Improved Sensed Data Dependability and Integrity in Wireless Sensor Networks

Zibouda Aliouat and Makhlouf Aliouat(✉)

Networks and Distributed Systems Laboratory, Computer Science Department,
Faculty of Sciences, University Ferhat Abbas Sétif 1, Sétif, Algeria
{zaliouat,maliouat}@univ-setif.dz

Abstract. The mainspring of a Wireless Sensor Network (WSN) is to capture data from an interesting deployment area and sending them out to the end user. However, one can ask to what extent the end user can have confidence in the use of these data? Especially when these collected data are employed in a crucial application that definitely excludes wrong data use. Thereby, WSN mission success is basically dependent on trustworthy of the data delivery process to the end user. To reach this goal, some obstacles, related to malicious node behavior or failure nodes, must be avoided or tolerated. Therefore, we propose, in this paper, a scheme to improve the dependability of the data sensed by sensor nodes in one hand and the reliable communication of these data to the sink in other hand. The proposal is based on a fault tolerant sensing process and the resilience to malware threat on the transmitted data from nodes to sink. The proposal was integrated to the well know Leach protocol and the performance evaluation, carried out on NS2 simulator, showed convincing results in terms of energy conserving, received data rate and node failure occurrence and attack detections.

Keywords: WSN · Dependable data sensing · Trusted data aggregation
Secure data transmission

1 Introduction

Dependable data sensing is of paramount importance in WSNs. Indeed, what should be the consequences when wrong data are used, particularly in deployed critical applications? To make suitable decision from application outcomes, the end users have to rely on consistent information gathered by sensor nodes from the area of interest. Otherwise, not only the application outcomes should be erroneous, but they could lead to a disaster if they should be used to generate a final critical decision.

Data should be correct since the time they are captured until the time they should be used by the WSN controller. To this end, data must be free from alterations purposely induced by an adversary or due to a sensor failed component or resulting from the unpredictable wireless environment.

Prospective wrong data received by an end user, after data aggregation process [1], may have been corrupted from diverse ways such as: a malfunctioning of node sensing unit, any impairment during transmission or alteration of data integrity caused by malware or intruder at any node acting as router located along the routing path. This is

A. Amine et al. (Eds.): CIIA 2018, IFIP AICT 522, pp. 460–471, 2018.
https://doi.org/10.1007/978-3-319-89743-1_40

for what is devoted the work reported in this paper; more precisely, coping with the tricky issue of node fault tolerance in case of unit sensing failure and security issue related to data integrity in case of network attack by intruders. The two problems have been treated separately in the specialized literature; they should be conjointly considered in order to reach end user confidence requirements in consuming collected data. We point out that node failures concern any flaws in correct data forwarding but more precisely incorrect data sensing generated by lack of energy or fault occurrence in sensing unit.

In the sequel, the paper is organized as follows: in Sect. 2, we give a related work dealt with the data aggregation issue; Sect. 3 presents our proposals for dependable data sensing and aggregation and performance evaluation of our proposals. We conclude the paper by a conclusion and future work.

2 Related Works

Several approaches have been proposed for reliably aggregating data in WSNs or securing them. These approaches coped only with routing problems like adversary attacks or node malfunctioning to forward data they have in charge.

De Cristofaro et al. [2] have proposed FAIR (Fuzzy-based Aggregation providing In-network Resilience for real-time Wireless Sensor Networks) an algorithm for robust data aggregation in real time. Witness nodes are used to confirm the outcomes of the aggregation process. The protocol seems to be robust but is only suitable for small WSNs and suffers from security lack and time overhead.

Wang et al. [3] have proposed EESSDA (Energy-Efficient and Scalable Aggregation Secure Data) a protocol using secure channel data encryption scheme. EESSDA uses several steps: creation of an aggregation tree rooted by the sink, creation of a secure channel between children and parents sharing a common key. Any node waits a certain time to receive data from its children, aggregates them with its own data and sends the new result to its parent. The protocol is costly and suffers from lack of security, because a compromised node near BS may jeopardize the aggregates confidentiality.

Shivakumar et al. [4] proposed ERDRA (Efficient and Reliable Data Routing for In-Network Aggregation in Wireless Sensor Networks) a protocol building a routing tree with the shortest path connecting all nodes for reliable data aggregation. The tree is used by coordinators, elected among nodes having detected a new event, to collect data and sending them along a reliable new path. Weaknesses: Compromising a channel involves to intercept all messages, time overhead.

In [5] Jose et al. have proposed a data aggregation protocol, ensuring confidentiality, authentication and freshness, suitable for the critical time applications. The message authentication is obtained via a key pair and a secret identification of each node. The protocol uses a tree aggregation including terminal nodes, parent nodes and sink. The protocol suffers from possible internal attacks, CPU overload and an unsafe key management mechanism.

Zhu et al. [6] proposed ECIPAP (Efficient Confidentiality and Integrity Preserving Aggregation Protocol), an effective protocol ensuring confidentiality and aggregated data integrity. A node getting sensed data sends them to its parent with a Mac. Each parent node aggregates the received value with its own and sends the result to the upper level until reaching the sink. The latter decrypts the received message and broadcasts the aggregated data allowing each node to verify if its own data have been added. Protocol weaknesses: Possible internal attacks and time overhead.

Lathamanju et al. [7] proposed a secure data aggregation algorithm improving network life time and ensuring safety and security via Diffie-Hellman algorithm. Nodes are classified as friends or malicious. When a node wants to send data, it initiates a request message to route. Each intermediate node forwards the request if the sender is not malicious. When the destination node receives the route request, it sends an ACK response and its public key. The source node evaluates the best route with the largest number of friend nodes to send data. Protocol Weaknesses: Time overhead and resources waste.

BabuKaruppiah et al. [8] have proposed NADSPSD (A Novel Approach to Detect the Shortest Path for Secure Data Aggregation using Fuzzy Logic in WSNs) an effective technique to detect the shortest path using fuzzy logic to secure the data aggregation. It is based on trust and residual energy of a node. The selection of the best route to send aggregated data is based on the combination of the path length, the available energy level and the node reputation. Protocol weakness: Cost and lack of security.

In [9], Jia et al. have proposed MDRN (Minimum Distance Redundant Nodes) a fault recovery protocol which is deployed on the receiver node that has knowledge of node locations and failed ones. By choosing an appropriate number of redundant nodes, the algorithm will provide an accurate recovery. The proposal is feasible and effective to deal with the coverage hole issue caused by failed node. The protocol is unable to deal with multiple faults and requires a significant nodes redundancy.

3 Proposed Protocols

WSNs are typically deployed in harsh unattended environment. The inherent node vulnerabilities may lead to an unauthorized modification of sensed data. To overcome this problem, we present a solution offering safe data aggregation and sensing fault tolerance capability.

3.1 Description of the First Proposal

The different proposals work with the following hypothesis: - A WSN with N sensor nodes organized in clusters with a Cluster Head (CH) as manager and aggregator for each cluster. Each node i has a unique identifier Idi. The Base Station (BS) is assumed to be robust and reliable, with inexhaustible resources. - A compromised node may send corrupted data to the BS.

Notations: Emin: minimal energy amount, Round: period for changing aggregators. Frame: temporal interval for data sensing, Vmax: maximal value, R: node communication range.

The first proposal named LEACH-FD (Low Energy Adaptive Clustering Hierarchy Fault Discovering) is a version improving the well-known LEACH Protocol [10] by including a filtering to eliminate incorrect data before their aggregation.

Proposed Protocol Steps: The algorithm takes place in rounds having approximately the same interval of time determined in advance. Each round consists of an initialization phase and a transmission phase. The general proposed algorithm is the following:

- Selecting of ClusterHeads (CHs) as done in LEACH protocol.
- Request dissemination by CHs to all nodes.
- A node may join a CH by sending it a join request.
- Formation of TDMA-schedule by CHs and Phase of data transmission to the CHs.
- Filtering received data and aggregation and sending them to the base station.

The different phases are described hereafter:

Initialization Phase: This leads to the clusters formation by electing a Cluster-Head (CH) for each cluster and establishing nodes channel access strategy within each cluster. This phase begins by local decision making to become cluster-head. Each node *Ni* chooses a random number *rn*, if $rn \leq$ t, the node *Ni* becomes cluster-Head as in [9]. When elected, a CH must inform its neighboring nodes of its new rank. The cluster member nodes managed by a CH are those nodes having joined CH according to the signal strength of the rank notification message sent out by CH. Each member node has to inform a CH of its decision to be membership. After that, the communications within a cluster can be made according to the TDMA communication protocol. For this, each CH establishes a TDMA schedule for its members.

Transmission Phase: Data transmission operation is repeated at each frame, where nodes send data to their respective CH once a frame during their own allocated slots. Outside their slots, nodes get into sleeping mode for conserving their energy [11].

Aggregation Step: We added an important preliminary step to the aggregation operation, in which, after receiving data sensed by all nodes, each aggregator has to locally perform a filtering of received data to eliminate potential erroneous sensed data before the final aggregation process. A suitable detection algorithm of outliers should detect most of the errors and the number of false positive must be as small as possible. It uses the median, which is classified statistically among the robust features for detecting outliers [12].

Simulation Results: For the sake of limited space, the details of simulation are given only for the third proposal. Therefore, the conducted simulations showed that the proposal works only if each cluster includes more than two sensors nodes in order to be able to compare the values sent out from a cluster. Ignoring the incorrect values is not enough, but we must determine the causes in order to avoid using them in the future. When a node *Ni* sends an abnormal value compared to other nodes *Nj*, $j \neq i$, this does not necessarily imply that *Ni* is failed. We can find a case where an event is triggered at a node level but other nodes have not yet discovered it. Therefore, we must add a method to ensure that the value sent is correct without waiting for the next frame since in the critical systems, delays in event discovery are crucial.

3.2 Description of the Second Proposal

We present a new protocol FDP (Fault Discovering Protocol), that uses neither the centralized approach nor the only distributed one but it combines the two. That is, the aggregators selection is done by the BS to ensure a global view of the network and a good energy management, but each node chooses its aggregator alone to reduce the load on the BS. This proposal is trust based where nodes history is used to decide if a node is faulty or not. The BS maintains a table of confidence in which each node has a certain degree of confidence. This value may change if sensed data by a node are not correct. It always uses sensed data filtering at the aggregators' level, but it adds a procedure to determine if a sensor is really faulty.

Proposed Algorithm: This algorithm is also carried out in rounds. Each round consists of an initialization phase, transmission phase and verification phase see Fig. 1.

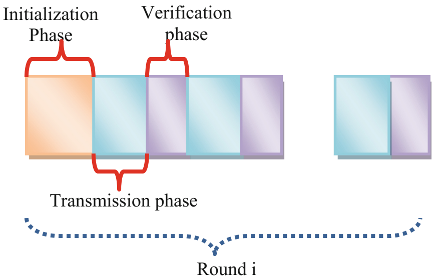

Fig. 1. 2nd proposal phases.

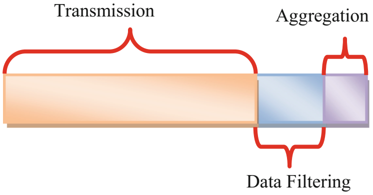

Fig. 2. Transmission phase.

Initialization Phase: This phase is composed of 3 sub-phases: aggregators selection, clusters formation and scheduling. At the beginning of each round, each node checks its residual energy if it is greater than a threshold value *Emin,* then it sends to BS a message containing its location X, Y and its energy level E.

- Aggregators Selection: In this phase, BS will retrieve all data sent by nodes and then will choose nodes with maximum value of confidence as aggregators. If there are two nodes with same value, the node with maximum energy will be chosen. Also, in case of equality, the node with the smallest index is selected. After that, BS will broadcast a message including the CH identities and their coordinates.
- Clusters formation: Upon reception of the information message sent by the BS, if the receiver node is a CH, it will do nothing, otherwise (standard node) it chooses from the list of the aggregator nodes the node closest to it as its leader. The node must inform its leader by sending it a JOIN message (belonging) containing the identity of the member node.

An Aggregator receiving JOIN message adds the sender to a list containing all member nodes. After that, any aggregator builds a TDMA-schedule and sends it to its members. Each one will know as well, its position in this TDMA-schedule and the data transmission time slot to its aggregator. The TDMA-schedule allows avoiding collisions between nodes which minimizes nodes energy consumption. This energy savings is also enhanced by node going to sleep out of its own time slot.

Transmission Phase: This phase also is decomposed in frames whose size is larger than that of the frame of the first proposal, because each frame contains 3 sub-phases: transmission, data filtering and accepted data aggregation (Fig. 2).

- Data Transmission: It is decomposed into several time slots, a slot in a frame is dedicated to a single node during which it transmits its data to the aggregator. Each node sends a message containing the sensed value and its residual energy.
- Aggregation Phase: When the aggregator receives data from a node *Ni*, it checks if the energy of *Ni* is less than *Emin*, if so, it will ignore the received value otherwise it will store it in a list. At the end of each frame, the aggregator will count the number of values in its list, if this number is greater than 3 then it will filter the data using the procedure used in the first proposal but this time when it finds an abnormal value it will send an alert message to the BS containing the node identity that has captured the incorrect value.

Verification Phase: It contains two sub-phases: data verification, data confirmation.

- Data Verification: This sub phase is carried out when the BS receives an alert message containing the ID of a node; it will send a request for that node to re-capture data in its proximity according to radius R.
- Confirmation phase: When a node receives a confirmation message, it will sense again and return the value captured for the second time. When the BS receives a confirmation message, it will compare the data received and if there is a difference between the data of the nodes, so the node is considered faulty. BS will decrement trust value of a node having sent incorrect values, if this value reaches 3 then that node is declared faulty. These operations are repeated until the round end.

The initialization phase is repeated at each round up to the simulation end caused either by the exhaustion of the simulation time set to 3600 s or by the condition “the number of remaining nodes in WSN is less than the number of aggregators” but the confirmation phase is made only in case of a problem.

Simulation Results: The simulation results show that we cannot surely conclude if the sensor node that sent incorrect values is failed or compromised. Also the FDP protocol does not cover the case of failed aggregator.

3.3 Description of the Third Proposal

This proposal FDAP (Fault Discovering and Attack Protocol) is to enhance the second proposal to which is added an encryption mechanism for node authentication.

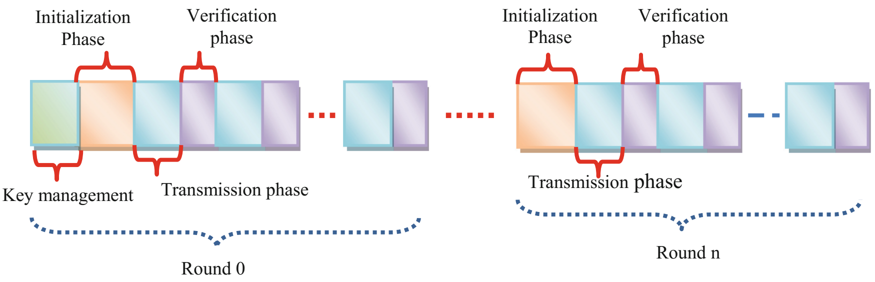

Fig. 3. Phases of the 3rd proposition.

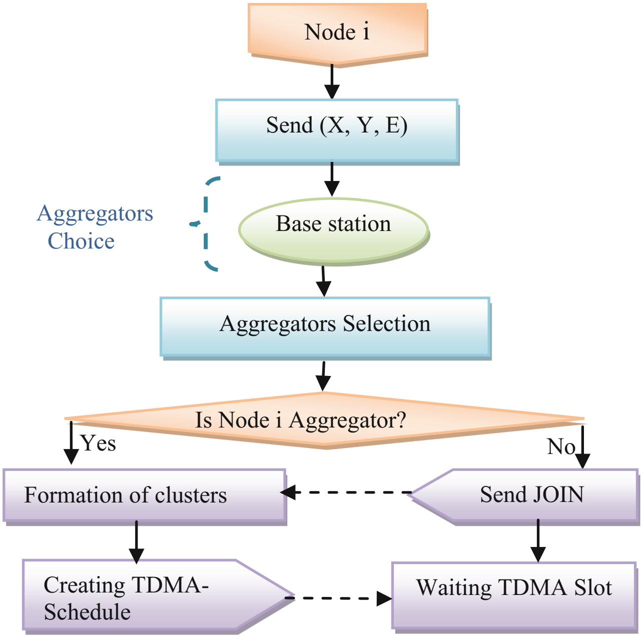

Fig. 4. Initialization phase flowchart.

Proposed Algorithm: Each round consists of an initialization phase (Fig. 4), a transmission phase (Fig. 5) and a confirmation phase (Fig. 6); this is before the phase of key management (Fig. 3). The main phases of the third proposal are as follows:

Key Management and Authentication Nodes: Key management is carried out a time the deployment is achieved and before nodes are operating as sensors. So, BS assigns to each node *Ni* a unique identifier *IDi* and a symmetrical key that BS shares with the node Ni and a large integer value M is also preset, where key > M. The other phases remain the same until the confirmation phase of detecting erroneous data. The verification phase remains the same except that it is added a technique of cryptography to ensure the nodes authentication:

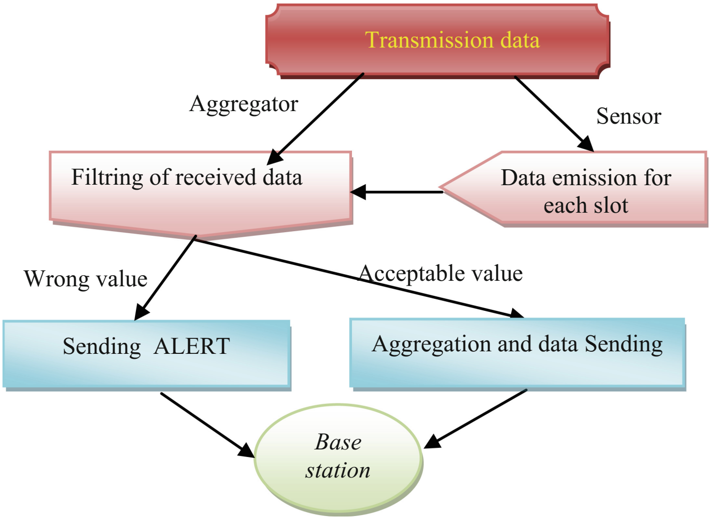

Fig. 5. Transmission phase flowchart.

- *Detection of compromised node:* When BS receives an alert message containing the ID of a node; it will send it a message containing a random value S asking it to encrypt the value using its private key. The BS will also send to the neighbors of this node a confirmation message in the same way as in the previous proposal. A node *Nj* which receives a confirmation message of its identity, uses the following expression to encrypt the received value: C = ENC (value received) = value received + key(Nj) MOD M. Then it sends a confirmation message containing the value C. When the base station receives this message, it will decrypt it using this expression: d = Dec(C) = C key(Nj) MOD M. Finally BS compares D and S if they are equal then the authentication is ensured otherwise the node Nj has been compromised.
- *Detection of faulty aggregator:* The base station can receive final erroneous aggregate values due to a faulty aggregator, so we add a procedure to cover this case. The BS will check the values received from the aggregators if it finds a value superior to *Vmax*, it will trigger the verification procedure. But now, it does not send to the nodes in the vicinity of the aggregator but to all members of its cluster. BS will then compare the new received values with those of the aggregator, if there is a large difference, BS will decrement confidence value of the aggregator. BS will then play the role of the aggregator during the remaining time of the current round.

3.4 Proposal Performance Evaluation

The performance evaluation of our final proposal is carried out through the well known simulator NS2 according to the network parameters summarized in the Table 1 hereafter. The metrics of performance evaluation used are detected anomalies, life time duration, energy consumed and the data amount received by the base station. The three

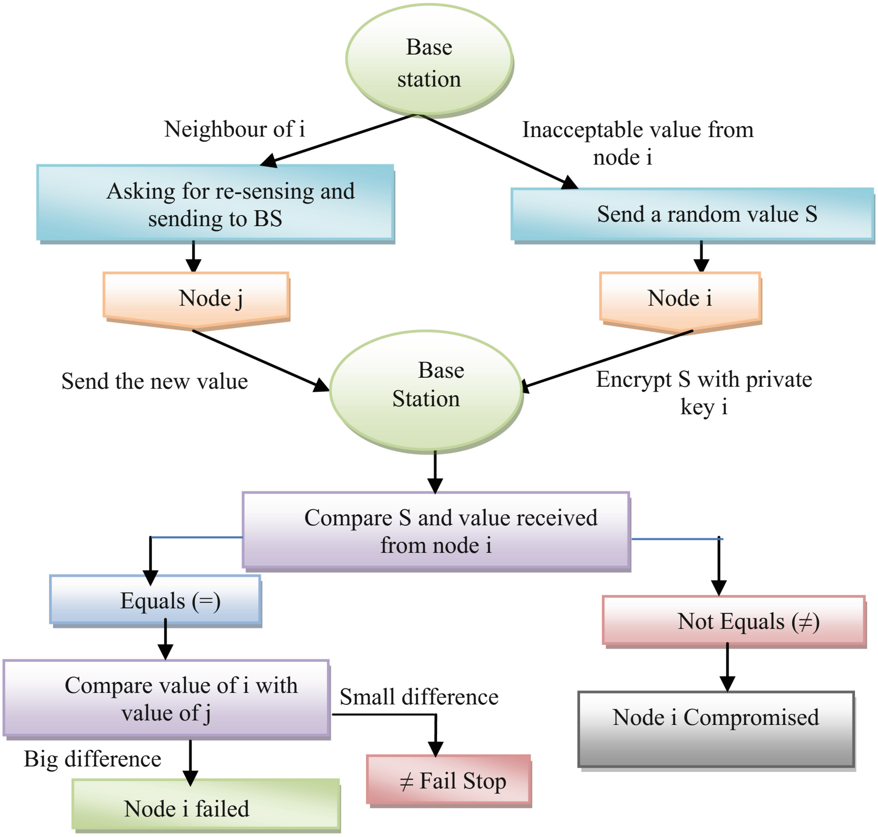

Fig. 6. Verification phase flowchart

Table 1. Parameters used in the simulation process

Parameters	Values	Parameters	Values
Size of the network	1000 m * 1000 m	Round time duration	30 s
Base station position	50, 175	Initial energy amount	2 J
Clusters number	5	Initial trust value	6
Sensor nodes number	100	Frame	3 s
Simulation time	3600 s	Round	30 s
		Radius R	10 m

algorithms are used to simulate the same scenario designed to sense the temperature in a forest and monitor fires (The scenario contains a failed node, an intruder and a fire). Table 2 shows the anomalies recorded by sensor nodes according to the three proposals. As a result, the first proposal detects no anomalies while the 2nd proposal detects only failures and fires but the 3rd proposal detects all the assumed anomalies (failures, fires and attacks).

Table 2. Anomalies detected during simulation operation

Proposals	Failure	Attack	Fire
1st Proposal	No	No	No
2nd Proposal	Yes	No	Yes
3rd Proposal	Yes	Yes	Yes

Comparison Between Different Proposal Versions: Figure 7 showed the energy consumed as a function of simulation time in different proposals. We note a great improvement obtained by FDP and FDAP regarding to Leach-FD version. Therefore, the strategy of aggregator's selection we proposed gives lower energy consumption which is vital for WSNs to last longer until achieving their missions. Also, we note that the use of cryptography in the third protocol does not consume a lot of energy. Figure 8 shows the network life time duration ensured par different proposals. It represents the number of survival nodes during the entire simulation. From the curves, we note that FDP and FDAP allow ensuring a survival of all nodes until point of time 320 and 350, and maintain a functionality of all nodes during this period and prolong the life time of the WSN, while the life time duration of nodes in improved leach-FD is limited to time 200 only. We justify this difference by the new technique of selecting the aggregators used by FDP and FDAP.

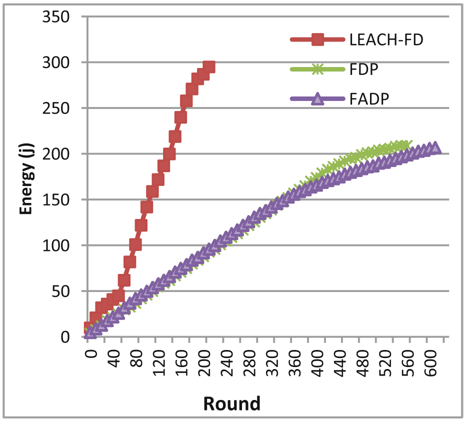

Fig. 7. Energy consumed by nodes.

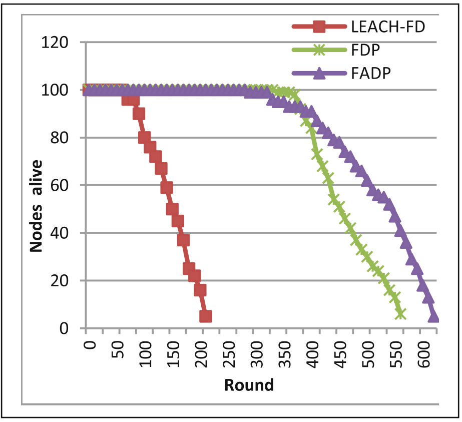

Fig. 8. Alive nodes number function of time.

Figure 9 represents the amount of data received by BS, which is expressed as a function of the number of messages sent by nodes to BS. The curves show that FDP and FDAP, ensures data transmission up to time 600 with a number of messages equals 11800. On the other hand, the transmission of data in leach-FD stops at time 210 with almost 17500. Therefore, no remarkable improvement regarding the amount of data received by BS is noted. The cause of data reduction is the elimination of data of the faulty sensors and the nodes that have a minimum energy because we are interested in the quality of the data more than the quantity.

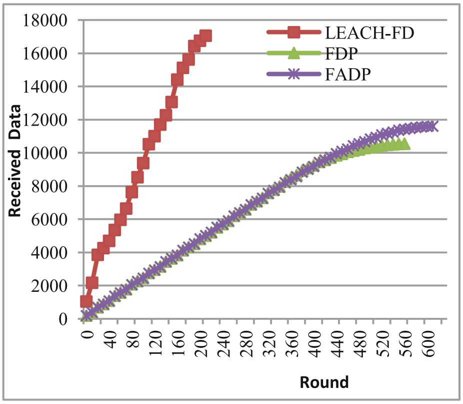

Fig. 9. Data amount received by BS.

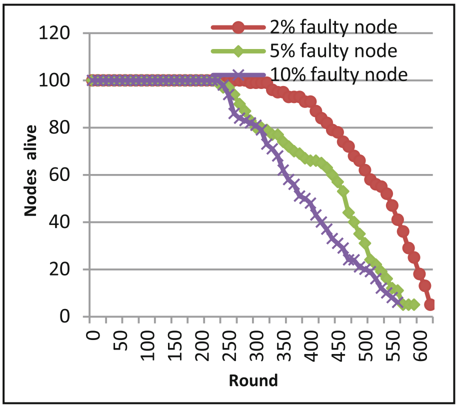

Fig. 10. Energy consumed in FDAP.

Proposals Behavior in Presence of Faults: The results obtained after variation of the number of failed nodes are illustrated by the Figs. 10, 11 and 12 with 2%, 5% and 10%. of failed nodes. Figure 10 represents the energy consumption as a function of simulation time in FDAP in the three cases: 2%, 5% and 10% of faulty nodes. The curves show a difference in energy consumption between the three cases. When a node fails, the number of messages, during the verification phase, increases leading to the increase of energy dissipation. Figure 11 represents the life time duration of WSN in FDAP. We note a reduction in the WSN life time each time the number of failed nodes has increased. Figure 12 represents the data received by BS in FDAP in the three different cases. The results showed that the number of failed nodes impacts negatively the performance of the protocol where we note a big difference in the amount of data received by BS because of the elimination of incorrect values.

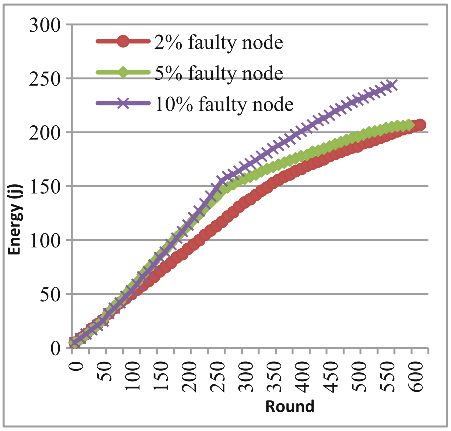

Fig. 11. Nodes life time duration in FDAP.

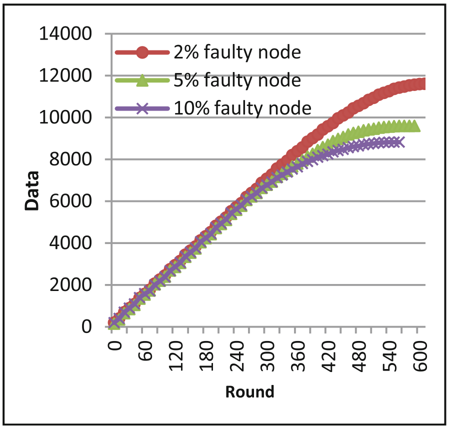

Fig. 12. Data amount received by BS in FDAP.

4 Conclusion

Since the decision taken by the end user of a WSN greatly depends on the data sensed by nodes from an area of interest, these data which have undergone an aggregation process, must be as dependable as possible to ensure the success of the network mission. That is why our protocol is designed to transmit only reliable data from healthy nodes avoiding so those that are faulty or compromised by intruders. The results of simulation carried out via the well known Network Simulator NS2 shown that our third proposal is able to detect all the considered anomalies like failure node or node attack. The convincing results have been obtained in an efficient way in terms of energy consumption and network life time duration. To be complete, we intend to include in our protocol the robustness of the sink to cope with both fault occurrences and attacks.

References

1. Ramar, C., Rubasoundar, K.: A survey on data aggregation techniques in WSNs. Int. J. Mob. Netw. Des. Innov. **6**(2), 81–91 (2015)
2. De Cristofaro, E., Bohli, J.M., Westhoff, D.: FAIR: fuzzy-based aggregation providing in-network resilience for real-time wireless sensor networks. In: Proceedings of ACM WiSec 2009, Zurich, Switzerland, 16–18 March 2009 (2009)
3. Wang, T., Qin, X., Liu, L.: Energy-efficient and scalable aggregation secure data a secure protocol. In: Distributed Sensor Networks, January 2013
4. Shivakumar, R., Jagadeesha, J., Babu, A., Rashmi, K.R.: An efficient and reliable data routing for in-network aggregation in wireless sensor network. Int. J. Emerg. Technol. Eng. **2**(4) (2015)
5. Jose, J., Princy, M., Jose, J.: PEPPDA: power efficient privacy preserving data aggregation for wireless sensor networks. In: ICE-CCN, March 2013
6. Zhu, L., Yang, Z., Xue, J., Guo, C.: An efficient confidentiality and integrity preserving aggregation protocol in wireless sensor networks. Int. J. Distrib. Sens. Netw. **10**(2), 1–8 (2014)
7. Lathamanju, R., Senthilkumar, P.: CRSR algorithm: a secure data aggregation algorithm in WSN. Int. J. Adv. Res. Electron. Commun. Eng. (IJARECE), **2**(9), 781–789 (2013)
8. BabuKaruppiah, A., Kannadhasan, S.: A novel approach to detect the shortest path for secure data aggregation using fuzzy logic in WSNs. Int. J. Eng. Comput. Sci. **2**(2), 506–510 (2013)
9. Jia, S., Bailing, W., Xiyuan, P., Jianfeng, L., Cheng, Z.: A recovery algorithm based on minimum distance redundant nodes in fault management in WSNs. Int. J. Control Autom. **6**(2), 175–184 (2013)
10. Heinzelman, W., Chandrakasan, A., Balakrishnan, H.: Energy-efficient communication protocol for wireless micro sensor networks. In: Proceedings of the 33rd Hawaii International Conference on System Sciences (2000)
11. Aliouat, Z., Aliouat, M.: Improved WSN capabilities through efficient duty-cycle mechanism. In: Yao, L., Xie, X., Zhang, Q., Yang, L.T., Zomaya, A.Y., Jin, H. (eds.) APSCC 2015. LNCS, vol. 9464, pp. 268–277. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-26979-5_20
12. Titouna, C., Aliouat, M., Gueroui, M.: FDS: fault detection scheme for wireless sensor networks. Wirel. Person. Commun. **86**(2), 549–562 (2016)