSs due to staying on the host to monitor, should be aware of all kinds of additional local information with security implementations. One of the advantages of HIDSs is that provide specific information about the "where, when and by whom the intrusion has been occurred?" [2].

2.3 Distributed Intrusion Detection Systems

These systems are briefly called DIDS. Distributed intrusion detection systems are composed of multiple NIDS or HIDS or a combination of these two types with a central management station. Each intrusion detection system existed in the network sends its reports to a central management station. Central station assesses the received reports and informs the responsible for system security. Another task of the central station is updating the database for detection rules of each intrusion detection system existed in the network. The resulting information is stored in the central management station.

3. Attacks in Intrusion Detection System

3.1 Denial of Service Attack

This type of attack, is one of the most common attacks. This attack can affect all layers of the OSI connection model. In this attack, the attacker after crashing a system, attempts to steal, change or influence to the source of information. In some cases, after the attack being done, the intended service will not be completely cut off and only its performance will impair. In this type of attack, the attacker reduces the energy of the network nodes, as a result, nodes due to the low level of energy cannot participate in allowed and necessary operations of the network such as sending authorized packages and provide service to other nodes of the network.

3.2Distributed Denial of Service Attack

In DDos attack, attacking program begin its implementation from several systems or systems that are in a common network domain. In this type of attack, all systems send data packets to the victim simultaneously and with maximum power. When a DDos attack occur, the bandwidth traffic is higher than usual traffic. Even if DDoS attack is detected there is no way to identify the attack packets correctly because the attackers clean all the evidence which show them guilty. As a result, users are not suspicious of them. Their aim is to make application layer services out of reach for legitimate users [4].

3.3 Sinkhole Attack

Depending on routing algorithm, sinkhole node tries to collect network traffic and then slowly remove or alter it which cause disturbance in the network. Sinkhole node produces spurious RREQs using order number and obtains order number of target node from RREQ. Sinkhole node receives RREQ packet and puts its sequential number as the highest and adds its ID to the packet and then distribute it. The node that receives this fake packet, records this route as a new route. As a result, nodes use fake route to send data packets which cause the network to mitigate and finally disappear [5].

3.4 Wormhole Attack

This attack does not need to compromise with the sensor in the network. It is not possible to prevent wormhole attack using cryptography methods. Wormhole attack occurs by two malicious nodes via out of band connection. Hostile proceeds to receive or eavesdrop of packets in a region, then it sends the packets to hostile tunnel which is in another part of the network through the long range wireless connection or using a direct wired link. After that, sending packet between two separate nodes, easily convince them that they are neighbors. Also the hostile using this attack convince the nodes in a hub that their distance to base station is only one or two hub [6].

4. Ad hoc network

In ad hoc networks, the flow of malicious attack is very dangerous because they will block not only the victim node but also disrupt entire network. For example, when the source node wants to send a packet to predetermined destination, attacker changes the packet or do eavesdropping or maybe throw the packet. However, regarding that the network is a critical component of today's world and without providing network security, communications will be disrupted so using some strategies, attacks should be prevented and detected. One of the topics in the field of sensor network is the error detection or intrusion detection and the methods of transferring and exchanging information between network nodes which is strongly dependent on limitations, available resources and provided facilities from other layers of the network. Selecting the appropriate routing algorithm with the nature and conditions of considered network, has a significant impact on assessment parameters of the network performance and the amount of its cost. As usual, attacks take palace in such networks which result in loss of network performance. With the comparing of some important metaheuristic algorithms it is possible to assess the accuracy of intrusion detection in these methods.

5. Routing Protocol In Manet

An intrusion detection system by monitoring the network traffic, supervise the activity of mistrustful and does necessary actions through alerting the system. Including blocking the user or source IP address to access the network. To improve the network security, the routing protocol should be used. DSDV routing protocol is used in this study. DSDV protocol is a table based routing protocol in which every node of the network has a routing table. Routing table entries containing the destination address, the next hub to the destination and the number of required hub to reach the destination as well as the number of sequence in the destination node.

Updating in DSDV takes place such as this: the route with newer number always have priority and routes with older sequence number are discarded. Also routes with the same sequence number have priorities if they could provide a route with a better metric. If the node receives a packet with the same sequence numbers that has reached earlier, the route is selected accordance with the hub minimum criteria. Also if updating message arrives late through the route with minimum hub, it shows that there may be congestion on the route and if the data packet is sent in certain congested route, congestion and delay increase greatly. Incremental packets are used to reduce the volume of traffic caused by the updating of routes in the network. The advantage of this method is avoidance of creation routing loops in the networks including mobile routers [7].

6. Metaheuristic Algorithms

Intrusion detection capability can be increased inspirati on by nature. In this paper, the performance of metah euristic algorithms such as genetic, particle swarm opt imization and glowworm optimization has been assess ed in the intrusion detection system. Metaheuristic alg orithm is an artificial intelligence that is based on gro up behaviors. Metaheuristic algorithms include differen t types of algorithms. A few examples of these types are: Ant Colony Optimization (ACO), Artificial Imm une System (AIST), Artificial Bee Colony (ABC), Ge netic Algorithm (GA), Particle Swarm Optimization, G lowworm Swarm Optimization (GSO).

6.1 Genetic Algorithm

Genetic algorithm is as one of the first metaheuristic methods which mimics the natural biological evolution rules. This algorithm is on top of the algorithms called population based algorithms. The algorithm is based on the repetition and its basic principles is adapted from genetics. Genetic algorithms usually are implemented as a computer simulation in which population of an abstract sample (chromosomes) from solution candidates in an optimization problem will lead to a better solution. One of the main features of genetic algorithm is that, it constantly works on chromosomes and solution space [8].

The initial population is a set of chromosomes in which each chromosome represents a point in the search space and a possible solution to the problem. Applying genetic operators on each population, a new population with the same number of chromosomes is formed. Then the fitness function is determined for them. Using operators such as selection, crossover and mutation that are most commonly used in genetic algorithms, the new generation can be created. The number of generations like chromosomes' population is determined in the algorithm initialization step and methods of setting the parameter.

The selection operator selects a number of chromosomes from the existing chromosomes in a population to reproduction. Graceful chromosomes have more chance to be selected for reproduction. In fact, chromosomes which makes the next generation are being selected by this operator.

$$p_i = \frac{f_i}{\sum_{j=1}^{pop-size} f_j} \quad (1)$$

$$l_i = \pi r^2 * p_i \quad (2)$$

Equation (1) is to obtain the probability of each chromosome selection. p_i is the probability of selecting ith chromosome and f_i is the value of fitness function of ith chromosome and the dominator shows the total amount of fitness function of all chromosomes. In equation (2), l_i is the length of ith chromosome. Then the crossover operator produces the child chromosome. This operator produces two parent chromosome genes in new chromosome (child). The number of chromosomes which are selected from the initial population as a parent for crossover operation is obtained from equation (3).

$$n_c = 2\left[\frac{p_c \times n_{pop}}{2}\right] \quad (3)$$

Where shows the crossover p_c percent which usually $0.8 \leq p_c \leq 0.95$. n_c is always even.

Mutation operator also selects a gene from a chromosome randomly and then alters content of that gene. The mutation operator guarantees that genetic algorithm does not fall in trap of local minimum point and covers all chromosomes which may be destroy during the performance of other operations such as selection and crossover.

The number of chromosomes which should be affected by mutation is obtained from equation (4).

(number of genes = $\pi_m \times n_{var}$) (4).

In this relation $1 \leq \pi_m \leq 0$ is the rate of mutation impact and n_{var} is the number a chromosome genes (the number of variables). The genetic algorithm ends when a fixed number of generations is obtained or the best degree of chromosomes merit is acquired or the better result is not achieved [8].

Genetic algorithm pseudocode:

Start: generating random population from n chromosome.
Merit: evaluating merit function called F(x) of each chromosome X in population.
New generation: repeating the following steps, a new complete generation is produced.
Selection: in this step, two parent chromosome from a population are selected according to merit function.
Crossover: in crossover the line which is laid on both chromosomes generates two children that have their parents' features. If there is no any crossover, child will be an exact copy of their parents.
Mutation: this operator makes a chromosome which has been selected randomly to mutate. As a result, new children are generated respect to parent chromosome mutation.
Loop: return to step 2. This loop will be repeated as long as a certain number of rules being created.

6.2 Particle Swarm Algorithm

Inspiration of behaviors in the nature, algorithms such as ant colony and bee colony and . . . have been provided. Particle swarm algorithm is classified in this category. Particle motion is due to the received information from its surrounding. Each solution is called a particle. Each particle is calculated by merit function which represents the merit of that particle. More the particle has merit it is more close to its goal, in fact, each particle moves toward a particle which has higher merit function. Each particle has a velocity that expresses motion of the particle. Each particle following the optimum particles, continues to move in the problem space. That is, a group of particles are generated randomly at the beginning of PSO and find their optimum solution by updating the generations [9].

At each step, each particle is updated using the two best values. The first is the best position that the particle has been yet succeed to reach it which called pbest. This position is determined and maintained. The latter is the best position that has been achieved ever by particle population. This position is displayed as gbest. After finding the best values, velocity and the position of each particle is updated using equations (5) and (6).

$$v_i = v_i + c_1 * rand[pbest_i - position] + c_2 * rand[gbest_i - position] \quad (5)$$

Equation (5) is composed of three parts which the first part is the current velocity of the particle and second and the third parts are the changing of the particle velocity and moving it towards the best personal experience and the best experience of the group. Rand is a random number between 0 and 1. C1 and c2 are learning factors which usually their values are considered as 2.

The main advantage of this method over the other optimization strategies is that the large number of swarmed particles cause the flexibility of the method against the problem local optimum solution.

$$position = position + v_i \quad (6)$$

The initial position of particles should be uniformly distributed throughout the space. So that they could be found in most of the places i.e. the first position of particles must be generated with uniform distribution and the change velocity of the initial direction should be considered as zero.

Particle swarm algorithm pseudo-code:

1. Swarm creation by P particles.
2. Setting initial value for position and velocity of each particle randomly.
3. Calculating merit function for each position.
4. Calculating pbest and gbest for each particle.
5. Repeat
   - Updating the velocity of each particle according the equation
   - Updating the position of each particle according the equation
   - Calculating merit function for each particle
   - Updating Pbest for each particle
   - Updating gbest for each particle
6. In the case of meeting stop conditions, it ends otherwise it goes to step 5.

6.3 Glowworm Algorithm

glowworm algorithm is as one of the newest benchmarking algorithms and has proved its ability to solve optimization problems. In the first phase of glowworm swarm optimization algorithm, glowwormes have been placed objective function space randomly and at first glowworm contains equal amount of luciferin and sensor range. The phase of motion is strongly dependent on the luciferin severity of glowwormes' orientation. In fact, the glowwormes move to the better position and function. Each glowworm has local domain of decision which has been surrounded by wide range of sensors and is displayed by r.

Light intensity in the specified distance r from light source follows $I \propto \frac{1}{r^2}$. In the other words, the light intensity decrease by increasing the distance r. Then in the updating phase of the luciferin, the value of luciferin changes accordance with the value of objective function in the current location of glowworm.

$$L_i(t) = (1-\rho)L_i(t-1) + \gamma J(x_i(t)) \quad (7)$$

In equation (7) which represents value of luciferin, ρ is indicator of luciferin decay constant that has a value between 0 and 1. γ is luciferin incremental constant and has a value between 0 and 1. The objective function is displayed with $J(x_i(t))$. $x_i(t)$ Represents the position of ith glowworm at tth repetition.

Then in the updating phase of decision local domain, the number of neighbors is changing and decision local domain must be updated in each repetition.

The attractiveness of a glowworm is proportional to the light intensity appeared by the adjacent glowworm. The attractiveness value of a glowworm is calculated by equation (8).

$$\beta = \beta_0 + e^{-\gamma r^2} \quad (8)$$

Where β_0 is attractiveness value at r=0.

In glowworm algorithm, calculations for decision making space is complicated. Updating the position of glowworm is dependent on decision making space and is in direct relation with luciferin [10]. There is no limitation in glowworm algorithm in the number of neighbors and distance. This algorithm has cognitive limitations which allows the glowworm to divide into subgroups and receive the highest value of the function. As a result, this algorithm is very efficient in finding several peak points in multi model functions and evolutionary optimization. For the maximization purposes, the brightness can be associated with the value of objective function. More the brightness of glowworm, more the value of objective function and the obtained solution is better and the cost is less.

GSO algorithm pseudo code

1. Objective function, variables and algorithm parameters $f(x)\ x = (x_1, x_2, x_3, \dots, x_d)^t$
   - Defining initial population of glowwormes $x_i \quad i = 1, \dots, n$
   - Calculating the brightness of each glowworm (I_i) using the equation $f(x_i)$
   - Defining impact factor Υ.
2. While the stop condition has not been met
   - For all glowwormes ($i = 1{:}n$)
   - For all glowwormes ($j = 1{:}n$) (inner loop)
   - If $I_i < I_j$ glowworm i move towards glowworm j.

End if

   - Calculating the impact of r using $\exp[-\Upsilon^r]$
   - Evaluating and selecting new solution and updating brightness

End loop j
End loop i
Sorting glowwormes based on brightness and selecting best glowworm
End loop

3. Return the results

7. Simulation

Since most systems have stochastic processes, simulation often uses random number generator to create input data which are random events of real world. Table 1 shows the simulation parameters.

Table 1. Simulation parameters

No	Parameters	Values
1.	Simulation region	500 x 500 m^2
2.	Transmission range	100 m^2
3.	Initial energy	0.5J
4.	MAC protocol	802.11

5.	Transmission power	20*0.000000001W
6.	Receiving power	20*0.000000001W
7.	Number of nodes	100
8.	Traffic Demand	1Mb/s – 5Mb/s
9.	Gateway candidate	5

7.1 Comparing permittivity of three algorithms: GA – PSO – GSO

To calculate permittivity, a counter has been set in the input section of information to the NS2 parameter corresponding to the node which up to 3000 times sends the packet with the specified length with different data rate of byte per second. This information is moving between specified source and destination node. By changing the data rate which has been set in the transmitter node, the throughput has become different. This scenario is based on the number of repetitions and D-dos and Dos attacks has been applied on them.

In this study, each step has been repeated 7 times to improve the reliability and protocol 802.11 is used to send packets.

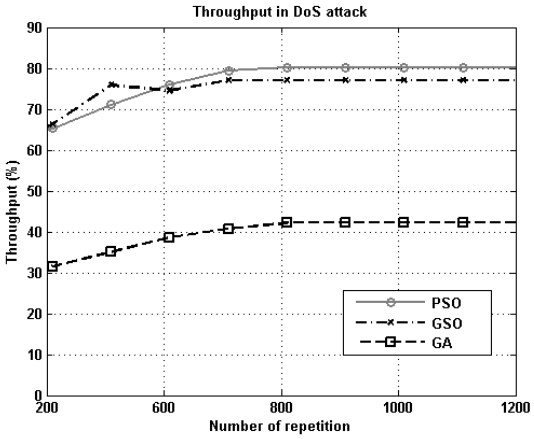

Figure 1. permittivity in the case of Dos attack

As shown in Figure 1, PSO algorithm has higher permittivity against applied DoS attack. GA algorithm by increasing the frequency of repetition has less permittivity against DoS attack compared to the PSO algorithm. GSO algorithm has the lowest permittivity compared to PSO and GA algorithms in the case DoS attack. As a result, PSO and GSO algorithms have high permittivity but there is a large gap between GA algorithm and other two algorithms.

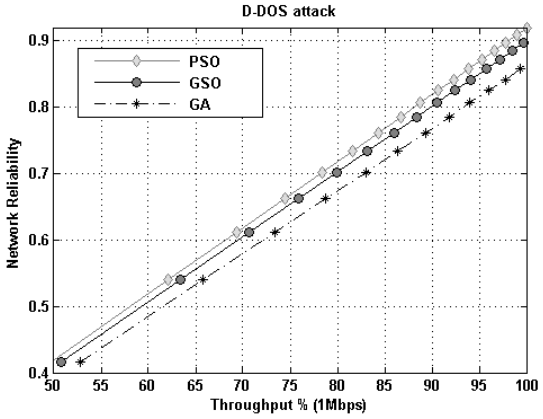

Figure 2. permittivity at the case of DoS attack

In Figure 2 permittivity on reliability can be seen. Data rate is considered as 1Mbps and PSO algorithm has higher reliability at D-dos attacks. By increasing sent data rate in GSO, its reliability at D-dos attack is less than PSO algorithm. GA algorithm has less reliability than PSO and GSO algorithms. In other words, PSO algorithm reliability is higher than other three algorithms at D-dos attack and this is a good feature for this algorithm. While GA algorithm has the lowest reliability.

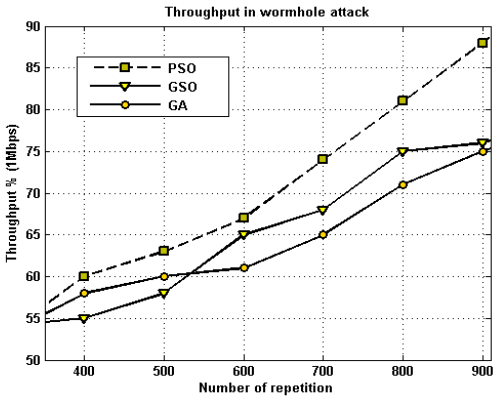

Figure 3. Permittivity in wormhole attack

Figure 3 shows the permittivity at wormhole attack for three different algorithms: PSO, GA and GSO. PSO algorithm has highest permittivity compared to other two algorithms, GA and GSO, which is one of the advantages of this algorithm. GSO algorithm has less permittivity than PSO algorithm at wormhole attack. GA algorithm by increasing the number of repetitions at wormhole attack, has lowest permittivity compare to other two algorithms.

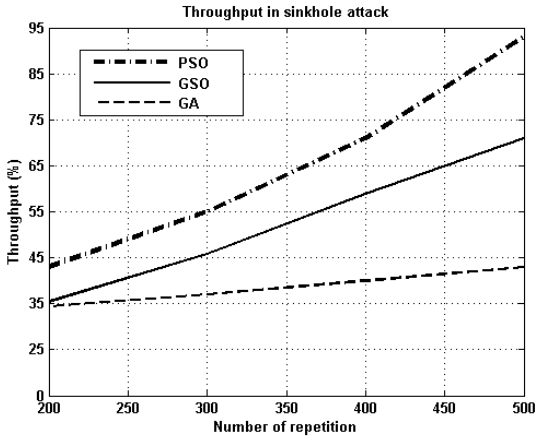

Figure 4. Permittivity at sinkhole attack

Figure 4 shows wormhole attack in three different alg orithms: PSO, GA and GSO. PSO algorithm which ha s been attacked by sinkhole, has highest permittivity c ompared to other two algorithms by increasing the nu mber of repetitions. GSO algorithm has less permittivi ty than PSO algorithm which has been attacked by si nkhole and compared to GA algorithm has higher per mittivity. GA algorithm has less permittivity at sinkho le attack compared to other algorithms in high freque ncy of repetitions. As a result, the highest permittivity belongs to PSO algorithm and lowest permittivity bel ongs to GA algorithm.

In this simulation, packets are being delivered betwee n transmitter and receiver. Permittivity was calculated based on the number of packets that have been delive red correctly to the destination.

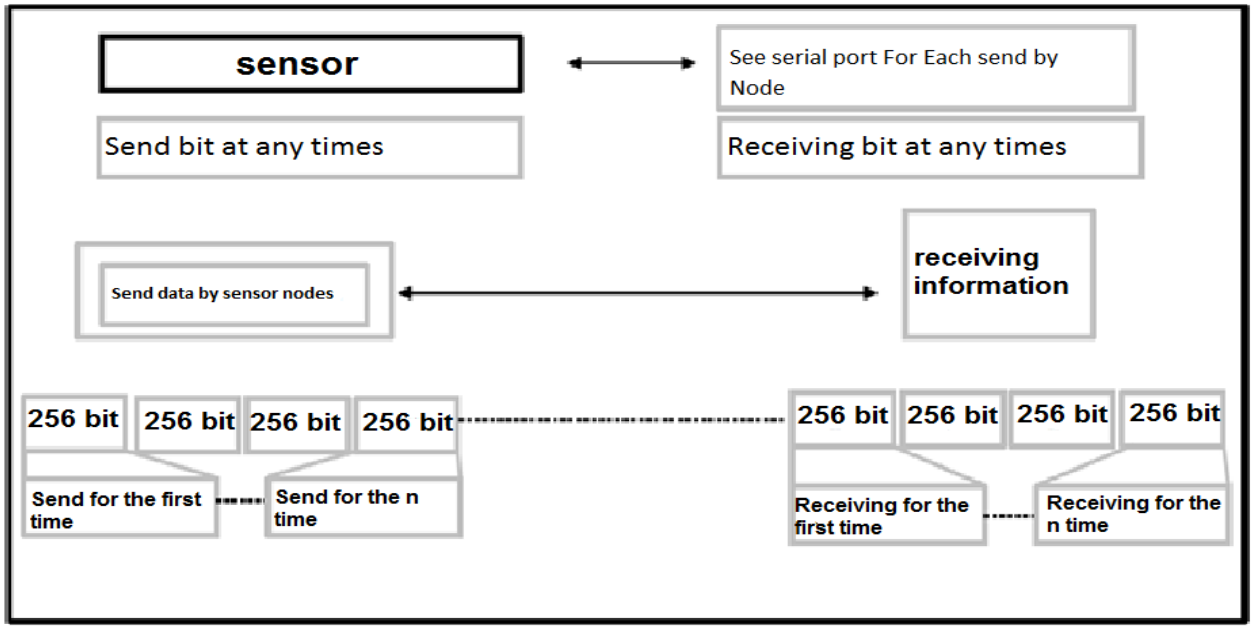

Figure 5. the number of repetitions.

Figure 5 describes the number of repetitions to send t he packet. Data is sent by the sensor. Packets has be en composed of bytes which have sent information an d the same bytes are received by the receiver. But a packet may be destroyed due to receiving an attack. I f packets to be delivered healthy to the destination, w e will obtain high permittivity. But if the received bit is less than the sent bit in received packet, permittiv ity will be decreased. In this model, packets simulatio ns are being received as queuing. They are placed at queue at first then they exit as FIFO. FIFO states tha t the packet which has arrived at first, is sent at first for sending.

7.2 Comparing energy consumption between three algorithms: GA - PSO – GSO

Energy consumption in sensor network in fact, is one of the most important challenges in the field of incr easing reliability. Also, due to the use of this type of networks in inaccessible environments, there is no po ssibility to recharge or replace the sensor nodes. So t he most important problem in the design of wireless sensor networks, is reducing energy consumption to in crease network lifetime.

(9)

$$Node_{RemEng} = Node_{RemEng} - (E_{tx} \times N_t + E_{rx} \times N_r + E_{ix} \times E_{sx})$$

Equation (9) is for calculating energy. E_tx represen ts the energy per sending the packet. Also N-t shows the number of transmitter nodes. E_rx represents the e nergy per receiving the packet and N_r is the number of receiver nodes. E_ix is the network's first energy which is a constant parameter in NS2 simulator. E_sx is the existing energy for network density and also is a constant parameter in NS2 simulator.

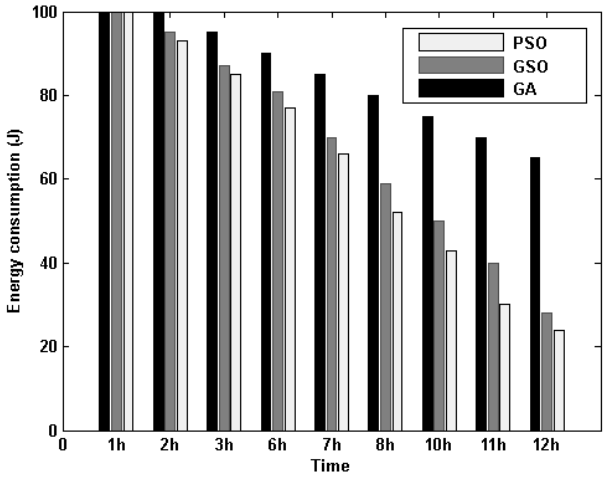

Figure 6. Energy consumption against the time.

Figure 6 shows the energy consumption. Energy consu mption of PSO algorithm increases over the time. On e of the reasons for high energy consumption in this algorithm is due to its complexity. GSO algorithm has less energy consumption than PSO algorithm over th e time. GA algorithm has less energy consumption th an other two algorithms, PSO and GSO. GA algorith m has less energy consumption due to its simplicity a nd less complexity.

7.3 Comparing lifetime of nodes in three different algorithms: GA – PSO – GSO

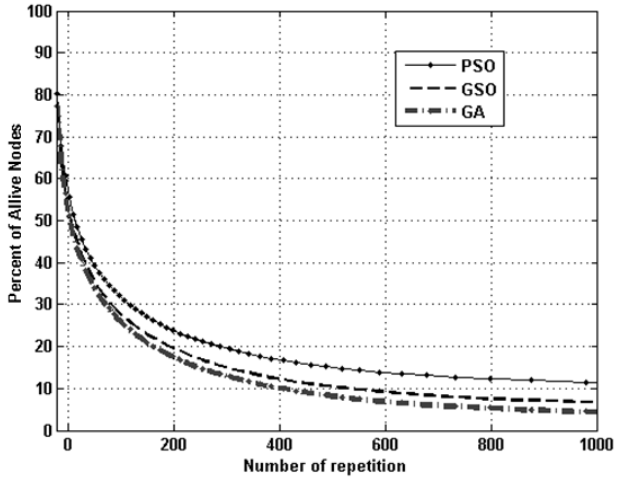

Figure 7. Assessing live nodes against the number of repetitions.

Figure 7 depicts the assessing of live nodes in three algorithms of GA, PSO and GSO against the number of repetitions. In PSO algorithm, the number of live nodes is larger than the number of repetitions. The nu mber of live nodes in GSO algorithm is less than PS O algorithm. GA algorithm due to low permittivity ha s lower live nodes compared to two other algorithms, PSO and GSO.

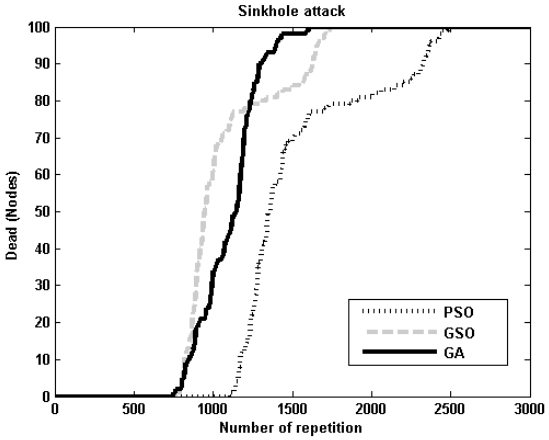

Figure 8. Assessing the dead nodes against sinkhole attack.

As it can be seen in Figure 8, when GA algorithm h ad been attacked by sinkhole the number of its dead nodes became more than two other algorithms, PSO a nd GSO. GSO algorithm has less dead nodes than G A algorithms while being attacked by sinkhole becaus e of having higher permittivity than GA algorithm. PS

O algorithm regarding the sinkhole attack, has the lo west dead node against the number of repetitions.

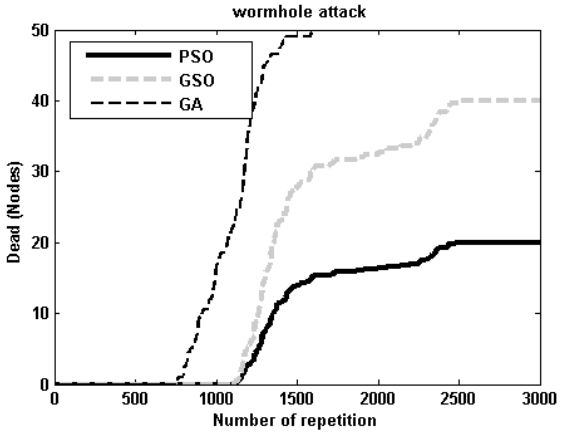

Figure 9. Assessing the dead nodes against wormhole attack.

As shown in Figure 9, GSO algorithm has more dead nodes than PSO algorithm while being under attack and in other words, its nodes have been removed bef ore than PSO algorithms. The nodes of GA algorithm were removed at shorter time compared to other two algorithms during the attack.

8. Conclusion

This study examined three metaheuristic algorithms of GA, PSO and GSO. Also DOS and D-dos attacks w ere applied to these algorithms. We evaluates energy and throughput of metaheuristic algorithms and assess lifetime of nodes in wireless sensor network. GA alg orithm has more less permittivity than PSO and GSO algorithms during DOS and D-dos attack. In terms o f energy consumption, this algorithm has less energy consumption compared to other two algorithms due to having simple equations which is considered as an a dvantage for this algorithm. The number of nodes whi ch arrived healthy to destination is very low in this a lgorithm. Applying DOS and D-dos attack to GA algo rithm will decrease the number of nodes which surviv ed in the repetitions.

In terms of permittivity, PSO algorithm has higher ra nk than two other algorithms, GA and GSO which thi s characteristic is one of the particular features of this algorithm. This algorithm has high energy consumpti on compared to other two algorithms due to its comp licated equations and this is considered as a disadvant age. PSO algorithm has less number of dead nodes c ompared to GA and GSO algorithms in repetitions of sending. During the DOS attack, the permittivity of GSO algorithm is less than PSO algorithm and much more than GA algorithm. In terms of energy consump tion, GSO algorithm due to complicated equations, but not as complicated as PSO algorithm, has high energ y consumption. The number of dead nodes of GSO al gorithm in repetitions of sending is more than PSO al gorithm and less than GA algorithm.

References

[1] Mostaque Md. Morshedur H. 2013. Network Intrusion Detection System Using Genetic Algorithm and Fuzzy Logic, International Journal of Innovative Research in Computer and Communication Engineering ,Vol. 1, Issue 7,pp 1435-1445.

[2] Poston H. 2012 .A Brief Taxonomy of Intrusion Detection Strategies. Aerospace and Electronics Conference (NAECON), 2012 IEEE National,pp. 255 – 263.

[3] Soryal J. , Saadawi T . 2012 . IEEE_802.11 Denial of Service Attack Detection in MANET, Wireless Telecommunications Symposium (WTS), pp. 1 – 8.

[4] Alqahtani S. Gamble R..2015 .DDoS Attacks in Service Clouds, 48th Hawaii International Conference on System Sciences,pp. 5331 – 5340.

[5] Basha M. , Vivekananda N. Bindu H. 2014.Evaluating the Effect of Attack on MANET Routing Protocols Using Intrusion Detection System .Vol. 5 ,Issue.2 ,pp 64-71.

[6] Gupta A. Ranga S.P. 2012 . Wormhole Detection Methods in Manet, International Journal of Enterprise Computing and business Syestems (Online) Ijecbs indi. Vol. 2 ,Issue 2 ,July,pp. 1-8.

[7] Basha M. , Vivekananda N. Bindu H. 2014.Evaluating the Effect of Attack on MANET Routing Protocols Using Intrusion Detection System .Vol. 5 ,Issue.2 ,pp 64-71.

[8] Neha Rai N. Rai kH. 2014 .Genetic Algorithm Based Intrusion Detection system,(IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 , pp. 4952-4957.

[9] Saxena.H, Richariya.V.(2014). “Intrusion Detection System using K- means, PSO with SVM Classifier”. International Journal of Emerging Technology and Advanced Engineering .ijetae, pp 653 – 657.

[10] Du M. Xiujuan Lei X. Wu ZH .2014. A Simplified Glowworm Swarm Optimization Algorithm, IEEE Congress on Evolutionary Computation (CEC) ,July 6-11, Beijing, China ,pp. 2861 – 2868.