Energy Efficiency Enhancement of TICK –based Fuzzy Logic for Selecting Forwarding Nodes in WSNs

Muhammad Ashraf[1] and Cho Tae Ho[2*]
[1]School of Information and Communication Engineering, Sungkyunkwan University,
Suwon 16419–Republic of Korea
[e-mail: ashraf84@skku.edu]
[2]College of Software, Sungkyunkwan University,
Suwon 16419–Republic of Korea
[e-mail: thcho@skku.edu]
*Corresponding author: Cho Tae Ho

*Received June 11, 2017; revised January 31, 2018; accepted May 2, 2018;
published September 30, 2018*

Abstract

Communication cost is the most important factor in Wireless Sensor Networks (WSNs), as exchanging control keying messages consumes a large amount of energy from the constituent sensor nodes. Time-based Dynamic Keying and En-Route Filtering (TICK) can reduce the communication costs by utilizing local time values of the en-route nodes to generate one-time dynamic keys that are used to encrypt reports in a manner that further avoids the regular keying or re-keying of messages. Although TICK is more energy efficient, it employs no re-encryption operation strategy that cannot determine whether a healthy report might be considered as malicious if the clock drift between the source node and the forwarding node is too large. Secure SOurce-BAsed Loose Synchronization (SOBAS) employs a selective encryption en-route in which fixed nodes are selected to re-encrypt the data. Therefore, the selection of encryption nodes is non-adaptive, and the dynamic network conditions (i.e., The residual energy of en-route nodes, hop count, and false positive rate) are also not focused in SOBAS. We propose an energy efficient selection of re-encryption nodes based on fuzzy logic. Simulation results indicate that the proposed method achieves better energy conservation at the en-route nodes along the path when compared to TICK and SOBAS.

Keywords: Wireless sensor networks, network security, energy, en-route filtering, fuzzy

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. NRF-2015R1D1A1A01059484)

http://doi.org/10.3837/tiis.2018.09.009 **ISSN : 1976-7277**

1. Introduction

Modern technological advancements have made the deployment of small-scale, low-cost and low-power distributed devices a reality. These devices are referred to as sensor nodes. Each sensor node has constrained data processing capability, restricted storage, low-power resources, and a small communication radius. Still, sensor nodes have the potential to thoroughly monitor a given physical environment. A group of such sensor nodes can coordinate with each other and perform specific functions. A collection of sensor nodes is known as a sensor network, which can be used in various applications, including health, aerospace, transportation vehicles, intelligent highways, and smart spaces [1, 2]. Since sensor nodes are left unattended and are deployed in hostile environments without any infrastructure, adversaries can easily compromise these nodes [3-6] and inject malicious data to disrupt the network. In addition, the sensor node carries limited and generally irreplaceable power sources [7]. Therefore, providing good energy efficiency and resilience against false injected data are of the utmost importance.

En-route-filtering schemes [8-14] are used to eliminate false injected information from wireless sensor networks. These schemes perform well at securing the network by filtering false injected reports, but may not necessarily increase the energy efficiency of the network. These schemes exchange keys to refresh or redistribute keys in the network [15], resulting in high communication costs. Time-based Dynamic Keying and En-Route Filtering (TICK)[16] was introduced to address these issues and minimize the communication cost. TICK achieves high energy savings by eliminating the exchange of control messages regarding keying or rekeying and further utilizes a small portion of these energy savings in the computation of local security services. Secure SOurce-BAsed Loose Synchronization (SOBAS) [17] exploits the TICK framework and provides re-encryption for forwarded reports at multiple preselected intermediate nodes that are at equal distances from one another.

We propose a fuzzy rule-based selection of intermediate encryption nodes. The proposed scheme uses fuzzy parameters such as the remaining energy level (REL), hop count (HC), and false positive rate (FPR) that improves the proximity of the encryption nodes with respect of one another in the presence of high false positive report insertion rate. In the presence of high FPR, the transmission energy can be saved by choosing encryption nodes closer to one another to filter false report earlier. In the absence of or low FPR, the computational energy can be saved by choosing lesser and relatively farther encryption nodes from one another. Whereas, the absence of an adaptive encryption node selection strategy, their number and distance remain fixed irrespective of the false positive rate and energy status of the intermediate nodes.

The proposed method presents the following contributions:

- Reduced false positive reports.
- Improved energy efficiency of the network through the selection of a suitable number of re-encryption nodes.
- Increased network lifetime.

The rest of this paper is organized as follows. Section 2 presents related works. A background overview of the TICK scheme and the motivation for the proposed method is provided in Section 3. Section 4 provides a comprehensive description of the proposed method. A performance evaluation of the proposed method is discussed in Section 5 and conclusions are presented in Section 6.

2.Related Work

Many en-route filtering schemes have been developed that are used to filter malicious data from WSNs,including statistical En-route Filtering (SEF) [8],Commutative Cipher based En-route Filtering (CCEF)[9],An Interleaved Hop-by-Hop Authentication (IHA)[10],Secure Ticket-Based En-route Filtering (STEF) [11], Probabilistic Voting-based Filtering (PVFS) [12],Dynamic En-route Filtering (DEF) [13], and Bandwidth-Efficient Cooperative Authentication for Sensor Networks (BECAN) [14].In SEF [9], different keyed Message Authentication Codes (MACs) are used to validate each sensed report. Therefore, the size of each report is increased due to the MAC overhead. Although Bloom-filters have been suggested to decrease the overhead of MACs, these have shown many flaws when implemented in static key management schemes [18]. A commutative cipher based en-route filtering (CCEF) scheme [9] verifies the false reports through an encrypted session key and un-encrypted witness key in the Query message. CCEF is based on a public key algorithms which needs more energy for the commutative ciphering and is unsuitable for wireless sensor networks. The IHA [10] scheme verifies and filters the malicious reports as long as the number of compromised nodes are less than t nodes. Its scope of filtering false injected reports is minimized specifically when the compromised nodes increases than t nodes. In the STEF protocol [11], the Base Station (BS) issues tickets and reports, and the valid tickets are forwarded. The drawback of this scheme is the unidirectional communication from the BS to the Cluster Heads (CHs), which is required for ticket traversal. A commutative cipher based en-route filtering (CCEF) scheme [12] verifies the false reports through an encrypted session key, and un-encrypted witness key in the Query message. CCEF is based on a public key algorithms which needs more energy for the commutative ciphering and is unsuitable for wireless sensor networks. In DEF [13], several nodes employ their authentication keys to endorse legitimate reports. Hence, DEF requires a significant amount of energy for authentication due to the use of separate secret keys. The BECAN [14] scheme is based on the graph characteristics of node deployment and the cooperative bit-compressed authentication technique. BECAN provides high security by the early detection of injected false data in the network, but it causes extra overhead at the forwarding nodes and consumes unnecessary energy resources due to its multi-report requirement. In [19], authors suggest a clustering algorithm on the basis of the type-2 fuzzy logic model instead of type-1 fuzzy logic (t1FL), to handle uncertain level decisions. The authors in [19] consider three fuzzy input parameters such as remain energy, distance to the BS, and the node concentration. The scheme proposed in [19] is proved to have better energy conserving performance than T1FL, LEACH-SH and LEACH-MH. In [20], Authors propose a fuzzy logic based data fusion technique to improve the QoS and the energy conservation in wireless sensor networks. The authors maintain that the fuzzy-based solution introduced in [20] is highly suited for applications that do not require strict real-time constraints and most useful in soft real-time contexts. Sensor nodes combine and transfer their data generating a lower traffic, instead of sending individual messages for each event separately.

Authors, in [21], propose a multi-attribute decision fusion model based on intuitionistic fuzzy set for dealing with uncertain data in wireless sensor networks. The proposed method achieves low energy consumption and low computation complexity and provides high classification accuracy contrary to that of traditional fuzzy fusion and fuzzy aggregation algorithm.

However, The proposed schemes in [19-21] are solely related to data aggregation and fusion in wireless sensor network using sophisticated fuzzy logic based techniques, where as we propose a fuzzy logic based source authentication technique which helps to very data reports at the intermediate nodes. Our proposed scheme enables the adaptive selection of a number of such verification nodes aimed at saving computational energy at low false positive report rate and increased security at higher false positive report insertion rate.

Selcuk et al. [16] presented a TICK scheme for WSNs that sends reports to the BS without sending rekeying messages. Sensor nodes encrypt each report with a dynamic key generated by their local time values. Although TICK saves more energy when compared to other schemes like DEF, SEF, STEF, and BECAN, it has relatively a high risk of classifying valid messages as malicious, which causes the rapid depletion of already limited energy resources. The working principle of SOBAS is based on that of TICK with three choices of re-encryption at the intermediate forwarding nodes: i) no re-encryption, ii) full re-encryption, and iii) selective re-encryption. However, SOBAS selects the nodes for re-encryption in the pre-deployment phase, and the number of selected nodes for re-encryption nodes remains fixed throughout the network lifetime; this is an impractical solution, as re-encryption activities consume more energy from the nodes. Therefore, we propose an adaptive selection of such nodes considering three input parameters, the remaining energy statuses of the intermediate nodes, false positive rate and the distance between the re-encryption nodes.

3. Background and Motivation

3.1 Background

The main objective of the TICK protocol is to achieve energy efficiency by eliminating the exchange of control keying messages, which cause rapid depletion of the sensor node energy. The working principle of the TICK protocol consists of three phases. In the first phase, when an event occurs, the source node utilizes its local time variable and generates a one-time dynamic key. The dynamic key is a function of the source node's local time (t_l) and a shared initialization vector (IV) which is given by.

$$K_1^t \; = F(t_l \, , IV) \tag{1}$$

Heret_l is the node's current local time value and IV is the initialization vector loaded in the node at the time of deployment.

The generated one-time dynamic key is then used for security services, such as encryption and authentication. Finally, the encrypted report is sent to the upstream forwarding nodes.

The intermediate forwarding nodes verify the incoming reports. To verify the report, the forwarding node decrypts the encrypted report and tries to find the time-based dynamic key. If the key is not original, it considers the report to be malicious and drops the report. Otherwise, it re-encrypts the report with its local time value and forwards the report towards the next forwarding node. In this way, the report reaches the BS. The BS also verifies the incoming report.

In SOBAS, three operational modes exist in the module responsible for encryption and decryption of the data to determine how to forward the incoming packet. In the first mode, *No-reEnc*, the original incoming packet is forwarded to the next node without any re-encryption whereas in the second mode, *Full-reEnc*, the received data packet is re-encrypted with the key associated with the local time at the current node before forwarding it to the next node. For the *Full-reEnc* mode, every intermediate forwarding node synchronizes itself loosely with the source. The benefit of the *No-reEnc* mode is the elimination of encryption computation, hence energy is saved by forwarding the original packet. However, a legitimate report may be classified as a false report if the current forwarding node is located too far away from the source node. The third mode is the *Selective-reEnc* (S-ReEnc) mode, in which packets are selectively re-encrypted over selected nodes along the data delivery path; note that these nodes are also loosely synchronized with the source as in *Full-reEnc*. *Selective-reEnc* is explicitly presented in SOBAS to solve the problem of classifying a legitimate incoming packet as malicious (i.e., false positive), a situation that may occur in the *No-reEnc* mode.

Energy consumed during transmission is directly proportional to the distance between the source and the recipient, as given by Equation (2).

$$E_{Tx} = E_{elec(k)} + E_{amp} \times k \times d^{a} \tag{2}$$

Where k is the size of data, d is the distance between the source and the recipient, E $_{\mathrm{elec(k)}}$ is the energy consumed to run the radio electronic and ε amp is the energy consumed to amplify the signal. A is 2 for the free-space. Therefore, it is very important to detect and drop false report at the early stage to save energy as well save the computation energy needless expended in the absence of false positive reports. We propose a Fuzzy logic-based adaptive selection of encryption nodes that saves the energy resources of the intermediate encryption nodes.

3.2 Motivation

Knowing that false report injection and false positive classification deplete more energy from the sensor nodes thus shortening the network life-time, it is important to design an energy efficient method capable of improving energy savings, maximizing the network life-time and providing maximum resistance against false report injection. TICK counters this type of attack using a completely new approach which is also employed by SOBAS. TICK presents a scheme that sends reports to the BS without exchanging different control keying messages to avoid stale keys [16]. However, TICK employs either *No-reEnc* or *Full-reEnc* whereas SOBAS [17] also includes *Selective-reEnc* along with the two existing non-re-encryption and full encryption strategies. Thus, SOBAS supplements the TICK operation framework by including loose synchronization between nodes and the selective re-encryption of data along the path. In the*No-reEnc* mode in either TICK or SOBAS, a valid message might be dropped because the clock drift between the two nodes is too large, and therefore the key falls outside of the time window of the forwarder. As the number of hops between the source and forwarding node increases, the clock drift is more unbounded, therefore the probability of a false positive increases. Secure Source-Based Loose Synchronization (SOBAS) [17] improves TICK by employing selective re-encryption and full- re-encryption modes and loose synchronization, thus reducing the likelihood of opportunity of malicious nodes injecting false data into the network. However, these modes are pre-determined, as in the aforementioned mode where every 3rd, 5th, or 7th forwarding node is selected for the re-encryption operation;

this is suboptimal because it selectively depletes more energy via odd-numbered encrypting and forwarding nodes in low attack situations. There is no need to have either *Full-reEnc* or *Selective-reEnc* with encryption nodes located more closely to one another if there are ideally few or no false positive reports. That is to say; it is only advisable to decrease the distance between re-encrypting nodes and increase the number of selected re-encryption nodes if the rate of false positives increases. Re-encryption increases the computational cost and therefore it only advisable to increase the re-encryption workload of nodes if needed. In the*Full-reEnc* mode, every forwarding node re-encrypts the incoming report with its local time value and increases the computational cost in the presence of a healthy environment and low attack rate. In addition, these modes do not consider the network conditions (i.e., the remaining energy of the intermediate nodes, hop count distance between the two consecutive re-encryption nodes, and false positive attack ratio). In both TICK and SOBAS, the selected nodes are always fixed and dedicated to re-encryption and continuously performing the re-encryption operation; this in turn continuously, depletes the energy and may result in early WSN failure.

Communication and report verification are two common reasons for energy consumption in WSNs. We propose an energy efficient method based on a fuzzy logic system to minimize the energy consumption of the forwarding nodes by considering the remaining energy level of the forwarding nodes, the hop counts between two consecutive re-encryption nodes, and the false positive ratio. In the presence of high FPR, the transmission energy can be saved by choosing encryption nodes more closer to one another to filter false report earlier. In the absence of or low FPR, the computational energy can be saved by choosing lesser and relatively farther encryption nodes from one another. Whereas, the absence of an adaptive encryption node selection strategy, their number and distance remain fixed irrespective of the false positive rate and energy status of the intermediate nodes.

4. Proposed Method

4.1 System Model and Assumptions

In our work, we assumed a large scale densely populated wireless sensor network comprised of a base station and many sensor nodes. Sensor nodes, with the same communication range and initial energy, remain static after they are deployed in the network. We also assume that all nodes are assigned a network-wide initialization vector (IV) and are synchronized in the pre-deployment stage. The BS is the decision maker and has more computational power, a high storage capacity, and unlimited power resources. In addition, the topology establishes initial routing paths through directed diffusion [16]. The scheme deals with false injected messages from an external node, and so insider attacks are not considered in this paper.

4.2Method Details

4.2.1Overview

The proposed method utilizes the functioning modules of TICK. The other two modules, i.e., time-based key management (TKM) and filtering-forwarding (FFWD) modules, remain the same as in [16]. The proposed method adaptively selects forwarding nodes for re-encryption to conserve energy of the forwarding nodes and reduce the probability of classifying a valid message as malicious. We aim to choose nodes that perform re-encryption through fuzzy logic and therefore only discuss the CRYPT module of the original schemes, i.e., TICK and SOBAS.

4.2.2Crypto (CRYPT) Module

This module obtains the dynamic key generated in the TKM module and performs the security operations required in the proposed method. The key from the TKM module is also verified in this module. If the verified key is not correct, the module obtains another key from the TKM and continues this operation until it finds the correct key. Otherwise, after all of the attempts to find the correct key is exhausted within the tick window (Tw), it considers the report as malicious and drops it in the FFWD module. This module uses the RC4 algorithm as the encryption technique because it needs less energy for the encryption process and is easy to implement [22, 23]. The CRYPT module detects injected false data.

Three working modes can be employed in this module that determines how to forward the incoming report. These modes are no-re-encryption, full-re-encryption and selective-re-encryption. In the no-re-encryption mode, each intermediate node sends the original report to the upstream forwarding nodes without a re-encryption operation. The no-re-encryption mode saves energy by forwarding the original report as it performs a one-time encryption operation and reduces the computational cost of the forwarding nodes. However, if the distance between the current forwarding node and the source node is large, it increases the probability of classifying the original report as malicious at the forwarding node. In the full-re-encryption mode, each intermediate node forwards the incoming report to the upstream nodes after performing the re-encryption operation. For the full-re-encryption mode, each forwarding node must create a new key when re-encrypting the incoming report using its current local clock value. The chance of considering a healthy incoming report as malicious is highly unlikely because each forwarding node refreshes its key in this mode, which makes it easier for the next receiving node to find the correct key associated with the current report. Finally, in the selective-re-encryption mode, the report is forwarded to the upstream nodes after performing the re-encryption operation at only some selected forwarding nodes, and the distance between any two consecutive such nodes is the same. It is worth noting that in the last two modes, re-encrypting nodes synchronize themselves loosely with the source, as illustrated in [17].

In this paper, we employ the selective re-encryption operation mode based on fuzzy logic which selects the re-encryption nodes in response to an increase in the false positive rate (FPR). As long as the FPR is low or ideally zero, it is not ideal to select each 3rd, 5th or so on nodes to save the network energy from the computation. However, since SOBAS is not adaptive and selects selective re-encryption modes during pre-deployment, it, therefore, cannot reduce the distance between two consecutive re-encryption nodes if the FPR increases. Therefore, the nodes between two consecutive re-encryption nodes also invest their energies in receiving and transmitting the report, which will be dropped by the next re-encryption node because it is unable to calculate the correct key associated with the report. Our proposed method selects more re-encrypting nodes in the presence of a higher FPR in order to save the energy used in transmitting and receiving malicious reports. However, in the presence of a low FPR, it selects fewer re-encryption nodes in order to conserve the overall energy of the network and reduce the computational cost.

4.2.3Fuzzy Logic System

A mathematical model for a WSN with capabilities of adapting to the changing network environment is not only complex and difficult to make but also unfeasible and unviable a solution. Such mathematical models are not extensible and can only cater to specific configurations of static networks. Environments in which WSN are deployed are usually very dynamic with environmental variables frequently changing values. Security model, Attacks' nature, and frequency, inter-node distances, energy status and size of the network are few variables which need to be constantly monitored. In such a case Fuzzy Rule-based techniques become a natural choice as they provide us with the best approach for dealing with the uncertainty of measurements performed in WSNs, which are affected by errors in precision and accuracy [25, 26]. Rule-based fuzzy systems are effective because of their utility for approximate reasoning when there is a degree of uncertainty in the reasoning process and imprecision in the data [27]. The inference engine makes use of fuzzy membership functions as inputs and fuzzy rules for its decision making.

In our proposed method, we employ a fuzzy rule-based scheme to adaptively select re-encryption nodes, which not only reduces the chances of considering a healthy incoming report as malicious, but also conserves the energy of the forwarding nodes in both cases (i.e., in no-attack situations, the energy consumed during re-encryption computations and massive attack situations, and the energy consumed in receiving and transmitting a false positive report). The BS regularly monitors the network status and decides how many nodes are to be selected for re-encryption based on three parameters (the remaining energy of the nodes on the routing path (REL), hop count distance between two current executive re-encryption nodes (HC), and false positive rate (FPR)) using a fuzzy logic system, as shown in Fig. 1. The number of nodes selected for re-encryption computations the incoming data depends on the nature of the fuzzy inputs. The forwarding nodes selected for re-encryption may increase or decrease based on the fuzzy decision.

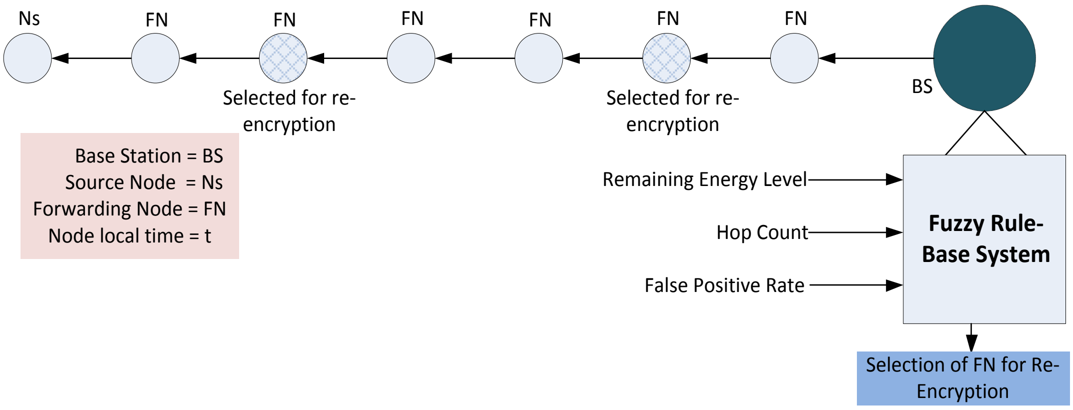
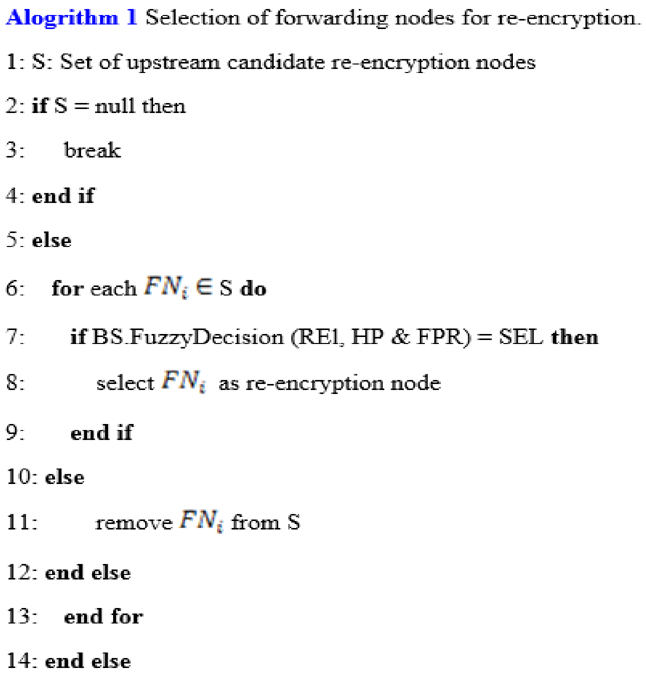


Fig. 1. Fuzzy rule-based selection of forwarding nodes.

Algorithm 1 describes the process of selecting nodes for re-encryption. Here, S is a set of upstream forwarding nodes and FN_i Represents the i^{th} data forwarding node between the source node and the BS. FuzzyDecision (*input_values*) produces the decision of whether or not to select the current node as a re-encryption node. Upon receiving a report, the source node transmits the report by forwarding nodes towards the BS. The BS informs all forwarding nodes between the source node and BS about the result of the fuzzy decision. The fuzzy decision is then made by considering various selected network factors, such as the remaining energy level, hop count, and false positive rate. The fuzzy decision is either one of two possible cases; SEL for selection and REJ is for rejection. If the result is SEL, then the selected forwarding node performs the re-encryption operation. Otherwise, the forwarding node sends the report towards the upstream forwarding node in contact with the BS without performing the re-encryption operation.



Alogrithm 1 Selection of forwarding nodes for re-encryption.

```
S: Set of upstream candidate re-encryption nodes
if S = null then
    break
end if
else
   for each $FN_i \in$ S do
      if BS.FuzzyDecision (RE1, HP & FPR) = SEL then
         select $FN_i$ as re-encryption node
      end if
else
         remove $FN_i$ from S
end else
   end for
end else
```

The general block diagram of the fuzzy logic system is shown in **Fig. 2**. A fuzzy system is basically composed of three phases:

1) Fuzzification
2) Fuzzy inferencing
3) Defuzzification.

In the first phase, fuzzification, the fuzzifier converts the inputs (crisp values) into a fuzzy set. This can be achieved by fuzzification. In the second phase, fuzzy inferencing, the inference engine processes the fuzzified values and draws fuzzy reasoning from the knowledge base. In the third phase, defuzzification, the defuzzifier converts the fuzzy output set into a crisp-value output.

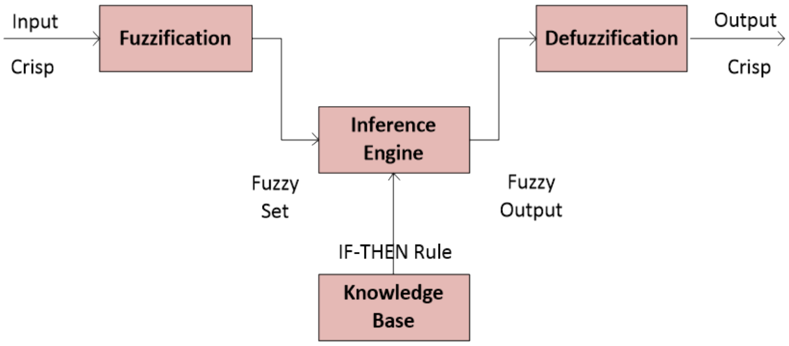


Fig. 2. Block diagram of the fuzzy logic system

Fig. 3 shows the proposed fuzzy logic system employed in the BS. The three phases of fuzzy logic, fuzzification, fuzzy inference, and defuzzification are presented in **Figs. 3**(a), **3**(b) and **3**(c), respectively.

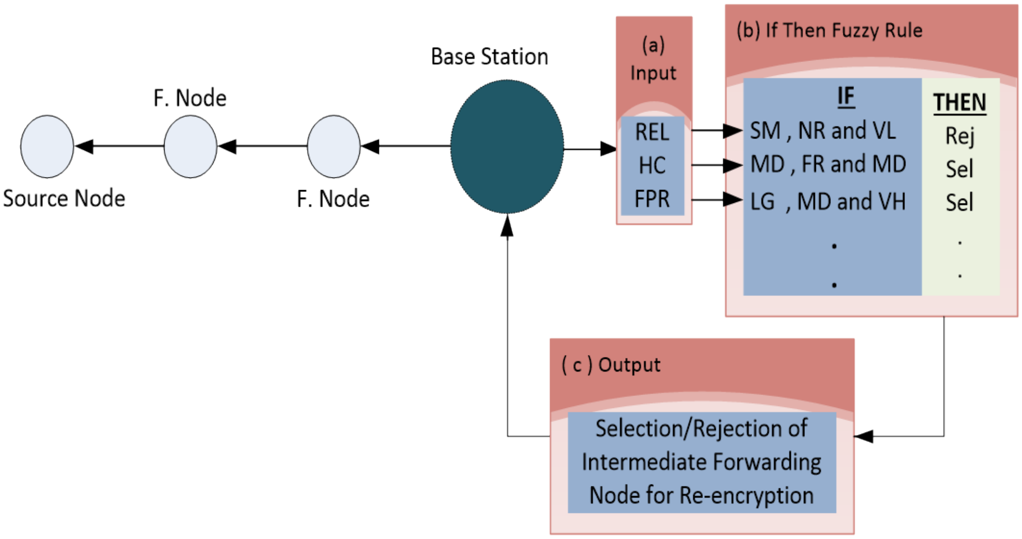


Fig. 3. Fuzzy logic system: (a) input; (b) fuzzy if-then rules, and (c) output.

1) *Fuzzification:*

To achieve a high energy gain at the forwarding nodes, three important input factors are considered in the proposed method. In our proposed protocol, Mamdani’s fuzzy inference method was used to select the forwarding node for re-encryption because it is the most frequently used inference technique [28, 29]. The fuzzy set that represents the first input variable (i.e., the remaining energy level) is shown in **Fig. 4**(a). We consider the remaining energy of the node because there will always be energy consumption at each node in each round and it is an important factor to consider when trying to achieve energy balance between the forwarding nodes. The linguistic variables for the fuzzy set are small, medium and large. If the REL of the forwarding node is very small, the node is discarded from consideration for re-encryption to extend the battery life of the node. The selection of the forwarding node for re-encryption depends on its energy state. The value of the input parameter Remaining Energy is calculated as:

$$Remaining\ Energy = \sum_{i=1}^{n} \frac{REL_i}{n} \quad (3)$$

Where REL$_i$ is the remaining energy of the intermediate nodes between the source node, and the BS and n is a total number of intermediate nodes between the source node and the BS.

The hop count (HC) is another vital factor in selecting re-encryption nodes for verification. The HC in our work is the distance between two consecutive re-encryption nodes. This factor is important because clock drift is directly proportional to the HC. Clock drift becomes unbounded if the distance between the two consecutive encrypting nodes increases, which may result in a healthy report dropped after is classified as a malicious report. Our proposed method selects the optimal number of re-encryption nodes. The linguistic variables for hop count are near, middle, and far. The fuzzy set for the hop count is depicted in **Fig. 4(b)**.

HOP-COUNT is calculated using the following equation:

$$HOP - COUNT = \frac{d_{i,i+1}}{d_{source,BS}} \times 100\ (\%) \quad (4)$$

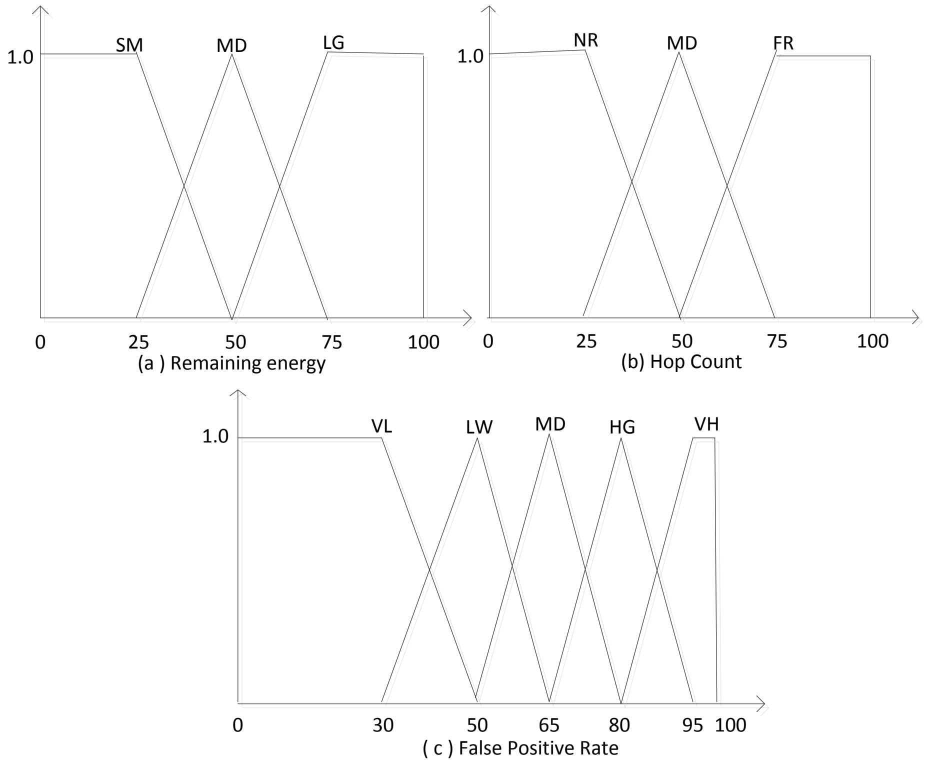
Where $d_{i,i+1}$ = The distance between two consecutive encryption nodes *i* and *i*+1. And $d_{source,BS}$ = The distance between the source and the BS.

In case of no intermediate encryption nodes between the source and the BS, HOP-COUNT = 100 (%), because there is no encryption node between the source node and BS. In case of full encryption, every intermediate node is an encryption node, therefore, $HOP - COUNT = \left(\frac{1}{d_{source,BS}}\right) \times 100\ (\%)$

The false positive rate (FPR) also plays an important role in selecting the number of forwarding nodes for re-encryption. If the FPR is high and the number of nodes for re-encryption operation is relatively low, then the en-route nodes will consume more energy while receiving and transmitting incoming false positive reports. The rate of false positives increases with increases in the distance between two consecutive re-encrypting nodes and the gradual drift in the clocks. The proposed method selects more nodes for the re-encryption operations in the case of a high FPR and vice versa. The linguistic variables for the FPR are very low, low, medium, high, and very high. The trapezoidal membership function was considered for the very low and very high variables. For the low, medium, and high variables, a triangular membership function was considered. The fuzzy set for the FPR is depicted in **Fig. 4(c)**. **Table 1** shows the membership functions of all the input variables. Our proposed method considers this factor when selecting the number of forwarding nodes for re-encryption to save the energy of the forwarding nodes. The FPR is calculated using Equation (5).

$$\mathrm{FPR} = \frac{\mathrm{R_{T.GEN}} - \mathrm{R_{BS.ack}}}{\mathrm{R_{T.GEN}}} \times 100 \quad (5)$$

Here, $\mathrm{R_{T.GEN}}$ is the total number of reports generated and $\mathrm{R_{BS.ack}}$ is the total number of reports acknowledged by the BS.



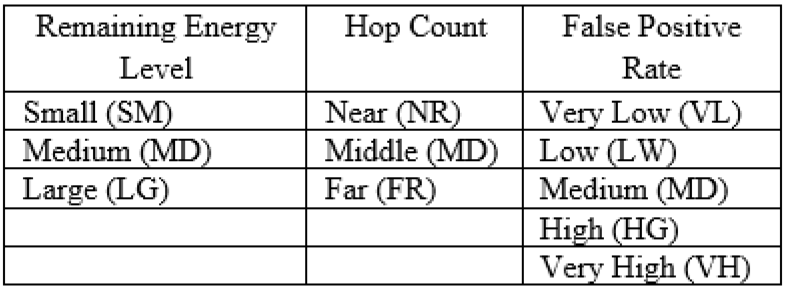
(a) Remaining energy

(b) Hop Count

(c) False Positive Rate

Fig. 4. Fuzzy set for the input variables.

Table 1. Membership functions for input variables.



Remaining Energy Level	Hop Count	False Positive Rate
Small (SM)	Near (NR)	Very Low (VL)
Medium (MD)	Middle (MD)	Low (LW)
Large (LG)	Far (FR)	Medium (MD)
		High (HG)
		Very High (VH)

2) *Inference Engine:*

We have 45 rules in the inference engine that are obtained from the three input factors selected in the proposed method. Both trapezoidal membership functions and triangular membership functions are used in the inference engine. The trapezoidal membership functions represent the boundary variables, and the triangular membership functions represent the intermediate variables. The two membership functions used in our inference engine are given in Equations (6) and (7).

$$f(x;a,b,c) = \max\{\min\left(\frac{x-a}{b-a},\frac{c-x}{c-b}\right),0\} \tag{6}$$

$$f(x;a,b,c,d) = \max\left\{min\left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}\right), 0\right\} \tag{7}$$

The form of the fuzzy rules is if A *AND* B *AND* C then R. A the represents remaining energy level, B represents the hop count, C represents the network attack ratio, and R represents the result.

For the two extreme cases, the given two fuzzy rules are:

- If the remaining energy level of a candidate node, that is currently being considered for selection is very small and is closer to the previous re-encryption node, and the false positive rate is small, then the current node is discarded from the candidate list.
- If the remaining energy level of a candidate node is very large and is far from the previous re-encryption node, and the false positive rate is large, then the candidate node is selected for re-encryption.

In our proposed method, membership functions of the input parameters have been selected based on the simulation results. For example, peaks of the membership function of FPR were determined by locating intersection points on the plot. Then, functions were tuned by combining fuzzy results until desired situations could be covered by the membership function. In real sensor networks, some membership functions may need to be modified due to more uncertainty in the parameters. Membership functions of input parameters can also be tuned based on expert knowledge of the network settings.

Some of the IF-THEN rules are given in **Table 2**.

Table 2. Fuzzy if-then rules.

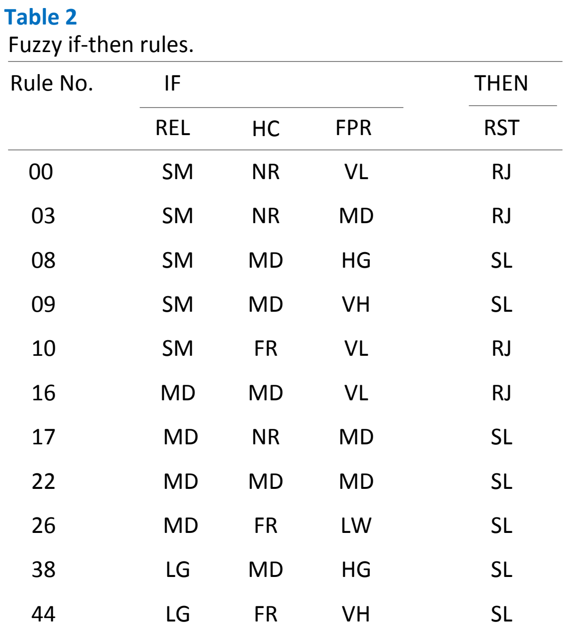


Table 2
Fuzzy if-then rules.

Rule No.	IF			THEN
	REL	HC	FPR	RST
00	SM	NR	VL	RJ
03	SM	NR	MD	RJ
08	SM	MD	HG	SL
09	SM	MD	VH	SL
10	SM	FR	VL	RJ
16	MD	MD	VL	RJ
17	MD	NR	MD	SL
22	MD	MD	MD	SL
26	MD	FR	LW	SL
38	LG	MD	HG	SL
44	LG	FR	VH	SL

3) Defuzzification

The output variable, i.e., Result (RST), is defuzzified by using the centroid method or CoA (Center ofArea) to produce a crisp value. The weighted average of the fuzzy set A is calculated by Equation (8).

$$Result = \frac{\int u_a(x) * xdx}{\int u_a\,(x)dx} \tag{8}$$

The COA method for defuzzification is most commonly used for implementation in wireless sensor network [27]. The Center of Area defuzzification method effectively calculates the best compromise between multiple output linguistic terms.

The output variable "Result (RST)" is composed of two membership functions, select (SL) and reject (RJ). The fuzzy set for Result is shown in **Fig. 5**. **Table 3** shows the membership functions for the output variable.

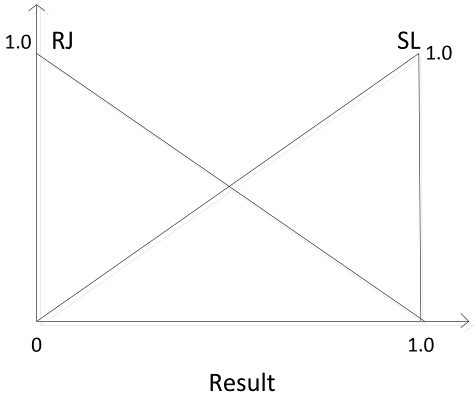
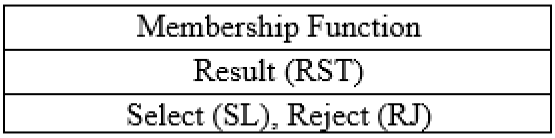


Fig. 5. Fuzzy set for the output variable.

Table 3. Membership functions output variables



Membership Function
Result (RST)
Select (SL), Reject (RJ)

5.Performance Evaluation

We compared the proposed method to the improved version of [16], which is the Secure Source-based loose synchronization for Wireless Sensor Networks (SOBAS) [17]. The authors of [16] improved TICK by employing a loose synchronization strategy and suggested a selective re-encryption mode along with full and no re-encryption strategies [17]. Our proposed method also efficiently employs selective re-encryption operations through fuzzy logic considering the network conditions (i.e., REL, HC, and FPR), which is more energy efficient than the pre-deployment determination of re-encryption nodes in SOBAS.

5.1 Energy consumption analysis

We consider six en-route nodes between the source node and the BS in **Fig. 6(a)**. In the case of [17], each third node performs the re-encryption operation. As these selected nodes are pre-determined and must perform verification and re-encryption operations continuously, they may, therefore, run out of energy, while other nodes still have more energy than ample energy. In the presence of a massive FPR, the energy depletion due to the transmission and reception of false positive reports increases with an increased distance between the two consecutive re-encrypting nodes and the higher FPR. The adaptive selection of re-encryption nodes reduces this distance and improves the network performance by reducing both the FPR and the over- all network energy consumption (previously caused by the increased FPR).

The proposed method selects re-encryption nodes with respect to the nature of the three network factors (i.e., REL, HC, and FPR) based on fuzzy logic systems. Consider the six en-route nodes between the source node and the BS in **Fig. 6(b)**. In the case of the proposed method, the remaining energy of the en-route nodes is considered while selecting the re-encryption node. The proposed method selects the node with the highest remaining energy level for the re-encryption operation. The proposed method also considers the distance between the source node and the recipient node to reduce the probability of classifying the healthy report as malicious; this is because of reports classified as a false positive, waste energy resources of the en-route nodes as they will ultimately be dropped en-route. The proposed method selects more nodes for re-encryption operations in the case of a high FPR to reduce the communication cost, whereas it selects a small number of forwarding nodes for re-encryption to reduce the overall computational costs of the network and to minimize the false positive classification rate.

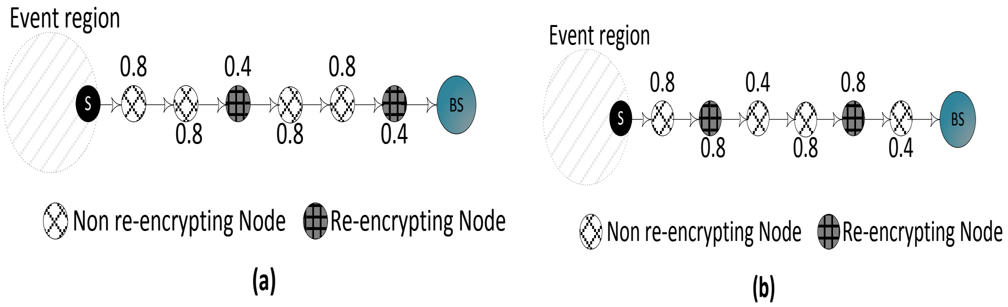


Fig. 6. Energy efficiency of (a) the original scheme and (b) the proposed method.

5.2Simulation Results

In this subsection, we evaluate the effectiveness of the proposed method through simulations. Network parameters and assumptions are described in Section 5.2.1, and simulation results regarding security and energy consumption are presented in Section 5.2.2.

5.2.1Simulation Parameters and Assumptions

We simulated the proposed method in a custom simulator developed in Microsoft Visual C ++ 2010. The network details and parameters are presented in **Table 4**.

Table 4. Simulation parameters

Parameter	**Value**
Number of nodes	100
Network size	100 m^2× 100m^2
Base station location	0 m^2× 0m^2
Range	30m
Report size	32Byte
Tick window (Tw)	16
Time off set	U [-3, +3]
Node's initial energy	5000 mj
Energy consumed to:	
Transmit/ Receive a report	66.7/ 59.6 µj
Decode/ Encode a report	3.3 µj
Generate a MAC	8.6 µj

The network is composed of a BS and 100 sensor nodes, where the nodes are randomly deployed in a field of area 100 m × 100 m. Each of the sensor nodes has a communication range R_i = 30m. The BS is located at the edge (bottom left) of the network and knows the IDs of the sensor nodes and their location information in advance. The sensor network used in our method is shown in **Fig. 7**.

The Directed Diffusion protocol [22] is used for routing. Upon detection of the first event from the event region, a path is created from the source node to the BS. Event sources are randomly selected. After establishing a secure session, all subsequent events in that area are reported to the BS following the same path for the duration of the session or until a node is depleted. Each sensor node has a fixed and limited sensing range and is battery powered with a fixed yet limited energy of 5000 mJ. Moreover, the communication links are bidirectional in the sense that if A can send a message to B, then B is also capable of sending a message back to A.

The energy consumption cost of the different operations is computed based on the values given in [16, 17]. E_{tx} ,E_{rx} and E_{sens} are the energy consumption costs of sending a report, receiving a report and sensing an event, respectively. E_{enc} ,E_{dec} and E_{mac} are the energy consumption costs of encryption, decryption, and the message authentication code, respectively.

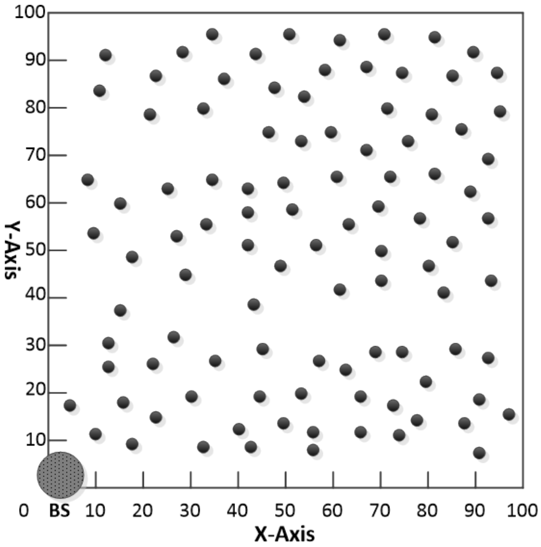
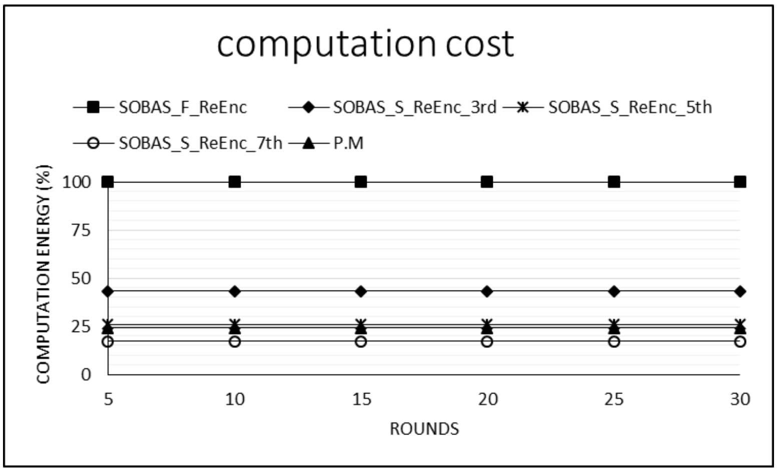


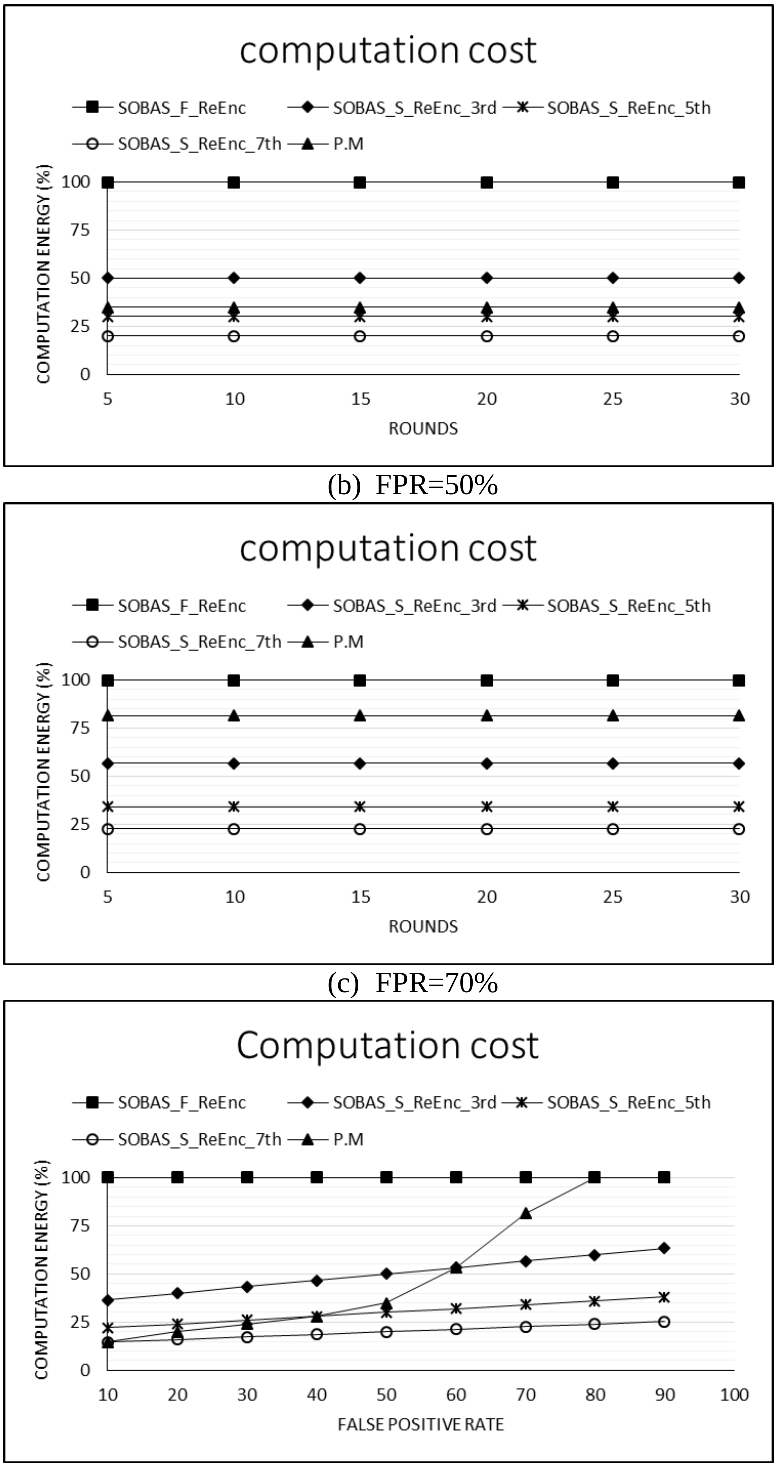
Fig. 7. Random deployment of sensor nodes in the network.

5.2.2Simulation Results for Security and Energy Consumption

To evaluate the performance of the proposed scheme, we compared its performance with that of SOBAS. Performance is measured by quantitative metrics of energy dissipation in both the cases, i.e., computational energy dissipation and transmission energy dissipation. The plots in **Figs. 8(a)**, **8(b)** and **8(c)** represent the computational energy consumed in re-encrypting reports when the FPR is 30%, 50% and 70% respectively. In **Fig. 8(d)**, we measured the computational energy consumed in re-encrypting the reports while the FPR is both variable and increasing with the passage of time. It is evident in **Fig. 8(a)** that the computational energy consumed during periods when the FPR is lower is much less than when the FPR is higher. The computational energy only increases with the increase in FPR.



(a) FPR=30%



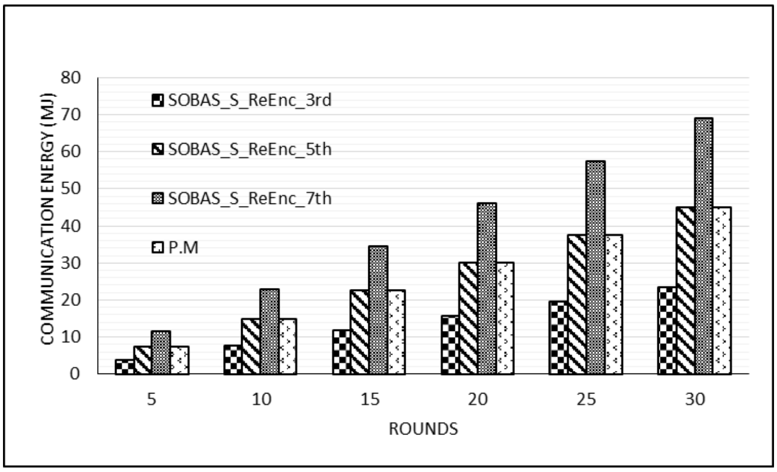
(b) FPR=50%

(c) FPR=70%

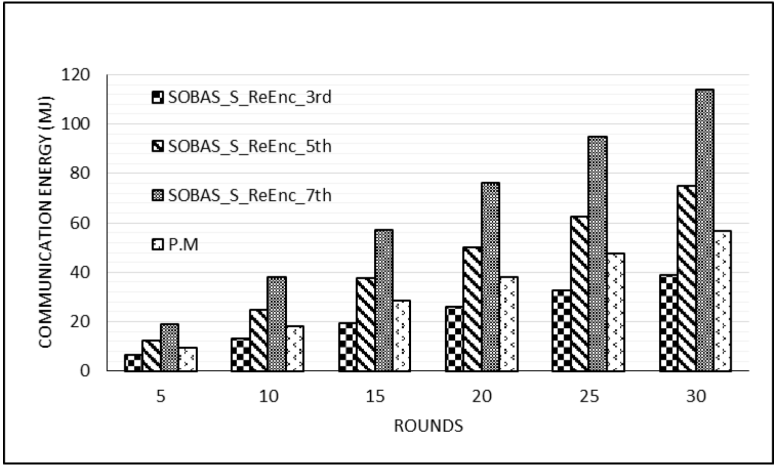
(d) Reports= 100

Fig. 8. Energy dissipation due to re-encrypting false positive reports

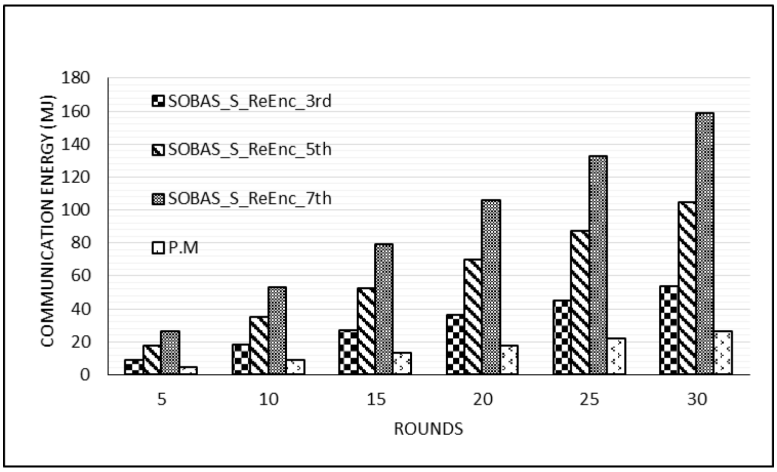
We also evaluated the energy consumption behavior of the existing scheme and our proposed scheme in the presence of different FPR rates and with different modes of encryptions, i.e., when each 3rd, each 5th or each 7th node performs encryption. The FPR increases with the passage of time if the distance between the two consecutive encrypting nodes is longer. Therefore, more energy is consumed in the transmission of a report classified as false positive when the hop-count distance between the encrypting nodes is longer. The false positive report is forwarded until the next encrypting node identifies it as false positive and filters it out. The plots in **Figs. 9(a)**,**9(b)** and **9(c)** represent the communication energy consumption in forwarding the false positively classified reports in the presence of 30%, 50%, and 70% FPR respectively. In **Fig. 9(d)**, we measured the communication energy consumed in forwarding the reports classified as false positive with respect to an increase in the FPR. The above plots show that greater energy efficiency is attained using the proposed method than SOBAS. The communication cost is gradually reduced when the reports are generated and forwarded to the BS as the FPR increases. This is because the proposed method allocates more forwarding nodes for re-encryption operations in the presence of a high FPR. The selection of a higher number of encryption nodes on the data forwarding path helps to drop false positive reports early and reduces the communication cost. Our proposed scheme allocates fewer forwarding nodes for re-encryption in the case of a low false positive ratio to reduce the overall computational cost in the network.



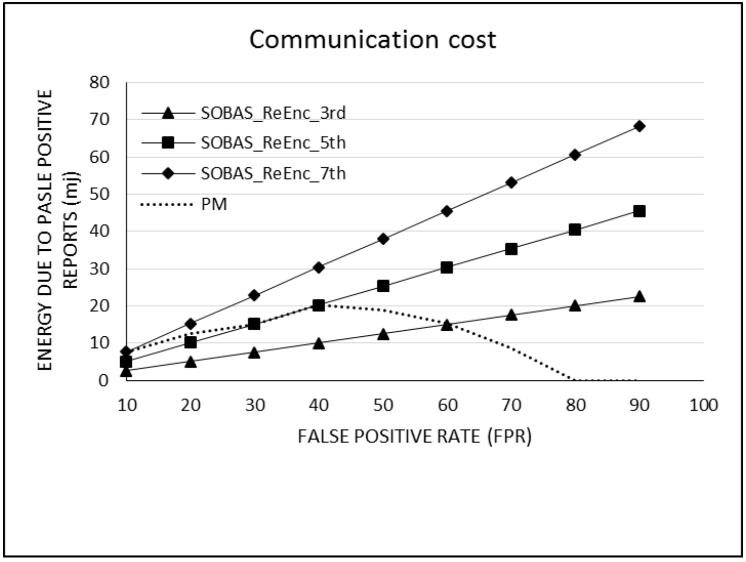
(a) FPR=30%



(b) FPR=50%



(c) FPR=70%



(d) REPORTS=100

Fig. 9. Communication energy consumption of forwarding false positive reports in the presence of different FPR values

Fig. 10 shows the total energy consumption behaviors of the existing and proposed schemes. The total energy consumption is comprised of two parts, the computational energy consumption (due to decryption/re-encryption operations) and the communication energy consumption in forwarding the incoming reports from the source node towards the BS. The total energy gain in the network is greater using the proposed method than is possible using SOBAS. Due to selecting a fixed number of forwarding nodes as re-encryption nodes in SOBAS, it will fail to adapt to the variable nature of the FPR. Therefore, if SOBAS employs a ReEnc_7th mode, it will consume more energy due to the communication of false positive reports when FPR is high. Similarly, if SOBAS employs a ReEnC_3rd mode in the presence of low an FPR, each 3rd node will always be engaged in re-encryption computation, and will, therefore, incur more computational energy. In contrast, the selected re-encrypting nodes are not fixed in the proposed method and are adaptively selected for the verification operation using fuzzy logic. Hence, the energies of these nodes do not deplete as much or as quickly in the existing scheme, and the energy of nodes tends to deplete in a uniform manner. In **Fig. 10**, the plots show the average gain of communication and computational energy consumptions with different FPR values during each round.

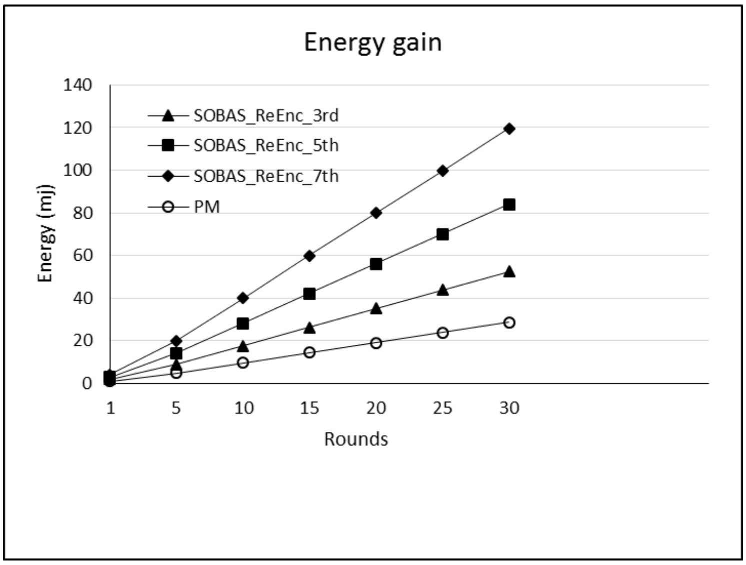


Fig. 10. Energy gain with respect to the number of rounds

6. Conclusion

Exchanging control messages regarding keying or rekeying consumes most of the energy in WSNs. These operations also increase the communication cost in sensor networks. In addition, malicious nodes can disturb the network functions by injecting malicious data. This type of attack causes a rapid reduction in the network's energy resources. To provide network security and minimize communication cost, TIme-Based DynamiC Keying and En-Route Filtering (TICK), and Secure SOurce-BAsed Loose Synchronization (SOBAS) were introduced. In TICK, nodes utilize their time values and generate time-based dynamic keys that are used to encrypt reports and avoid the overhead associated with regular keying of messages. SOBAS is an extension to TICK, wherein three modes are proposed, i.e., i) no re-encryption at the intermediate nodes, ii) re-encryption at selected intermediate nodes and iii) re-encryption at all the intermediate nodes.

Although this method is more energy efficient than TICK which employs only no-encryption and full re-encryption modes at all the intermediate nodes, a valid message may still be dropped if the clock drift between two nodes is too large, i.e., if the distance between two re-encrypting nodes is greater than a threshold. In this case, the key may fall outside the time window of the next forwarding node, which has to first decrypt the message through finding a correct key associated with the report. As the number of hops between nodes increases, the clock drift becomes more unbounded. Therefore, the probability of false positive increases, which causes more energy consumption in forwarding a false positive report until they are dropped at an intermediate node.

In addition, network conditions (i.e., the residual energy of the en-route nodes, false positive rate and hop count distance between the two re-encrypting nodes) are not considered in TICK, which is equally important when choosing the selective re-encryption nodes. To address these issues, we proposed an energy efficient method based on a fuzzy logic system that intelligently selects re-encryption nodes, given the consideration of the network parameters: the remaining energy of the forwarding nodes, the hop-count distance between the encrypting node, and the false positive rate. In the presence of high FPR, the transmission energy can be saved by choosing encryption nodes more closer to one another to filter false report earlier. In the absence of or low FPR, the computational energy can be saved by choosing lesser and relatively farther encryption nodes from one another. The proposed method reduces the probability of valid reports being classified as malicious reports and, at the same time, improves the overall energy conservation efforts of the network by selecting the proper number of encryption nodes only when needed. The simulation results validate the robustness and efficacy of the proposed method and serve to strengthen the security and utility of WSNs. In future, we aim to carry out further study to investigate and implement the idea of source authentication at intermediate nodes in a data aggregation based sensor network.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 08, pp.102-114, Aug, 2002. Article (CrossRef Link)

[2] P. Rawat, K.D. Singh, H. Chaouchi and J.M. Bonnin, "Wireless sensor networks: a survey on recent developments and potential synergies," *The Journal of Supercomputing*, vol. 68, no. 01, pp.1-48, April, 2014. Article (CrossRef Link)

[3] A. Guermazi, A. Belghith, M. Abid and S. Gannouni, "KMMR: An efficient and scalable key management protocol to secure multi-hop communications in large scale wireless sensor networks," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 2, pp. 901-923, February, 2017. Article (CrossRef Link)

[4] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures," in *Proc. of 1st IEEE International Workshop on Sensor Network Protocols and Applications*, vol. 34, no. 04, pp.113-127, May, 2003. Article (CrossRef Link)

[5] Q. Yang, X. Zhu, H. Fu and X. Che, "Survey of security technologies on wireless sensor networks," *Journal of Sensors*, (pages 09), Dec, 2014. Article (CrossRef Link)

[6] H. Y. Lee and T. H. Cho, "Fuzzy adaptive selection of filtering schemes for energy saving in sensor networks," *IEICE Transactions on Communications*, vol.E90-B, no.12, pp.3346-3353, Dec, 2007. Article (CrossRef Link)

[7] T. Kim and H. Lee, "Performance evaluation of the RIX-MAC protocol for wireless sensor networks," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 2, pp. 746-784, February, 2017. Article (CrossRef Link)

[8] F. Ye, H. Luo, S. Lu and L. Zhang, "Statistical en-route filtering of injected false data in sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp.839–850, April, 2005. Article (CrossRef Link)

[9] Yang, H. Yang and S. Lu, "Commutative cipher based en-route filtering in wireless sensor networks," in *Proc. of Vehicular Technology Conference*, vol. 2, pp. 1223-1227, October 2004. Article (CrossRef Link)

[10] S. Zhu, S. Setia, S. Jajodia and P. Ning, "An Interleaved Hop-by-Hop Authentication Scheme for Filtering of Injected False Data in Sensor Networks," in *Proc. of IEEE Symposium on Security and privacy*, pp. 259–271, May, 2004. Article (CrossRef Link)

[11] C. Kraub, M. Schneider, K. Bayarou and C. Eckert, "Stef: A secure ticket-based en-route filtering scheme for wireless sensor networks," in *Proc. of 2nd Int. Conf. on Availability, Reliability and Security*, pp. 310–317, April, 2007. Article (CrossRef Link)

[12] F. Li, A. Srinivasan and J.Wu, "PVFS: a probabilistic voting-based filtering scheme in wireless sensor networks," *International Journal of Security and Networks*, vol. 3, no 3, pp.173-182, August, 2008. Article (CrossRef Link)

[13] Z. Yu, and Y. Guan, "A dynamic en-route filtering scheme for data reporting in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 1, pp. 150-163, February, 2010. Article (CrossRef Link)

[14] R. Lu, X. Lin, H. Zhu, X. Liang, and X. Shen, "Becan: A bandwidth-efficient cooperative authentication scheme for filtering injected false data in wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 1, pp. 32-43, January, 2012. Article (CrossRef Link)

[15] L. Eschenauer and V.D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. of Proceedings of the 9th ACM Conference on Computer and Communications Security*, pp. 41–47, November, 2002. Article (CrossRef Link)

[16] A.S. Uluagac, R. A. Beyah and J. A. Copeland, "Time-based dynamic keying and en-route filtering (TICK) for wireless sensor networks," in *Proc. of IEEE Global Telecommunications Conference*, pp. 1-6, Dec, 2010. Article (CrossRef Link)

[17] A. S. Uluagac, R. A. Beyah and J. A. Copeland, "Secure source based loose synchronization (SOBAS) for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, pp. 803–813, April, 2013. Article (CrossRef Link)

[18] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu and M. Galloway, "A survey of key management schemes in wireless sensor networks," *Computer Communications*, vol. 30, no. 11, pp. 2314–2341, September, 2007. Article (CrossRef Link)

[19] P. Nayak and B. Vathasavai, "Energy efficient clustering algorithm for multi- hop wireless sensor network using type-2 of fuzzy logic," *IEEE sensor journal*, vol. 17, no. 14, July 2017. Article (CrossRef Link)

[20] M. Collotta, G. Pau and A. V. Bobovich, "A fuzzy data fusion solution to enhance the QoS and the energy consumption in wireless sensor networks," *Wireless Communications and Mobile Computing*, pp.10, 2017. Article (CrossRef Link)

[21] Z. Zhang, Z. Hao, S. Zeadally, J. Zhang, B. Han and H. Chao, "Multiple attributes decision fusion for wireless sensor networks based on intuitionistic fuzzy set," *IEEE Access*, vol. 5, pp. 12798-12809, 2017. Article (CrossRef Link)

[22] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *Proc. of ACM MOBICOM*, pp. 56–67, 2002. Article (CrossRef Link)

[23] K. Islam, W. Shen and X. Wang, "Wireless sensor network reliability and security in factory automation: a Survey," *IEEE Transactions on Systems, MAN and Cybernetics*, vol. 42, no. 6, pp. 1243 – 1256, November, 2012. Article (CrossRef Link)

[24] M. Passing and F. Dressler, "Experimental performance evaluation of cryptographic algorithms on sensor nodes," in *Proc. of IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, pp. 882–887, October 9-12, 2006. Article (CrossRef Link)

[25] M. Akram and T. H. Cho, "Energy efficient fuzzy adaptive selection of verification nodes in wireless sensor networks," *Ad Hoc Networks*, Vol. 47, pp. 16–25, September 2016. Article (CrossRef Link)

[26] S. M. Nam and T.H. Cho, "A fuzzy rule-based path configuration method for LEAP in sensor networks," *Ad Hoc Networks*, vol. 31, pp. 63–79, August, 2015. Article (CrossRef Link)

[27] M. Akram and T. H. Cho, "Energy efficient adaptive verification node selection-based path determination in wireless sensor networks," *Symmetry*, vol. 9, no. 10, pp. 01–25, October, 2017. Article (CrossRef Link)

[28] H. Wang, C. Tang, Z. Zhao and H. Tang, "Fuzzy logic based admission control for on-grid energy saving in hybrid energy powered cellular networks," *KSII Transactions on Internet and nformation Systems*, vol. 10, no. 10, pp. 4724-4747, October, 2016. Article (CrossRef Link)

[29] P. Nayak and A. Devulapalli, "A fuzzy logic-based clustering algorithm for WSN to extend the network lifetime," *IEEE Sensors Journal*, vol. 16, no. 1, pp.137–144, January, 2016. Article (CrossRef Link)

Muhammad Ashraf received a B.E. degree in Computer Systems Engineering from Balochistan University of Engineering and Technology, Pakistan and an M.S. degree in Computer Engineering from University of Engineering and Technology Taxila, Pakistan in 2007 and 2013, respectively. He is currently a Ph.D. scholar in the College of Information and Communication Engineering at Sungkyunkwan University, South Korea. His research interests include wireless sensor networks and network security.

Tae Ho Cho received a Ph.D. degree in Electrical and Computer Engineering from the University of Arizona, USA, in 1993, and B.S. and M.S. degrees in Electrical Engineering from Sungkyunkwan University, Korea and the University of Alabama, USA, respectively. He is currently a Professor in the College of Information and Communication Engineering at Sungkyunkwan University, Korea. His research interests include wireless sensor networks, intelligent systems, modeling and simulation, and enterprise resource planning.