Journal of Computer Science

Original Research Paper

Credence Aware Data Aggregation for Wireless Sensor Networks

[1]Swathi S, [2]Yogish H K, [3]Deepa Yogish and [4]Asha N

[1]Department of Computer Science and Engineering, Nagarjuna College of Engineering and Technology, Ramaiah Institute of Technology, Bengaluru, India
[2]Department of Information Science and Engineering, Ramaiah Institute of Technology, Bengaluru, India
[3]Department of Artificial Intelligence and Machine Learning, Donbosco Institute of Technology, Bengaluru, India
[4]Department of Master of Computer Applications, Mysuru, India

Article history
Received: 31-01-2022
Revised: 19-03-2022
Accepted: 31-03-2022

Corresponding author:
Swathi S
Department of Computer Science and Engineering, Nagarjuna College of Engineering and Technology, Ramaiah Institute of Technology, Bengaluru, India
Email: s.swathieswar@gmail.com

Abstract: To ensure data's reliability and credibility in Wireless Sensor Networks (WSNs), we provide an effective Credence-aware in-network aggregation design in persistent wireless sensor networks. This approach was motivated by a well-studied reputation and Credence relationships within social sciences. The proposed method uses an efficient CSDA algorithm to get more accurate results in terms of response time, penalty weights, the number of nodes, detection accuracy, etc. During the aggregating process, the Credence evaluation technique obtains benefits by identifying sensor node reliability, distinguishing illegal nodes, and filtering out erroneous data. The main objective of the work is to offer the most accurate answer possible to the user also while ensuring network health by identifying possibly compromised nodes. Experimental results show strategy is effective.

Keywords: Security, Wireless Sensor Networks, Aggregation, Credence

Introduction

Since increasing usage of Wireless Sensor Networks (WSNs) throughout everyday tasks grows in all of the commercial and defense sectors, developing a highly effective approach for secure query processing is becoming extremely relevant. Among the main query types for obtaining and analyzing sensor data, aggregation queries are resource-constrained enough to integrate within wireless sensor nodes and typical tree-based structures, for example, depending on the local sensor readings, an aggregation node evaluates a partial aggregate result. in addition to the readings provided by their children nodes, eventually sends the outcome to a higher-level parent node. Throughout this procedure of in-network aggregation, every node simply needs to send a single message of a fixed size to its parent, saving valuable bandwidth resources from restricted WSNs.

The safety component of the majority of existing sensor query methods is assumed that sensor nodes agree and are not deceptive. Wireless sensors are utilized in a variety of hostile conditions, including the battlefield and they are subject to a variety of threats. When a node is hijacked or hacked, the premise that all nodes are always cooperative is incorrect. Furthermore, because of the complicated structure with unexpected undesirable behaviors faced on WSNs, Conventional encryption and verification processes may give a limited degree of assurance but cannot provide a practical solution. For instance, when a node with appropriate encryption keys is easily compromised, this may easily implant fake sensor values or alter the aggregation value. The receiving nodes may utilize message encryption and verification to determine whether the messages from a specific node and were not been modified throughout propagation, However, they are unable to establish whether the sensor reading received is accurate. When using such a network aggregation approach, the problem becomes even worse since each node must conduct local aggregate depending on sensor readings received since each node must perform local aggregation derived from remotely sensed readings Whereas if the aggregation node purposefully updates the aggregate result and then subsequently transmits any modified data through the network, the receiving node is unaware. In most circumstances, a compromised aggregator has a bigger security impact than fake sensor readings

Outlier identification (Wu *et al*., 2007) is a technique that compares collected data to a set of values basis on previous domain expertise about the physical process being observed to determine if it corresponds. The data generator, on the other hand, is unable to detect a falsified sensor reading due to a lack of domain expertise. Incorrect sensor readings could be generated and then recognized

automatically if domain knowledge is specified. For instance, spatially adjacent sensor data may be employed to determine anomalous (false) results on spatially continuous processes like temperature. Temporal measurements can be used to describe temporally continuous phenomena such as humidity.

In rare situations, an outlier detector that relies entirely on geographically or temporally close observations may mistakenly label a valid reading as an exception. Consider how such a WSN may be used to detect a fast-moving vehicle. In an instance that sensor node A recognizes a vehicle, depending on present nor prior sensor node A data can produce equivalent vehicle detection accuracy in this instance, spatially and temporally near observations are typically favorable. When node A identifies a vehicle, which is certainly possible it has already been noticed by another neighboring node nearby. This type of historical data can be used to identify fake sensor readings. Fortunately, the majority of physical events detected by such a WSN are temporally, geographically, or spatiotemporally consistent, an outlier detection approach may be applied.

As partial aggregation results of different sub-networks include significantly greater uncertainty about sensor data from nearby sub-networks, an outlier detector can identify incorrect sensor data, However, it is unable to determine whether a partial aggregate result is incorrect. Gathering raw readings and evaluating them in a centralized area is a straightforward approach. The naïve technique, on the other hand, can considerably increase wireless transmission latency while sacrificing the benefits of in-network aggregation processing.

Credence-Aware In-Data Aggregate technique for resilient WSNs under this study by which Credence assessment technique can be applied to determine the integrity of sensor nodes, differentiate unauthorized nodes from normal nodes, as well as filter out fake data throughout the fusion process. This technique has differentiated itself in the social sciences by its high reputation and Credence model. The primary purpose of this study by provide the most specific response to the user while also monitoring network health and identifying possibly compromised nodes. In the social sciences, the method is distinguished by a high reputation and a Credence model. The objective of this study is to offer the user the most accurate answer possible while monitoring network health by identifying potentially affected nodes.

Review of Literature

Paper (Hu and Li, 2011) authored by B. Sun, X. Jin, K. Wu, Y. Xiao Suggested the mechanism based on the EKF (Extended Kalman Filter) for detecting the FID (False Injected Data). This method monitors the given sensor node which helps in predicting the aggregated value in the future. Here, a range is determined to detect the FDI. The method of EKF is also used for creating the LDM (Location Detection Mechanism). LDM helps in finding the difference between the emergency event and the malicious events. However, the FDI (False Data Injection) is considered only during the data forwarding.

Paper (Cao and Yu, 2011) H. Cam, S. introduces the Data Aggregation and Authentication (DAA) protocol, which integrates FDI with DA as well as confidentiality. To back the DA with the FDI, a monitoring algorithm is also introduced. The data aggregator's monitoring nodes do the DA as well as compute its message Authentication Code (AC) to verify the data at their respective pair-mates. Between the two data aggregators, the SN (Sensor Node) checks the DA upon this Encrypted Data (ED). The Data Packet (DP) is coupled to two messages-AC, each of which contains a T+1 authentication code. Up to T comprised nodes, the DAA detects the FD (False Data) injected, and these data are not taken further hence these are omitted.

Paper (Bidai *et al*., 2011) Yue-Hsun Lin, Shih-Ying Chang, and Hung-Min Sun proposed a method for multiple applications namely Concealed Data Aggregation (CDA) - work in a multiple application also known as CDAMA. This approach is intended for use in multiple application environments. In this case, The BS (Base Station) in this scenario eliminates the Application Specific (AS) information through the collected ciphertexts, limiting the risks of compromising attacks in a sole application environment. Eventually, it diminishes any destruction caused by illegal (unauthorized) gatherings. This method, however, is only relevant when the number of applications is less.

To get rid of the above issue, (Ozdemir, 2007) Chien-Ming Chen, Yue-Hsun L have been presented, the method known as Recoverable Concealed-DA (Data Aggregation) and firmly known as RDCA. This method is applicable for a large number of WSNs (unlike the previous method). The special feature of the scheme is that the BS recovers all the sensing data instead of summarized results. However, the Transmission Overhead (TO) remains acceptable. The ASS (Aggregation Signature Scheme) is used to guarantee the data's validity and integrity; such a system is less costly than the other model (compared above).

To provide security by mapping the reputation and the Credence of the node, paper (Sun *et al*., 2007) Mohsen Rezvani, Student Member, IEEE, Aleksandar Ignjatovic, Elisa Bertino, Fellow, IEEE, Sanjay Jha, proposes improvised Iterative Filtering, the approach is known as SDAT, which stands for SDA technique for WSN in the presence of Collision Attack. The data that arrives from the Comprised-Nodes is filtered here. Filtration is implemented based on the data's Credence worthiness, which is determined by computing the difference between the two rounds, i.e., data sensed in the present round and data sensed in the prior round. The accuracy of the IF algorithm is mostly determined by the initial Credence provided to each node. In the first round, all nodes are

Credenceed and fairly weighted. This phenomenon gives the attacker the ability to insert corrupt data. The main problem with this technique is that the CA is handled alongside the assumption of AN (Aggregator Nodes). As a result, threats are only addressed at the source nodes.

Cam and Ozdemir (2007) Choudhari *et al*. (2017) Mundada proposed an advanced collision attack against the several existing iterative-filtering algorithm, these algorithms are mainly based on the False Data (FD) injection. The method implied is the advanced version of iterative filtering, the algorithm is presented along with the novel scheme for revocation as well as the CD (Collision Detection), these are based on the initial approximation of the aggregated value and the difference between each reading.

Raha *et al*. (2011), introduced a novel optimization policy to balance the trade-off between energy and security aspects.

Choudhari *et al*. (2017), the conventional methods adopted for securing the WSN vulnerability-based attacks introduces delay, which brings congestion in the routing flow as well as influence the quality of service.

Unfortunately, the previous study did not account for more complex attack circumstances. False data insertion may be used to conduct highly complex attacks against WSNs using several compromised nodes. When the attackers may have a better understanding of the aggregation process and its configuration.

Credence-Aware in-Data Aggregation Approach

System Modeling

A network system is analyzed in this case, with the specified nodes arranged using the clustering algorithm (Cam and Ozdemir, 2007). In our system model, we utilized a single linked cluster with a huge amount of nodes.; the main goal in developing this model would be to gather data from various sensing nodes and built an overlay network to make it much more flexible, wherein two nodes can interact (i.e., exchange information) with one another. An undirected graph is used to depict an overlay network. Consider the undirected network $U = (X, Y)$, where X denotes the node X, Y denotes the edges (links) and B_m is the collection of neighbors of the given node m.

Suppose $A_m(0)$ becomes a node within the network's initial state; these phases represent the private information of each node, which means that the security of the node in its initial phases is a major focus of our work. In first section, the general agreement on secure data aggregation is provided which aids in the development of our algorithm.; the second part, Monitoring the nodes discusses monitoring deceitful or corrupt nodes; The remaining sections of this study, cover our suggested security technique, Efficient- Consensus-based Data Aggregation.

Figure 1 illustrates the process of our suggested technique, which consists of six stages. In the first stage, data is collected through sensor nodes, and in the second stage, the General Secure Data Aggregation Consensus is employed to provide security while also adding noise. The third stage is crucial because it allows our system to monitor the nodes. This can be accomplished using regulations or through monitoring nearby nodes, however monitoring over nodes allows more stability, therefore we selected the same. Our algorithm is then performed, so data aggregation is completed safely before being transferred to the Base Station.

General Secure Data Aggregation Consensus (GSDAC)

Every time a node communicates, it adds noise to the current state to ensure security. The noise added is shown in the equation below, i.e., Eq. 1:

$$a^{+}_{m}(l) = a_m(l) + \Theta_m(l), l \epsilon X \tag{1}$$

Now, $a_m(l)$ indicates the node's current state. During iteration I, Θ_m denotes noise which is used as Random Variable (RV):

$$a_m(l+1) = V_{mm} a^{+}_{m}(l) + \sum V_{mn} a^{+}_{m}(l), m \epsilon X, m \epsilon X \tag{2}$$

Equation 2 is a revised version of Eq. 1 and Eq. 3 would be the end result:

$$V_{mn} = \begin{cases} \dfrac{1}{\left[1 + \max\left\{\left[B_m\right], \left[B_n\right]\right\}\right]' \left(1 - \sum_{1 \epsilon N_i} V_{mr}\right)} & n \in B_n \\ 0, & \overset{m=n}{otherwise} \end{cases} \tag{3}$$

Equation 3 may be resolved in a distributed fashion as well. Equation 2 is represented in a matrix form as Eq. 4.

Equation 3 may be resolved in a distributed fashion as well. Equation 2 is represented in a matrix form as Eq. 4:

$$a(l+1) = V\left(a(l) + \Theta_m(l)\right) \tag{4}$$

In the above equation, ϵP^d, $V \epsilon P^{dXd}$ which satisfies a and V in Eq. 4:

$$\begin{aligned} A &= \left[a_1, a_2, \ldots \ldots, a_n\right]^z \\ \Theta &= [\Theta_1, \Theta_2, \ldots \ldots, \Theta_n]^z \\ V &= \left[V_{mn}\right]_{dxd} \end{aligned} \tag{5}$$

The discarding of corrupt nodes is required to establish the perfect average and secure consensus and this may be accomplished using two general methods.

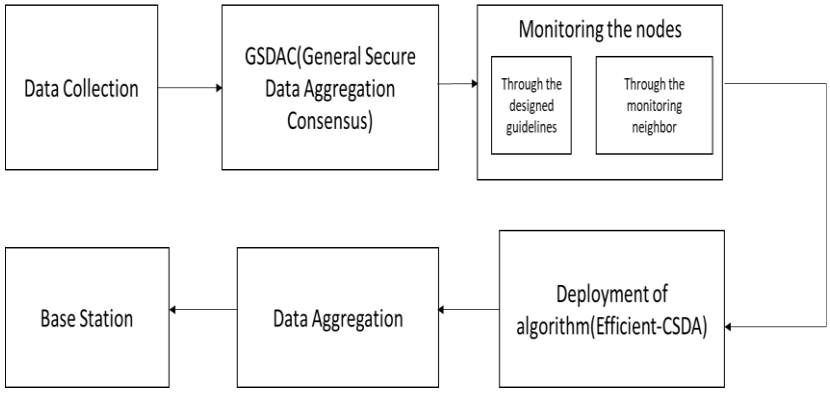


Fig. 1: Architecture for credence aware data aggregation approach

Monitoring the Dishonest Nodes

The security of network nodes should be supervised using one of two models. The first model is dimension expansion. The current states are divided into two different portions and these two parts, together with the extra noise, are delivered to the Neighbor Set. A set of criteria is in place to monitor the nodes and determine whether any misconduct is detected.

3.3.1 Monitoring the nodes following the established standards:

To keep track of the corrupt nodes, dimension expansion is utilized. The nodes are first divided into two distinct parts (Eq. 4 and 5) and then delivered to the neighboring nodes with the distortions:

$$a^1{}_m(0) = ½\, a_m(0) + e_m \quad (6)$$

$$a^1{}_m(0) = ½\, a_m(0) - e_m \quad (7)$$

e_m is selected at a random variable from the range of 0 <Y<1

Monitoring using a Neighbor Node

The aggregator asks a specific node to monitor a neighbor node at any moment, which would be a unique method of monitoring corrupt nodes. However, a few conditions must be satisfied to monitor.

Condition 1: $\left|\Theta^e{}_m\ (l)\right| \leq {}^1/_2 \alpha\rho^l$, *where* $\Theta^e{}_m(l)$ is computed by:

$$\Theta^e{}_m(l) = a^{e+}{}_n(l) - \left[V^e{}_{nn} a^{e+}{}_m(l-1) + \sum_{r\in B^e{}_n} V^e{}_{nr} a^{e+}{}_r(l-1)\right] \quad (8)$$

And $V^r{}_n$ is determined using the equation 3 for l . $l \in B^+$

Condition2: $a^+{}_n(0) - a^\wedge{}_n(0) \leq {}^5/_4 \alpha\rho$

If a preceding condition exists, then node j is the corrupt node.

Condition3: $\frac{a_n^+(0)}{2} - a_n^+(0) \leq \frac{5}{4\alpha\rho}$

Efficient-CSDA Algorithm

Step1: Random_Vector_Generation

Step2: initialization of $a^1_m(0)$ and $a^2_m(0)$ using the equation

Step3: initialization of $a^+_m(0)$ and $a^{g+}_m(0)$

Step4: value transmission of the step3 to their neighbor

Step5: in case if the given node m is chosen by aggregator himself for monitoring the neighboring node n, given data is formulated and denoted with e, T^g_r, T^g_n, B^g_n for the given value of e=1 or e=2 and $r \in B^g_n$.

Step6: set $\delta^g_m(0) = U^g_m(0)$

Step7: initializing r=1

Step8: while r<MAX_IT_NO do

Step9: if the node I is selected, the received value $a^g_m(l-1)$,afterwards the IS(Information set) $M^g_n(l-1)$ is used for monitoring the neighbor node.

Step10: report aggregator(if the node is found to be corrupt)

Step11: updation of $a^g_m(l) = B^g_{mm} a^{g+}_m(l-1) + \sum_{r\in B^g_m} V^g_{mr}\, x^{g+}_i(l-1)$

Step12: put $a_m(l) = a^1_m(l) + a^2_m(l)$

Step13: random _selection of $\delta^g_m(l)$

Step14: set $\theta^g_m(l)$ by formulating $\theta^g_m(l) = \delta^g_m(l) - \delta^g_m(l-1)$

Step15: put l=l+1

Step16: end_while_loop

Results and Discussion

The performance of our suggested model is shown in this section of the research. Our algorithm is examined to determine the results produced and the results are then compared to the current to illustrate that our suggested algorithm is appropriate.

Consider the sensor nodes are extensively placed to detect a specific target. Unlike compromised nodes, when a normal node starts transmitting an alert, its neighbors will start sending an alert after a brief delay. Moreover, after a specific number of cycles, typical alarming nodes will stop delivering alerts. The node that has been identified or misidentified as a malicious node gets deactivated from the entire process. The detection is turned off for 200 cycles, or when just about 25% of all nodes are recognized as fraudulent. Every outcome is based on 1000 individual simulations on average.

A sensor node deployment in a simulated environment is shown in Fig. 3. In a square plane, sensor nodes are evenly distributed. A sensor node might be malicious, normal, or alert-generating.

The detection algorithm's performance is measured using three measures. The response time, computed as overall detection phases of properly discovered malicious nodes, indicates as quickly malicious nodes can be identified. The detection rate, which is the proportion of malicious nodes that have been discovered to the overall number of malicious nodes, is used to determine the effectiveness of our scheme. The misdetection ratio is the proportion of properly recognized and misdirected nodes among all discovered nodes, Essentially, there are two aspects to these misdirected nodes: The number of normal nodes that have been taken for malicious nodes and the number of malicious nodes that have been mistaken for normal nodes. Short response times, high detection rates, and a low misdetection ratio are all sought in a malicious node detection strategy. We investigated the three metrics using simulations with various settings.

Weights on the System's Performance

During the first simulation, a detection algorithm is used to establish the optimal weight penalty. Both the attack and alarm probability remain 0.04. For a total of 10 cycles, normal nodes transmit alarms and wait for alerts to terminate. As previously stated, a detection threshold (0.4) is often specified.

The findings shown in Fig. 2 depict the results, that show weight penalties ranging between 0.02 to 1.0 and sensor node counts varying from 100 to 400. The increased weight penalty results in a shorter response time and a higher detection ratio. Intuitively, the penalty value represents the susceptibility to detecting variance in data collected. However, when the weight penalty grows, the misdetection ratio rises as well, especially once the penalty ratio reaches 0.08 or higher. Taking all of the tradeoffs between reaction time, detection accuracy, and misdetection rate into account, it is appropriate to fix these weight penalties within a range between (0.04-0.1).

Although the number of sensor nodes evolved between 9 to 900, the reaction time, detection, and misdetection ratios remained generally consistent; especially when there were more than 64 nodes. As a result of this discovery, the efficiency of the defined WTE-based detection algorithm is excellent, while this study well in a wide range of network sizes while compromising little performance. Performance is almost unaffected by network size, especially when it is large enough, for example, greater than 64.

Figure 3 depicts the effect of penalty weight selection. Choosing a larger value ($\theta = 0.1$) helps the method to detect malicious nodes quicker and more efficiently than using a smaller value ($\theta = 0.05$), as seen in Fig. 3(a) and the upper two curves in Fig. 3(b).

However, as indicated by the lower two curves in Fig. 3(b), this quicker response is accomplished at the cost of a larger misdetection rate Fig 3(b). This illustrates that the penalty weight parameter's sensitivity may be modified by the system operator to match the needs of different applications, proving the balance between detection performance and misdetection ratio.

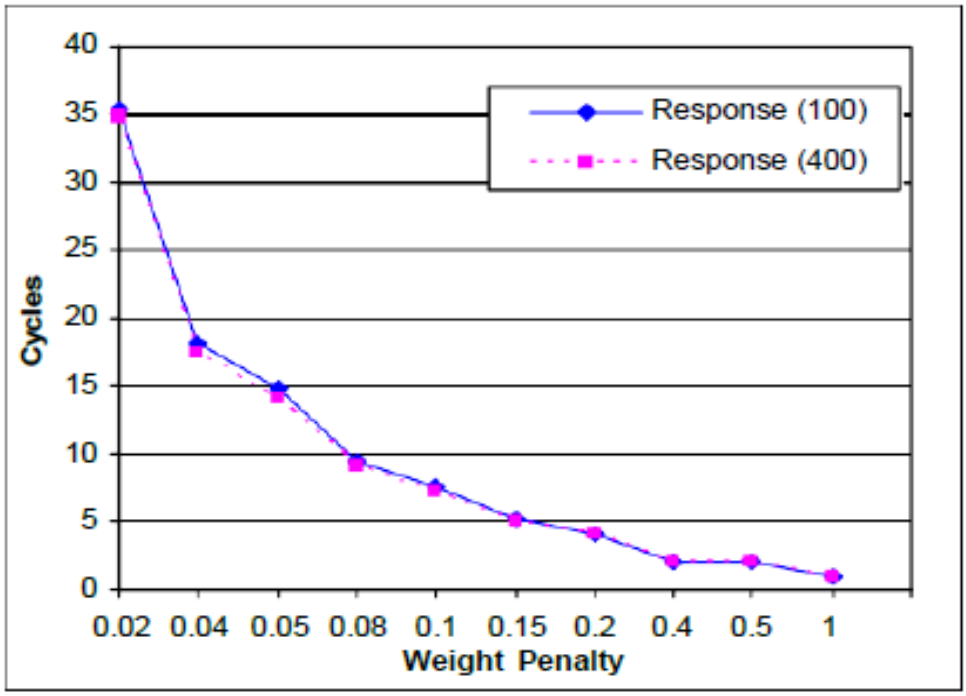
Furthermore, for the 100 node and 400 node scenarios, the performance is assessed with a weight penalty of 0.05 for various attack probabilities. The probability of an attack is determined by dividing malicious nodes by the overall number of sensor nodes in the network that might be compromised. This indicates the amount of fake data injected into the network by the attacker.

According to the findings of the Byzantine General Problem investigation (Ozdemir, 2007), when the malicious node numbers exceed genuine ones, loyalty generals are unable to determine who is the rebel. Moreover, if no authentication system is used, the number of rebel generals has to be fewer than 1/3 of the overall number of generals for the loyal generals to agree on the right action.

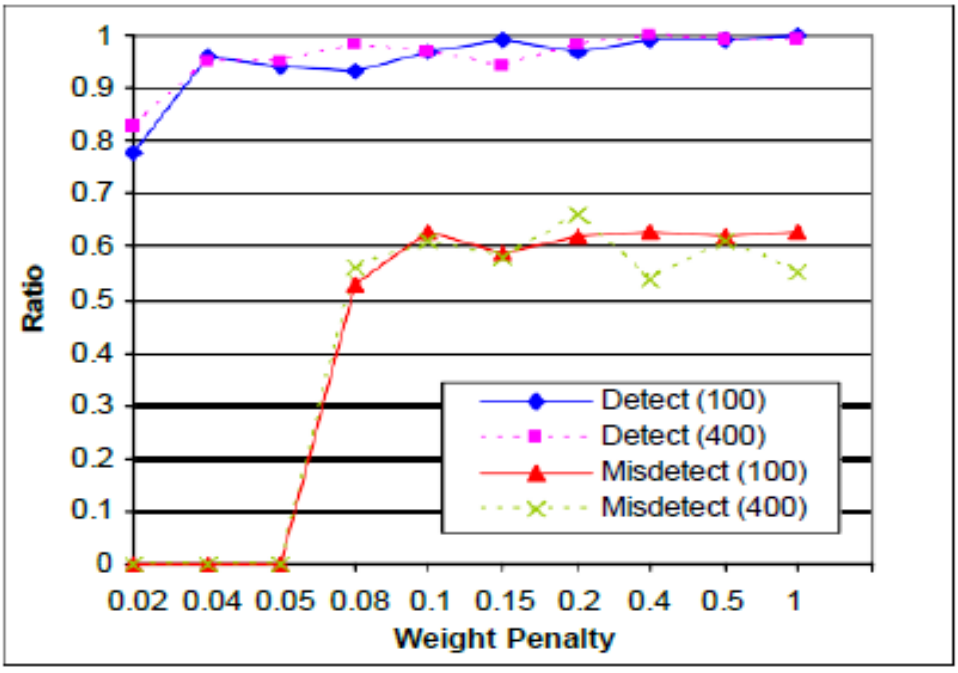
Similarly, in this situation, when the number of malicious nodes exceeds 25% of total nodes, Experts will be unable to detect the "bad guys" with certainty. The upper bound for the number of compromised nodes in our simulation was 30% of the total number of nodes. As a result, an attack probability of one means that 25% of the sensor nodes are compromised.

The response time increases significantly as the attack probability increases, as seen in Fig. 4(a). As more malicious nodes arrive, it appears that the collected data can be more influenced by incorrect facts. Though detection ratios exhibit relatively minor changes, when attack probability increases, the misdetection ratio decreases dramatically, as seen in Fig. 4(b). This would lead to a little increase among malicious nodes, lowering the false positive rate.

In the presence of various nodes with high compromise probability, the reaction time, detection and misdetection ratios remain constant, as per the results reported above. It shows that the proposed detection approach is successful across both big networks and situations with a high attack potential. These experimental results reveal that the previously mentioned factors have a substantial influence on the detection algorithm's performance.

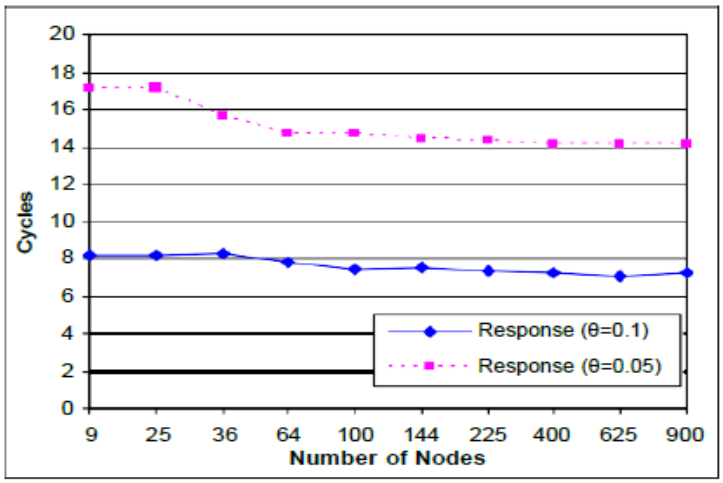


(a)

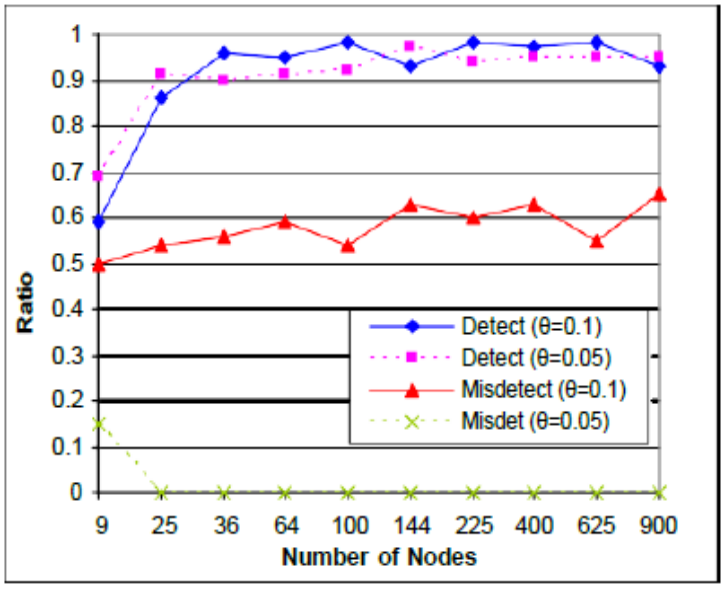


(b)

Fig. 2: The effect of different penalty (a) response time Vs. penalty weights; (b) detection accuracy Vs. penalty weights

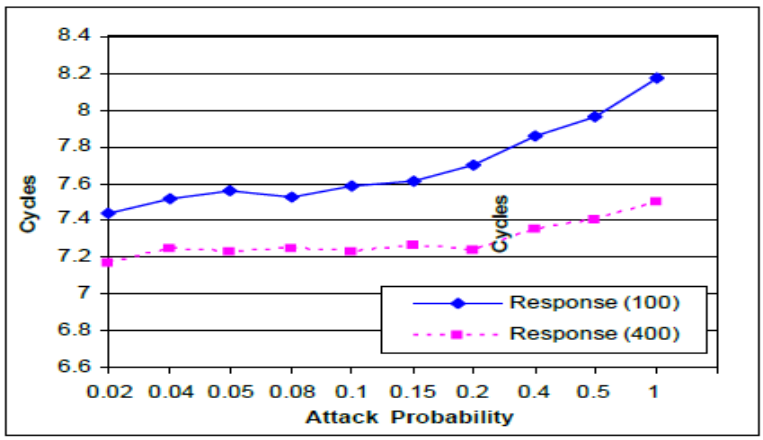


(a)

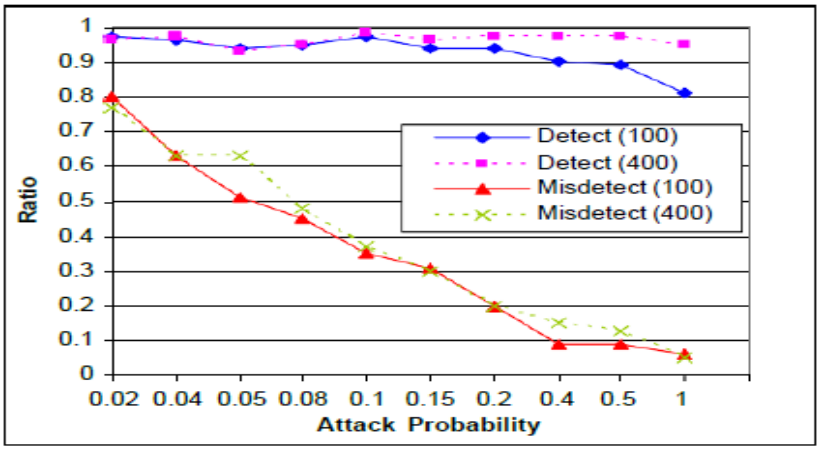


(b)

Fig. 3: Illustration of the system scalability (a) response time Vs. the number of nodes; (b) detection accuracy Vs. number of nodes



(a)



(b)

Fig. 4: Illustration of the attack probability (a) response time Vs. compromise probability (b) detection accuracy Vs. comprom is probability

Conclusion

We introduced a new weighted-Credence evaluation-based approach to identify corrupted or misbehaving nodes across wireless sensor networks. The fundamental notion is that FNs provide Credence worthiness to every cluster node, only if the node provides completely irrelevant information, implying that the node is being compromised or is still no longer functioning; the FN decreases that node's Credence level. This will be easier and less difficult to keep track of nodes, so compromising the majority of the nodes should be much more difficult whenever the base stations are compromised. Our technique has excellent scalability and can be used in both small and big-sized WSNs. Only one change when applying it to larger WSNs is that the number of FNs is increased. Essentially, it's a node-clustering problem. Our technique is reliant on the notion of base stations can be depended on properly. However, when an intruder gains control over base stations, he or she is free to attack the WSN in either way they find appropriate; however, it is outside the scope of the study. An important assumption is the vast majority of sensor nodes are operational. Legal nodes will be recognized as malicious and separated if there are more compromised nodes than regular nodes. In this study, we just provided preliminary data that confirmed the validity and efficiency of our method. An additional extensive analysis of the system's performance could be investigated as the research progresses and additional questions will be answered.

Author's Contributions

All authors equally contributed in this study.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Bidai, Z., Haffaf, H., & Maimour, M. (2011, April). Node disjoint multi-path routing for ZigBee cluster-tree wireless sensor networks. In 2011 International Conference on Multimedia Computing and Systems (pp. 1-6). IEEE. https://ieeexplore.ieee.org/abstract/document/5945672

Cam, H., & Ozdemir, S. (2007). False data detection and secure data aggregation in wireless sensor networks. Security in Distributed, Grid, Mobile, and Pervasive Computing, 129-157.

Cao, G., & Yu, F. (2011, August). The analysis of load balance for wireless sensor network using compressive sensing. In 2011 14th IEEE International Conference on Computational Science and Engineering (pp. 100-105). IEEE. https://ieeexplore.ieee.org/abstract/document/6062859

Choudhari, E., Bodhe, K. D., & Mundada, S. M. (2017, February). Secure data aggregation in WSN using iterative filtering algorithm. In 2017 International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) (pp. 1-5). IEEE. https://ieeexplore.ieee.org/abstract/document/7975603

Hu, C., & Li, X. (2011, September). A clustering algorithm based on geography region for WSN. In 2011 International Conference on Electrical and Control Engineering (pp. 480-483). IEEE. https://ieeexplore.ieee.org/abstract/document/6057631

Ozdemir, S. (2007, November). Secure and reliable data aggregation for wireless sensor networks. In International symposium on ubiquitious computing systems (pp. 102-109). Springer, Berlin, Heidelberg. https://link.springer.com/chapter/10.1007/978-3-540-76772-5_8

Raha, A., Babu, S. S., Naskar, M. K., Alfandi, O., & Hogrefe, D. (2011, December). Trust integrated link state routing protocol for Wireless Sensor Networks (TILSRP). In 2011 Fifth IEEE International Conference on Advanced Telecommunication Systems and Networks (ANTS) (pp. 1-6). IEEE. https://ieeexplore.ieee.org/abstract/document/6163648

Sun, B., Jin, X., Wu, K., & Xiao, Y. (2007, June). Integration of secure in-network aggregation and system monitoring for wireless sensor networks. In 2007 IEEE International Conference on Communications (pp. 1466-1471). IEEE. https://ieeexplore.ieee.org/abstract/document/4288917

Wu, K., Dreef, D., Sun, B., & Xiao, Y. (2007). Secure data aggregation without persistent cryptographic operations in wireless sensor networks. Ad Hoc Networks, 5(1), 100-111. doi.org/10.1016/j.adhoc.2006.05.009