Hindawi
Wireless Communications and Mobile Computing
Volume 2017, Article ID 7623742, 15 pages
https://doi.org/10.1155/2017/7623742

Research Article

Performance Comparison of Reputation Assessment Techniques Based on Self-Organizing Maps in Wireless Sensor Networks

Sabrina Sicari,[1] Alessandra Rizzardi,[1] Luigi Alfredo Grieco,[2] and Alberto Coen-Porisini[1]

[1]Dipartimento di Scienze Teoriche e Applicate, Università degli Studi dell'Insubria, Via Mazzini 5, 21100 Varese, Italy
[2]Department of Electrical and Information Engineering, Politecnico di Bari, Via Orabona 4, 70125 Bari, Italy

Correspondence should be addressed to Luigi Alfredo Grieco; alfredo.grieco@poliba.it

Received 9 June 2017; Revised 19 September 2017; Accepted 4 October 2017; Published 28 November 2017

Academic Editor: Pierre-Martin Tardif

Many solutions based on machine learning techniques have been proposed in literature aimed at detecting and promptly counteracting various kinds of malicious attack (data violation, clone, sybil, neglect, greed, and DoS attacks), which frequently affect Wireless Sensor Networks (WSNs). Besides recognizing the corrupted or violated information, also the attackers should be identified, in order to activate the proper countermeasures for preserving network's resources and to mitigate their malicious effects. To this end, techniques adopting Self-Organizing Maps (SOM) for intrusion detection in WSN were revealed to represent a valuable and effective solution to the problem. In this paper, the mechanism, namely, Good Network (GoNe), which is based on SOM and is able to assess the reliability of the sensor nodes, is compared with another relevant and similar work existing in literature. Extensive performance simulations, in terms of nodes' classification, attacks' identification, data accuracy, energy consumption, and signalling overhead, have been carried out in order to demonstrate the better feasibility and efficiency of the proposed solution in WSN field.

1. Introduction

A Wireless Sensor Network (WSN) includes a wide number of sensor nodes in charge of acquiring various information from the environment where they are placed in. Such information are then transmitted by means of multihop wireless connectivity to a collection center, named sink, in order to be stored and processed, in accordance with the scope of the specific application domain [1]. Mostly due to the wireless nature of the radio channel, WSNs are exposed to multiple kinds of attack [2], which could potentially compromise the following: (i) the confidentiality and the integrity of the exchanged data; (ii) the nodes' identity; (iii) the network's resource consumption (e.g., the nodes' batteries); (iv) the correct execution of the established routing protocol. As a consequence, such threats could hinder the correctness of the data collected by the sink. Moreover, sensor nodes present other weaknesses: (i) they have power and computational constraints, which limit the resources they can employ for performing the network activities; (ii) they are often deployed in unsupervised sites, thus increasing the risk of external and uncontrolled attacks.

Many solutions have been proposed to cope with such issues [3], but few completely address all the threats just described. What emerges is the need of embedding orthogonal security systems into the network stack to monitor sensor node behavior and detect anomalies. In this way, the network would be able to quickly respond to the attacks by isolating the malicious nodes, save resources, and preserve data confidentiality and integrity. In this direction, the use of machine learning techniques seems to represent a valuable, accurate, and efficient solution. In fact, several trust and reputation systems based on different machine learning techniques (e.g., use of neural networks, clustering, and learning automata) have been proposed in the last years [4]. Such techniques allow the sink to analyze the sensor nodes' behavior with the final aim of inferring about their trustworthiness.

In this context, the secure framework, named GoNe, presented in [5] and targeted to sensor nodes' reputation assessment in WSN, has been conceived. It adopts a well-known machine learning technique, based on Self-Organizing Maps (SOM) [6]. The reputation assessment mechanism based on SOM, also known as Kohonen network, is conceived for unsupervised neural architectures and is especially suitable for environments which present a huge amount of data, such as WSN, in particular those deployed in hostile context, difficult to reach in a reasonable time. SOM are also relatively fast and not expensive in terms of computational consumption. Based on this assumption, the use of SOM is considered in the presented scenario as the best technique for reaching the required purposes and contributing to the existing solutions in WSN intrusion detection field. The improvements regard various requirements, such as the following: the detection and blocking of malicious attacks, the energy saving, the end-to-end data security, and the congestion control.

In order to demonstrate GoNe robustness and effectiveness in terms of false positive/negative rate of node classification, data accuracy, energy consumption, packet delay, and overhead, this paper proposes a performance comparison with respect to the relevant solution presented in Bankovic et al. [7], which is the only one based on SOM available in literature, apart from GoNe. It is worth noting that GoNe presents various peculiarities with respect to existing works, such as the following:

(i) It guarantees, in addition to the detection of multiple kinds of attacks, the security and the privacy of the data managed by the WSN (i.e., in terms of confidentiality, anonymity, and integrity).

(ii) It adopts a data aggregation technique based on homomorphic encryption [8] for reducing the amount of transmitted information and avoiding network congestion.

(iii) It is able to perform a periodic assessment of the nodes, while Bankovic et al. technique runs this task only once.

Such features, along with the results obtained by extensive simulations, make GoNe a valuable, efficient, and promising solution in recognizing malicious attack, preserving network's resources, and protecting the data in WSN. In fact, GoNe not only correctly classifies the reputation of the sensor nodes but also mostly avoids the cases of false negatives/positives. At the same time, power consumption of sensor nodes, packets' arrival delay, and packets' losses are drastically reduced. Finally, it allows identifying a wider range of attacks with respect to Bankovic et al. and guarantees more accurate data to the sink.

The rest of the paper is organized as follows: Section 2 investigates the state of the art and highlights the contribution of this paper with respect to the available literature; Section 3 describes the network reference scenario and explains the threat model. Section 4 introduces the solutions proposed for GoNe and Bankovic et al. techniques; Section 5 presents the simulation scenarios and the results, while Section 6 ends the paper and draws future research.

2. Related Works

In a WSN, sensor nodes have to acquire, store, process, and communicate information sensed by a target environment. Such activities must be performed in a controlled and secure manner, in particular in relation to the application context and the level of sensitivity associated with the involved data. A WSN is exposed to different kinds of attacks, which regard both to the violation of the data transmitted or stored in the nodes themselves and to network failures, such as packets' flooding or blocking forwarding. A Denial of Service (DoS) attack may be suspected if a node receives a high number of packets from a neighboring node. Another kind of situation may occur when a node presents a high percentage of dropped packets and few packets forwarded; in this case the node could perform a neglect and greed attack or could be faulty. Other attacks may be directed to the routing protocols or to the node identity (i.e., clone, sybil) [2, 7]. As an example, in networks using hierarchical routes, a node may start to send packets to all nodes in its range, instead of the neighboring nodes established by the routing algorithm, thus increasing network traffic and causing higher network resource consumption.

Once an attack is detected, it is necessary to mitigate its actions or isolate the ill-behaved node or nodes from the network. Hence, to ensure security in WSNs, three strategies could be adopted, as described in the next sections.

2.1. Attacks to Confidentiality, Integrity, and Identity.

The first strategy regards the confidentiality and the integrity of the information handled within the network. Some countermeasures are ciphering the data from the time when they are stored in the sensing node and transmit them in an encrypted form until they reach the sink, therefore guaranteeing an end-to-end encryption mechanism. Several cryptography schemes have been proposed in WSN field, such as [8, 9]. A previous work by the authors is the secure and energy-efficient framework, named *SETA* (a SEcure sharing of TAsks in clustered Wireless Sensor Networks) [10], conceived for preserving data integrity and confidentiality in a clustered WSN. *SETA* allows evaluating if data has been violated or not; in this way, the sink can discard the compromised information. The main drawback of *SETA* is that it is vulnerable to threats to node resources or to the routing protocol (e.g., flooding, DoS). Such issues are solved by GoNe.

The second strategy is about the identity issue; in addition to the encryption techniques mentioned before, other solutions exploit node locations [11–14], which help in determining the malicious nodes on the basis of their positions with respect to the normal ones. As an example, *Verifiable Multilateration* (*VM*) [12] uses localization in order to evaluate the reliability of the sensor nodes and identifying possible outliers. Note that [13, 14] are tailored to secure wireless communications of mobile devices, but the defined methods could be applied to WSN.

2.2. Malicious Node Detection.

The third strategy is related to the monitoring of network activities, since the countermeasures presented above are not sufficient to detect malicious attacks towards nodes resources or changes in the traffic flow. To face such issues, several Intrusion Detection Systems (IDS) have been proposed [15]. IDS typically logs information about network behavior and reports an alarm in case of anomaly; an IDS may rely on the following techniques:

(i) Signature based detection: the current features of network behavior are compared with predefined attacks patterns of misbehavior.

(ii) Anomaly based detection: the ordinary network behavior is determined as a baseline on which anomalies are detected; therefore, the IDS can adapt itself to the specific environment issues; what is out of the ordinary can be defined with respect to the history or training data.

(iii) Reputation based detection: a reputation manager has to detect nodes exhibiting a selfish behavior rather than violating security.

Focusing on the reputation based detection, literature provides many solutions addressing the WSN context, which enables performing an assessment of the behavior of the nodes [16–18], by means of both trust and reputation systems. It is worth remarking that reputation and trust concepts are different. In particular, reputation is a measure of node behavior, which is influenced by their network activities, the accuracy of the provided data, and so on, while trust is a more subjective measure and it is mostly based on past experience. Whatever parameter is considered (reputation or trust), a mechanism able to dynamically classify the nodes belonging to the network and to reconfigure itself in order to support network changes and also the different kinds of attacks previously discussed must be defined. Note that the cases in which a node is recognized as malicious even if it is well-behaved (i.e., false negative) are not few. There are different reasons, such as the proximity to a malicious node or simply a fault; in the last situation, the node may restart to work normally.

Among the available approaches, able to assess node behavior, [19] presents an overview of the main trust and reputation models and suggest some guidelines for the development of standardized solutions. The same authors in [20] propose a privacy-aware trust and reputation model with the scope of advising a domain when it has to decide whether to exchange some necessary information with another domain or not, depending on its trustworthiness and reputation. In [21], a decision-making mechanism for trust assessment in multiagent systems has been carried out and a test-bed has been built able to evaluate and compare different trust models. The work [22] provides a survey about the existing trust methodologies sharing between trust models for securing routing and for securing data; moreover it categorizes various types of attack and challenges related to trust schemes. In [11] it is suggested to perform a probabilistic location verification algorithm in order to retrieve trustworthy data from sensor nodes and create several trust levels in the network. The authors of [23, 24] exploit bioinspired algorithms for their reputation system. In [18] a trust management system is proposed to defend against attacks inside the WSN which uses beta distribution to evaluate the different nodes' credibility. Finally, many works are based on computational intelligence techniques [25] or propose Fuzzy logic approaches for detecting intrusion in WSN [26–30] or learning algorithms and neural networks (i.e., clustering, SOM, and learning automata) [5, 7, 31–33].

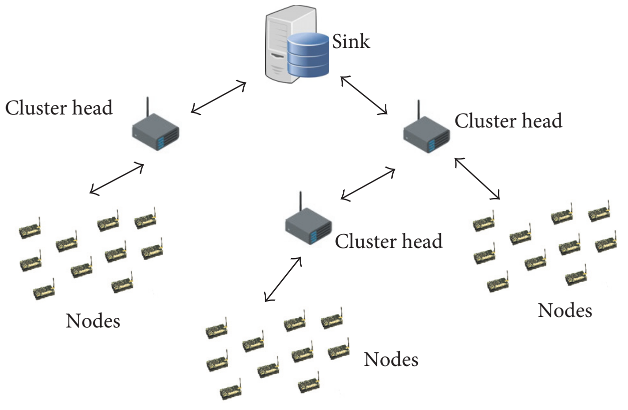


Figure 1: Network reference architecture.

The focus, with GoNe, is on the last approach, due to the potentialities of learning algorithms and neural networks in recognizing in an inexpensive and fast way the misbehavior happening within the WSN. Note that such a feature also concerns large-scale environments, as demonstrated in [34], which shows a performance comparison among different IDS techniques. It reveals that the approaches based on neural networks outperform the others. For such a reason, this work started in this direction.

The goal of this paper is twofold: on the one side, the aim is to demonstrate that GoNe overcomes the other existing solutions based on SOM reputation mechanism in terms of network performance (i.e., overhead, delay, lost packets, and energy consumption), data accuracy, and node classification; on the other side, a more practical comparative approach is presented, with respect to most of the existing works, whose comparison among different techniques is limited to a theoretical discussion (i.e., qualitative and not quantitative). Finally, this work also points out the potentialities of SOM in IDS field.

3. Reference Scenario

The reference scenario for both GoNe and Bankovic et al. is a clustered WSN, based on a wireless multihop mesh backbone [35]. Each cluster is made of a variable number of sensors and one mesh router acting as Cluster Head (CH), directly or indirectly connected to the sink through the wireless mesh backbone (see Figure 1). Sensor nodes are very constrained in terms of energy and processing resources, while CHs can be assumed to be (i) grid powered (or with a huge energy availability) and (ii) able to run more complex algorithms. Such considerations are more relevant for GoNe than for Bankovic et al., since GoNe adopts such a hierarchy to assign the different tasks among nodes, in the following way:

(i) Sensor nodes will only execute the sensing and the data encryption activities.

(ii) CHs verify the integrity of the received data and, in case of no violation, aggregate the data according to the congestion level of the network, as proposed in the secure framework, named SETA [10].

(iii) The sink collects the data provided by the WSN and assesses the reputation level of the sensor nodes, thus detecting the misbehaving ones, as described in Section 4. This happens in both the frameworks (i.e., GoNe and Bankovic et al.). In fact, the sink has no resources constraints and therefore it can store all the historical information regarding WSN behavior for ongoing and future analysis.

To guarantee an end-to-end security, in GoNe some information contained in the transmitted packets is encrypted by using the *Message Digest MD5* algorithm with proper keys. *MD5* has been adopted due to its suitability for WSN in terms of memory usage and resource power consumption [36], for example, with respect to *SHA* and *AES*, which require a higher computational effort.

As regards the data integrity requirement, the countermeasure adopted by GoNe is a hashing procedure. More in detail,

(1) for each received message, the CHs calculate the hash of the sensed or aggregated data contained in the message itself;

(2) the obtained output is encrypted with the hash key associated with the node that generated the message (also extracted from the message);

(3) if such a result matches the field that contains the encrypted hash of the data into the received message, a security violation has not occurred.

Note that since also the hash is encrypted, another security level is added in order to avoid attacks which can modify both the hash and the related data. Concerning secure aggregation, sensor nodes make use of homomorphic stream ciphers, as presented in [8], which allow the CHs to aggregate data without deciphering them. The approach presented in GoNe adopts data aggregation at CH level to avoid traffic congestion: when the transmission queue builds up, data therein are aggregated to keep the queue length under its maximum limit. In fact, the aggregation strategy iteratively arranges the enqueued packets in a proper number of aggregation groups, whose contents are merged into a single packet. For further details about the encryption technique, the key management, and the aggregation process, please refer to [10].

The messages, exchanged in the WSN containing the sensed data, are denoted by $m_{n,q}$ where n refers to the node that generated and transmitted the message, whereas q uniquely identifies the message among those generated by the node n. In this way, the sink can keep track of the messages generated and transmitted over the network.

The communications are supposed to be multihop from the sensor nodes to the CHs and, then, to the sink or vice versa. The information related to such a path is denoted in the message $m_{n,q}$ as L_n, that is, a list of the nodes which forwarded the data itself. As regards Bankovic et al., the authors do not specify how communications happen within their proposed framework.

3.1. Threat Model.

The threat model considered in this paper is detailed hereby, supposing that sensor nodes may be deployed in unsafe environments. Furthermore, nodes are assumed to have the same communication ranges, while the size of the packets exchanged among nodes is fixed (see simulation scenario in Section 5).

Therefore, WSN may be exposed to eavesdropping and masking attacks; they are counteracted, in GoNe, by means of a two-level encryption approach exploiting a hashing procedure, as introduced in Section 3, while, in Bankovic et al., no action is undertaken in order to directly face such a kind of misbehavior (i.e., the sink is unable to recognize violated packets). Note that, in GoNe, the hash calculation includes the current timestamp (i.e., the instant time of packet generation), thus preventing replay attacks. Moreover, sensor nodes are not synchronized (i.e., they do not have perfect clocks and they do not generate packets at the same instant time). Such features, along with the different keys owned by sensor nodes, cause that the same event (e.g., the detection of a particular condition) reported by different nodes will generate diverse encrypted hash. Such a feature greatly improves the resistance of GoNe to brute-force attacks, aimed at discovering the keys used by nodes for data encryption. Instead, Bankovic et al. technique only relies on the capability of SOM for recognizing replay attacks.

Other kinds of attacks included in this analysis are those directed to the routing protocol or the network resources, for example, flooding, DoS, and wormhole [22, 37, 38]. Both GoNe and Bankovic et al. techniques face them by means of SOM functionalities, which are examined in depth in Section 4.

4. Reputation Frameworks

4.1. Self-Organizing Maps.

The reputation algorithm used both by GoNe and by Bankovic et al. is based on SOM. SOM is a kind of Artificial Neural Network (ANN), which is trained by means of unsupervised learning [6]. It uses a neighborhood function to preserve the topological properties of the input space; in this way, both similarities and anomalies of the node behavior can be recognized. For such reasons SOM is particularly suitable for WSN application case. A value for the reputation score in the range $[0, 1]$ is finally derived for each sensor node, where 0 is the lowest possible value and means that no confidence is associated with the node, while 1 is the highest possible value and means that there is a complete confidence in that node.

More in detail, SOM is able to organize various features into an internal representation, consisting of two principal layers (see Figure 2):

(i) The input layer: it takes the features $V_1 \cdots V_n$ as input signals for the neurons of the SOM; in particular, each neuron is directly connected to all the neurons in the output layer (as suggested by the arrows in Figure 2)

(ii) The output layer: it provides the reputation score update. At each iteration of the algorithm the weights, $[W_1 \cdots W_n]$, among the input and output neurons, are calculated and updated; such an adjustment is a

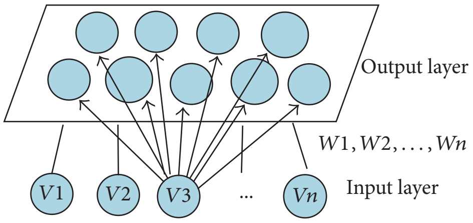


Figure 2: Self-Organizing Map scheme.

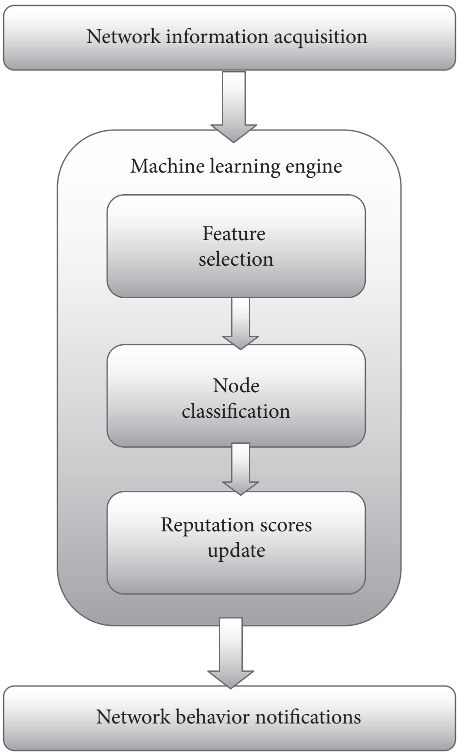


Figure 3: Main steps of the machine learning algorithm.

linear combination of input vector and current weight vector, as shown by the scoring function in Equation (1).

$$W(t+1) = W(t) + R(t)(V(t) - W(t)), \tag{1}$$

where W is the weight, t represents the instant time, R is a positive number less than 1, named *learning rate* (which decreases with time), and V is the current input vector. A scheme of a typical SOM is shown in Figure 2, while, in Figure 3, a flow chart of the main steps of the machine learning algorithm is sketched. As shown in Figure 3, the network nodes themselves acquire information about their activities; then the machine learning engine, whose operations have just been described, processes such information in order to extract the useful features and, as a consequence, classify the sensor nodes and update their reputation. The final step consists in informing the network nodes of the changes that happened.

Note that a key feature of SOM is that the neighborhood nodes participate in the process of adaptation (i.e., learning). For this reason, SOM finds application in many contexts, such as recognition and identification (e.g., medical diagnosis, face recognition), data mining, monitoring, and control (e.g., e-mail spam filtering, vehicle control), and forecasting and prediction (e.g., financial applications).

In the remainder of this section, GoNe and Bankovic et al. frameworks will be detailed.

4.2. GoNe. GoNe aims not only at identifying possible data violations, as described in Section 3 about integrity verification, but also at accurately detecting the malicious nodes. To this end, the machine learning engine of Section 4 is introduced at the sink level, with the scope of isolating misbehaving nodes by evaluating their reputation during the network activity. In particular, SOM takes as inputs the following features:

(i) Regarding the network usage, the number of messages received/generated/forwarded/dropped by a node, the average packet arrival time, and the number of messages received by specific neighbor nodes

(ii) Regarding the computing resources, the memory, and the CPU utilization.

Note that the features related to the network usage allow monitoring the unusual traffic in a given neighborhood. In fact, as pointed out in Section 4.1, SOM are able to recognize anomalies in the monitored areas of the map itself. In order to activate the protocol, each node has to periodically send to the CH of its cluster a packet with the following fields:

$$p_{s,q} = \left(n_{s,q}, P_r, P_g, P_f, P_d, Mem, Cpu\right), \tag{2}$$

where

(i) $n_{s,q}$ is the couple (n_s, q_s). n_s identifies the sensor node which generated the packet; q_s identifies such a message among those (of the same type, not considering the packets $m_{n,q}$ containing the sensed data, presented in Section 3) transmitted by n_s. Note that this field is kept unchanged among transmissions;

(ii) P_r is the number of packets received by the node n_s until the instant t_n, when such a message was generated;

(iii) P_g is the number of packets generated by the node n_s until t_n;

(iv) P_f is the number of packets forwarded by the node n_s until t_n;

(v) P_d is the number of packets dropped by the node n_s until t_n;

(vi) Mem is the percentage of filling of the node's buffer at the time t_n;

(vii) Cpu is the CPU utilization until t_n, measured in MIPS (Million Instructions Per Second).

In order to guarantee the anonymity of the nodes which generated the packets and the confidentiality and integrity of the transmitted information (which could be maliciously modified), all the fields contained in $p_{s,q}$ are encrypted by the sensor nodes with a group signature [39, 40] shared only with the sink; such a scheme allows the group's members (in this case, the nodes which belong to the clusters) to sign the messages on behalf of the whole group without revealing the node identity; only the group manager (i.e., the sink) can open the signature and trace the original signer. Another parameter considered by the sink is the average packets arrival time of the nodes, indicated as $P_{\mathrm{avg}}[k]$, where k represents the number of nodes in the considered cluster. Note that all this information represents the input signals for the SOM neurons.

Once SOM has calculated the score for each sensor node, the other two tasks have to be executed by the sink: (i) the node classification in one of the three categories (i.e., *normal*, *unknown*, and *malicious*) and (ii) the update of the reputation scores, which has to be communicated to all the sensor nodes, in order to activate the proper countermeasures in case of nodes classified as malicious. In the initial phase of the network, the scores are set to 0.5, which is the average value between the two limits 0 and 1. Nodes are then classified in three categories, depending on the scores determined by the machine learning algorithm:

(i) *Normal*, when the associated score is greater than 0.6

(ii) *Unknown*, when the associated score is in the range [0.4, 0.6]

(iii) *Malicious*, when the associated score is less than 0.4.

The classification ranges [0; 0.4)–[0.4; 0.6]–(0.6; 1] have been determined through simulations, which demonstrated that such ranges optimize the node classification in terms of false positive/negative rate (more details are available in [5]).

At each iteration of the classification phase, once the sink notices relevant updates in the node reputation, it informs the CHs about the changes of confidence towards the detected nodes, and then the CHs inform the sensor nodes of their cluster through a proper message, as follows:

$$sr = \left(r_{i,q}, repList\,[n_s]\,[rep_s]\right), \tag{3}$$

where

(i) $r_{i,q}$ is the couple (r_i, q_i). r_i identifies the CH which generated the packet; q_i identifies such a message among those (of the same type) transmitted by r_i. Note that this field is kept unchanged among transmissions;

(ii) $repList[n_s][rep_s]$ represents the list containing the couples of values referred to node/reputation, in which n_s represents a particular node belonging to the CH's cluster, while rep_s is the reputation score associated with the node, provided by the sink.

Figure 4 summarizes the steps of the GoNe approach just described, emphasizing the tasks performed by the machine learning engine.

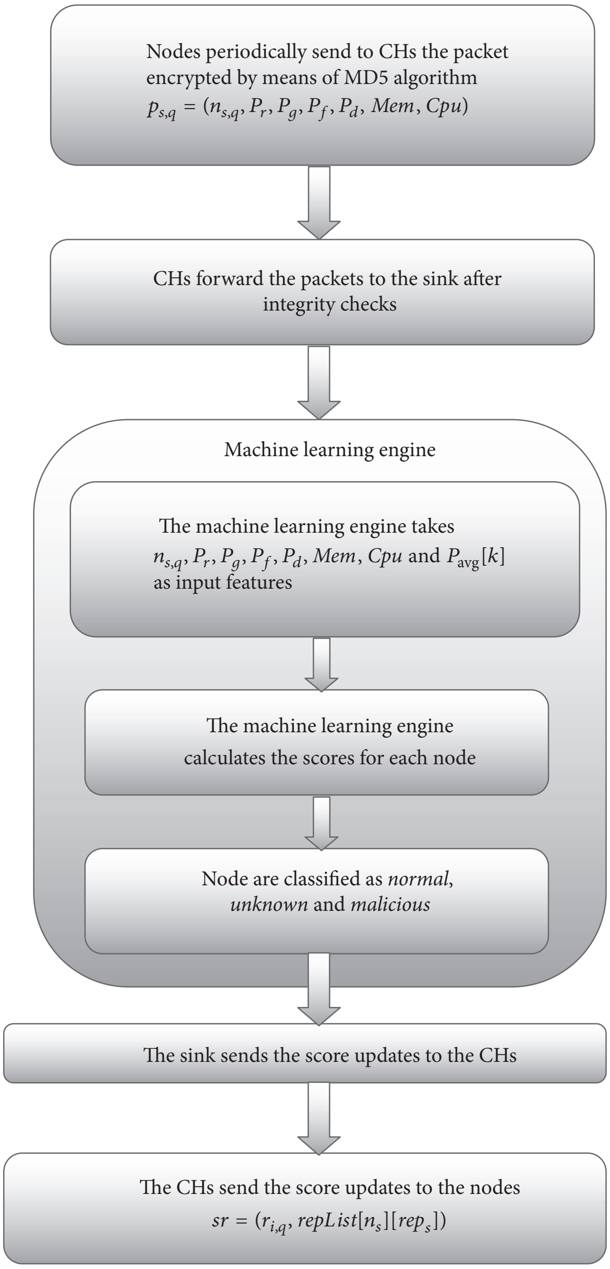

Nodes periodically send to CHs the packet encrypted by means of MD5 algorithm $p_{s,q} = (n_{s,q}, P_r, P_g, P_f, P_d, Mem, Cpu)$
CHs forward the packets to the sink after integrity checks
Machine learning engine
The machine learning engine takes $n_{s,q}, P_r, P_g, P_f, P_d, Mem, Cpu$ and $P_{\mathrm{avg}}[k]$ as input features
The machine learning engine calculates the scores for each node
Node are classified as *normal*, *unknown* and *malicious*
The sink sends the score updates to the CHs
The CHs send the score updates to the nodes $sr = (r_{i,q}, repList[n_s][rep_s])$

Figure 4: Main steps of GoNe approach.

The field n_s, which represents the node identifier, has been encrypted by the sink with the proper group key and forwarded by the CH guaranteeing the end-to-end anonymity. Note that the CHs are considered trusty. Once a sensor node receives such a type of packet, it has to store the retrieved information in its own local table T; this table aims at containing the couples node reputation assigned to all the nodes belonging to the same cluster. As a consequence, at each data packet reception, sensor node verifies the reputation scores stored in T regarding the nodes which are in the field L_n. As just said in Section 3, L_n is the list of the nodes which handled the data. Also the list of the nodes contained in L_n are encrypted with the group key; therefore the nodes belonging to the network are able to establish the associated score. Three different situations may occur:

(i) In case of nodes classified as *normal*, the node processes the packet in the standard way.

(ii) In presence of nodes classified as *unknown*, the CH does not aggregate the data coming from such nodes, in order to preserve the data accuracy; such a kind of packets is forwarded to the sink, which will decide whether to use them or not.

(iii) If almost a node in the fields L_n is identified as *malicious*, then the packet is immediately dropped, as well as all the other kinds of message sent by the nodes recognized as malicious, in order to isolate them from the network.

Note that the scope of the learning algorithm is to minimize, if not avoid, the number of nodes classified as *unknown* [41].

4.3. *Bankovic et al.*

In Bankovic et al. [7] the malicious node detection is also based on SOM, but, unlike GoNe, the considered features concern

(i) the identification of the outlying nodes;

(ii) the identification of the data generated and transmitted by the no-outlying nodes.

As regards the first feature, the average Euclidean distance *MD* of each node to the other (or its closest neighborhood) is calculated; therefore, the node or the nodes for which *MD* is significantly greater than the others is/are declared to be outlying node/nodes and the corresponding inputs are considered to be anomalies. Regarding the second feature, the quantization error *QE* of each data with respect to the data provided by the neighboring nodes and the CH is calculated; hence, if the *QE* of a node is greater than the rest of the same node, it is considered to be the proof of an anomaly in the current input.

Starting from the *MD* values, the reputation value *repMD* is defined in

$$repMD = \frac{\left(\max MD_{\text{value}} - anoScMed\right)}{\max MD_{\text{value}}}, \quad (4)$$

where

(i) $\max MD_{\text{value}}$ is the maximum median distance of the calculated *MD;*

(ii) *anoScMed* is the *MD* value for the best matching unit of the current input.

Note that *repMD* takes values in the range $[0, 1]$ and, in particular, the nodes which are closer to the rest have a higher reputation and vice versa.

```
if (repMD < 0.5)
    rep = repMD;
else
    rep = repQE;
```

Listing 1: Current reputation.

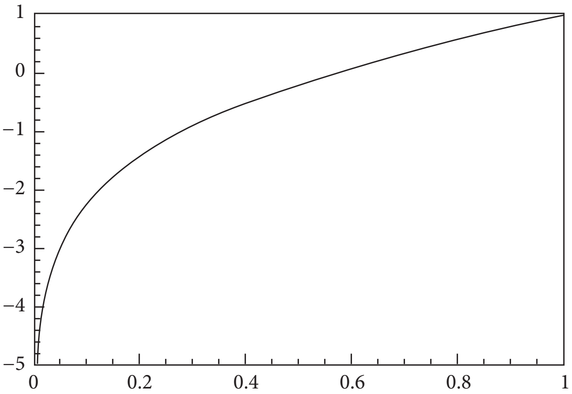


Figure 5: Reputation update function.

As regards the reputation value *repQE*, the median *medQE* of *QE* values of all the nodes is calculated, and then the *QE* value for each data input QE_{value} is evaluated in order to obtain the reputation value *repQE*, as presented in

$$repQE = \frac{QE_{\text{value}}}{medQE}. \quad (5)$$

As a consequence, the current reputation value *rep* for each node is established, as indicated in Listing 1.

It is worth remarking that also Bankovic et al. chose 0.5 as threshold, which is the median value between 0 and 1. The final step is the update of the node reputation, which takes into account the previous reputation value, as presented in

$$cumQE = rep_{(t-1)i} + rep_{ti} + \log\left(0.99 * rep_{ti}\right), \quad (6)$$

where

(i) *cumQE* represents the cumulative *QE*; note that such a result could take values greater than 1 or less than 0; in such cases *cumQE* is truncated to 1 or 0, respectively;

(ii) $rep_{(t-1)i}$ is the reputation of the ith node at the time $t-1$;

(iii) rep_{ti} is the reputation of the ith node at the time t;

(iv) The logarithmic function represents the cumulative distribution of reputation in a way in which, as shown in Figure 5, for values lower than 0.3 the reputation falls down quickly, while for values higher than 0.65 the function significantly arises; finally, the reputation presents small changes for values in the range [0.5, 0.65].

Note that, with respect to GoNe, the classification only distinguishes nodes in *normal* and *malicious* ones, since the threshold is fixed to 0.5.

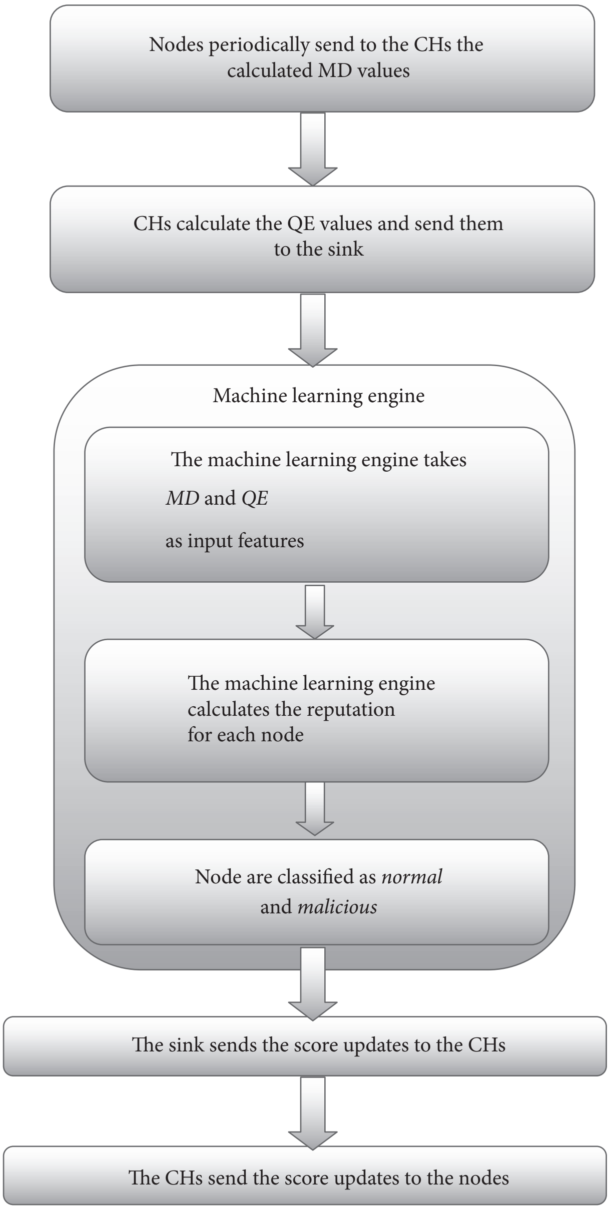

Nodes periodically send to the CHs the calculated MD values
CHs calculate the QE values and send them to the sink
Machine learning engine
The machine learning engine takes *MD* and *QE* as input features
The machine learning engine calculates the reputation for each node
Node are classified as *normal* and *malicious*
The sink sends the score updates to the CHs
The CHs send the score updates to the nodes

Figure 6: Main steps of Bankovic et al. approach.

Figure 6 summarizes the steps of the Bankovic et al. approach just described; the tasks performed by the machine learning engine are properly pointed out.

5. Performance Comparison

From the discussion of Sections 3 and 4, the behavior of GoNe and Bankovic et al. revealed several differences between the two approaches:

(i) GoNe adopts an encryption mechanism for data sensed and transmitted by sensor nodes, while Bankovic et al. transmit in clear way the information within the network.

(ii) GoNe puts in place an integrity verification system at CH level, while Bankovic et al. technique does not supervise the integrity and confidentiality of the transmitted information.

(iii) GoNe uses aggregation technique at CH level for reducing the traffic, while Bankovic et al. technique does not control network congestion situations.

On such basis, the overhead and also the benefits of GoNe with respect to a framework, like Bankovic et al., which do not adopt any of these mechanisms, are compared.

Furthermore, both GoNe and Bankovic et al. techniques use SOM potentialities, but they do that in a different way; in particular, they use diverse features and the node classification has a diverse level of accuracy, since GoNe establishes three categories (i.e., *normal*, *unknown*, and *malicious*), while Bankovic et al. technique only two ones (i.e., *normal*, *malicious*). In this case, it needs to check if such classifications are effective and in what measure. It is worth noting that, in [5], GoNe has been compared with the other two schemes defined by the authors (i.e., *SETA* [10] and *VM* [12]). In this paper, a further validation of the GoNe effectiveness is fulfilled, by means of a comparison with the solution of Bankovic et al., with respect to the following metrics:

(i) Data accuracy: evaluated by comparing the hypothetical environmental temperatures estimated by the sink and the actual temperatures acquired by nodes; such temperatures are random-generated during the system running, as well as the message-passing towards the sink

(ii) Delay of packet arrival at the sink: representing the time elapsed between the packet generation by a sensor node and its reception at the sink

(iii) Power consumption of sensor nodes: estimated using the real-time monitoring toolkit named *Energino* [42]

(iv) Overhead due to the reputation algorithm in terms of percentage of signalling messages with respect to the total messages transmitted by the network

(v) Lost messages

(vi) Number of nodes correctly classified as *normal*, *unknown* and *malicious*, thus pointing out the false positive/negative rate (i.e., malicious/normal nodes correctly detected)

(vii) Percentage of kinds of attack recognized (i.e., integrity, resources, and routing).

To evaluate such performance indexes, the Omnet++ simulator has been adopted [43]. Hereby GoNe and Bankovic et al. algorithms have been implemented by means of C++ language. In fact, Omnet++ has already been used in [5], while Bankovic et al. technique does not recommend any network simulator. The setup of the simulated scenarios is summarized in Simulation Parameters. In order to exploit the header compression gain due to 6LoWPAN standard [44], messages are encapsulated in a IPv6 over IEEE 802.15.4 stack [45]. Moreover, in order to allow the proper functioning of

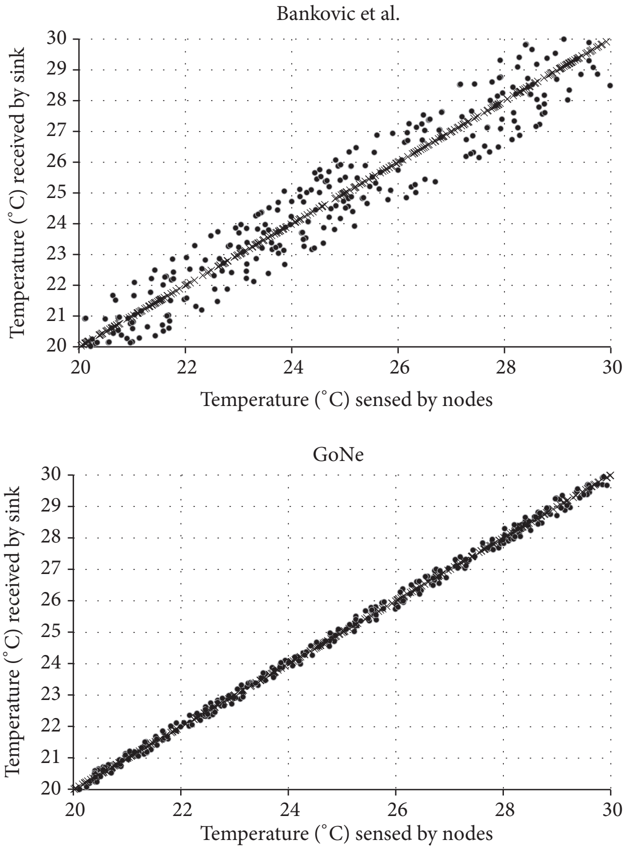


Figure 7: Data accuracy.

Bankovic et al. scheme, a fixed position in the form of (x, y) coordinates is assigned to each network entity (i.e., sensor nodes, CHs, and sink).

As regards malicious attacks, several models are randomly simulated, such as attacks to data integrity [10], to resources, and to routing behavior, as explained in Section 3.1. The outcomes are presented for different percentages of malicious nodes (up to 40% of the total node number) and when not specified, they refer to a percentage of 20%. Since GoNe and Bankovic et al. techniques aim at recognizing malicious or broken nodes on the basis of their behavior during network activities, no external malicious entities are included in the presented simulations. However, if an external attack is performed towards one or more sensor nodes, then GoNe and Bankovic et al. techniques should recognize the misbehavior, since the activity of the nodes object of the attack would be compromised. As a consequence, such nodes will see their reputation scores get lower and the network will be able to counteract the malicious attacks in the same way as for any ill-behaved sensor nodes (e.g., by isolating them from the WSN communications).

5.1. Simulation Results. Starting from the accuracy evaluation of the data received by the sink (Figure 7), from simulations in the two network topologies (i.e., 100 and 200 nodes) and varying the percentage of malicious nodes, it emerges that GoNe is able to guarantee better results, since it adopts an encryption scheme and an efficient data integrity verification on all the data before sending them to the sink, whereas Bankovic et al. technique does not encrypt the information transmitted over the network, and only the machine learning algorithm executes a control on the data (see Section 4.3). For such reasons, the data is more widespread around the bisector in the case of Section 4.3 compared to GoNe.

As regards the delay of packets arrival at the sink, Figure 8 shows the corresponding Cumulative Distribution Function (CDF). Bankovic et al. technique worsens packet delay with respect to GoNe; this is mainly due to the fact that GoNe adopts an aggregation mechanism, thus reducing the waiting times in the buffers. Such a behavior is the same both in a network without malicious nodes and in presence of malicious ones. Therefore, it can be concluded that such a behavior does not directly depend on the efficiency of the two machine learning algorithms.

The energy consumption is evaluated for the transmission and reception operations of sensor nodes and, as shown in Figure 9, GoNe and Bankovic et al. techniques present similar results. In fact, both the frameworks are able to isolate misbehaving nodes, thus reducing their action of compromising the network resources. This demonstrates the efficiency of the two approaches; however, GoNe would present lower energy consumption if sensor node did not perform the data encryption operations.

The overhead of GoNe and Bankovic et al. techniques is presented in Figures 10 and 11, respectively. Such an overhead aims to represent the effects of the reputation mechanism during simulation, in terms of the percentage of messages related to the score evaluation with respect to the total number of packets transmitted over the network (i.e., messages referring to reputation evaluation and message containing the sensed data, detailed in Section 4). It is assumed that malicious nodes are ill-behaved from the beginning of the simulation; what emerges for both the frameworks is the following: (i) without malicious nodes the overhead is concentrated at the beginning of the simulation (i.e., all the nodes are normal; therefore the reputation algorithm settles quickly); (ii) with the increase of malicious nodes the peak overhead is lower, but its long term value is higher than before, since the reputation algorithm needs a certain time to recognize the malicious behavior. Moreover, the peaks of detection for GoNe are nearer to the beginning of the simulations than those of Bankovic et al. This may be due to the fact that GoNe has smaller delays than Bankovic et al. technique, thus speeding up the transfer of information into the network. Note that the overhead could change during network activity if, for example, new nodes become malicious or some nodes are damaged.

Since both GoNe and Bankovic et al. techniques face attacks to traffic and resources by isolating the malicious nodes, the differences in the percentage of lost packets (Figure 12) may not be totally due to the malicious behavior, but, instead, to the different management of network entities operated by the two schemes themselves. In fact, GoNe mitigates also the congestion situations (by means of data

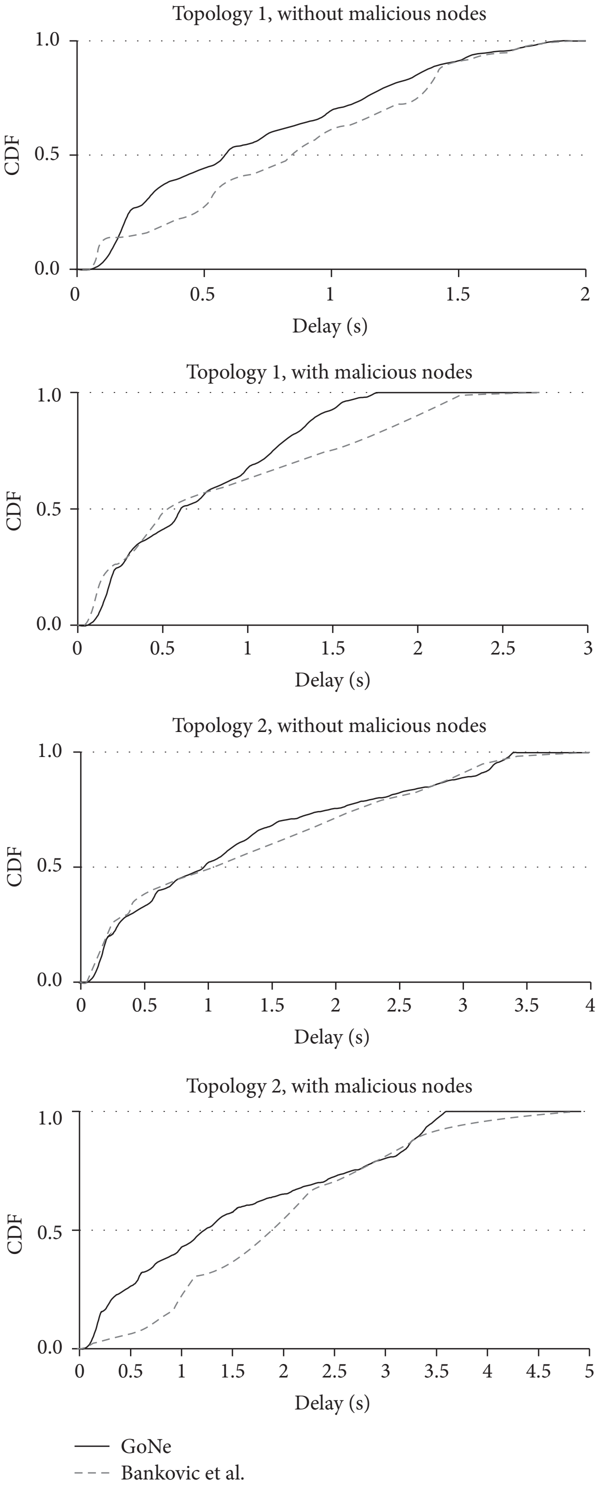


Figure 8: Delay of packets.

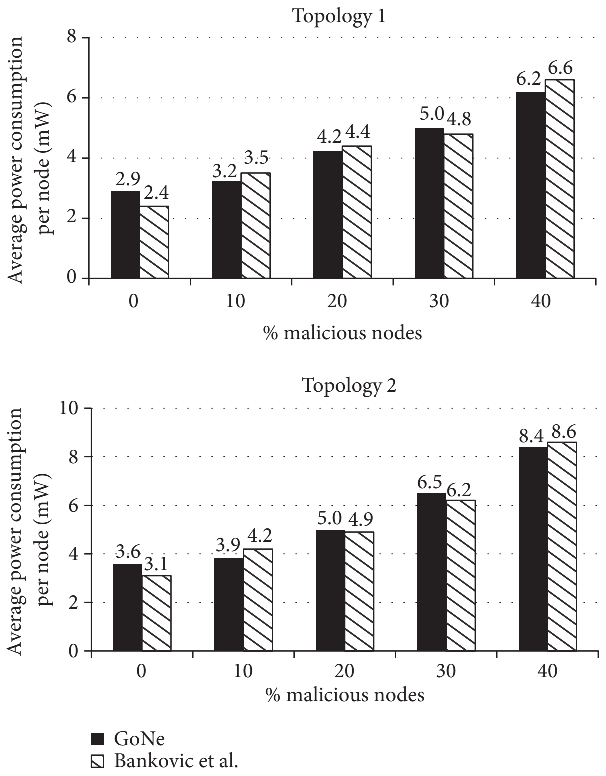


Figure 9: Mean energy consumption of sensor nodes.

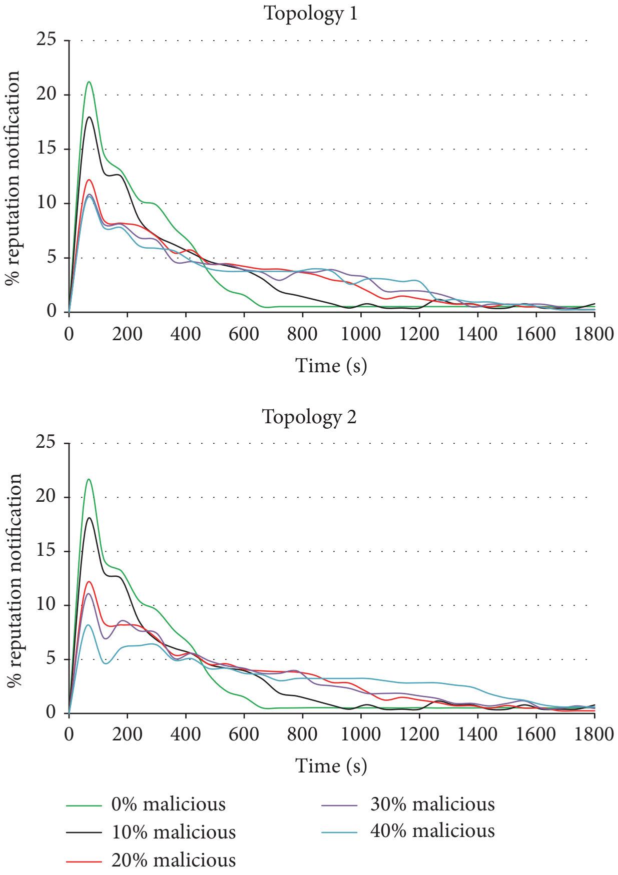


Figure 10: Overhead of GoNe.

aggregation), which are the main causes of the packet lost, and drops the violated packets before reaching the sink, thus reducing the useless network traffic.

Figures 13 and 14 compare the node classification of GoNe and Bankovic et al. techniques. What emerges, for the different percentage of malicious nodes included in the WSN, is

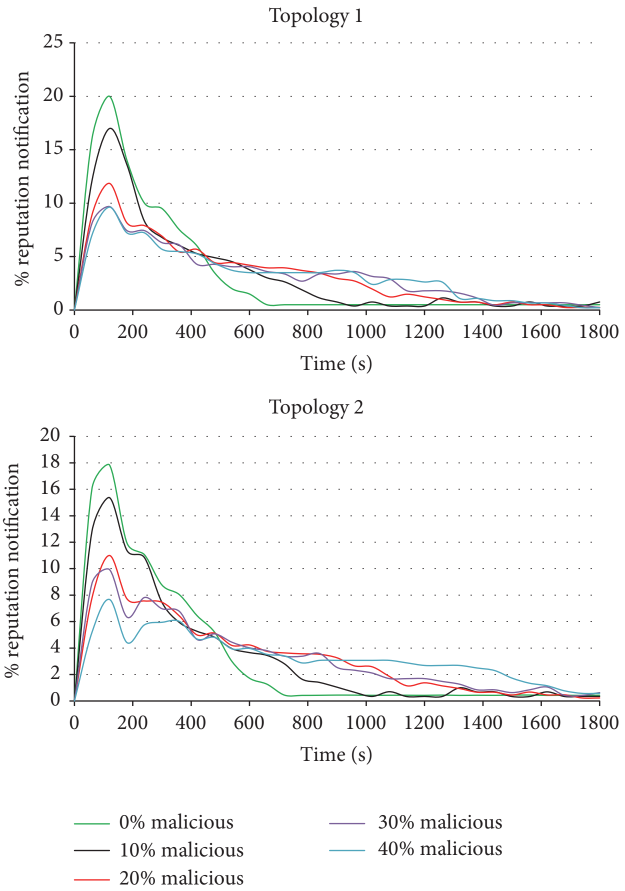


Figure 11: Overhead of Bankovic et al.

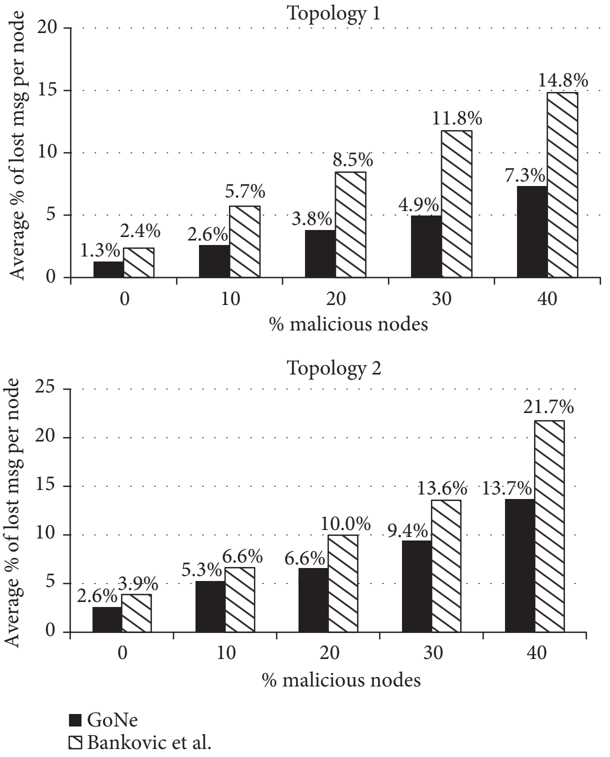


Figure 12: Lost packets.

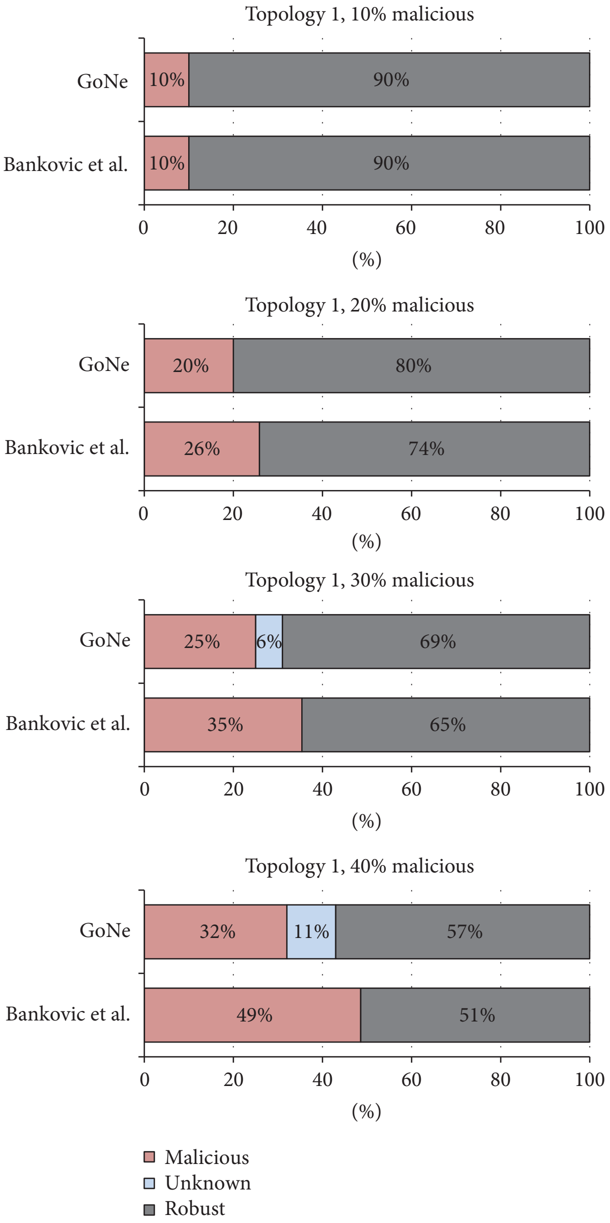


Figure 13: Node classification—Topology 1.

that both the frameworks mostly classify the sensor nodes in the correct way. More in detail, in many cases, Bankovic et al. technique classifies as malicious more nodes than the existing ones, thus generating several misclassifications (i.e., false negatives), while GoNe classifies some nodes as *unknown*, avoiding misclassifications and leaving the sink the task to assess the data provided by them. It is important to note that GoNe prevents the generation of false negatives as well as false positives. Moreover, GoNe guarantees the identification of more nodes as *Robust* with respect to Bankovic et al. With regard to Bankovic et al., such an approach also uses the information related to the node localization; in this way, Bankovic et al. technique is able to identify the outliers. Otherwise, GoNe does not consider them and is unaware

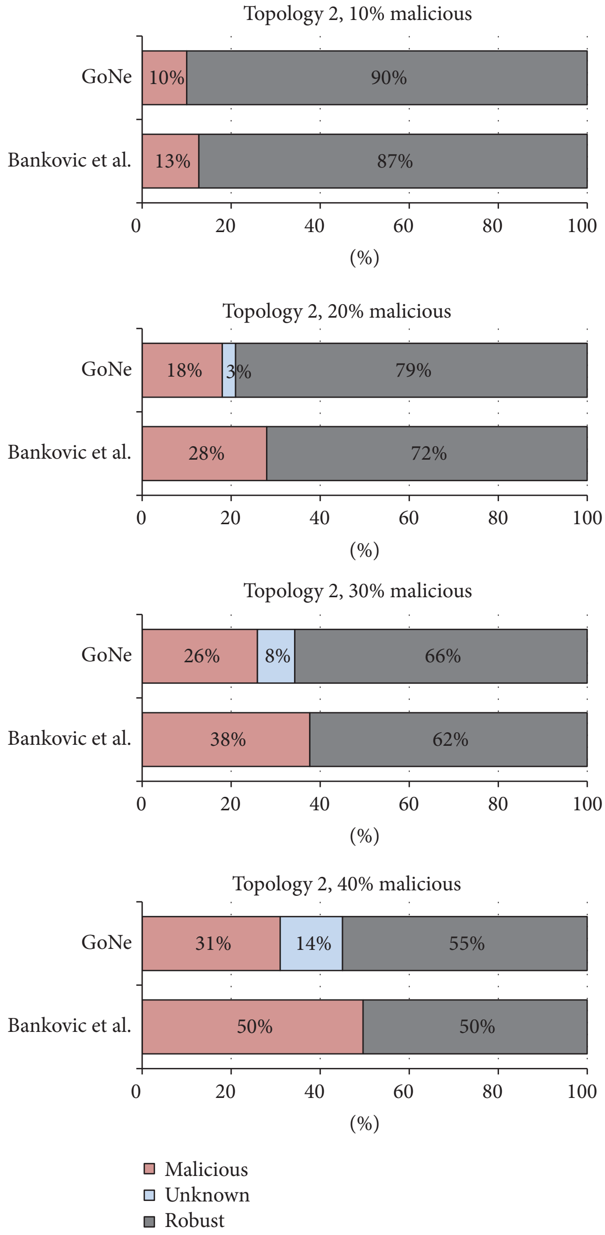


Figure 14: Node classification—Topology 2.

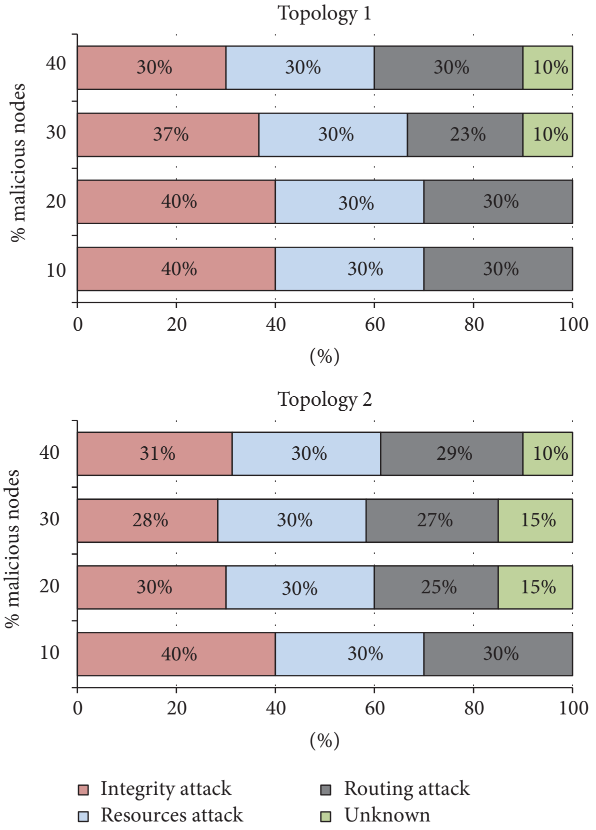


Figure 15: Attacks evaluation for GoNe.

of the position of the nodes themselves but clearly identifies them by means of their identifiers.

Finally, besides the classification of malicious nodes, it has been evaluated if the two frameworks are also able to identify the specific kind of attack for each malicious node detected. The simulated attacks are related to the data integrity, the network resources, and the routing protocol. Figure 15 shows the percentages of the different kinds of recognized attack for GoNe, which respect the percentage of malicious behaviors included in the simulation network scenarios (approximatively, 40% integrity attacks, 30% attacks to resources, and 30% attacks to routing protocols), whereas, as just explained, Bankovic et al. technique is not able to recognize attacks to data integrity; moreover, no routing protocol has been specified in [7], and therefore only the detection of resource attacks has been evaluated (Figure 16).

6. Discussion and Conclusions

Malicious node detection plays an important role in providing security in a WSN. Many solutions available in literature address such an issue by proposing reputation algorithms, able to assess the trustworthiness of the sensor nodes belonging to the network. In general, an analysis of the node behavior and of the data provided to the sink is carried out, in order to assign a score to the reputation of each node. On the basis of the assigned scores, the sink may counteract different kinds of possible attacks (e.g., by isolating one or more nodes from the network communications). Machine learning techniques are widely adopted in the existing reputation mechanism. Among them, the most popular intrusion detection schemes are based on Fuzzy C-Means clustering, Backpropagation Neural Network, Self-Organizing Maps, Wavelets, Agglomerative Clustering, and Bayesian classifier [34]. In particular, Self-Organizing Maps (SOM) emerge as a technique suitable for constrained and unattended environments, like WSN, in which the node behavior may be not predictable a priori. Furthermore, SOM are also able to quickly react to changes in the monitored area, and thus its adoption should allow the WSN to counteract malicious attacks in a reasonable time. Note that, due to its self-adaptation capabilities, a WSN based

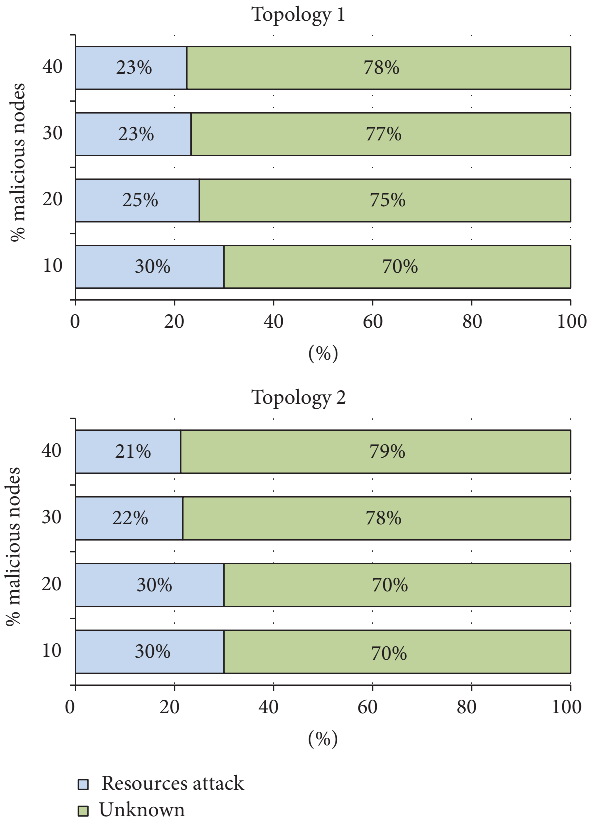


Figure 16: Attacks evaluation for Bankovic et al.

on SOM finds application in many contexts, characterized by frequent environment changes and different patterns of behavior, for example, data mining or image recognition in surveillance areas or in medical diagnosis, vehicle control, and ambient monitoring in hostile environments.

For the aforementioned reasons, SOM has been adopted in the framework GoNe for recognizing the malicious attacks in a clustered WSN. A preliminary validation of GoNe performance has been provided in [5] by means of a comparison with *SETA* framework, which did not adopt any mechanism for intrusion detection. However, this is not enough for ensuring its robustness and effectiveness with respect to other existing solutions still available in literature. Therefore, in this paper, a comparison with the scheme proposed by Bankovic et al., also based on SOM, has been carried out. The performance of the two frameworks has been extensively analyzed by means of simulations, which revealed the following results:

(i) Both the schemes presented an acceptable level of node classification; this means that the network is mostly able to recognize in a correct way the malicious and the normal nodes; however, Bankovic et al. scheme generates some false negatives.

(ii) Both the schemes are able to limit the power consumption of sensor nodes, thus proving their suitability in resource constrained environments.

(iii) By means of the adoption of integrity verification and congestion control mechanisms, GoNe, with respect to Bankovic et al. scheme, is able to

(a) provide more accurate data;
(b) limit the packet delays;
(c) reduce the packet losses.

(iv) GoNe allows recognizing a wider range of attacks with respect to Bankovic et al. scheme, since Bankovic et al. technique is not able to recognize violations to data integrity.

Another important remark is about the adopted architecture. In fact, both the schemes act in a clustered WSN, with the final goal of reducing the resource consumption of sensor nodes by introducing the more powerful CHs. As proved by simulations, such a scope has been achieved. However, many solutions in intrusion detection field adopt a flat architecture. The scheme GoNe is also suitable for this kind of network configuration, because the node assessment is directly performed by the sink and not by CHs. But the presence of CHs allows, as just said, reducing the network load near the sink and, as a consequence, the energy waste at sensor node level, by actuating a sharing of tasks. This feature represents the main relevant drawback of adopting a flat architecture instead of a clustered one. As regards Bankovic et al. technique, the authors do not specify the viability of their scheme in a flat scenario, but similar conclusions may be drawn as for GoNe. Another final point, not clarified by the authors, is if Bankovic et al. technique performs the node assessment only once during network running or if this task is periodically executed. This represents a crucial aspect, which would allow Bankovic et al. technique to make a better evaluation of node behavior during the time.

As a future work, a comparison of GoNe with other node reputation schemes not based on SOM is planned, in order to evaluate GoNe performance with respect to different machine learning mechanisms. Furthermore, GoNe will be integrated in a more complex Internet of Things framework, including not only sensor nodes but also other kinds of devices and communication technologies (e.g., RFID, NFC, and Bluetooth). Finally, it will be interesting to analyze and evaluate the performance of GoNe scheme in real network scenarios, such as a case of environmental monitoring or an application in underwater networks [46].

Simulation Parameters

A:	Network area, 400 m^2
N:	Number of nodes, 100, 200
C:	Number of clusters, 3
D_c:	Depth of connections, 5
M:	Percentage of malicious nodes, up to 40%
P:	Interval time of data generation, 1 s
P_{Max}:	Max packet size, 93 bytes
br:	Bit rate, 250 kbps
C_m:	CH buffer size, 20 kB
Q_n:	Node buffer size, 10 KB
t_S:	Duration of simulation, 1800 s.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[2] F. G. Mármol and G. M. Pérez, "Security threats scenarios in trust, reputation models for distributed systems," *Computers & Security*, vol. 28, no. 7, pp. 545–556, 2009.

[3] M. Xie, S. Han, B. Tian, and S. Parvin, "Anomaly detection in wireless sensor networks: a survey," *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1302–1325, 2011.

[4] I. Butun, S. D. Morgera, and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 266–282, 2014.

[5] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, "GoNe: Dealing with node behavior," in *Proceedings of the 5th IEEE International Conference on Consumer Electronics - Berlin, ICCE-Berlin 2015*, pp. 358–362, Berlin, Germany, September 2015.

[6] T. Kohonen, "The Self-Organizing Map," *Proceedings of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[7] Z. Bankovic, D. Fraga, J. Manuel Moya et al., "Improving security in WMNs with reputation systems and self-organizing maps," *Journal of Network and Computer Applications*, vol. 34, no. 2, pp. 455–463, 2011.

[8] C. Castelluccia, E. Mykletun, and G. Tsudik, "Efficient aggregation of encrypted data in wireless sensor networks," in *Proceedings of the 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous '05)*, pp. 109–117, IEEE Computer Society, July 2005.

[9] K. Lauter, "The Advantages of Elliptic Curve Cryptography for Wireless Security," *IEEE Wireless Communications Magazine*, vol. 11, no. 1, pp. 62–67, 2004.

[10] S. Sicari, L. A. Grieco, A. Rizzardi, G. Boggia, and A. Coen-Porisini, "SETA: a secure sharing of tasks in clustered wireless sensor networks," in *Proceedings of the 2013 IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob 2013*, pp. 239–246, Lyon, France, October 2013.

[11] E. Ekici, S. Vural, J. McNair, and D. Al-Abri, "Secure probabilistic location verification in randomly deployed wireless sensor networks," *Ad Hoc Networks*, vol. 6, no. 2, pp. 195–209, 2008.

[12] A. Coen-Porisini and S. Sicari, "Improving data quality using a cross layer protocol in wireless sensor networks," *Computer Networks*, vol. 56, no. 17, pp. 3655–3665, 2012.

[13] M. K. Domenic, Y. Wang, F. Zhang, I. Memon, and Y. H. Gustav, "Preserving users' privacy for continuous query services in road networks," in *Proceedings of the 2013 6th International Conference on Information Management, Innovation Management and Industrial Engineering, ICIII 2013*, pp. 352–355, China, November 2013.

[14] I. Memon, I. Hussain, R. Akhtar, and G. Chen, "Enhanced privacy and authentication: an efficient and secure anonymous communication for location based service using asymmetric cryptography scheme," *Wireless Personal Communications*, vol. 84, no. 2, pp. 1487–1508, 2015.

[15] R. Mitchell and I.-R. Chen, "A survey of intrusion detection in wireless network applications," *Computer Communications*, vol. 42, pp. 1–23, 2014.

[16] G. Han, J. Jiang, L. Shu, J. Niu, and H. Chao, "Management and applications of trust in Wireless Sensor Networks: a survey," *Journal of Computer and System Sciences*, vol. 80, no. 3, pp. 602–617, 2014.

[17] B. Zhang, Z. H. Huang, and Y. Xiang, "A novel multiple-level trust management framework for wireless sensor networks," *Computer Networks*, vol. 72, pp. 45–61, 2014.

[18] W. Fang, C. Zhang, Z. Shi, Q. Zhao, and L. Shan, "BTRES: beta-based Trust and Reputation Evaluation System for wireless sensor networks," *Journal of Network and Computer Applications*, vol. 59, pp. 88–94, 2016.

[19] F. G. Mármol and G. M. Pérez, "Towards pre-standardization of trust, reputation models for distributed, heterogeneous systems," *Computer Standards and Interfaces*, vol. 32, no. 4, pp. 185–196, 2010.

[20] F. G. Mármol, J. Girao, and G. M. Pérez, "TRIMS, a privacy-aware trust and reputation model for identity management systems," *Computer Networks*, vol. 54, no. 16, pp. 2899–2912, 2010.

[21] D. Jelenc, R. Hermoso, J. Sabater-Mir, and D. Trček, "Decision making matters: a better way to evaluate trust models," *Knowledge-Based Systems*, vol. 52, pp. 147–164, 2013.

[22] Y. Yu, K. Li, W. Zhou, and P. Li, "Trust mechanisms in wireless sensor networks: attack analysis and countermeasures," *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 867–880, 2012.

[23] H. Marzi and M. Li, "An enhanced bio-inspired trust and reputation model for wireless sensor network," *Procedia Computer Science*, vol. 19, pp. 1159–1166, 2013, Proceedings of the 4th International Conference on Ambient Systems, Networks and Technologies.

[24] Z. Banković, D. Fraga, J. M. Moya et al., "Bio-inspired enhancement of reputation systems for intelligent environments," *Information Sciences*, vol. 222, pp. 99–112, 2013.

[25] S. Shamshirband, N. B. Anuar, M. L. M. Kiah, and A. Patel, "An appraisal and design of a multi-agent system based cooperative wireless intrusion detection computational intelligence technique," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 9, pp. 2105–2127, 2013.

[26] H. Kumarage, I. Khalil, Z. Tari, and A. Zomaya, "Distributed anomaly detection for industrial wireless sensor networks based on fuzzy data modelling," *Journal of Parallel and Distributed Computing*, vol. 73, no. 6, pp. 790–806, 2013.

[27] S. Shamshirband, A. Patel, N. B. Anuar, M. L. M. Kiah, and A. Abraham, "Cooperative game theoretic approach using fuzzy Q-learning for detecting and preventing intrusions in wireless sensor networks," *Engineering Applications of Artificial Intelligence*, vol. 32, pp. 228–241, 2014.

[28] S. Shamshirband, N. B. Anuar, M. L. M. Kiah et al., "Co-FAIS: cooperative fuzzy artificial immune system for detecting intrusion in wireless sensor networks," *Journal of Network and Computer Applications*, vol. 42, pp. 102–117, 2014.

[29] M. Akram and T. H. Cho, "Energy efficient fuzzy adaptive selection of verification nodes in wireless sensor networks," *Ad Hoc Networks*, vol. 47, pp. 16–25, 2016.

[30] H. Jadidoleslamy, M. R. Aref, and H. Bahramgiri, "A fuzzy fully distributed trust management system in wireless sensor networks," *AEÜ - International Journal of Electronics and Communications*, vol. 70, no. 1, pp. 40–49, 2016.

[31] Z. Yu and J. J. P. Tsai, "A framework of machine learning based intrusion detection for wireless sensor networks," in *Proceedings of the 2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, SUTC 2008*, pp. 272–279, Taichung, Taiwan, June 2008.

[32] Z. Banković, J. M. Moya, Á. Araujo, D. Fraga, J. C. Vallejo, and J.-M. De Goyeneche, "Distributed intrusion detection system for wireless sensor networks based on a reputation system coupled with kernel self-organizing maps," *Integrated Computer-Aided Engineering*, vol. 17, no. 2, pp. 87–102, 2010.

[33] A. H. FathiNavid and A. B. Aghababa, "A protocol for intrusion detection based on learning automata in forwarding packets for distributed wireless sensor networks," in *Proceedings of the 4th International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2012*, pp. 373–380, Sanya, China, October 2012.

[34] H. H. Soliman, N. A. Hikal, and N. A. Sakr, "A comparative performance evaluation of intrusion detection techniques for hierarchical wireless sensor networks," *Egyptian Informatics Journal*, vol. 13, no. 3, pp. 225–238, 2012.

[35] I. F. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer Networks*, vol. 47, no. 4, pp. 445–487, 2005.

[36] M. Passing and F. Dressler, "Experimental performance evaluation of cryptographic algorithms on sensor nodes," in *Proceedings of the 2006 IEEE International Conference on Mobile Ad Hoc and Sensor Sysetems, MASS*, pp. 882–887, can, October 2006.

[37] S. Md Zin, N. Badrul Anuar, M. Laiha Mat Kiah, and A.-S. Khan Pathan, "Routing protocol design for secure WSN: review and open research issues," *Journal of Network and Computer Applications*, vol. 41, no. 1, pp. 517–530, 2014.

[38] R. Di Pietro, S. Guarino, N. V. Verde, and J. Domingo-Ferrer, "Security in wireless ad-hoc networks - A survey," *Computer Communications*, vol. 51, pp. 1–20, 2014.

[39] D. Chaum and E. van Heyst, "Group Signatures," in *Proceedings of the 10th Annual International Conference on Theory, Application of Cryptographic Techniques*, vol. 547 of *Lecture Notes in Computer Science*, pp. 257–265, Springer, Berlin, Heidelberg, 1991.

[40] J. Camenisch and M. Stadler, Efficient group signature schemes for large groups, Advances in Cryptology, CRYPTO '97, 1294, 410-424, 1997.

[41] N. Basilico, N. Gatti, M. Monga, and S. Sicari, "Security games for node localization through verifiable multilateration," *IEEE Transactions on Dependable and Secure Computing*, vol. 11, no. 1, pp. 72–85, 2014.

[42] K. Gomez, R. Riggio, T. Rasheed, D. Miorandi, and F. Granelli, "Energino: a hardware and software solution for energy consumption monitoring," in *Proceedings of the 2012 10th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks, WiOpt 2012*, pp. 311–317, May 2012.

[43] Omnet++ Simulator - Official Website, 2016, http://www.omnetpp.org/.

[44] J. Hui and P. Thubert, Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks, 2011, Sep., IETF RFC 6282.

[45] M. R. Palattella, N. Accettura, X. Vilajosana et al., "Standardized protocol stack for the internet of (important) things," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1389–1406, 2013.

[46] N. Mohamed, I. Jawhar, J. Al-Jaroodi, and L. Zhang, "Sensor network architectures for monitoring underwater pipelines," *Sensors*, vol. 11, no. 11, pp. 10738–10764, 2011.

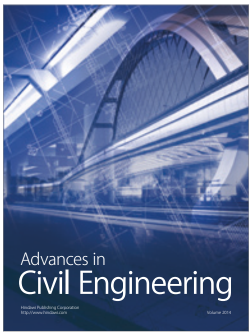
Journal of
Engineering

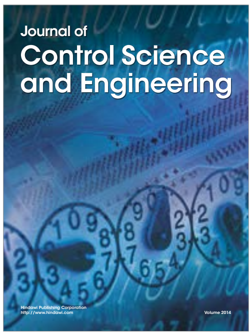
The Scientific World Journal

International Journal of
Rotating Machinery

Journal of
Sensors

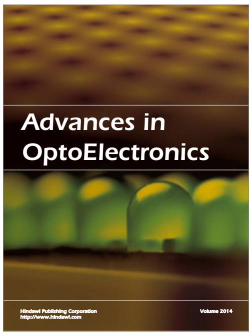
International Journal of
Distributed Sensor Networks


Advances in
Civil Engineering


Journal of
Control Science and Engineering

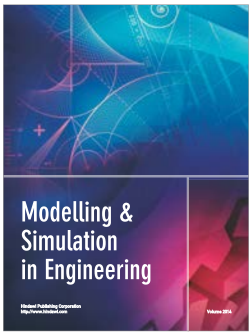
Journal of
Robotics

Journal of
Electrical and Computer Engineering

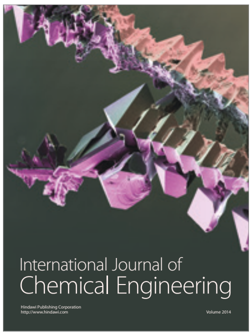

Advances in
OptoElectronics

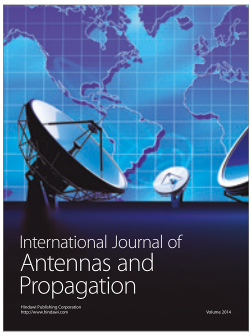
VLSI Design

International Journal of
Navigation and Observation

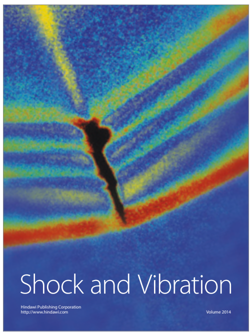

Modelling &
Simulation
in Engineering

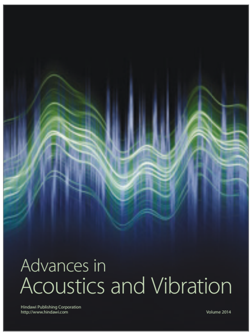
International Journal of
Aerospace Engineering


International Journal of
Chemical Engineering


International Journal of
Antennas and Propagation

Active and Passive
Electronic Components


Shock and Vibration


Advances in
Acoustics and Vibration