future internet

MDPI

Article

A New Lightweight Watchdog-Based Algorithm for Detecting Sybil Nodes in Mobile WSNs

Rezvan Almas Shehni [1,‡], Karim Faez [2,‡], Farshad Eshghi [3,*,†,‡] and Manoochehr Kelarestaghi [3,‡]

1 Department of Computer Engineering and Information Technology, Qazvin Azad University, Qazvin 15195-34199, Iran; r.shehni@qiau.ac.ir
2 Department of Electrical Engineering, Amirkabir University of Technology, Tehran 15916-34311, Iran; kfaez@aut.ac.ir
3 Department of Electrical & Computer Engineering , Faculty of Engineering, Kharazmi University, Tehran 15719-14911, Iran; kelarestaghi@khu.ac.ir
* Correspondence: farshade@khu.ac.ir; Tel.: +98-912-497-0182
† Current address: Department of Electrical & Computer Engineering , Faculty of Engineering, Kharazmi University, Tehran, Iran.
‡ These authors contributed equally to this work.

Received: 15 November 2017; Accepted: 8 Decmber 2017; Published: 21 Decmber 2017

Abstract: Wide-spread deployment of Wireless Sensor Networks (WSN) necessitates special attention to security issues, amongst which Sybil attacks are the most important ones. As a core to Sybil attacks, malicious nodes try to disrupt network operations by creating several fabricated IDs. Due to energy consumption concerns in WSNs, devising detection algorithms which release the sensor nodes from high computational and communicational loads are of great importance. In this paper, a new computationally lightweight watchdog-based algorithm is proposed for detecting Sybil IDs in mobile WSNs. The proposed algorithm employs watchdog nodes for collecting detection information and a designated watchdog node for detection information processing and the final Sybil list generation. Benefiting from a newly devised co-presence state diagram and adequate detection rules, the new algorithm features low extra communication overhead, as well as a satisfactory compromise between two otherwise contradictory detection measures of performance, True Detection Rate (TDR) and False Detection Rate (FDR). Extensive simulation results illustrate the merits of the new algorithm compared to a couple of recent watchdog-based Sybil detection algorithms.

Keywords: mobile WSN; security; watchdog node; Sybil attack

1. Introduction

Nowadays we are witnessing the emergence of new WSN applications in different fields such as military, urban services, the environment, medicine, explorations and Intrusion Detection Systems (IDS). WSNs comprise a large number of small sensor nodes featuring small memory and low power. The broadcast nature of wireless and unattended operation of WSNs necessitate the implementation and improvement of security schemes [1,2].

Operation disruption in hostile wireless networks can be realized in Physical (PHY) or higher layers. In the former case, a malicious node tries to harm communication between wireless nodes by broadcasting jamming signals [3] or imposing any other kind of interference [4] whereby normal wireless nodes become unable to interpret receiving signals. in the latter case, malicious nodes try to either deceive normal wireless nodes through disseminating fake information (e.g., Sybil, wormhole, impersonation, and etc. attacks) or overwhelm/disable normal wireless nodes using, for instance, Hello flood attack [2].

In this paper, we focus on the Sybil attack which is one of the most important attacks in WSNs. In this attack, an illegal node or a legal node captured by enemy, called a "malicious node", identifies itself by releasing several fake IDs or IDs fabricated from other legal nodes. The fake IDs represent some non-existing nodes known as Sybil nodes. As a result, legal nodes think they have many legitimate neighbors. Malicious nodes can affect routing and operational protocols such as data aggregation, voting, resource allocation, misbehavior [5,6] .

In general, Sybil attack detection techniques can be categorized into centralized and decentralized methods. Centralized methods feature a central node which is responsible for node identity management. In relevant detection methods, the information in central nodes are used for Sybil attack detection. Decentralized methods make use of some pre-authenticated nodes, sometimes called watchdogs, trusted, etc., which administrate the attack detection operation. These watchdogs can be fixed or mobile.

From a networking layer stack point of view, Sybil attack detection techniques can be grouped into PHY-layer-based and upper-layers-based techniques. The former make use of the parameters of the radio signal and the second layer's node identity information. On the other hand, the latter techniques are based on the communication of the data which is formed in the upper layers. Nevertheless, the identity information is still required in upper-layer based techniques. PHY-layer-based category is further divided into location and non-location-based techniques. Location-based techniques mostly involve analysis of Received Signal Strength Indicator (RSSI), Time Difference of Arrival (TDoA), and Angle of Arrival (AoA). One of the techniques under non-location-based category, as far as we could identify, is the Radio resource testing [6]. The techniques under the upper-layers-based category are further divided to neighborhood-based [7,8], code attestation-based [9–11], authentication-based (puzzle solving technique [12] for peer-to-peer networks, Identity certificate technique [13]), and identity registration-based [14]. Figure 1 illustrates the Sybil attack detection techniques categorization.

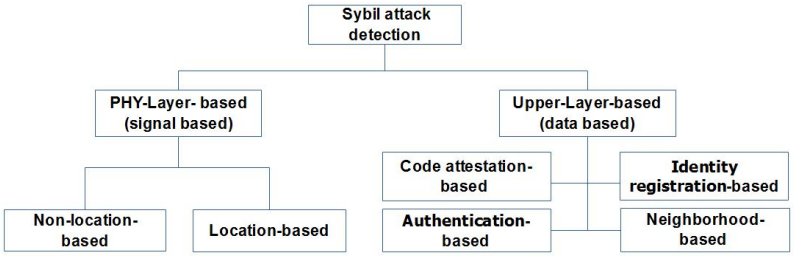


Figure 1. Categorization of Sybil attack detection algorithms in WSNs.

In mobile WSNs, Sybil nodes corresponding to a particular malicious node appear/disappear simultaneously in/from some neighborhoods. This behavior (misbehavior) can be used for Sybil nodes detection. At the same token, Watchdog-based techniques have been extensively employed to detect misbehaviors in WSNs [15,16]; thus, it is plausible to think of it as a tool for Sybil nodes detection as well. In this paper, we propose a lightweight, sufficiently accurate, and practical watchdog-based algorithm for detecting Sybil nodes in mobile WSNs. Our algorithm, considered as a neighborhood -based technique, does not need any centralized base station and does not require transmission of neighborhood data (neighborhood table) from normal nodes to watchdog nodes.

Due to being computationally lightweight, in addition to WSNs, the proposed algorithm can be well adapted to new emerging applications which involve less complex nodes such as Internet of Things (IoT), smart home/car, and health-care.

This paper is organized as follows. Section 2 discuses the related works. Problem statement is presented in Section 3. Section 4 is dedicated to the description of the proposed algorithm, followed by simulation results and performance evaluation in Section 5. Concluding remarks are drawn at the end.

2. Related Works

Sybil attack was introduced for the first time in [17] for peer- to- peer networks. In [2], it was noted that this attack can also be a dangerous threat to routing algorithms in WSNs and can be. Newsome et al. [6] present a detailed analysis of the Sybil attack in WSNs alongside some attack detection mechanisms. Also, the taxonomy of Sybil attack are introduced in the same work which is referenced to by most researchers in the field. In what follows we review some important related works within the categorization framework introduced in Figure 1.

Focusing on PHY-layer-based approaches, in [6], several approaches in Sybil attack detection have been discussed. The first one is radio resource testing, a non-location-based technique, which relies on the assumption that a node can communicate with each of its neighbors through pre-assigned channels. When a node wants to verify one of its neighbors, it could choose a channel randomly to listen. If the neighbor that was assigned that channel is legitimate, it should hear the message. This detection technique might not work in the case of an attacker equipped with a multi-radio transceiver capable of concurrently communication through different channels.

Abbas et al. [18] propose a Sybil attack detection scheme in Mobile Ad hoc Networks (MANETs) based on monitoring and differentiating between the entry and exit RSS behavior of legitimate nodes and the Sybil attackers.

There are other techniques introduced in [6] which belong to the upper-layer-based category and will be discussed in due place. Most of Sybil attack detection approaches are based on location-verification based which use some related techniques, such as RSSI or TDOA, to distinguish between Sybil and normal nodes. Since these methods depend on some parameters of receiving signals, most probably tainted with noise and multi-path phenomena, they might end up being less reliable. The RSSI-based location determination has been used In [19] to detect Sybil nodes. Four detector nodes which are able to hear the packets from all areas of the network, cooperatively evaluate the location of the node sending the packet. Node IDs found to be sending packets from the very same location are assumed to be Sybil IDs. In [20] a solution for Sybil attack detection is proposed based on TDOA between a source node and three beacon nodes which detect the location of Sybil nodes. Since three Beacon nodes are used to calibrate the time measurements, there is some communication overhead present. Another Sybil detection system has been proposed in [21] which relies on the raging capabilities of Ultra-Wide Band (UWB) in the PHY layer. Each node periodically monitors its distance from each possible pair of its neighbors. An alarm is triggered when two or more nodes are being located in the same area. The locally computed ranging estimation is used to measure the distance between neighbors. As the authors mention, the proposed technique induces lack of compliance with old-fashioned WSNs. In this work node mobility has not been considered.

Turning our focus on upper-layer-based approaches, code attestation-based techniques are another approach mentioned in [6]. these differentiate the code running on a malicious node with that of a legitimate node [9,10]. A compromised node detection algorithm, based on code attestation, is proposed in [11] and called Unpredictable Software-based Solution (USAS). USAS administers attestation on a randomly selected nodes rather than all, in order to decrease checksum computation time. Since it only attests nodes one hop from the base station, if a compromised node is far from the base station (more than one-hop), it might not be detected.

An authentication-based approach, also introduced in [6], is a version of random key per-distribution.

In random key pre-distribution "a random set of keys or key-related information is assign to each sensor node, so that in the key set-up phase, each node can discover or compute the common keys it shares with its neighbors; the common keys will be used as a shared secret session key to ensure node-to-node secrecy". So the technique involves associating the node identity with the keys assigned to the node, and thereafter Key validation of an claimed identity. As [6] claims himself, "the problem with this approach is that if an attacker compromises multiple nodes, he can use every combination of the compromised keys to generate new identities". He presents the solution to be

indirect node validation. however, it should be mentioned that indirect node validation imposes a large operational overhead on the network. Another problem with all authentication-based techniques, as is the case with this one, is the requirement of secure code pre-distribution channels which are not addressed clearly.

Another authentication-based approach is a light-weight identity certificate method which uses one-way key chains and Merkle hash trees to defeat Sybil attacks [13]. This method requires a significant amount of memory for storing information. The authors claim to have overcome this issue by means of the low level Merkle hash tree cryptographic.

The last authentication-based approaches to be reviewed, use puzzle solving techniques to detect Sybil attackers [12,22]. Another way of authentication is using puzzle-based computational mechanisms [12,22]. The main idea, herein, is that the attacker should not be able to solve a subject puzzle. There is an inherent communication overhead due to puzzle dissemination and receiving puzzle solutions back. As to the best of our knowledge, the puzzle solving techniques have not been reported in WSNs so far.

Identity registration is an centralized approach which is introduced in [6]. All nodes are registered in a trusted central authority (such as a base station) and the list of legitimate IDs are distributed amongst all nodes. To prevent the Sybil attack, each ID should be checked against the list of legitimate IDs. An important concern is that list of legitimate nodes must be protected from being maliciously modified. If the attacker is able to add IDs to this list, he will be able to add Sybil nodes to the network.

The last category of Sybil attack detection algorithms to be reviewed is the neighborhood-based methods. This category is of special interest to us since our proposed algorithm belongs to it. Ssu et al. [8] propose a neighborhood–based method for detecting Sybil nodes which uses the fact that a malicious node produces similar neighbors lists corresponding to its different Sybil IDs. Each node constructs a critical set of neighboring IDs using similarities of different neighbor lists that it receives. IDs which transmit neighbors lists containing the critical set are labeled as Sybil. In the case of mobile nodes, the neighbors lists will change so often that they result in high communicational overhead.

In [23] a Sybil attack detection method is proposed for MANETs based on cooperative monitoring. The packet receiver or forwarder can provide a proof that the sender transmitted the packet at the claimed location and time using a signature field that is unique for a non-malicious node. The results of these observations are periodically communicated between nodes. From these observations path similarities for packets originating from Sybil IDs are extracted and acted upon. The proposed algorithm involves computationally intensive procedures for signature operations, communication overhead for communicating observations, and hardware cost associated with employing directional antennas. Therefore, it seems not to be suitable for WSNs.

Another neighborhood-based method which can serve as a comparison basis for our proposed algorithm is stated in [24]. In [24] a Sybil attack detection method is proposed based on observed transmissions. This method, called Passive Ad hoc Sybil Identity Detection (PASID), uses the fact that all Sybil IDs of a single malicious node must move together because they are bound to a single physical node. Therefore, the Sybil nodes could be detected by periodically observing the network. It assumes that there is a single malicious node in the network that fabricates IDs. A subset of the legitimate nodes observe all received transmissions over time intervals. Then, the observing nodes exchange their information to identity the nodes which were heard simultaneously in the same duration. Finally, a graph-based profile of co-heard nodes is constructed where the weighted edge between two vertexes denotes their affinity. Piero et al. [24] do not study scenarios with multiple malicious nodes.

Finally, in [25], a watchdog node labels other nodes it sees as they move around. For instance, in a 4-watchdog scenario, watchdog nodes have certain labels like 00, 01, 10, and 11. Each watchdog assigns and stores its corresponding label to nodes which appear in its neighborhood. Periodically, watchdog nodes exchange their assignment information (moving_history) with each other to update the so called bit_label of their neighbor nodes. At the end, each watchdog node detects Sybil IDs by investigating bit patterns in its bit_label. Because of the periodic information exchange between

watchdog nodes, and since each watchdog node has to store the whole bit_label of its neighboring nodes, the protocol imposes a lot of communication and memory overheads. Also, an error in bit_label in one of the watchdog nodes is propagated to other watchdog nodes. At the end, it seems that labeling may non-linearly increase computational, communication, and memory overheads for an increased number of watchdog nodes.

The latter two protocols [24,25] are best fit to serve as bases for comparison against our proposed protocol due to similarity in assumptions, specifically the attack model and the mobility-based-computation considerations.

3. Problem Statement and Attack Model Assumptions

In what follows we describe the network assumptions and attack models employed. The subject network consists of two sets of nodes, normal sensor nodes (SN) which perform sensing, routing, and data aggregation, and watchdog nodes (WD) which are responsible for network monitoring and detecting Sybil IDs. It is further assumed that, in order to conceal their presence, the watchdog nodes do not send any messages while overhearing the transmissions of their neighbors.

Each node has a unique ID and is not aware of its geographical location. All nodes (normal and watchdog) have the same wireless range and move according to Random-Way-Point mobility model [7] during the network lifetime. Regarding the attack models, we have considered the "Direct, Simultaneous and Fabricated IDs" Sybil attack models as described in [6]. The subject network is insecure due to the presence of some malicious nodes (MN) that fabricate some IDs, representing Sybil IDs. the malicious nodes broadcast Hello Packets using these fake IDs. This is intended to disrupt routing operation in the network.

4. Proposed Algorithm

The following entities are the core to the proposed algorithm.

- **A^{k}_{co-prs}:** This is an upper-triangle $H \times H$ matrix which contains the co-presence status of all node pairs at time index k. The elements of A^{k}_{co-prs} are in the form of $xy : x, y = 0/1$ where 0 and 1 represent absence and presence respectively (Figure 2a). In Figure 2, H refers to the total number of Sybil and normal-node IDs which is equal to:

$$H = N + M * S \tag{1}$$

  where $N = |SN|$, $M = |MN|$, and S equals the number of Sybil IDs per malicious node.
- **C^{k}_{co-prs}:** This matrix which is structurally similar to A^{k}_{co-prs} (upper-triangle $H \times H$) scores the co-presence of each node pair at time index k (Figure 2b) and is updated according to A^{k-1}_{co-prs}, A^{k}_{co-prs} and the co-presence state diagram model.
- **Co-presence state diagram model:** This diagram shows how a transition between co-presence states of ID pairs updates the elements of the C^{k}_{co-prs} matrix (Figure 3).

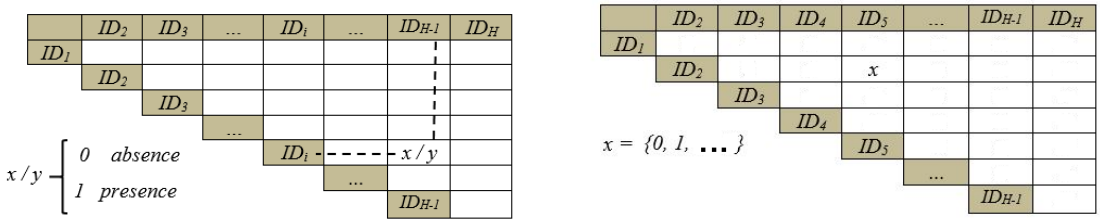


Figure 2. Structure of the WD matrices: a) A^{k}_{co-prs}, b) C^{k}_{co-prs}.

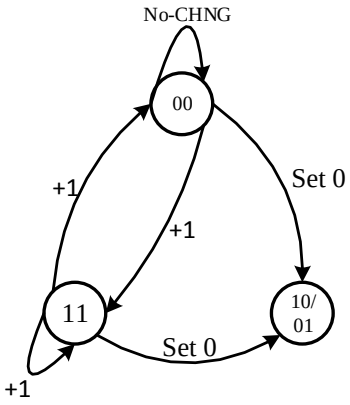


Figure 3. Co-presence state diagram model of ID pairs.

The algorithm is described in steps, with numbered references to the flowchart of Figure 4, as follows.

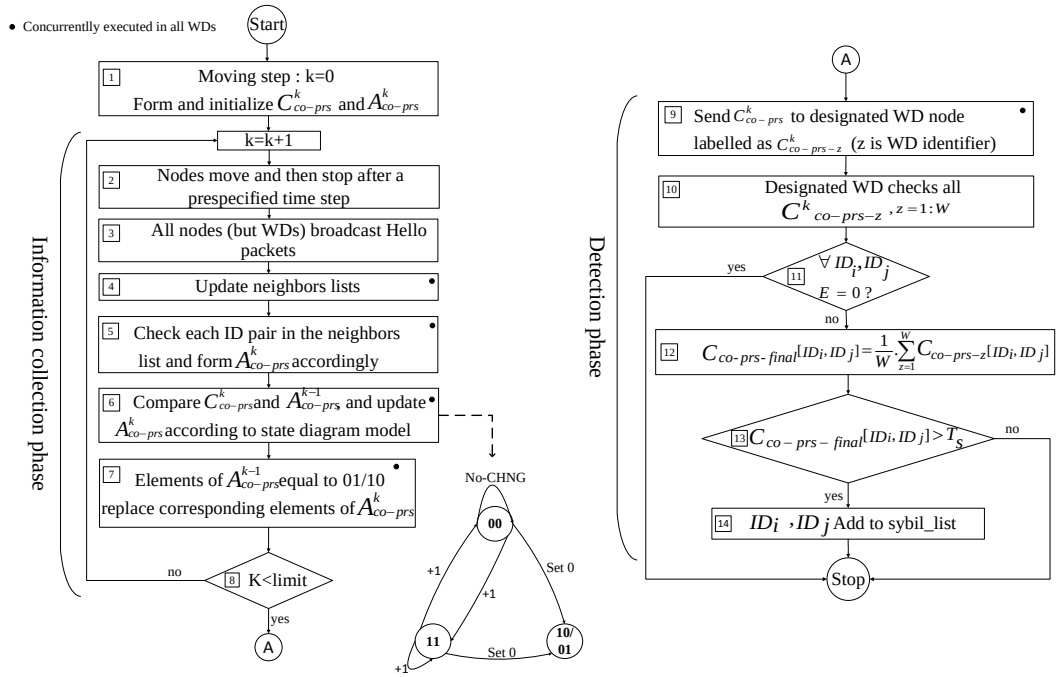


Figure 4. Flowchart of the proposed algorithm.

- **Initialization phase**

  Step I (1): Each WD constitutes of matrices A^k_{co-prs} and C^k_{co-prs} and fills the former with 00s corresponding to the general co-absence status.

- **Information collection phase**

  Step II (2, 3, 4): Hello Packets are broadcasted by normal and malicious nodes at fixed time intervals (movement steps). Following each movement, all WD nodes update their neighbors lists by overhearing these Hello Packets.

  Step III (5): Each WD forms the new A^k_{co-prs} based on its current neighbors list as follows. Two-digit binary numbers 00, 11, and 10/01 correspond to co-absence, co-presence, and

alternate-presence statuses respectively. Specifically, if ID_i *and* $ID_j \notin$ *neighbors* $-$ *list*, the content of element (ID_i, ID_j) of matrix A^k_{co-prs} is set to 00. On the other hand, if ID_i *and* $ID_j \in$ *neighbors* $-$ *list*, this content is set to 11. Finally, if $ID_i \in$ *neighbors* $-$ *list* and $ID_j \notin$ *neighbors* $-$ *list* (or vice versa), the content of the corresponding element of A^k_{co-prs} is set to 10. These values also represent the states in the co-presence state diagram model.

Step IV (6): By comparing the contents of the corresponding elements of A^{k-1}_{co-prs} and A^k_{co-prs}, the content of the corresponding element of C^k_{co-prs} is updated according to the transitions in the state diagram. For instance, when $A^{k-1}_{co-prs}[ID_i, ID_j] = 00$ and $A^k_{co-prs}[ID_i, ID_j] = 11$, according to the state diagram, $C^k_{co-prs}[ID_i, ID_j]$ must be increased.

Step V (7, 8): In this step, A^k_{co-prs} is updated so that the elements of A^{k-1}_{co-prs} which are equal to 01/10 replace the corresponding elements of A^k_{co-prs} (trap states are preserved).

Steps II to V are repeated for a predetermined number of times representing the network's lifetime.

- **Detection phase**

  Step VI (9, 10, 11, 12): All WDs send their co-presence information C^k_{co-prs} (labeled as $C^k_{co-prs-z}$) to a designated WD node . The designated WD then checks the elements of $C^k_{co-prs-z}$ for each WD and creates a $C_{co-prs-final}$ matrix according to:

$$C_{co-prs-final}[ID_i, ID_j] = \frac{E.Func}{W} \tag{2}$$

wherein

$$Func = \sum_{z=1}^{W} C^k_{co-prs-z}[ID_i, ID_j] \tag{3}$$

and

$$E = \begin{cases} 0 & \text{if } \exists z \neq y \mid C^k_{co-prs-y}[ID_i, ID_j], \\ & \quad C^k_{co-prs-z}[ID_i, ID_j] = 0\,, \\ & \quad y, z = 0, 1, ..., n\,, \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

Step VII (13, 14): The designated WD examines the elements of $C_{co-prs-final}$ against a predetermined Sybil threshold, T_s. If $C_{co-prs-final}[ID_i, ID_j]$ is greater than T_s, ID_i and ID_j are added to its internally maintained Sybil list. T_s is a representation of how often on average two non-sybil IDs are expected to co-appear in one WD's neighborhood. T_s can be specified by trial and error to generate satisfactory true and false detection results. The designated WD finally broadcasts the Sybil list to other WDs to act upon.

Notes :

- There are rare circumstances where the proposed algorithm falsely detects normal nodes as Sybil IDs (false negative). In particular, this happens when a malicious node and a normal node simultaneously move in and out of a WD's neighborhood and because the proposed algorithm operates based on co-appearance detection .
- While the detection phase is implemented in the designated WD node, it is no different from other WDs. If the designated WD fails, provisions could be put in place (as a future work) to replace it with another WD. Thus, the algorithm can be thought of as being somewhat protected from the single-point-of-failure problem.

A Typical Example: As a typical example, assume a scenario in which there is a sensor network consisting of four normal nodes (ID_1, ID_2, ID_6, and ID_7), four WDs (W_1, W_2, W_3, and W_4 one of which plays the role of the designated WD), and one malicious node (labeled M, generating three Sybil IDs (namely ID_3, ID_4, and ID_5) all moving around according to the Random-Way-Point movement model.

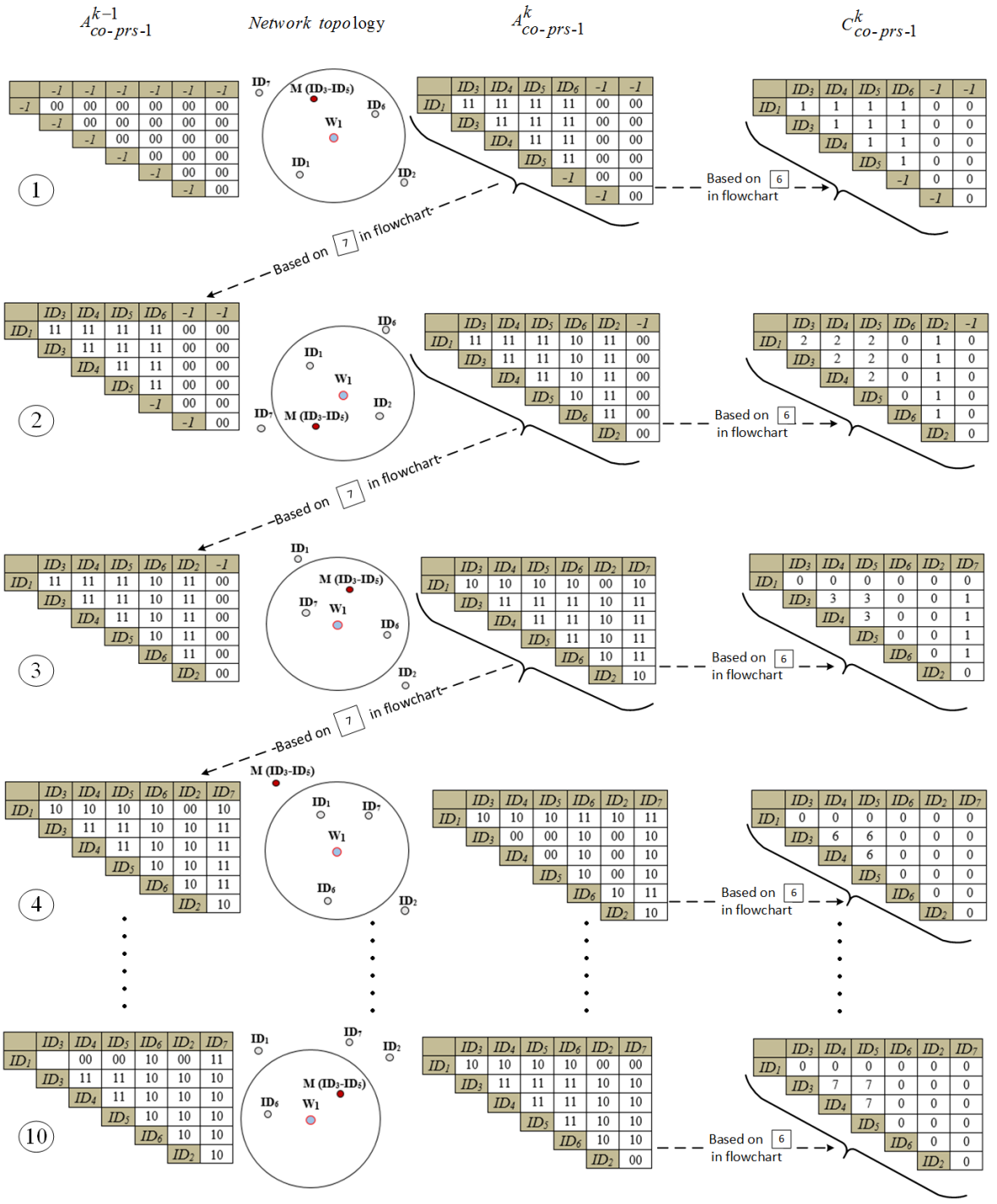


Figure 5. A pictorial representation of the advancement of the information collection phase in a typical watchdog, herein, W_1.

- Information collection phase: Figure 5 shows the information collection phase of the algorithm illustrating how it proceeds step-by-step as is implemented in W_1. The same procedure is repeated in $W_2 - W_4$ not shown here for the sake of brevity.

  In each row (corresponding to a specific time step), the left most column illustrates the former IDs-co-presence status of the network shown by matrix $A^{k-1}_{co-prs-1}$. The second column from the left illustrates the network topology after applying one-step movement.

Similarly, the third column from the left shows the current (post-movement) IDs-co-presence status of the network presented by matrix $A^{k}_{co-prs-1}$. By comparing $A^{k-1}_{co-prs-1}$ and $A^{k}_{co-prs-1}$, the scoring matrix $C^{k}_{co-prs-1}$ is updated according to the state diagram of Figure 3, as shown in the forth column. This algorithm proceeds in the following row (corresponding to the next time step) starting with a $A^{k-1}_{co-prs-1}$ generated from $A^{k}_{co-prs-1}$ in the preceding row and manipulated by $\boxed{7}$ in the flowchart of Figure 4. This phase is terminated by reaching the simulation step limit which is equal to 10 in this example.

- Detection phase: The designated WD receives the $C^{k}_{co-prs-z}$ matrices from the other three WDs. Then the designated WD uses the received information, and its own $C^{k}_{co-prs-z}$ matrix to form the $C_{co-prs-final}$ matrix according to Equation (2). Finally, each element of the $C_{co-prs-final}$ matrix is compared against the Sybil threshold, $T_s = 1$, to detect and announce the Sybil IDs as illustrated in Figure 6.

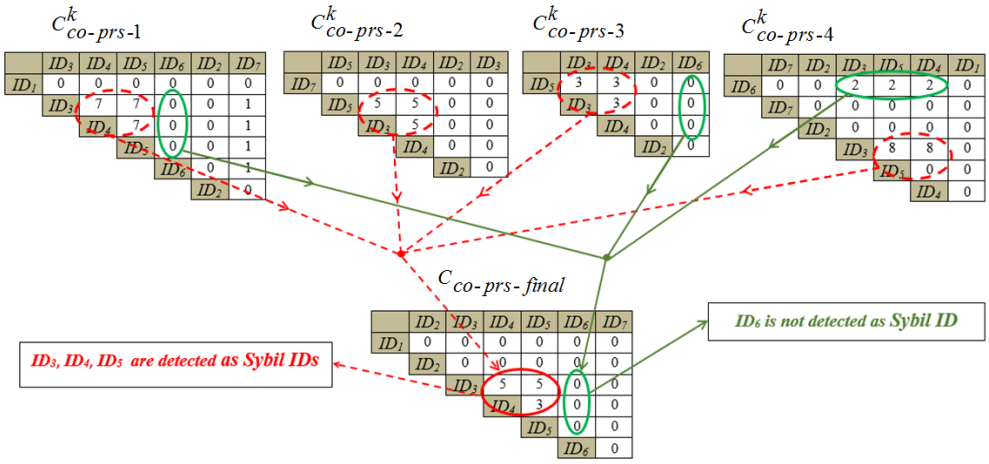


Figure 6. A pictorial representation of the advancement of the detection phase (Sybil IDs announcement) in the designated WD which is selected from $W_1 - W_4$ beforehand.

5. Simulation Results and Performance Evaluation

In this section, first we start with introducing the proper measures of performance needed for evaluating our algorithm. This is followed by describing the simulation setup. Finally, the merits of our algorithm is studied in the simulation results section.

5.1. Measures of performance

Measures of performance commonly used for evaluating the efficiency of detection algorithms are as follows.

- True Detection Rate (TDR): the percentage of Sybil nodes detected by a detection algorithm.
- False Detection Rate (FDR): the percentage of normal nodes detected as Sybil nodes erroneously.
- Memory overhead: the amount of memory consumption for algorithm implementation.
- Communicational overhead: the amount of extra algorithm-specific control-information required for algorithm implementation.
- Computational load: the number of computational operations needed for implementing an algorithm.

5.2. Simulation Setup

The proposed algorithm is simulated using J-SIM ([26]) where its physical layer employs free-space and Two-ray ground models. Regarding the topology of the network, there are a total number of N

normal nodes, W watchdog nodes, and M malicious nodes all of which move according to the 2D Random-Way-Point process and with a random speed confined to a maximum speed limit. Initially, all nodes are randomly located over the network area. Each malicious node fabricates S number of Sybil IDs. The value of the above parameters and other simulation parameters appear in Table 1. Regarding the Sybil threshold, T_s, the adoption of value 1 represents an astringent choice.

Table 1. Simulation Parameters.

Parameter	Value/Fixed	Value/Variable
No. of normal nodes, N	300	$100 \rightarrow 400$
No. of watchdog nodes, W	4	$4 \rightarrow 10$
No. of Sybil IDs/malicious node, S	14	$10 \rightarrow 22$
No. of malicious nodes, M	5	$1 \rightarrow 10$
Max speed of nodes	5 m/s	-
Topology size	100 m $\times$ 100 m	-
Wireless radio range	10 m	-
Simulation time step	50 ms	-
Sybil threshold, T_s	1	-

5.3. Simulation Results

In this section, we evaluate our algorithm by setting up four tests, and thereafter we compare our algorithm with two more relevant recent works.

5.3.1. Evaluation

The TDR and FDR of the new algorithm are evaluated by setting up five tests. These tests differ by the choice of the independent and fixed parameters. To ensure the validity of the results, each point is the average of 30 repetitions to achieve a 95% confidence interval.

At the end, the three remaining measures of performance, memory/communicational overhead, and computational load are discussed qualitatively.

- **Test 1:** This test is designed to evaluate the new algorithm's TDR and FDR against the number of movement steps for different number of Sybil IDs as a parameter varying from 8 to 20. The value of the remaining parameters are fixed as they appear in Table 1.
  Figure 7a shows that TDR increases as time goes by. This is completely expected since more information is collected in longer periods of time. After 120 movement steps, for any number of Sybil IDs, good detection rates of at least 95% are achieved.
  Similarly, we expect FDR to improve by collecting more information over time towards higher movement steps. As Figure 7b illustrates, regardless of the number of Sybil IDs, FDR sufficiently nears zero at movement steps close to 160.
- **Test 2:** In this test, TDR and FDR have been evaluated against the number of movement steps for different number of normal nodes varying from 100 to 400 (The values of the remaining parameters are fixed as they appear in Table 1).

  As shown in Figure 8a and very consistent with the results of Figure 7a, good TDR results are obtained after 120 number of movement steps, no matter what the number of normal nodes is Similar to what we observed in Figure 7b of Test 1, Figure 8b shows that FDR decreases with increasing movement steps until reaching almost zero at movement steps close to 200.
- **Test 3:** Through this test it is illustrated how TDR and FDR are affected by varying the number of malicious nodes from 1 to 10. The fixed parameters are $S = 10, W = 4$, and $N = 200$ (for the values of other parameters refer to Table 1).

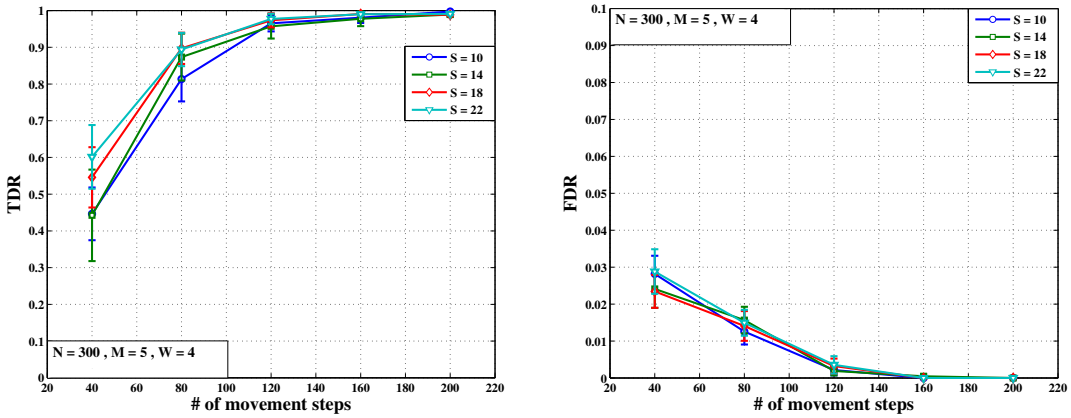
  In Figure 9a,b, almost perfect TDRs and FDRs are achieved for all number of malicious nodes from movement steps greater than or equal to 120 on.

- **Test 4:** Values of TDR and FDR against the number of Sybil IDs for different number of watchdog nodes is the subject of this test. The results correspond to movement step equal to 160 where the system has already reached its steady state. As before, the other fixed parameters are as they appear in Table 1.

  Figure 10a suggests that while increasing the number of WDs results in better TDRs, the improvement is negligible after some case-dependent WD population ($W = 6$ and greater). Figure 10b illustrates perfect FDR at the steady state for all choices of WD population.
- **Test 5:** In this test, the effect of scalability on the TDR/FDR performance of the proposed algorithm is verified (Figure 11). To make it a fair comparison, all node populations (normal, malicious, and watchdog) grow proportionally with the network area. The value of population parameters and the snapshot instance (Movement step) are mentioned in the figure. The maximum network size adopted is constrained by the limitations of J-SIM. The other fixed parameters are as they appear in Table 1.

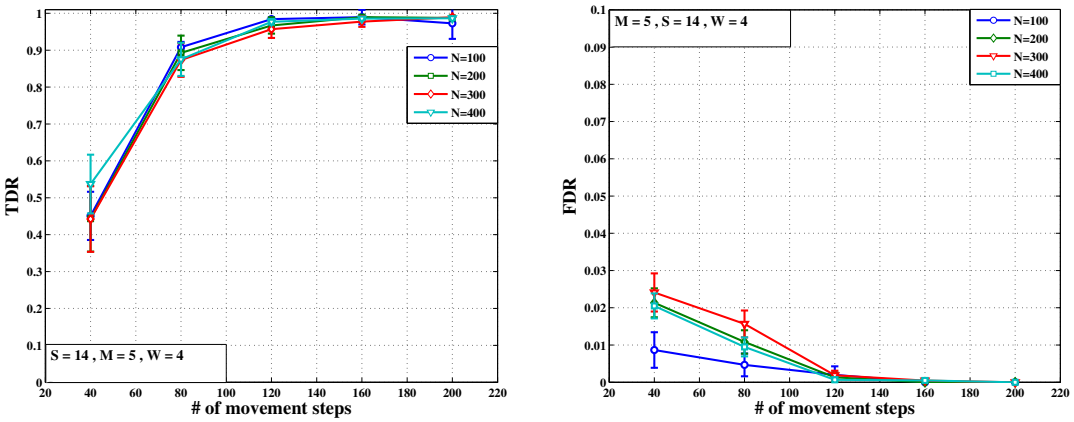
  The TDR/FDR results of Figure 11 show very insignificant variations with respect to the network size. This is somewhat expected since in wireless networks it is the routing that is mostly susceptible to scalability which is not a concern herein.

  Below we give a summary of the test results of the proposed algorithm.



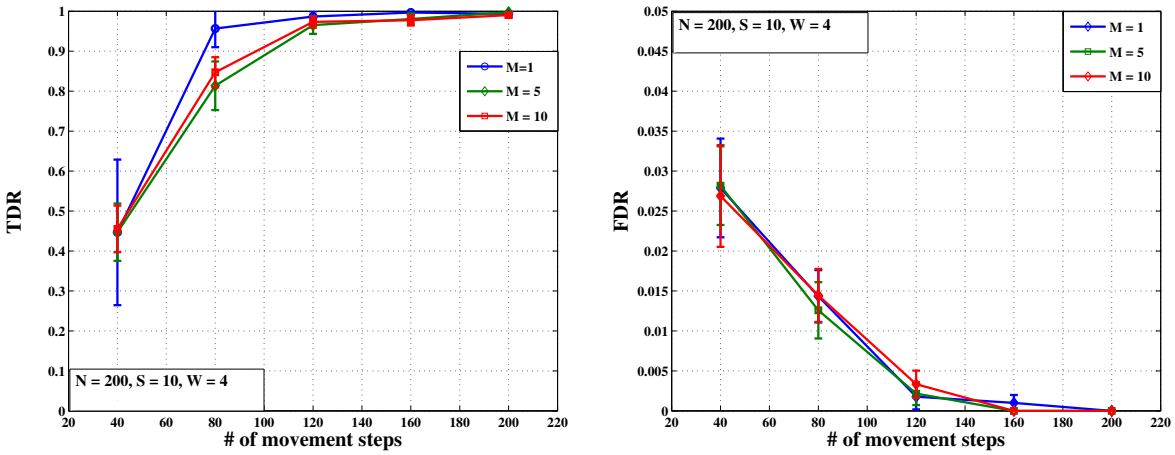
(**a**) (**b**)

Figure 7. Variation of TDR (**a**) and FDR (**b**) versus time for different number of Sybil IDs.



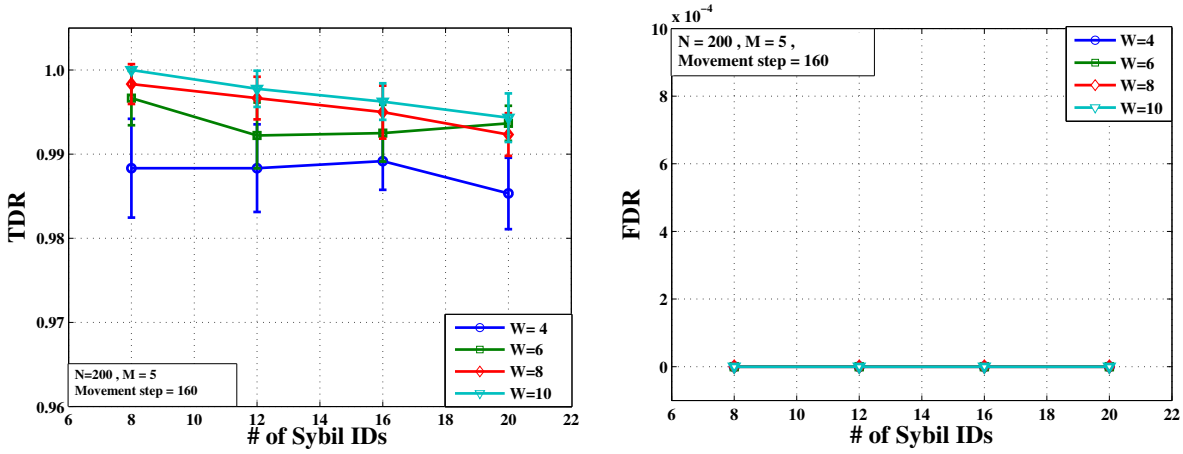
(**a**) (**b**)

Figure 8. Variation of TDR (**a**) and FDR (**b**) versus time for different normal node populations.



(**a**) (**b**)

Figure 9. Time variations of TDR (**a**) and FDR (**b**) for different malicious node populations.



(**a**) (**b**)

Figure 10. The effect of WD population on the variation of TDR (**a**) and FDR (**b**) versus the number of Sybil IDs.

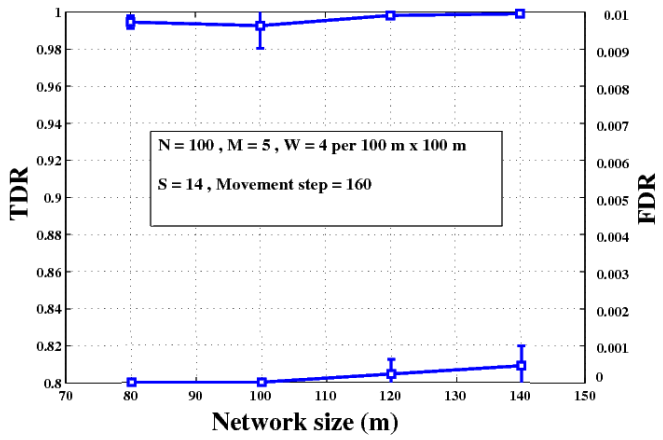


Figure 11. Scalability evaluation of the proposed algorithm.

- Observing Figures 7 and 9, TDRs and FDRs surely reach their steady-state values (upon sufficient information collection) and interestingly almost at the same time (at movement step greater than or equal to 120) regardless of the values of the network parameters.
- Comparing Figures 7 and 8, the convergence rates of TDR and FDR are more sensitive to the number of Sybil IDs per malicious node and the number of normal nodes respectively, before reaching their steady states. This is expected because of the very definition of TDR and FDR in Section 5.1.
- Somewhat related to the previous observation, Figures 7a and 9a show that increasing the total number of Sybil IDs by increasing S and M have opposite impacts on TDR. In fact, increasing S makes the Sybil pattern more noticeable and in favor of TDR. As illustrated by Figures 7b and 9b increasing the total number of Sybil IDs does not have much effect on FDR.
- The results in Figure 10a suggest that, in steady state, there is a turning point of WD population above which not much TDR gain is achieved.

Regarding the three remaining measures of performance, in the proposed algorithm, the memory overhead is equal to the amount of memory needed for storing upper-triangle matrices A^k_{co-prs} and C^k_{co-prs} which is equal to $O(H^2)$ (refer to Equation (1)). It should be noted that this memory consumption occurs only in WDs.

To do their jobs, WDs overhear the Hello Packets which are already in use in mobile sensor networks. After forming and updating C^k_{co-prs}, each WD sends its final C^k_{co-prs} to the designated WD just once, in the detection phase. So the communicational overhead is $O(H^2)$. It is very important to note that in calculating the communicational overhead, we do not consider Hello-Packets related overhead (and its consumed energy) since Hello-Packets broadcasting mechanism is embedded in any mobile wireless routing and is not additionally imposed by our proposed detection algorithm.

The algorithm includes arithmetic and comparison operations on upper-triangle matrices (with $H \times H$ dimensions) in WDs and the designated WD leading to a computational complexity of $O(H^2)$.

5.3.2. Comparison

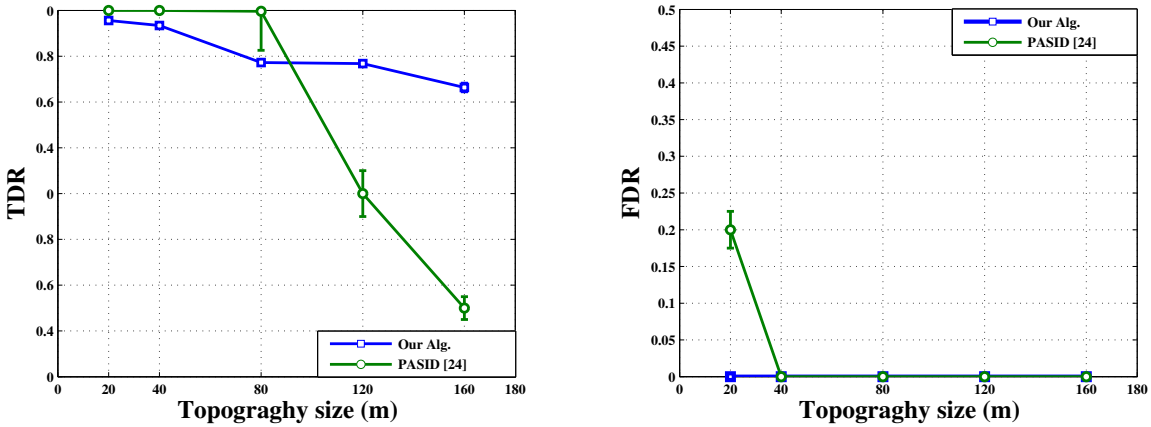
In this section, we compare our algorithm with two methods in [24,25] in terms of TDR and FDR. These works are chosen due to the similarities in the adopted attack models and approach.

- **Comparison 1:** To compare our algorithm with PASID, we have to set the simulation parameters as per [24]. Therefore, in this part, we assume $M = 1$, $max\ speed = 0.2$ m/s, $wireless\ radio\ range = 10$ m, and the results are calculated at $movement\ step = 200$. Figure 12 illustrates TDR and FDR for PASID and our algorithm against the network size (herein called topography size as in [24]). To be consistent with results in [24], each point of the new proposed algorithm's results is an average over 240 combinations of $N = 5, 10, 25, 40$, and $S = 5, 10, 20$, and $W = \lfloor N/2 \rfloor$ where the latter corresponds to the best case result reported in [24]. The results of PASID are extracted from [24]. The new algorithm performs better than PASID in terms of FDR for all network sizes. In terms of TDR, the new algorithm has a significant edge over PASID in large networks while slightly under-performing it in smaller ones.
- **Comparison 2:** In this test we compare our algorithm with another relevant algorithm in [25]. Our algorithm uses a somewhat similar information collection strategy as in [25] while using a completely different detection rule. To make our results comparable, we adopt the same values for our simulation parameters as those in [25].

Figure 13 compares TDR and FDR variations versus time for different number of Sybil IDs. It should be noted that [25] provides the results just for a limited number of movement steps (time period) and no confidence interval consideration. Figure 13a suggests that initially TDR and its improvement rate in [25] is better than our algorithm. However, at some point our algorithm not only catches up but also starts out-performing [25]. Both algorithms expectedly show slightly

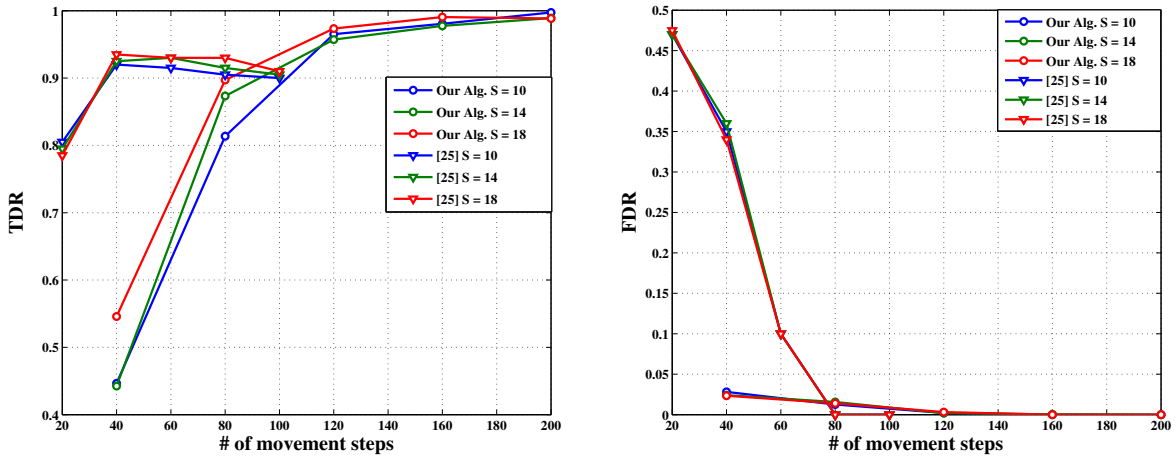
better performance by increasing the number of Sybil IDs per malicious node (S). Regarding FDR, our algorithm starts significantly better and the difference in performance vanishes through time. Consistent with its definition, FDR in Figure 13b shows no sensitivity to variation of S. The FDR of our algorithm is initially much better than the FDR of the algorithm in [25]. However, both algorithms perform asymptotically perfect in the long term. Time variations of TDR and FDR with varying malicious node population for both algorithms are illustrated in Figure 14. We observe almost the same trends as in Figure 13 with the following particularities. TDR in [25] seems to decline after some time as apposed to the new algorithm wherein TDR keeps improving constantly. Moreover, only TDR in the new algorithm and only FDR in the algorithm of [25] show sensitivity to the malicious node population in the transient state. To sum up:

- Regarding TDR, although the detection process is slightly late in picking up due to its conservative approach (Equation (2) in step V) to somewhat compensate for erroneous receptions , it performs reliably and asymptotically better.
- Introducing the trap state (01/10) in the state diagram of Figure 13 and applying the detection rule of Equation (4) in step V result in an adequate FDR even in short term.



(**a**) (**b**)

Figure 12. Performance comparison of PASID and our new proposed method in terms of TDR (**a**) and FDR (**b**) versus the topography.



(**a**) (**b**)

Figure 13. Performance comparison of [25] and our new proposed method in terms of TDR (**a**) and FDR (**b**) versus number of movement steps for varying number of Sybil IDs.

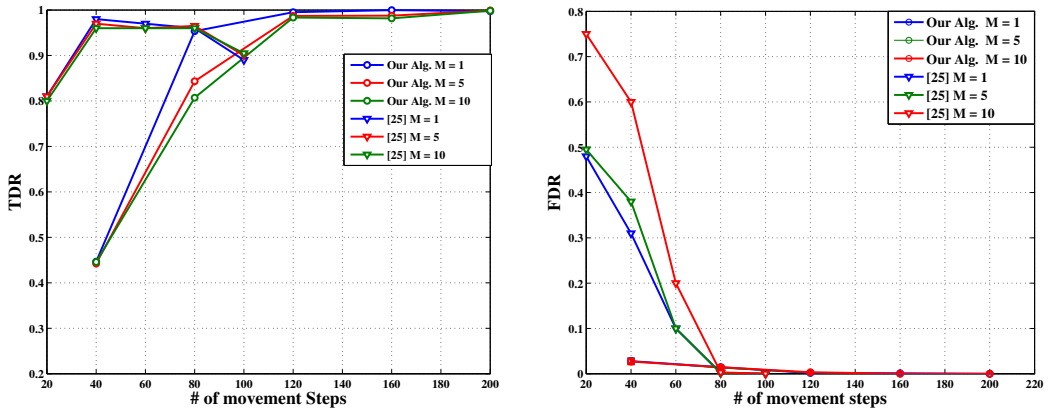


Figure 14. Performance comparison of [25] and our new proposed method in terms of TDR (**a**) and FDR (**b**) versus number of movement steps for varying number of malicious nodes.

Finally, we would like to point out that the diagrams representing variations of TDR and FDR versus the number of normal nodes in Figure 10 of [25] are not usable for comparison purposes.

6. Conclusions

In this paper, a new computationally lightweight method for detecting Sybil IDs in mobile WSNs is proposed. The proposed algorithm employs watchdog nodes which overhear Hello Packets exchanges between nodes. Each watchdog node uses a newly introduced co-presence state diagram to produce partial detection information. A designated watchdog node aggregates all partial detection information and uses a new detection rule to generate the final Sybils list. The new algorithm features low extra communication overhead, as well as a satisfactory compromise between two otherwise contradictory detection measures of performance, TDR and FDR.

The performance of the new algorithm is evaluated and compared with other similar important recent watchdog-based algorithms using extensive simulations. The simulation results illustrate the merits of the algorithm collectively.

Author Contributions: The authors contributed equally to this work.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

WSN	Wireless Sensor Network
TDR	True Detection Rate
FDR	False Detection Rate
PHY	Physical
RSSI	Received Signal Strength Indicator
TDoA	Time Difference of Arrival
AoA	Angle of Arrival
IoT	Internet of Things
MANET	Mobile Ad hoc Network
PASID	Passive Ad hoc Sybil Identity Detection
UWB	Ultra-Wide Band
USAS	Unpredictable Software-based Solution
WD	Watchdog Node
SN	Sensor Node
MN	Malicious Node

References

1. Akyildiz, I.F.; Su, W.; Sankarasubramaniam, Y.; Cayircl, E. A survey on sensor networks. *IEEE Commun. Mag.* **2002**, *40*, 102–114.
2. Karlof, C.; Wagner, D. Secure routing in wireless sensor networks: Attacks and countermeasures. *Ad Hoc Netw.* **2003**, *1*, 293–315.
3. Sagduyu, Y.E.X.; Berry, R.A.; Ephremides, A. Jamming Games in Wireless Networks with Incomplete Information. *IEEE Commun. Mag.* **2011**, *49*, 112–118.
4. Tsiropoulou, E.E.; Baras, J.S.; Papavassiliou, S.; Qu, G. On the Mitigation of Interference Imposed by Intruders in Passive RFID Networks. In Proceedings of the 7th International Conference GameSec, New York, NY, USA, 2–4 November 2016.
5. Chen, X.; Makki, K.; Yen, K.; Niki, P. Sensor Network Security: A Survey. *IEEE Commun. Surv. Tutor.* **2009**, *11*, 5–73 .
6. Newsome, J.; Shi, E.; Song, D.; Perrig, A. The Sybil Attack in Sensor Networks: Analysis & Defenses. In Proceedings of the Third International Symposium on Information Processing in Sensor Networks, Berkeley, CA, USA, 26–27 April 2004.
7. Bettstetter, C.; Resta, G.; Santi, P. The Node Distribution of the Random Waypoint Mobility Model for Wireless Ad Hoc Networks. *IEEE Trans. Mob. Comput.* **2003**, *2*, 257–269.
8. Ssu, K.F.; Wang, W.T.; Chang, W.C. Detecting Sybil attacks in wireless sensor networks using neighboring information. *Comput Netw.* **2009**, *53*, 3042–3056.
9. Seshadri, A.; Perrig, A.; Doorn, L.; Khosla, P. SWAtt: Software-based attestation for embedded devices. In Proceedings of the IEEE Symposium on Security and Privacy, Berkeley, CA, USA, 12 May 2004.
10. Seshadri, A.; Perrig, A.; Doorn, V.; Khosla, P. SCUBA: Secure Code Update By Attestation in Sensor Networks. In Proceedings of the 5th ACM Workshop on Wireless Security, Los Angeles, CA, USA, 29 September 2006.
11. Jin, X.; Putthapipat, P.; Pan, D.; Pissinou, N.; Kami Makki, S. Unpredictable Software-based Attestation Solution for Node Compromise Detection in Mobile WSN. In Proceedings of the Workshop on Advances in Communications and Networks, Miami, FL, USA, 6–10 December 2010.
12. Borisov, N. Computational puzzles as Sybil defense. In Proceedings of the IEEE international Conference on Peer-to-Peer Computing (P2P), Cambridge, UK, 6–8 September 2006.
13. Zhang, Q.; Wang, P.; Reeves, D.; Ning, P. Defending against Sybil attacks in sensor networks. In Proceedings of the 25th IEEE International Conference Distributed Computing Systems Workshops, Columbus, OH, USA, 6–10 June 2005.
14. Sharmila, S.; Umamaheswari, G. Node ID based detection of Sybil attack in mobile wireless sensor network. *Int. J. Electron.* **2012**, *100*, 1441–1454.
15. Marti, S.; Giuli, T.J.; Lai, K.; Baker, M. migration watchdog Mitigating Routing Misbehavior in Mobile Ad Hoc Networks. In Proceedings of the 6th Annual International Conference on Mobile Computing and Networking, Boston, MA, USA, 6–11 August 2000.
16. Sun, B.; Osborne, L.; Yang, X.; Guizani, S. Intrusion detection techniques in mobile ad hoc and wireless sensor networks. *IEEE Wirel. Commun.* **2013**, *14*, 56–63.
17. Douceur, J.R.; Druschel, P.; Kaashoek, M.F.; Rowstron, A. The Sybil attack. In Proceedings of the International Workshop on Peer-to-Peer Systems, Cambridge, MA, USA, 7–8 March 2002.
18. Abbas, S.; Merabti, M.; Llewellyn-Jones, D.; Kifayat, K. Lightweight Sybil Attack Detection in MANETs. *IEEE Syst. J.* **2013**, *7*, 236–248.
19. Demirbas, M.; Song, Y. An RSSI-based scheme for Sybil attack detection in wireless networks. In Proceedings of the International Symposium on World of Wireless Mobile and Multimedia Networks, Washington, DC, USA, 26 June 2006.
20. Wen, M.; Li, H.; Zheng, Y.-F. TDOA-based Sybil attack detection scheme for wireless sensor. *J. Shanghai Univ.* **2008**, *12*, 66–70.
21. Sarigiannidis, P.; Karapistoli, E.; Economides, A.A. Detecting Sybil Attacks in Wireless Sensor Networks using UWB Ranging-based Information. *Expert Syst. Appl.* **2015**, *42*, 7560–7572.
22. Li, F.; Mitial, P.; Cocsar, M.; Borisov, N. SybilControl: Practical Sybil defense with computational puzzles. In Proceedings of the 19th ACM Conference on Computer and Communications Security, Raleigh, NC, USA, 16–18 October 2012.

23. Tangpong, A.; Kesidis, G.; Hsu, H.; Hurson, A. Robust Sybil Detection for MANETs. In Proceedings of the International Conference on Computer Communications and Networks (ICCCN), San Francisco, CA, USA, 3–6 August 2009.
24. Piro, C.; Shields, C.; Neil Levine, B. Detecting the Sybil Attack in Mobile Ad hoc Networks. In Proceedings of the Secure-comm and Workshops, Baltimore, MD, USA, 28 August–1 September 2006.
25. Jamshidi, M.; Zangeneh, E.; Esnaashari, M.; Meybodi, M. A lightweight algorithm for detecting mobile Sybil nodes in mobile wireless sensor networks. *Comput. Electr. Eng.* **2016**, *40*, 1–13.
26. Sobeih, A.; Hou, J.C.; Kung, L.; Ning, L.; Zhang, H.; Chen, W.; Tyan, H.; Lim, W. J-Sim: A simulation and emulation environment for wireless sensor networks. *IEEE Wirel. Commun.* **2006**, *13*, 104–119.