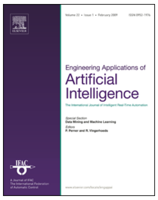
Engineering Applications of Artificial Intelligence 32 (2014) 228–241

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks

Shahaboddin Shamshirband [a,b,*], Ahmed Patel [c,d], Nor Badrul Anuar [b], Miss Laiha Mat Kiah [b], Ajith Abraham [e,f]

[a] *Department of Computer Science, Chalous Branch, Islamic Azad University (IAU), 46615-397 Chalous, Iran*
[b] *Department of Computer System and Technology, Faculty of Computer Science and Information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia*
[c] *School of Computer Science, Centre of Software Technology and Management (SOFTAM), Faculty of Information Science and Technology (FTSM), Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor Darul Ehsan, Malaysia*
[d] *School of Computing and Information Systems, Faculty of Science, Engineering and Computing, Kingston University, Kingston upon Thames KT1 2EE, United Kingdom*
[e] *Machine Intelligence Research Labs (MIR Labs), Scientific Network for Innovation and Research Excellence, USA*
[f] *IT4Innovations,VSB – Technical University of Ostrava, Czech Republic*

ARTICLE INFO

Article history:
Received 17 June 2013
Received in revised form
18 November 2013
Accepted 3 February 2014
Available online 3 March 2014

Keywords:
Game theory
Cooperative game IDPS
Fuzzy Q-learning
Intrusion detection and prevention systems
Security
WSN

ABSTRACT

Owing to the distributed nature of denial-of-service attacks, it is tremendously challenging to detect such malicious behavior using traditional intrusion detection systems in Wireless Sensor Networks (WSNs). In the current paper, a game theoretic method is introduced, namely cooperative Game-based Fuzzy Q-learning (G-FQL). G-FQL adopts a combination of both the game theoretic approach and the fuzzy Q-learning algorithm in WSNs. It is a three-player strategy game consisting of sink nodes, a base station, and an attacker. The game performs at any time a victim node in the network receives a flooding packet as a DDoS attack beyond a specific alarm event threshold in WSN. The proposed model implements cooperative defense counter-attack scenarios for the sink node and the base station to operate as rational decision-maker players through a game theory strategy. In order to evaluate the performance of the proposed model, the Low Energy Adaptive Clustering Hierarchy (LEACH) was simulated using NS-2 simulator. The model is subsequently compared against other existing soft computing methods, such as fuzzy logic controller, Q-learning, and fuzzy Q-learning, in terms of detection accuracy, counter-defense, network lifetime and energy consumption, to demonstrate its efficiency and viability. The proposed model's attack detection and defense accuracy yield a greater improvement than existing above-mentioned machine learning methods. In contrast to the Markovian game theoretic, the proposed model operates better in terms of successful defense rate.

1. Introduction

Wireless Sensor Networks (WSNs) provide an ideal schema for gathering data as opposed to sensor nodes and data transmission through wireless networks. These types of networks range in applications from the military (Bekmezci and Alagöz, 2009) and health care monitoring (Darwish and Hassanien, 2011) to disaster response (Shamshirband et al., 2010). The existing application designs for wireless sensors afford greater flexibility in establishing communications and increasing system automation, but are deficient in security and privacy (Naserian and Tepe, 2009; Sun et al., 2013; Xu, 2010). The core weakness of these sensor nodes lies in the limited-resource devices, i.e. power and processing units (Aslam et al., 2011). For this reason, vulnerability to various security threats is notably high. Meanwhile, adversaries may possess passive and active access to secret information, such as keys stored in a compromised node by eavesdropping (Schaffer et al., 2012) or Denial of Services (DoS) attacks. Thus the wireless medium becomes overloaded and the probability of packet collisions within the interfering signal's range increases, causing, in both cases, additional sensor node energy consumption (Tan et al., 2013).

In mitigation security attacks, the Soft Computing (SC) approach incorporates Intelligent Intrusion Detection Systems with Preventions (IIDPSs) to detect and impede abnormal traffic patterns that diverge from the modeled, expected, normal traffic behavior (Abraham et al., 2007; Anuar et al., 2013; Arun Raj Kumar and Selvakumar, 2013; Baig et al., 2013). A simple inspection packet mechanism was proposed by Tsunoda et al. (2008) to avoid stateful inspection against Distributed Reflective Denial of Service (DRDoS) attacks. Munoz et al. (2013) utilized fuzzy Q-learning for

* Corresponding author. Tel.: +60 146266763.
E-mail addresses: shahab1396@gmail.com, shamshirband@um.edu.my (S. Shamshirband).

http://dx.doi.org/10.1016/j.engappai.2014.02.001
0952-1976 © 2014 Elsevier Ltd. All rights reserved.

congestion detection to drop packets that differ from normal features. Misra et al. (2011) utilized a learning automaton to prevent Denial-of-Service attacks. While IDS-based SC approaches, such as misuse, anomaly and hybrid methods, display relatively reasonable performance regarding detection accuracy and minimal resource consumption, they fail to detect "unknown" attacks (Shamshirband et al., 2013). Hence, applying cooperative-based soft computing McGregory, 2013 security techniques that protect the wireless sensor network infrastructure by maximizing detection accuracy remains a challenge ((Shamshirband et al., 2013). As such, an attempt is made through this research to address the problem of security by applying the cooperative game-based fuzzy Q-learning mechanism.

The philosophy behind the cooperative game system is hereby employed to integrate anomaly detection into WSNs (Shamshirband et al., 2013; Patel et al., 2013; Huang et al., 2013; Shen et al., 2012). The game notion takes into account the results of the strategy selected by the players involved. In our scheme, however, the players (i.e., sink node, base station and attackers) are exchanged with actual wireless sensor nodes to detect attacks and defend against attackers by means of the cooperative game mechanism. In identifying the diversity of adversaries potentially encountered by a node, a Fuzzy Q-learning (FQL) algorithm is applied to reinforce players' self-learning abilities and provide detector players with an incentive function to protect the most vulnerable sensor nodes that represent possible security threats.

The remainder of the paper is structured as follows. In Section 2 related studies are presented. Section 3 describes the proposed model and its methodology. The model integrates a cooperative game theory with fuzzy Q-learning and aims to detect Distributed Denial-of-Service (DDoS) in WSN. The game model design is provided in Section 4 by introducing the player strategies, payoff functions, along with the reward and utility function. Section 5 highlights the fuzzy Q-learning algorithm, while Section 6 presents the simulation setup and a performance analysis discussion, particularly the detection and defense accuracy, network lifetime as well as energy consumption. Finally, Section 7 concludes the manuscript with suggestions for future research work.

2. Related studies

Data transmission within a WSN necessitates the fulfillment of five requirements associated with security and energy consumption, namely data privacy, authentication, integrity, a distributed denial-of-service (DDoS) attack in terms of flooding attack, and energy exhaustion (Huang et al., 2013). A multitude of DDoS attacks has been designed, which can be categorized as synchronized packets in transmission control protocol (TCP SYN) flooding, User Datagram Protocol (UDP) flooding, and Internet Control Message Protocol (ICMP) flooding. A flooding attack employs overwhelming volumes of packets to deplete the victim network's resources including the processing capability among network terminals. It may be assumed that the victim system's memory stack becomes saturated and no new demands can be processed (Zhou et al., 2010). In a WSN, flooding is more damaging on account of unstable wireless links, unbalanced usage of network resources, and weaker network devices, in which sensors always have processing and energy capability constraints. Sensor nodes near access points (i.e., sink nodes or cluster head routers) are normally more heavily loaded (Feiyi et al., 2007).

Among the numerous proposed network routing protocols over the past years, hierarchical routing protocols significantly contribute to a system's scalability, lifetime, and energy efficiency (Akkaya and Younis, 2005; Anisi et al., 2012). In hierarchical networks, nodes are allocated to different roles, such as CHs and cluster members. The upper level nodes, or the cluster heads (CHs), manage and collect data from the grouped, lower level nodes (cluster members). Each CH gathers data from the cluster members within its own cluster, aggregates this information, and then transmits it to the sink. All hierarchical routing protocols endeavor to select the finest CH and cluster the nodes into suitable groups in order to conserve energy (Lung and Zhou, 2010).

Although hierarchical protocols have innate weaknesses such as requiring time synchronization, potentially producing non-optimal routing, and utilizing higher overhead for cluster management, they reveal attractive advantages regarding WSN constraint management. Compared with flat protocols, hierarchical protocols offer a more feasible solution to handling large-scale networks with their enhancements to share limited wireless channel bandwidth, balance node energy consumption and reduce communication expense more optimally (Lung and Zhou, 2010; Liu, 2012). For instance, BEE-C (da Silva Rego et al., 2012) is a hierarchical routing algorithm bio-inspired by the behavior of bees for Wireless Sensor Networks (WSN), which aims to conserve the energy of sensor nodes. The BEE-C is based on the LEACH (Low Energy Adaptive Clustering Hierarchy) and LEACH-C (LEACH Centered) protocols, which are both prominent WSN protocols identified in literature. BEE-C is applied to sensor networks with continuous data dissemination. The results indicate a number of BEE-C advantages over LEACH and LEACH-C. In the search for an efficient approach to generate clusters, the well-understood hierarchical clustering algorithm is adopted in this paper by proposing a distributed hierarchical clustering algorithm that is discussed in Section 3.1.

Several countermeasures (Li et al., 2009), such as source-end defense points (Mirkovic and Reiher, 2005), core-end defense techniques (Chen and Hwang, 2006), victim-end defense (Wang et al., 2007) and adaptive probabilistic filter scheduling (Seo et al., 2013) have been developed to mitigate damage caused by flooding attacks in a routing protocol. Implementing firewalls, rate limitation and access control lists (ACLs) on routers may avert ongoing flooding attacks. End-to-end authentication should be well designed to ensure that each user is certified prior to access to any network resource or the wireless channel (Das, 2009). Traditional security strategies, like firewall and cryptography, are alternatives to preventing external intruders and satisfying data confidentiality, authentication, and integrity. Conventional strategies are essentially impractical in completely protecting network resources (i.e., energy resources) from increasingly sophisticated internal attacks (Qiu et al., 2013).

A different security approach incorporates Intrusion Detection and Prevention Systems (IDPSs) to detect and impede internal intrusions (Shamshirband et al., 2013). The Traditional Artificial Intelligence (TAI) (i.e., fuzzy set, neural network, genetic algorithm and artificial immune system) adapts to the IDS to capture the network's traffic activity through sensors and analyze it. For instance, the fuzzy data modeling-based wireless sensor network runs its dynamic rule settings to process all malicious events observed by the sensor and implements intrusion detection techniques (Kumarage et al., 2013). The Computational Intelligence (CI) classifiers (i.e., neuro-fuzzy, learning automata against DoS attack (Misra et al., 2009), game theory and reinforcement learning) regulate the multiagent to generate an iterative process of observing attack patterns, adjusting to the mathematical model, and predicting future attacks (Alpadin, 2010). More recently, the multi agent-based computational intelligence (MCI) IDS has been employed in wireless sensor networks to alleviate a DDoS attack through a cooperative agent scheme (Shamshirband et al., 2013).

The game theory is an applied mathematics branch that deals with the way rational entities or agents make decisions in the application of WSNs (Huang et al., 2013), cognitive radio networks (Elias et al., 2011), and ad hoc networks (Naserian and Tepe, 2009). It affords an array of mathematics tools for modeling and analyzing the interactions among rational groups, whereby rationalism is founded on the profit or reward perceived by the entities (Shoham and Leyton-Brown, 2009). Anomaly based WSN in the game-theoretic approach is a tremendously difficult task on account of the distributed nature of numerous players in WSNs. A large number of players additionally results in difficulty achieving equilibrium in a competitive game. To deal with a type of attack in WSN, Naserian and Tepe (2009) included an assortment of games, such as a non-cooperative, two-player, and non-zero-sum to their stratagem. In this game arrangement, better decisions are made according to the principles offered by payoff prevailing conditions. Shen et al. (2011) took into account the signaling game to create an IDPS game exhibiting the interaction between an attacker and cluster head in a WSN. The Bayesian Nash Equilibrium (BNE) scheme in conjunction with the mixed-strategies for outstanding detection policies served as the basis for their model. Thus, an ideal fundamental shield tactic to protect WSNs was achieved, while the probability of detecting attacks was simultaneously considerably enhanced.

A multi agent system utilizes the reputation security mechanism to perform dynamic role assignment based on the following three parameters: reputation, bootstrap time and energy. The approach evicts highly non-cooperative and malicious nodes from the network (Misra and Vaish, 2011). An adaptive learning routing protocol employs a learning automata algorithm for efficient malicious node detection (Rolla and Curado, 2013). The multilayer reinforcement learning framework assisted by the Hidden Markov Model (HMM) was proposed to solve real-time detection in a complex state space (Andersen et al., 2009). The results indicated that the network's cost function could be optimized if the agents collaborated repeatedly. In our proposed scheme, the cooperative game is implemented into IDPS to generate the benefits of a fuzzy Q-learning algorithm with a value function to mitigate the flooding attack issue in a WSN with respect to detection and defense accuracy. Resource loss, accuracy of attack detection via sensors, and service inaccessibility at critical times are among the challenges posed, and through this research, an effort is made to confront the security setbacks by applying the cooperative game-based fuzzy system and reinforcement learning mechanism.

3. Proposed model

3.1. WSN model

In the present research study, Fig. 1 illustrates the distributed network with hierarchical routing, which consists of clusters (C), their coordinators, or Cluster Heads (CHs), as well as the member sensor nodes (S). In the current scheme, the Cluster Head (CH) is assumed to be a Sink Node (SN) in each cluster. The SN monitors the behavior of sensor nodes by collecting data from the member sensor nodes and transmitting the critical status – the attack information of the sensor nodes, to a Base Station (BS). Each cluster is mapped into distributed system formation while the set of sensor nodes is mapped into each cluster grouping. Although only one BS is shown in Fig. 1, practically there could be several implemented in a real operational WSN.

The route from a sensor node to a BS is a deemed distributed-hierarchical path that creates a hierarchical system with numerous routes, which is the main feature of cluster-based WSNs. Sensor nodes function independently to avoid the collapse of all sensor nodes in case one of them fails. The sensor node redundancy approach increases the overall reliability in distributed hierarchical systems. Fig. 1 illustrates how sensor nodes send collected data from a sink node to a BS via other adjacent sink nodes, and the BS receives data only if all SNs within the routing formation are actively functioning. Hence, a set of clusters on a route is counted as a set of independent distributed-connected elements. Attacks in this scenario can target the WSN in multiple ways, with DDoS attacks potentially originating either from the Internet or neighboring wireless sensor sources.

3.2. Methodologies and techniques used

The game-based detection and defense mechanism operate to detect DDoS attacks, where the sink node and base station adapt to select the best strategy of detecting an immediate attack and respond to it. Regardless of whether the attacks are carried out on a regular or irregular basis, the IDPS can adjust its learning parameters through fuzzy Q-learning to identify future attacks.

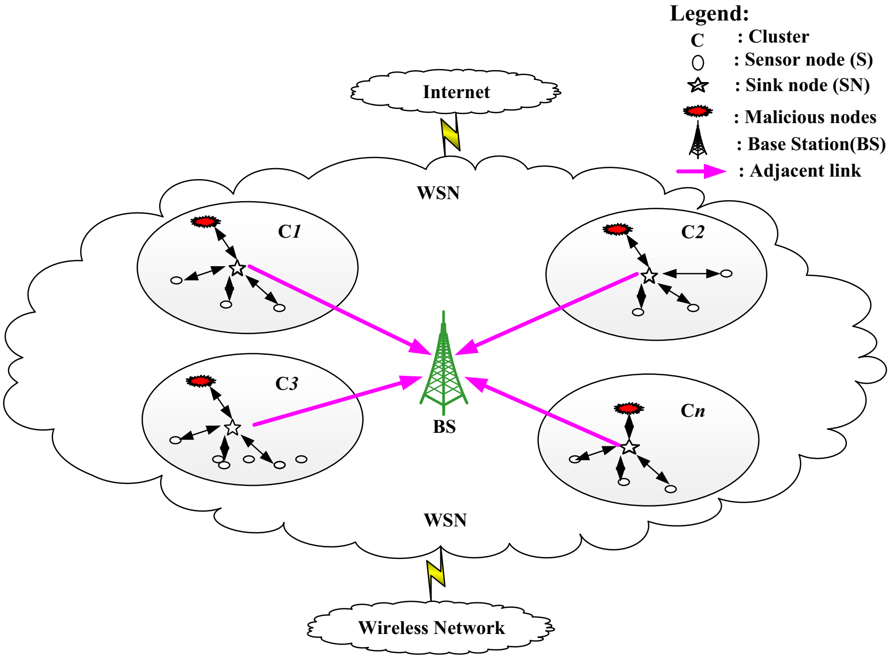


Fig. 1. A distributed hierarchical system perspective of a WSN with Internet and other wireless network connections.

The architecture of the proposed game-based FQL is dual, that is, it has two phases (Fig. 2).

Phase 1: In the primary stage of the game scenario, player 1 (the sink node) utilizes the fuzzy Q-Learning algorithm to identify the level of disruption done by the attacking player to the victim node, causing anomalies such as low access or damage. For attacker player detection, the sink player adopts three strategies, namely catch, missed, and low catch, as elaborated in Section 4.1.1 (player strategies applied to the sink node). Finally, the sink node transmits an alarm event that contains malicious node information to the base station (player 3) by an adjacent link connected to the base station (Fig. 1). The malicious information is preprocessed by the sink node to travel from phase 1 to 2 based on the alarm event beyond the default value threshold, to prepare a countermeasure strategy against the attacker through a defense mechanism.

Phase 2: In the second phase of the game scenario, player 2 (base station) employs the fuzzy Q-learning algorithm to confirm the malicious node's behavior. It checks the memory of player 1 or looks it up in a table and compares it with its memory in order to defend against the attacker. The detection player (sink node) and defense player (base station) coordinate their defense with each other to shield the wireless sensor nodes against the attacker player (attack/intrusion).

To highlight the proposed game-based FQL, the sink node and the base station are allocated a corresponding reward/incentive functional value, which is retained by the Fuzzy Q-learning IDPS. As such, a node's evolving fuzzy state may be recorded and quantified through the fuzzy reward utility function as discussed in Section 4.1.2 (the player payoff function). When a node encounters an attack or receives an anonymous message, the sink node sends the related severity alarm event evidence and messages to the BS, who then analyzes the critical data to adjust the FQL parameters. Based on the sink node information, the base station decides which nodes are under attack or at risk and elects whether to safeguard them or not. The BS has previously set a severity alarm event threshold rate, ν. Once the severity alarm value acquired by a node exceeds ν, the FQL IDPS deems the node under attack or at risk and strengthens its defenses to secure the cluster area in which the node is detected at the associated base station.

3.3. *Possible attack categories*

In this research study, the *Open System Interconnect* (*OSI*) *model* is classified into *five layers* (Akyildiz et al., 2002): physical layer, link/MAC layer, network layer, transport layer, and application layer. The attacks for each layer are analyzed by focusing on the flooding attack and its potential defenses. In the proposed scheme, a specific kind of DDoS attack is created with respect to a flooding attack that affects cluster heads. The generated attack sends flooding UDP packets to diminish the cluster head's energy. Table 1 indicates the impact of such attacks on the WSN layers as well as the defense mechanism.

In this work, only a DDoS attack in the application layer is considered. It is characterized by the presence of an attacker, and is known as a UDP flooding attack. In the proposed model, a UDP flooding attack occurs based on a random function to compromise the CH in each cluster. This kind of DDoS attack is aimed at exhausting CH energy by sending flooding packets in a fraction of time (Ghosal and Halder, 2013).

4. The architecture of cooperative game-based FQL IDPS

The proposed game-based defense strategy is primarily a combination of the cooperative game theory and fuzzy Q-learning algorithm. The game-based detection and defense mechanism operate to detect DDoS attacks, where the sink node and base station adapt to select the best strategy of detecting an immediate attack and respond to it. Regardless of whether the attacks are carried out on a regular or irregular basis, the IDPS can adjust its learning parameters through fuzzy Q-learning to

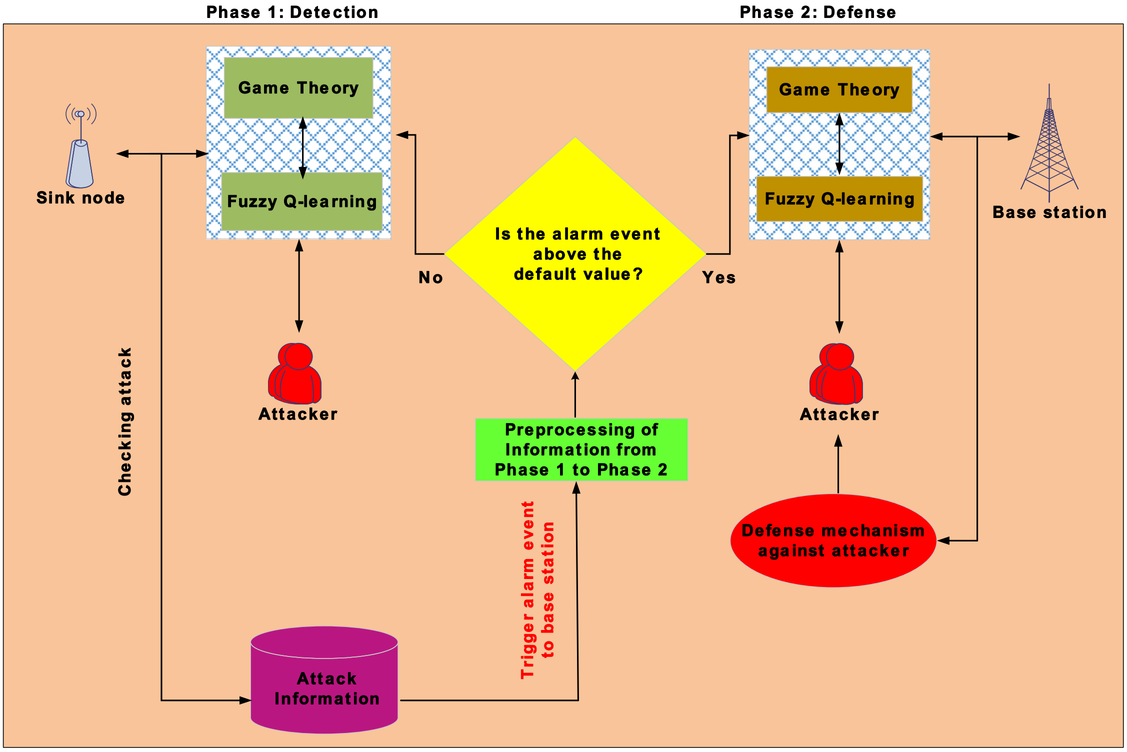

Phase 1: Detection
Phase 2: Defense
Game Theory
Fuzzy Q-learning
Sink node
Game Theory
Fuzzy Q-learning
Base station
Is the alarm event above the default value?
No
Yes
Attacker
Attacker
Checking attack
Preprocessing of Information from Phase 1 to Phase 2
Defense mechanism against attacker
Trigger alarm event to base station
Attack Information

Fig. 2. Model of a cooperative game-based IDPS and an attacker.

Table 1
Classification of denial-of-service attacks and defense at each protocol layer.

Protocol layer	Attack	Defense mechanism
Application layer	Overwhelming (McGregory, 2013)	Sensor tuning Data aggregation
	Path-based DoS (Li and Batten, 2009)	Authentication and anti-replay protection
	Deluge (reprogramming) attack	Authentication and anti-replay protection Authentication streams
Transport layer	SYN (synchronize) flood (Bicakci and Tavli, 2009)	SYN cookies
	Desynchronization attack (Xing et al., 2010)	Packet authentication
Network layer	Spoofing, replaying, or altering routing control traffic or clustering message (Qazi et al., 2013)	“Authentication and anti-replay protection secure cluster formation”
	Hello floods (Khalil et al., 2010)	“Pairwise authentication” “Geographic routing”
	Homing, black-hole attack (Khalil et al., 2012)	Header encryption Dummy packets
Link/MAC (medium access control)	Jamming (Law et al., 2005)	Authentication and anti-replay protection
	Denial of sleep (Law et al., 2009)	Authentication and anti-replay protection Detect and sleep Broadcast attack protection
Physical layer	Jamming (Li and Wang, 2012)	Detect and sleep Route around jammed regions
	Node tampering or destruction (Xing et al., 2010)	Hide or camouflage nodes Tamper-proof packaging

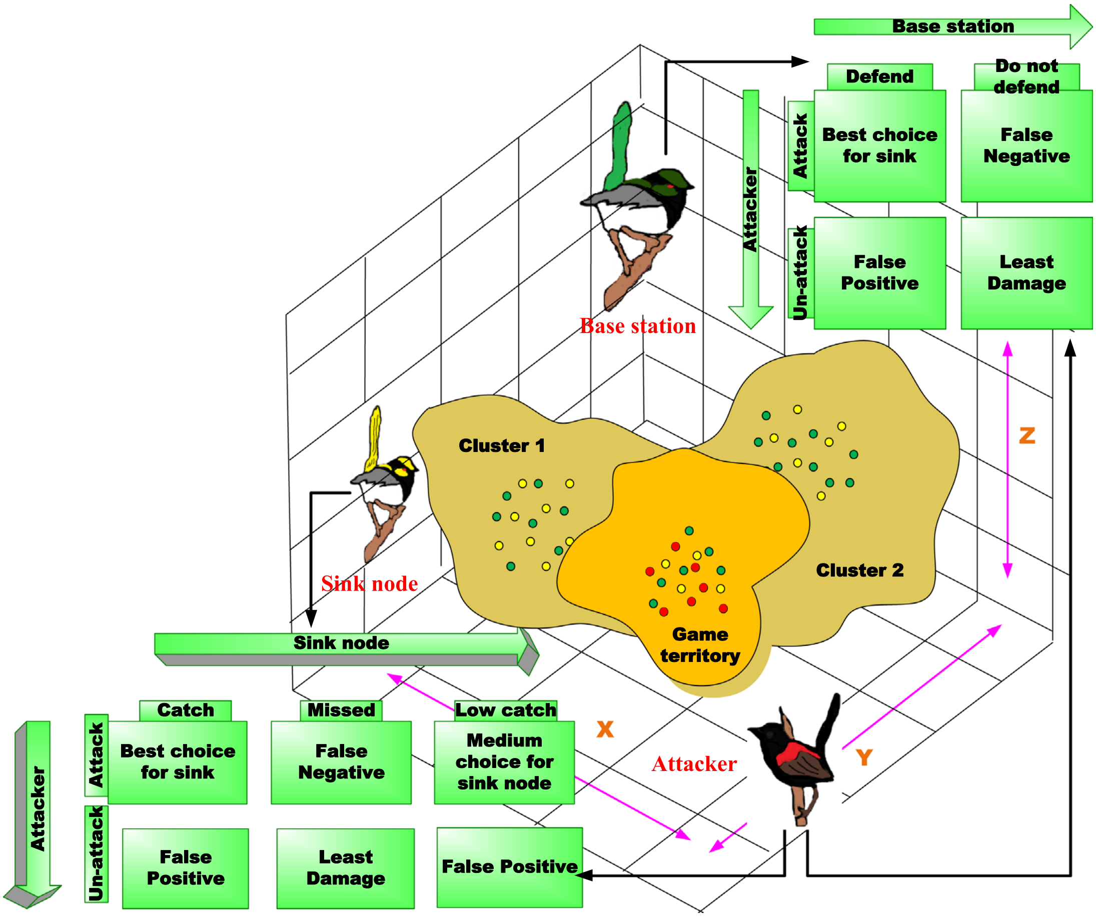


Fig. 3. Game-based defense system architecture.

identify future attacks. A comprehensive description of the theoretical and practical operation of the game theory and Q-learning modes, mainly concerning Fuzzy Q-learning, is provided later. Cooperative game based architecture in a WSN is shown in Fig. 3 and the block diagram of Fuzzy Q-leaning optimization system is proposed in Fig. 4.

In the primary stage of the game scenario, player 1 (the sink node) utilizes the FQL algorithm to evaluate the contents of the attacker player's level of access (i.e. low access, or damage). With regard to detection, the sink node player assumes three strategies, namely *catch*, *missed* or *low catch*. Upon completing the first stage, the sink node transmits an alarm to the base station (player 3) when

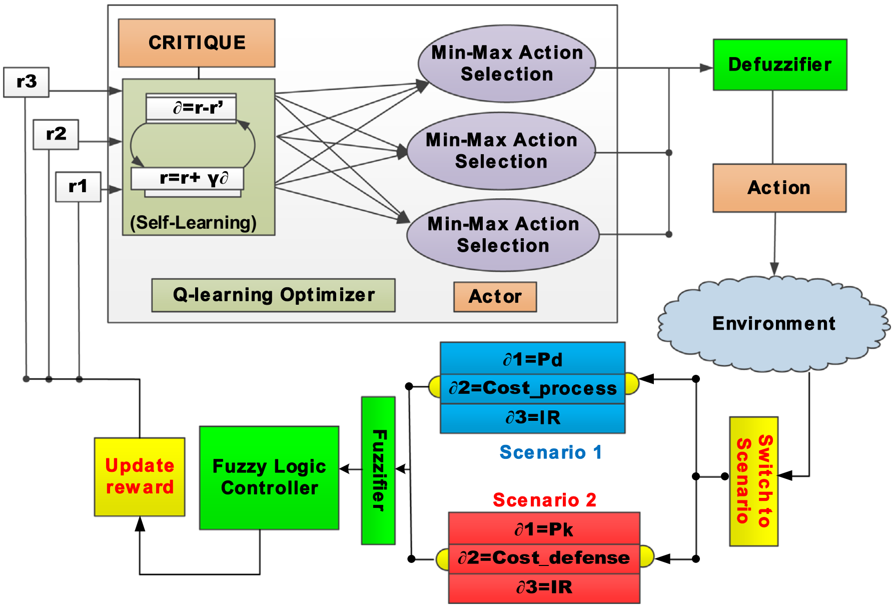


Fig. 4. Block diagram of the FQL optimization system.

Table 2
Game play between a sink node (IDS1) and an attacker.

Game play between sink node and attacker	Sink		
	Catch	Missed	Low catch
Attacker			
Attack	($a11$,$b11$)=Best choice for sink	($a12$,$b12$)=False Negative	($a13$,$b13$)=Medium choice for sink node
No attack	($a21$,$b21$)=False positive	($a22$,$b22$)=Least damage	($a23$,$b23$)=False positive

the attacker assaults the sensor node. In the second phase of the game scenario, player 3 (the base station) employs the FQL algorithm to evaluate the attack records in order to defend against the attacker. The detection player (the sink node) and defense player (the base station) take part in a game via a 3D game interface to shield the wireless sensor nodes against the attacker player (the attack).

4.1. Game design

In the proposed game theory method, it is assumed that the sink node can identify abnormalities in view of IDS1. Accordingly, in computer-generated WSNs, the sink player or cluster head diffuses the alarm to the base station (IDS2) upon perceiving an anomaly. When the IDS1 receives an anomaly message, it acquaints itself with this sort of attack using the FQL detection algorithm and archives the information in its attack record database. The IDS2 attempts to respond to these attack records. The fundamental concepts of the proposed game theory, player strategies and player payoff function are introduced next.

4.1.1. The player strategies

The interactions between the G-FQL and the attackers are split into two main categories, as seen in Tables 2 and 3 respectively. The first category represents the game between the attacker and sink node players, while the second type denotes the game between the attacker and the base station player. The game play strategy between a sink node and an attacker with respect to IDS1 comprises:

(1) *Best choice for sink:* The sink node chooses to identify the attacker, and the invader opts to attack;
(2) *False negative:* The sink node chooses not to identify the attacker, and the attacker strikes;
(3) *Medium choice:* The sink node chooses to identify the attacker with *low catch*, and the attacker attacks;
(4) *False positive:* The sink node elects to detect the attacker, and the attacker chooses not to attack;
(5) *Least damage:* The sink node chooses not to identify the attacker, and the attacker chooses not to attack;
(6) *False positive:* The sink node chooses to identify the attacker with *low catch*, and the attacker chooses not to attack.

The game strategy between a base station and an attacker concerning IDS2 is defined as follows:

(1) *Best choice for base station:* BS elects to defend and the attacker decides to attack;
(2) *False positive:* BS elects to defend, and the attacker chooses not to attack;
(3) *False negative:* BS elects not to defend, and the attacker attacks;
(4) *Least damage:* BS elects not to defend, and the attacker chooses not to attack.

4.1.2. The player payoff function

In this research, a payoff value is defined as a player reward function if it protects the WSN. In other words, when the IDPS fails to defend the WSN in case an invader attacks, the payoff of the player would be different. The three player payoffs are expressed as A, B, and C, where a_{ij}, b_{ij}, and c_{ij} denote the sink node, the attack and the base station payoff, respectively. Table 4 displays the

Table 3
Game play between a base station (IDS2) and an attacker.

Game play between base station and attacker	Base station	
	Defend	**Do not defend**
Attacker		
Attack	(a11,c11)=Best choice for sink	(a12,c12)=False Negative
No attack	(a21,c21)=False Positive	(a22,c22)=Least Damage

Table 4
The payoff matrix and utility functions.

Payoff function	Payoff matrix	Utility function	Description of Utility function
Attacker's payoff function	$A=[a_{ij}]_{2*3}$	$a_{ij}=IR-Cost_{processing}$	$IR=\frac{Number\ of\ malicious\ attacks}{Total\ malicious\ attacks\ sent}$ $Cost_{processing}=processing\ time\ for\ attack$
Sink node's payoff function	$B=[b_{ij}]_{2*3}$	$b_{ij}=P_d-Cost_{process\ detect}$	$P_d=\left(\frac{Correct\ attack\ detection}{Total\ detection\ and\ no\ detection}\right)$ $Cost_{process\ detect}=Cost\ of\ attack\ detection\ during\ sink's\ processing$
Base station's payoff function	$C=[c_{ij}]_{2*2}$	$C_{ij}=P_k-Cost_{defend}$	$P_k=Cost\ of\ killing\ attacks$ $Cost_{defend}=power\ cost\ during\ defense\ against\ attack$

Table 5
Notations associated with the reward functions of a sink node and base station.

T	T={0, 1, …, k −1} denotes the set of time in a Markov process
$\boldsymbol{S}$	The fuzzy state space of a sensor node, where the initial state is S_0, and the next state of si is si+1 for $I \in T$
$\boldsymbol{D1}$, $\boldsymbol{D2}$	The set of detection strategies
−$\boldsymbol{R1}$,−$\boldsymbol{R2}$	The payoff incurred at a false negative incident

payoff matrix, the utility function as well as a description of the utility function.

4.1.2.1. *Attacker payoff function.*

The attacker's payoff matrix $A=[a_{ij}]_{2*3}$ is defined as follows:

$$A_{ij}=\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}_{i*j} \tag{1}$$

where $a_{11}=IR-Cost_{processing}$ represents $(Number\ of\ malicious\ attacks/Total\ malicious\ attacks\ sent)-(processing\ time\ for\ attack)$, which is when an attacker and the sink node choose the same sensor nodes to attack and detect, respectively (AS1, SS1). The attacker's original utility value of $U(t)$ will be deducted from the cost of attacks. $a_{12}=IR-Cost_{processing}$ represents an instance when the attacker attacks and the sink node do not detect it correctly. However, $a_{13}=IR-Cost_{processing}$ means that an attacker attacks and the sink node detects a compromised node with a low rate of detection. $a_{21}=Cost_{processing}$, which signifies that an attacker does not attack at all, but the sink node falsely detects the attacker. By subtracting $IR=(Number\ of\ malicious\ attacks/Total\ malicious\ attacks\ sent)$ from the original utility function, $a_{22}=Cost_{processing}$ stands for when the attacker and sink node choose two different strategies, neither of which causes an attack nor detects an attack correctly, respectively. In this case the cost of attacking one node from the original utility is ignored. When $a_{23}=Cost_{processing}$, it signifies that the attacker does not attack and the sink node detects the attack with low probability/performance.

4.1.2.2. *Sink node payoff function.*

By denoting the sink node's payoff with matrix $B=[b_{ij}]_{2*3}$ we get:

$$B_{ij}=\begin{bmatrix} b_{11}=P_d-Cost_{process\ detect} & b_{12}=Cost_{process\ detect} & b_{13}=P_d-Cost_{process\ detect} \\ b_{21}=P_d-Cost_{process\ detect} & b_{22}=Cost_{process\ detect} & b_{23}=P_d-Cost_{process\ detect} \end{bmatrix}_{i*j} \tag{2}$$

where

- $P_d=\left(\frac{Correct\ attack\ detection}{Total\ detection\ and\ not\ detection}\right)$
- $Cost_{process\ detect}$ is the cost of attack detection during sink processing

4.1.2.3. *Base station payoff function.*

By describing the base station's payoff functions with matrix $C=[c_{ij}]_{2*2}$, it is defined as

$$C_{ij}=\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}_{2*2} \tag{3}$$

where $C_{11}=P_k-Cost_{defend}$ denotes $(Cost\ of\ killing\ attacks)-(the\ cost\ of\ defending\ against\ attack)$, which is when a base station and attacker choose the same sensor nodes to attack and defend, respectively.

4.1.3. *Reward function analysis*

Based on the three-player game, two constant reward values are defined: $R1$ as the gain of the IDS1 when the sink node identifies the WSN and reward value $R2$, or positive reward, as the gain of the IDS2 when the base station protects the WSN. If the sink node does not identify the WSN during the attack, the reward of the IDS1 would be –$R1$ (a negative reward). Likewise, if the base station fails to defend the WSN during an attack, the payoff of the

IDS2 would be –*R*2. An explanation of the correlated reward/incentive functions of a sensor node and base station is provided in Table 5. To detect a potential future DDoS attack on a sensor node, Fuzzy Q-learning is applied to enhance the self-learning ability of the IDS1 and IDS2 processes. The Fuzzy Q-learning supplies the IDPS with a learning mechanism, but the self-learning ability of the Q-learning IDS can evolve during the learning process, something that takes learning time, especially at the beginning. Through such self-iterative learning, IDSs are capable of protecting sensor nodes from recognizable potential attacks in ongoing active WSNs.

Fuzzy Q-learning is a discrete-time fuzzy-based Markovian procedure. When the process is at time *t* and fuzzy state *FSt*, the Decision Maker may choose to perform a fuzzy action. The process responds with a corresponding fuzzy reward for the decision maker at time $(t+1)$ and moves to fuzzy state $Fst+1$. The interaction details and information are as follows. Based on the Fuzzy Q-learning concept, a function $fx(1)\colon FS1 \to FD1 \times FA1$ is defined to demonstrate the detection and attack strategies for node *x* at a specific interval in IDS1. For instance, $fx(state\ 1)=(d1,\ a1)$ depicts $(d1,\ a1)$, which is a combination of the detection and attack strategies when the node sink transits from state *st* to $st+1$, and the reward established by *x* is defined as $R1(fx(st))$ as given by Eq. 4:

$$R1(f_x(S_t)) = \begin{cases} 0 & (\text{if } \ P_d = low \text{ and } Cost_{processing} = low) \text{ or } (\text{if } \ IR = low \text{ and } Cost_{processing} = low) \\ R1 & (\text{if } \ P_d = high \text{ and } Cost_{processing} = low) \text{ or } (\text{if } \ IR = high \text{ and } Cost_{processing} = low) \\ -R1 & (\text{ if } \ P_d = low \text{ and } Cost_{processing} = high) \text{ or } (\text{if } \ IR = low \text{ and } Cost_{processing} = low) \end{cases} \tag{4}$$

In the first case of Eq.(4), no detection and no attack are defined. Accordingly, the reward is fixed at 0. The second case defines when the sink node detects an attack with high accuracy, and its reward would be *R*1. In the last case, $(\text{if } P_d = low \text{ and } Cost_{processing} = high)$ or $(\text{if } \ IR = low \text{ and } Cost_{processing} = low)$, where the sink node uses strategy P_d with low processing cost and high detection accuracy to identify attack strategy *IR* with low attack and low processing cost, the reward is –*R*1. The first term (P_d) represents the gain of employing the sink node's strategy to detect attack strategy *ai*, and $(Cost_{processing})$ represents the cost of using strategy. The second term, (IR), represents the gain of utilizing the attack strategy with the processing cost for the attacker.

In the IDS2 scenario, the reward function incorporates the shield policy and attack strategy when the BS transits from state s_t to ${}_{st+1}$, and the reward received by the base station is defined as $R2(fx(st))$ as given in Eq. 5:

$$R2(f_x(S_t)) = \begin{cases} 0 & (\text{ if } P_k = low \text{ and } Cost_{defend} = low) \text{ or } (\text{if } IR = low \text{ and } Cost_{processing} = low) \\ R2 & (\text{if } P_k = high \text{ and } Cost_{defend} = low) \text{ or } (\text{if } IR = high \text{ and } Cost_{processing} = low) \\ -R2 & (\text{if } P_k = low \text{ and } Cost_{defend} = high) \text{ or } (\text{if } IR = high \text{ and } Cost_{processing} = low) \end{cases} \tag{5}$$

In Eq. (5), the first case defines no defense and no attack. Therefore, the reward is set to 0. In case two, when the base station defends against an attack with high defense strategy, its reward would be *R*2. The last case indicates that the base station uses strategy P_k with high processing cost and low cost of defending against an attack strategy, therefore the reward is –*R*2. The first term (P_k) represents the base station's gain of using strategy to eradicate the attack strategy *ai*, and the $(Cost_{processing})$ signifies the cost of using strategy. The second term (IR) denotes the gain of applying the attack strategy with the processing cost for the attacker.

It is assumed that the state of node *x* is s_0 at $t=0$. If the defense and detect strategies d_1, d_2 are taken against an attack strategy *a*, the state of node *x* evolves from s_0 to s_1, and node *x* (with respect to the sink node and base station) receives a reward $R\ (fx(s_0))$ and so on (Eq.(6)). In Q-learning, the state of node *x* transits from s_0 to s_1 and eventually to *sp* where $1 \ll p \ll k-1$, where *k* signifies the efficiency of IDS1 using the *di* strategy in detecting and defending against an *aj* attack strategy. Thus, the accumulated reward received by *x* is

$$R_x^p = \sum_{t=0}^{p} \gamma^p \ R(f_x(s_p)) \tag{6}$$

where $\gamma \in [0,\ 1)$ is the discount rate parameter. An attack strategy, and the objective of IDS2, is to choose a suitable defense policy against an assault to accumulate rewards. It is noteworthy that R_x^p will be calculated as two sub-rewards, such as *R*1 for the base station and *R*2 for the sink node. An instance of the reward function given to the cluster head (sink node) and attacker situation is the total amount of positive reward signals received when no attack has occurred and no alarm is raised (True Negative), and the number of correct invasion cases detected by the system (True Positive). The game theory phases are as follows:

- Phase 1**:** The sink node monitors message attacks through the game-based FQL operation as the first step defined by IDS1 (see Table 2) and then conveys the message to the base station for the second step function defined by IDS2 (see Table 3).
- Phase 2: Upon receiving an abnormal signal from the sink node, IDS2 uses its detection fitness test in conjunction with the knowledge database to assess attack patterns and severity. This evaluation permits IDS2 to regulate the overall defense strategy in order to mitigate the DDoS attack. The IDS2 function uses the fuzzy game theory principle to select an appropriate defense tactic to shield the message-consuming sensor node. The IDS2 also informs the affected sink node that it needs to protect itself against the offending attack pattern.
- Phase 3: The sink node verifies the current state of IDS play with the sensor node. If the sink node still detects an irregularity, it is likely that the IDS2 operation opted for the wrong defense strategy, and in turn, the sink node advises the IDS2 to revise its detection strategy. If the attack pattern alert count at the sensor node decreases in number, the sink node systematically endeavors to confirm the current state of IDS play with the sensor node until the attack condition is resolved and returns to the correct state of defense strategy.

- Phase 4: The sink node notifies IDS2 that the attack at the sensor node has been successfully counteracted and the attack has ceased.
- Phase 5: The IDS2 thus concludes defending the sensor node.

4.2. Utility function

To appraise the efficacy of the associations determined by the G-FQL and to determine the applicability of the rule at every point in time, Eq. (7) was utilized in this work, as suggested by Huang, et al. (2013). In Table 6 the parameters of the utility function are described:

$$U=\rho*SP-\beta*FN-\theta*FP \tag{7}$$

The game principle approach entails detection accuracy with low time complexity, which only afterward begins to formulate a shield policy. The major drawback of the game theory is that if attacks are recurring over a short period, a considerable amount of time is consumed in the detection phase, something that weakens the defense. It can be said that the detection precision may be low while the false alert rate is high. This problem is a worst-case scenario but can be addressed using the Cooperative-FQL proposed by Shamshirband et al. (2013). Its principal contribution is identifying the probability of future attacks aimed at a wireless sensor node. For frequent attacks occurring over a short time, multiagent-based FQL was adopted to deal with the excessive time spent on detection. The aim of the proposed FQL is to obtain high detection accuracy with a low false alarm rate.

5. Fuzzy Q-learning algorithm

To overcome the required complex detection and defense time, as well as detection precision issues in our game theory method, the FQL algorithm proposed by Shamshirband et al. (2013) is applied in this paper to detect probable future points of attack in advance. To optimize Q-learning algorithm performance from the action selection method and reward function perspectives, fuzzy min-max methods were employed. In the proposed scheme, the fuzzy min-max action selection and reward function with conventional Q-learning are evaluated. High detection accuracy performance was demonstrated. For this reason, FQL was employed to reinforce a system's learning capability.

The FLC inputs are provided by two scenarios through the switching process. In the first scenario, which is the game between a sink node and attacker, $P_d=(Correct\ \ attack\,detection/Total\,detection\ and\ no\ detection)$, the cost of attack detection during sink processing $(Cost_{process\ detect})$ as per sink node utility function and $IR=(Number\ of\ malicious\ attacks/Total\ malicious\ attacks\ sent)$, as well as $Cost_{processing}=processing\ time\ for\ attack$ with respect to the attacker utility function, correspond to the fuzzy state of network $S1\ (t)$ from the first scenario $S_1(t)=[Pd, Cost\_process, IR]$. In the second scenario that is the game between the base station and an attacker, $Pk=cost\ of\ killing\ attack$, $Cost\ \ defend{=}Power\ cost\ during\ defence\ against\ attack$ adapts as a base station utility function and $IR=(Number\ of\ malicious\ attacks/Total\ malicious\ attacks\ sent)$ and $Cost_{processing}=processing\ time\ for\ attack$, with regard to the attacker utility function, correspond to the fuzzy state of network $S2(t)$ from the first scenario: $S_2(t)=[Pk, Cost\_defend, IR]$.

The FLC output, given by the increment in states, represents the action of the sink node and the base station $A(t)$. The reward signal, $R\ (t)$, is built from FLC and is measured in both modes of adjacency in order to test if the sensors experience attacks in detection mode and the base station correctly defends against attacks. The linguistic variables *Pd, Cost_ Process*, and *IR* act as input for the first scenario, while the linguistic variables *Pk, Cost_defend*, and *IR* serve as input for the second scenario.

The *Detect Confidence* $(DC1)$ behaves as output for the first scenario and the *Defend Confidence* $(DC2)$ acts as output for the second scenario. They are both applied in the experiments (Table 7).

Two fuzzy sets are identified in all inputs and outputs, whose linguistic terms are ‘Low’ (L) and ‘High’ (H). The fuzzy reward is elaborated in Section 4.1.3. Hence, the objective is to determine the total reward value over time. If the defense and detect strategy *di* is used against the attack strategy *aj* at time *p* and the state of node *x* transits from S_t to Sp_{+1}, the Q-learning function for IDS1 is Q: $S\times D\times A\rightarrow R$ as given in Eq. 8:

$$Q(S_p, d_i, a_j)\leftarrow Q(S_p, d_i, a_j)+\ \alpha[R(f_x s_p)+\gamma R_x^{p+1}-Q(S_p, d_i, a_j)] \tag{8}$$

where $\alpha\ \in\ (0,\ 1]$ is the learning rate factor. In this scheme, the Q-function is applied in dual situations, such as IDS1 and IDS2.

Table 6
Utility function parameters.

Parameters	Explanation
U	Is a utility
ρ	Symbolizes the weight of effective prediction, $q=0.75$
SP	Characterizes the true confidence rate of attack patterns
β	Signifies the weight of failed estimates (attack but no defense), $b=1$
FN	Represents false negative of attack patterns – there are attacks but no defense
θ	Denotes the weight of failed predictions (defense but no attack), $h=1$
FP	Represents false positive of attack patterns – there is defense but no attack

Table 7
Linguistic variables for fuzzy set input and output.

Type of scenario	Variable	Attribute	Membership function		
Attacker and sink node	Input	Pd	Low	Med	High
		Cost process	Low	Med	High
		IR	Low	Med	High
	Output	*Detection confidence* $(DC1)$	Low	Med	High
Attacker and base station	Input	*Pk*	Low	Med	High
		Cost defense	Low	Med	High
		IR	Low	Med	High
	Output	*Defense confidence* $(DC2)$	Low	Med	High

In each state, a cluster head (sink node) gets rewarded by the reward function using the Q-learning method, and the base station also obtains the reward. G-FQL attains the final reward value of each player. A learning rate of zero means the system does not learn anything new, but a value of 1 would prompt the system to adjust its accuracy strategy as it self-learns from new attacks and update the information in its knowledge base. If the reward value is below the threshold, v, FQL IDS1 deems node x secure; otherwise it considers the node insecure and takes suitable detection action against the attack. Simultaneous to this evaluation, FQL IDS2 takes appropriate defensive action against any potential attacks.

6. Simulation and analysis

6.1. Simulation setup

To appraise the performance and check the connection between G-FQL and the routing protocol, NS-2 is simulated. In this work only the Distributed Denial-of-Service (DDoS) attack is considered. DDoS is characterized by the presence of an attacker and is called a flooding attack, and it causes noise in wireless communication by sending flooding packets as well as exhausts energy (Ghosal and Halder, 2013). The noise disrupts communication between nodes in the network, preventing them from entering 'sleep mode' due to the medium getting flooded with messages.

The Low Energy Adaptive Clustering Hierarchy (LEACH) protocol was utilized in the simulation, as it most closely reflects WSN in practice and it is also applicable to dealing with energy consumption concerns in WSNs. The simulations were run for 1000 s with LEACH as the routing protocol, the initial access point energy was 100 J, the effective transmission range of the wireless radio for the access point was 100 m, the sink node transmission range was 100 m, the common node transmission range was 50 m and the transport protocol is given in Fig. 5. In addition, the cooperative game-based IDPS with fuzzy Q-learning, was employed to hasten the simulation.

Table 8 presents the WSN configuration along with the set of parameters applied in NS-2. However, in practical WSN security operation, minimizing energy usage to conserve energy and maximize detection accuracy as much as possible is vital when designing and running G-FQL within efficient IDPS. The results obtained from the proposed algorithm are compared with those from Fuzzy Logic Controller, Q-learning, and Fuzzy Q-learning (Shamshirband et al., 2013) as well as the Markovian Game (Huang et al., 2013).

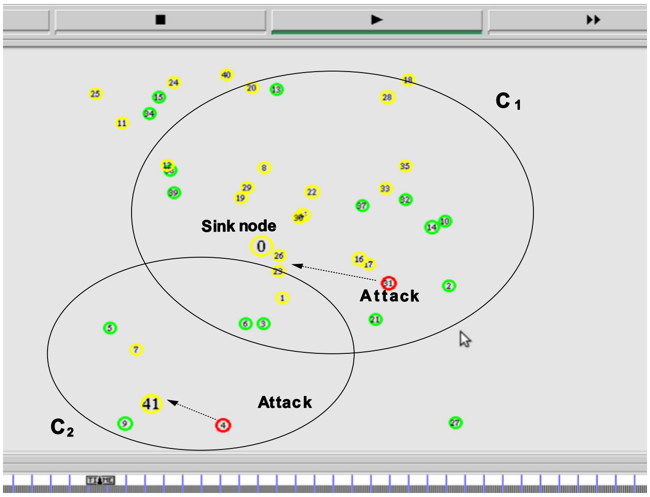


Fig. 5. Simulated WSN environment.

Table 8
Wireless sensor network parameters in NS-2.

Wireless sensor network parameters	Values
Access point	1
Common nodes	200
Sink node in each cluster	1
Routing protocol	LEACH
Scenario size	100*100
Simulation time	1000 s
Transport protocol	UDP
Access point initial energy	100 J
Access point transmission range	100 m
Sink node initial energy	10 J
Sink node transmission range	70 m
Common node initial energy	10 J
Common node transmission range	50 m

6.2. Generating and analyzing the flood attack strategy

The purpose of this section is to analyze the quantitative behavior of attacks in the UDP protocol layer. In the present experiments, normal UDP traffic was initially considered, after which the attack intensity under flood attacks with UDP traffic was explored. Subsequently, the total energy consumed before and after attack was examined. The accuracy of detection and defense as a result of executing the G-FQL algorithm was finally assessed. To generate an attack, a random function was employed, which selected subject nodes from each cluster to attack. The selected nodes adjusted their functions to send flooding packets to the cluster head. Algorithm 1 displays the attack strategy.

Algorithm 1. Attack strategy.

```
1. Start
2. Min($r$)=0%% Initial round simulation (Max($r$)=$n$)
3. While ($r< >n$)
4. Decide $r$ round's cluster head randomly
5. Cluster head advertises schedule time to all its common
   nodes
6. Generate attack node randomly
7. Attack node receives schedule time message from its cluster
   head
8. Attack node starts to compromise victims
8.1. Attack node sends flooding packets to its cluster head in
   this round
8.2. Victim (cluster head) receives data more quickly than
   normal state, so its energy will decrease rapidly
9. End.
```

In the experiment, an attack with UDP attack intensity was implemented. Fig. 6 indicates flooding attack intensity per packet length. Greater attack intensity percentage obviously occurred between 200 and 300 s, at which time packet length also reached elevated values. In Fig. 7 it appears that UDP attack intensity affected the WSN energy, besides the fact that energy was consumed in proportion to attack intensity. For example, for attack intensity between 100 and 150 s, the most energy was consumed.

In the present research work, three sets of experiments were conducted to examine the effects of attack detection accuracy and defense rate against attacks based on the Fuzzy Logic Controller, Q-learning algorithm, Fuzzy Q-learning and Game theory-based Fuzzy Q-learning algorithms. The cost function was calculated according to Eq. (7).

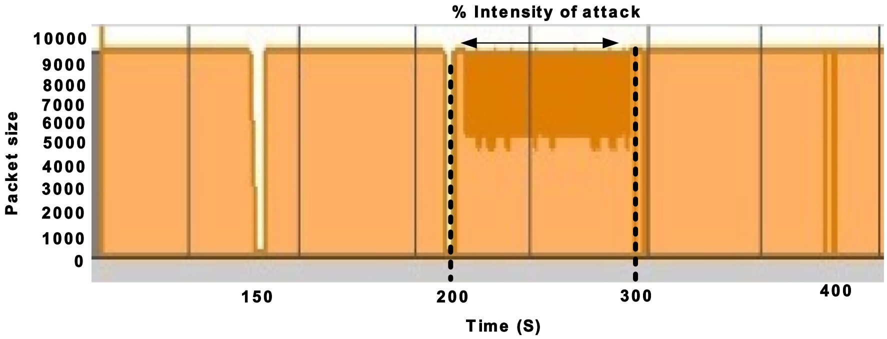


Fig. 6. Effects of UDP attack intensity on packet size.

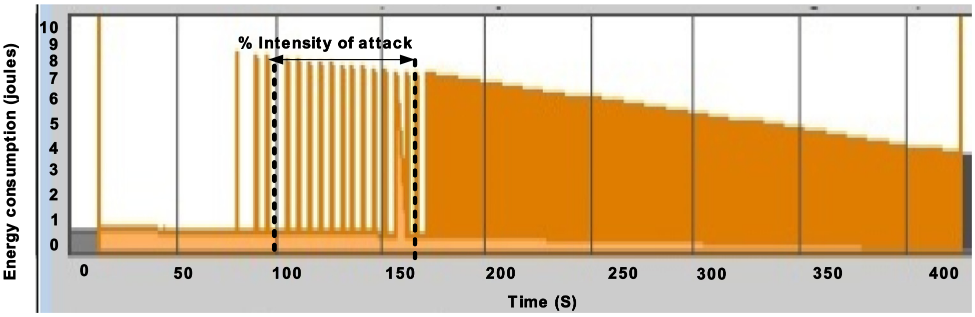


Fig. 7. Victim node's energy level over time.

Table 9
Simulation results of the detection algorithm for DDoS attacks.

Percentage of attack (%)	FLC				Q-learning				FQL				Game-based FQL			
	SP (%)	FP (%)	FN (%)	Utility function	SP (%)	FP (%)	FN (%)	Utility function	SP (%)	FP (%)	FN (%)	Utility function	SP (%)	FP (%)	FN (%)	Utility function
1	49.50	1.90	2.40	56.38	76.00	1.40	1.20	54.40	80.10	1.20	1.10	57.78	83.20	1.20	1.10	60.10
5	49.80	1.98	2.80	56.07	76.70	1.60	1.40	54.53	81.20	1.40	1.30	58.20	83.40	1.30	1.20	60.05
10	50.01	2.00	3.20	55.71	76.90	1.90	1.70	54.08	82.50	1.90	1.70	58.28	84.30	1.50	1.60	60.13
15	51.20	2.04	3.60	56.56	77.80	2.10	2.00	54.25	83.70	2.10	2.00	58.68	85.60	1.70	1.80	60.70
20	50.90	2.40	3.90	55.38	78.00	2.40	2.20	53.90	83.90	2.40	2.20	58.33	87.90	1.90	2.00	62.03
25	51.90	2.80	4.10	55.93	78.90	3.10	2.70	53.38	84.20	2.60	2.30	58.25	88.30	2.10	2.30	61.83
30	52.70	2.90	4.20	56.68	80.20	3.40	3.00	53.75	85.80	2.80	2.60	58.95	89.70	2.40	2.50	62.38
35	49.40	3.00	4.70	51.70	82.80	3.90	3.20	55.00	86.40	2.90	2.70	59.30	90.50	2.60	2.70	62.58
40	49.50	3.01	5.00	51.37	82.90	4.20	3.80	54.18	87.70	3.20	3.00	59.58	91.70	3.10	3.00	62.68
45	50.02	3.20	5.30	51.38	83.70	4.90	4.10	53.78	88.50	3.40	3.20	59.78	92.40	3.20	3.40	62.70
50	51.04	3.50	5.60	51.90	83.90	5.20	4.80	52.93	89.60	3.90	3.50	59.80	94.20	3.30	3.70	63.65
55	50.30	3.70	5.80	50.48	84.90	5.60	5.10	52.98	90.40	4.10	4.00	59.70	96.50	3.50	3.80	65.08
60	49.30	3.70	5.90	49.08	85.00	5.80	5.70	52.25	92.40	4.50	4.30	60.50	98.20	3.70	3.90	66.05
Average	51.20	2.78	4.35	**53.74**	80.59	3.50	3.15	**53.80**	85.88	2.80	2.60	**59.01**	89.68	2.42	2.54	**62.30**
Std. dev.	1.03	0.66	1.15	**2.76**	3.37	1.50	1.47	**0.75**	3.71	1.01	0.98	**0.83**	4.86	0.87	0.98	**1.86**

6.3. Analysis of the game-based FQL IDPS in terms of detection accuracy

The proposed game-based Fuzzy Q-learning (G-FQL) algorithm with the cost function $U = \rho * SP - \beta * FN - \theta * FP$ was compared with existing soft computing methods (Fuzzy Logic Controller, Q-learning, and Fuzzy Q-learning) with respect to the attack detection precision of modeled Denial-of-Service attacks. A comparison between the average utility function and G-FQL with cost maximization indicates that the latter yielded an improvement of 3.29% with 1.86 standard deviation as opposed to the FQL algorithm with 0.83 (Table 9).

It is evident that G-FQL with a cooperative mechanism attained the utmost detection accuracy gain. It can also be inferred from Fig. 8 that detection accuracy per percentage of attack is higher with the G-FQL algorithm than the other methods.

In Fig. 8, the X-axis shows the percentage of malicious nodes in an attack, and the Y-axis indicates the accuracy rate. At higher attack frequencies, the proposed method (Game-based FQL) displays greater accuracy scores.

6.4. Analysis of game-based FQL IDPS in terms of defense rate

The proposed Game-FQL method was weighed against that of Huang et al. (2013), who used the game theory and Markovian IDS with an attack-pattern-mining algorithm. According to Huang et al.'s (2013) empirical results, the defense rate effectiveness of non-cooperative-based Markovian IDS with an attack-pattern mining algorithm for 60% of malicious nodes in a network and two sink nodes ranged between 72% and 97% (Fig. 9). With the proposed game-based FQL IDPS, the successful defense rate was

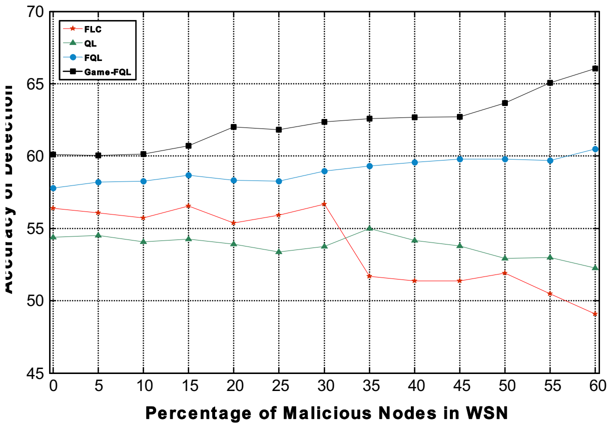


Fig. 8. Comparison of detection accuracy values.

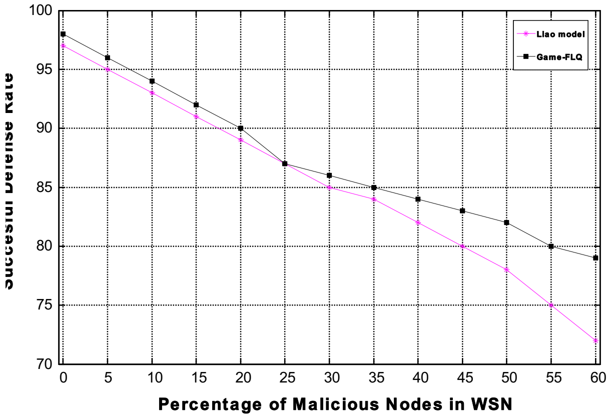


Fig. 9. Game-based FQL in terms of accuracy of defense rate under attack trends.

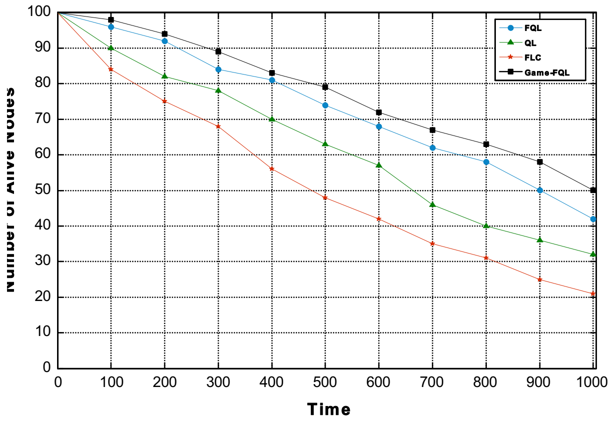


Fig. 10. Number of live sensor nodes during simulation runtime.

between 79% and 98%, as per Fig. 9 as well. It can be concluded that integrating the game theory with the Fuzzy Q-learning algorithm outperforms individual defense schemes.

Fig. 9 points out that the successful defense rate values for Huang et al.'s model (2013) and the proposed methods decreased from 100% to 87% when the anomaly percentage increased. However, the proposed method gained the advantage of a successful defense rate due to the higher percentage of malicious nodes detected compared to Liao's lower success rate. It can thus be deduced that by integrating the game theory with the Fuzzy Q-training method, performance surpasses that of any other individual defense approach.

6.5. *Analysis of game-based FQL IDPS in terms of number of live nodes*

This experiment was conducted to evaluate the performance of the Game-FQL algorithm in terms of number of live nodes during the simulation runtime. In the current scheme, the number of sensor nodes was 200. Fig. 10 displays the number of live nodes for different algorithms throughout simulation runtime. The simulation outcomes indicate the number of live nodes at the end of the simulation time (1000 s), according to which, the number of live sensor nodes in the proposed Game-FQL method is significantly greater than existing algorithms. Game-FQL maintains 50 live nodes against an attack in comparison to 42, 32, and 21 live nodes for FQL, QL, and FLC, respectively.

The procedure of adjusting rules according to FLC-based DDoS attacks is more time-consuming, and the attacker defeats a high number of nodes during FLC detection (Baig and Khan, 2010). Q-learning-based DDoS attack detection is capable of handling minor-class DDoS attacks, but the multi-objective procedure or major features of a DDoS attack consume maximum resources, especially in a real-time environment (Liu, 2008). Fuzzy Q-learning-based DDoS attack detection utilized the min-max fuzzy method to enhance the classification scheme. The min-max fuzzy classifiers perform well with a reduced dataset, but inaccurately when the high volume of traffic increases further and the fuzzy IDPS may crash. In addition, prior knowledge of data distribution is required for the FQL algorithm. In the Fuzzy Q-learning algorithm, observation is limited by one single classifier (Shamshirband et al., 2013). Therefore, this algorithm fails due to high volumes of real-time traffic. In the currently proposed method, the cooperative policy evaluates the proficiency of an agent to optimize the cost function based on weight assignment mechanisms for real-time DDoS attack detection. The countermeasure mechanisms result as modules to be applied in Game-FQL architecture and system implementation to accelerate the detection and defense learning process in a fraction of the usual time. Thus, the Game-FQL preserves a greater number of sensor nodes during simulation.

6.6. *Analysis of game-based FQL IDPS in terms of energy consumption over time*

In this experiment, the energy consumed by the Game-FQL algorithm during DDoS attacks on sensor nodes in comparison to FLC, QL, and FQL is studied. Fig. 11 provides the comparison between the mentioned algorithms in terms of total energy consumed by sensor nodes.

In existing detection, the players (sink node and base station) partake in activities such as local sensing and data reporting, which consume additional energy. The overhead of energy consumption may be considerable if the number of cooperating players or the amount of sensing results in the report is very large. Thus, energy efficiency needs to be considered in cooperative sensing schemes. To address this issue, the cooperative game-based FQL method enhances energy efficiency via optimization.

6.7. *Analysis of the energy consumed by different deployed nodes in the game-based FQL IDPS*

The impact of number of deployed sensor nodes on energy consumption is shown in Fig. 12. It is observed that with an increasing percentage of deployed nodes, the proposed Game-FQL is able to consume the total amount of energy in comparison with FQL, QL, and FLC.

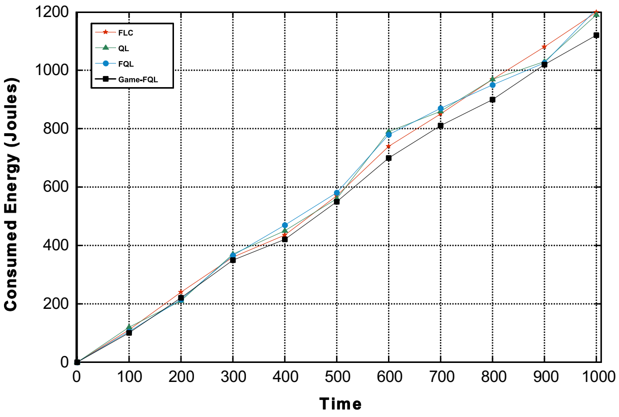


Fig. 11. Total energy consumption versus number of sensor nodes under malicious attack.

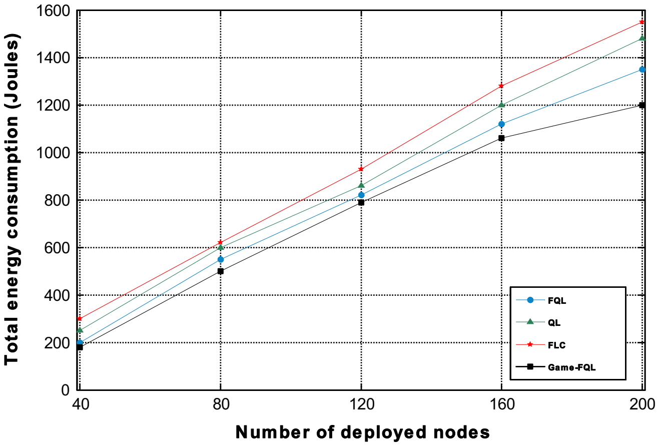


Fig. 12. Total energy consumption versus different numbers of sensor nodes deployed in a network.

Finally, Fig. 12 depicts the total energy consumed with varying numbers of sensor nodes deployed in the network. The experiment was run for 40, 80, 120, 160, and 200 nodes. As expected, when more nodes are present in the network, the energy consumption rate is lower than other comparable methods. This is attributed to the fact that the proposed Game-based FQL agents prefer to maximize their own utility function by means of cooperating learning algorithm to avoid the energy consumption by sensors from each cluster. However, it would be interesting for the cooperative Game-FQL solution to be implemented, for instance, to “BEE-C: A bio-inspired energy efficient cluster-based algorithm for data continuous dissemination in Wireless Sensor Networks” (da Silva Rego et al., 2012), to verify the energy consumption for intrusion detection and prevention.

7. Conclusion

In this paper, the interaction between attackers, sink nodes and the base station was studied, after which a novel Game-based FQL, cooperative game theoretic defense mechanism was proposed. This system combines the cooperative-based game theory with fuzzy Q-learning algorithmic elements. As such, the cooperation between the detection sink node player and response base station players is reinforced to defend against an incoming DDoS attack that may cause congestion and downtime in WSN wireless communication as a result of flooding packets. The Game-FQL model is a triple-player game strategy construed as two-player, providing double defense against a single attacker. It adds confidence and establishes a reputation as extremely apt in tracking an attacker and defending the system. This strategy-based cooperative game adapts to continuous self-learning of past attacks and the behavior in the fuzzy Q-learning decision making process to overcome the attacker. By defining incentives for cooperation and disincentives for fraudulent behavior, it has been determined that repeated interaction sustains cooperation, builds confidence and enhances reputation, something additionally offered by Game-FQL. Game theory-based Fuzzy Q-learning (Game-FQL), as a mechanism in IDPS, is an invaluable tool for increasingly securing next-generation complex heterogeneous computing and networking environments against sophisticated attacks and attackers, beyond what is encountered today. A future initiative is to extend the proposed Game-FQL mechanism by incorporating data from various attack types and sources to further enhance its decision making capabilities in order to thwart existing or new attacks. Also as part of future research work on complementing Game-FQL, studying a network evolutionary algorithm, such as the imperialist competitive algorithm, is considered of utmost importance.

Acknowledgment

This paper is financially supported by the Malaysian Ministry of Education under the University of Malaya High Impact Research Grant – UM.C/HIR/MOHE/FCSIT/12.

References

Abraham, A., Jain, R., Thomas, J., Han, S.Y., 2007. D-SCIDS: Distributed soft computing intrusion detection system. J. Netw. Comput. Appl. 30, 81–98.
Akkaya, K., Younis, M., 2005. A survey on routing protocols for wireless sensor networks. Ad Hoc Netw. 3, 325–349.
Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cayirci, E., 2002. Wireless sensor networks: a survey. Comput. Netw. 38, 393–422.
Alpadin, E., 2010. Introduction to Machine Learning. MIT Press.
Andersen, K.T., Zeng, Y., Christensen, D.D., Tran, D., 2009. Experiments with online reinforcement learning in real-time strategy games. Appl. Artif. Intell. 23, 855–871.
Anisi, M.H., Abdullah, A.H., Razak, S.A., Ngadi, M.A., 2012. Overview of data routing approaches for wireless sensor networks. Sensors 12, 3964–3996.
Anuar, N.B., Papadaki, M., Furnell, S., Clarke, N., 2013. Incident prioritisation using analytic hierarchy process (AHP): Risk Index Model (RIM). Secur. Commun. Netw. 6, 1087–1116.
Arun Raj Kumar, P., Selvakumar, S., 2013. Detection of distributed denial of service attacks using an ensemble of adaptive and hybrid neuro-fuzzy systems. Comput. Commun. 36, 303–319.
Aslam, N., Phillips, W., Robertson, W., Sivakumar, S., 2011. A multi-criterion optimization technique for energy efficient cluster formation in wireless sensor networks. Inf. Fusion 12, 202–212.
Baig, Z.A., Khan, S.A., 2010. Fuzzy logic-based decision making for detecting distributed node exhaustion attacks in wireless sensor networks. In: ICFN ‘10. Second International Conference on Future Networks, 2010. , pp. 185-189.
Baig, Z.A., Sait, S.M., Shaheen, A., 2013. GMDH-based networks for intelligent intrusion detection. Eng. Appl. Artif. Intell. 26, 1731–1740.
Bekmezci, I., Alagöz, F., 2009. Energy efficient, delay sensitive, fault tolerant wireless sensor network for military monitoring. Int. J. Distrib. Sensor Netw. 5, 729–747.
Bicakci, K., Tavli, B., 2009. Denial-of-service attacks and countermeasures in IEEE 802.11 wireless networks. Comput. Stand. Interfaces 31, 931–941.
Chen, Y., Hwang, K., 2006. Collaborative detection and filtering of shrew DDoS attacks using spectral analysis. J. Parallel Distrib. Comput. 66, 1137–1151.
da Silva Rego, A., Celestino, J., dos Santos, A., Cerqueira, E.C., Patel, A., Taghavi, M., 2012. BEE-C: a bio-inspired energy efficient cluster-based algorithm for data continuous dissemination in Wireless Sensor Networks. In: 18th IEEE International Conference on Networks (ICON), 2012, pp. 405–410.
Darwish, A., Hassanien, A.E., 2011. Wearable and implantable wireless sensor network solutions for healthcare monitoring. Sensors 11, 5561–5595.
Das, M.L., 2009. Two-factor user authentication in wireless sensor networks. IEEE Trans. Wireless Commun. 8, 1086–1090.
Elias, J., Martignon, F., Capone, A., Altman, E., 2011. Non-cooperative spectrum access in cognitive radio networks: a game theoretical model. Comput. Netw. 55, 3832–3846.

Feiyi, H., Yang, Y., Liwen, H., 2007. A flow-based network monitoring framework for wireless mesh networks. IEEE Wireless Commun. 14, 48–55.

Ghosal, A., Halder, S., 2013. Intrusion Detection in Wireless Sensor Networks: Issues, Challenges and Approaches, Wireless Networks and Security. Springer, pp. 329–367

Huang, J.-Y., Liao, I.E., Chung, Y.-F., Chen, K.-T., 2013. Shielding wireless sensor network using Markovian intrusion detection system with attack pattern mining. Inf. Sci. 231, 32–44.

Khalil, I., Awad, M., Khreishah, A., 2012. CTAC: Control traffic tunneling attacks' countermeasures in mobile wireless networks. Comput. Netw. 56, 3300–3317.

Khalil, I., Bagchi, S., Rotaru, C.N., Shroff, N.B., 2010. UnMask: Utilizing neighbor monitoring for attack mitigation in multihop wireless sensor networks. Ad Hoc Netw. 8, 148–164.

Kumarage, H., Khalil, I., Tari, Z., Zomaya, A., 2013. Distributed anomaly detection for industrial wireless sensor networks based on fuzzy data modelling. J. Parallel Distrib. Comput. 73, 790–806.

Law, Y.W., Hoesel, L.v., Doumen, J., Hartel, P., Havinga, P., 2005. Energy-efficient link-layer jamming attacks against wireless sensor network MAC protocols. In: Proceedings of the 3rd ACM Workshop on Security of Ad Hoc and Sensor Networks. ACM, Alexandria, VA, USA, pp. 76–88.

Law, Y.W., Palaniswami, M., Hoesel, L.V., Doumen, J., Hartel, P., Havinga, P., 2009. Energy-efficient link-layer jamming attacks against wireless sensor network MAC protocols. ACM Trans. Sens. Netw. 5, 1–38.

Li, B., Batten, L., 2009. Using mobile agents to recover from node and database compromise in path-based DoS attacks in wireless sensor networks. J. Netw. Comput. Appl. 32, 377–387.

Li, N., Zhang, N., Das, S.K., Thuraisingham, B., 2009. Privacy preservation in wireless sensor networks: a state-of-the-art survey. Ad Hoc Netw. 7, 1501–1514.

Li, Z., Wang, W., 2012. Node localization through physical layer network coding: Bootstrap, security, and accuracy. Ad Hoc Networks 10, 1267–1277.

Liu, L., 2008. System and Method for Distributed Denial of Service Identification and Prevention. Google Patents.

Liu, X., 2012. a survey on clustering routing protocols in wireless sensor networks. Sensors 12, 11113–11153.

Lung, C.-H., Zhou, C., 2010. Using hierarchical agglomerative clustering in wireless sensor networks: an energy-efficient and flexible approach. Ad Hoc Netw. 8, 328–344.

McGregory, S., 2013. Preparing for the next DDoS attack. Netw. Secur. 2013, 5–6.

Mirkovic, J., Reiher, P., 2005. D-WARD: a source-end defense against flooding denial-of-service attacks. IEEE Trans. Dependable Secure Comput. 2, 216–232.

Misra, S., Abraham, K.I., Obaidat, M.S., Krishna, P.V., 2009. LAID: a learning automata-based scheme for intrusion detection in wireless sensor networks. Secur. Commun. Netw. 2, 105–115.

Misra, S., Krishna, P.V., Agarwal, H., Saxena, A., Obaidat, M.S., 2011. A learning automata based solution for preventing distributed denial of service in internet of things, internet of things (iThings/CPSCom). In: 2011 International Conference on and 4th International Conference on Cyber, Physical and Social Computing. IEEE, pp. 114–122.

Misra, S., Vaish, A., 2011. Reputation-based role assignment for role-based access control in wireless sensor networks. Comput. Commun. 34, 281–294.

Muñoz, P., Barco, R., de la Bandera, I., 2013. Optimization of load balancing using fuzzy Q-learning for next generation wireless networks. Expert Syst. Appl. 40, 984–994.

Naserian, M., Tepe, K., 2009. Game theoretic approach in routing protocol for wireless ad hoc networks. Ad Hoc Networks 7 (3), 569–578.

Patel, A., Taghavi, M., Bakhtiyari, K., Celestino, J., Jr., 2013. An intrusion detection and prevention system in cloud computing: a systematic review. J. Net. Comput. Appl. 36, 25–41, http://dx.doi.org/10.1016/j.jnca.2012.08.007.

Qazi, S., Raad, R., Mu, Y., Susilo, W., 2013. Securing DSR against wormhole attacks in multirate ad hoc networks. J. Netw. Comput. Appl. 36, 582–592.

Qiu, W.-d., Zhou, Y.-w., Zhu, B., Zheng, Y.-f., Gong, Z., 2013. Key-insulated encryption based group key management for wireless sensor network. J. Cent. South Univ. 20, 1277–1284.

Rolla, V.G., Curado, M., 2013. A reinforcement learning-based routing for delay tolerant networks. Eng. Appl. Artif. Intell. 26, 2243–2250.

Schaffer, P., Farkas, K., Horváth, Á., Holczer, T., Buttyán, L., 2012. Secure and reliable clustering in wireless sensor networks: a critical survey. Comput. Netw. 56, 2726–2741.

Seo, D., Lee, H., Perrig, A., 2013. APFS: adaptive probabilistic filter scheduling against distributed denial-of-service attacks. Comput. Secur. (In Press)

Shamshirband, S., Anuar, N.B., Kiah, M.L.M., Patel, A., 2013. An appraisal and design of a multi-agent system based cooperative wireless intrusion detection computational intelligence technique. Eng. Appl. Artif. Intell. 26, 2105–2127.

Shamshirband, S., Kalantari, S., Bakhshandeh, Z., 2010. Designing a smart multi-agent system based on fuzzy logic to improve the gas consumption pattern. Scientific Research and Essays 5 (6), 592–605.

Shen, S., Han, R., Guo, L., Li, W., Cao, Q., 2012. Survivability evaluation towards attacked WSNs based on stochastic game and continuous-time Markov chain. Appl. Soft Comput. 12, 1467–1476.

Shen, S., Li, Y., Xu, H., Cao, Q., 2011. Signaling game based strategy of intrusion detection in wireless sensor networks. Comput. Math. Appl. 62, 2404–2416.

Shoham, Y., Leyton-Brown, K., 2009. Multiagent Systems: Algorithmic, Game-theoretic, and Logical Foundations. Cambridge University Press

Sun, D., Huang, X., Liu, Y., Zhong, H., 2013. Predictable Energy Aware Routing based on Dynamic Game Theory in Wireless Sensor Networks. Comput. Electr. Eng. 39, 1601–1608.

Tan, H., Ostry, D., Zic, J., Jha, S., 2013. A confidential and DoS-resistant multi-hop code dissemination protocol for wireless sensor networks. Comput. Secur. 32, 36–55.

Tsunoda, H., Ohta, K., Yamamoto, A., Ansari, N., Waizumi, Y., Nemoto, Y., 2008. Detecting DRDoS attacks by a simple response packet confirmation mechanism. Comput. Commun. 31, 3299–3306.

Wang, H., Jin, C., Shin, K.G., 2007. Defense against spoofed IP traffic using hop-count filtering. IEEE/ACM Trans. Netw. 15, 40–53.

Xing, K., Srinivasan, S., Rivera, M.M., Li, J., Cheng, X., 2010. Attacks and countermeasures in sensor networks: a survey. In: Huang, S.C.H., MacCallum, D., Du, D.-Z. (Eds.), Network Security. Springer, US, pp. 251–272

Xu, X., 2010. Sequential anomaly detection based on temporal-difference learning: principles, models and case studies. Appl. Soft Comput. 10, 859–867.

Zhou, C.V., Leckie, C., Karunasekera, S., 2010. A survey of coordinated attacks and collaborative intrusion detection. Comput. Secur. 29, 124–140.