ISCP : An Instantaneous and Secure Clustering Protocol for Wireless Sensor Networks

Miguel Landry Foko Sindjoung, Alain Bertrand Bomgni, Elie Tagne Fute,
Clémentin Tayou Djamegni
Department of Mathematics and Computer Science, University of Dschang,
Cameroon
E-mail : miguelfoko@gmail.com, alain.bomgni@gmail.com, eliefute@yahoo.fr,
dtayou@gmail.com

Gérard Chalhoub
LIMOS-CNRS, University Clermont Auvergne, BP 10448, F-63000,
Clermont-Ferrand, France
E-mail : gerard.chalhoub@uca.fr

Received : January 2, 2018 Accepted : March 28, 2018 Published : March 31, 2018
DOI :10.5296/npa.v10i1.12574 URL : https ://doi.org/10.5296/npa.v10i1.12574

Abstract

A Wireless Sensor Network (WSN) is an Ad-hoc network populated by small hand-held commodity devices, running on batteries called stations or sensors. These sensors are deployed in an area called a perception zone in order to study one or more phenomena. Generally, the perception zone is an area where access is almost impossible for humans. Given the absence of a previously defined infrastructure, deployed sensors need to organize themselves to ensure not only their connectivity but also effective management of their residual energy and the security of data that transit to them. The residual energy management is very important since we know that communications over long distances are always very energy-consuming for sensors, which most often do not have a secondary source of energy. Multi-hop communication is generally used to connect sensors in order to ensure efficient use of their energy. This being the case, multi-hop communication can be established by partitioning the network into clusters. Subsequently, network security management is also important because most WSN must circulate confidential information. In order to avoid malicious intrusions, all operations involved must be done in a secure manner. In this paper, we propose a secure clustering protocol which connects all the sensors of the network.

Keywords : Ad-hod network, Energy efficiency, Instantaneous Clustering, Secure Clustering,

Wireless Sensor Network.

1 Introduction

The recent evolution of wireless communications technologies and the emergence of portable computing units is driving researchers to make efforts to realize the purpose of networks : access to information anywhere and anytime. When these computing units are networked, they are called ad-hoc networks to refer to the absence of pre-existing infrastructure that allows their networking. Moreover when the portable computing units are able to capture the information in their environment for a processing or a transfer of the said data, we talk about sensors, and in this case, the networking of the sensors forms an ad-hoc network also known as wireless sensor network (WSN). That is, WSN is an ad-hoc network populated by small hand-held commodity devices, running on batteries called stations or sensors.

WSNs are used in various domains (military, environmental, medical, agricultural, surveillance, etc.)[1]. This variety of application fields raises several issues that need to be addressed for items processing from these networks. Indeed, sensor networks being ad hoc networks, it is important to ensure that the deployment gives the possibility to ensure both the good interconnection between sensors (in order to minimize energy spent by the latter during transfer), and the good way to route in secure manner the captured items to the base station, which station is aimed to collect items from the sensors of the network. That being said, several works in the literature are in the direction of saving the energy of sensors to extend the lifetime of the network [1, 2, 3, 4, 5], notably, the previous works use the clustering technique to save the energy spent by sensors during their work.

The clustering technique consists in dividing the network into subgroups called clusters. Inside the clusters, a leader called Cluster Head (CH) is elected. The idea of forming clusters and electing CH is to minimize the number of long range transmissions by the sensors, so only CHs will be responsible for routing the items of the cluster to the base station.

The realization of secure protocols for wireless sensor networks is not an easy task [6], mainly because of the limited capacity (storage and computing) of sensors that populate the network. Indeed, traditional security approaches assume that the best way to set up a security policy in the network is to use a fairly robust encryption system (consisting of very long keys), but such encryption systems can not be used in wireless sensor networks because their storage and computation capacity do not allow it [6]. A security policy for sensor networks is therefore difficult to implement because it is subject to more constraints compared to the management of security in traditional networks.

Several clustering protocols for wireless sensor are presented in the literature [7, 8, 9, 10, 11, 12, 13]. While some of these protocols show a new method of partitioning the network with a view to minimize the energy consumed by the sensors [7, 8, 9], some others handle the security aspect [10, 11, 12, 13] during the clustering in order to secure communications in the network. However, since the WSN infrastructure is not previously defined and that the WSN is facing significant hardware constraints, it is important to always look for new mechanisms to improve existing ones.

To reduce the amount of energy spent by sensors during the clustering phase, the authors of [7] presented an instantaneous clustering protocol (ICP) for wireless sensor networks, but their contribution did not take into account the security aspect. In this article we improve the work of the latter by integrating the security management during the clustering.

The rest of this document is organized as follows. Section 2 presents a state of the art of clustering and secure protocols in WSNs. Section 3 presents our secure clustering algorithm. We then present a comparison between our protocol and some other protocols in section 4, after that, we present the results of our simulations in section 5. Section 6 shows the conclusion and future works.

2 Related Work

In this section, we present a state of the art on some works that exist and which are related to our problematic, that is, section 2.1 presents an inventory of the situation and some existing clustering protocols while the section 2.2 presents the state of the art on security and some secure clustering protocols in WSN

2.1 Clustering of sensors

In WSNs, the clustering of sensors is an efficient technique to ensure scalability, self organisation, energy conservation, access to medium, routing, etc. [14]. Given a WSN, clustering consists in dividing the network in small groups called clusters. If all the sensors of a cluster can communicate with each other in the same cluster, this cluster is also called a clique. Note that clustering can be done following two different approaches [15, 16, 17, 18, 19, 20, 21, 22] :

— **The Cluster Head First approach** : this approach states that the clustering process is performed by choosing the CH first, and then cluster members are chosen based on this CH.

— **The cluster First approach** : this second approach consists of first choosing the members of the cluster. These members are then charged to elect their CH.

$Heinzelman$ et $al.$ [23] present LEACH (Low-Energy Adaptive Clustering Hierarchy), a clustering protocol based on the Cluster Head First approach. Indeed, LEACH forms clusters before using the obtained CHs to route items to the base station. In that protocol, CHs are first chosen based on a probability computed internally in each sensor. In that way, each sensor may decide to become CH or not, depending on its calculated probability. When it becomes CH, the sensor makes a broadcast in the network and the non-CH sensors connect themselves to it or to another CH depending on the distance between them. The main drawback of this protocol is the number and the type of communications in the network. Mainly, the CHs communicate directly with the base station, which has the disadvantage of exhausting their residual energy faster.

$Lindsey$ et $al.$[24] present a Power-Efficient GAthering in Sensor Information Systems (PEGASIS) that improves the protocol presented in [23]. They propose a voracious algorithm that, once executed by sensors, arranges them in a chain within which a single sensor transmits to the base station. In this case, the number of long-distance communication is reduced and therefore, energy consumption is reduced. However, authors assume that all sensors know the entire network, which in our mind is an unrealistic condition.

$Wadaa$ et $al.$ [25] defined a clustering protocol for WSNs. In that protocol, the network is seen as a circle which can be divided into angular sectors and coronas, thus, a cluster is the intersection between a corona and an angular sector. At the end of the clustering algorithm, all sensors are in a single cluster. However, the security aspect is not assured during the clustering.

Sun et $al.$[26] proposed a Cluster First clustering algorithm which uses the messages exchanged between sensors to form cliques. The particularity of this protocol is that after the clustering, all the sensors are in a single clique (thus no isolated sensors), but, this protocol shows some limits, particularly concerning the number of messages exchanged in the network ; indeed, this number is very high which is detrimental to the lifetime of the entire WSN.

Recently, $Linghe$ et $al.$[7] have proposed a Clustering algorithm based on the Cluster Head First approach. Due to a predistributed probability in all the sensors in the pre-deployment phase, each sensor executes an algorithm to determine which cluster head to join. Using the same principle as LEACH[23] and PEGASIS[24], this protocol offers a better performance compared to these last two protocols which are reference protocols in WSNs with regard to Clustering. Note that the number of messages exchanged during the clustering presented by $Linghe$ et $al.$ is very negligible compared to that of Sun et $al.$. However, the security aspect is not taken into account.

In the literature, several other clustering protocols for WSNs have been presented, we can enumerate the works done in [8, 9, 27] which are protocols based on the Cluster Head First approach while the protocols presented in [2, 3] are based on the Cluster First approach. Unfortunately, none of these protocols considers the security aspect in its implementation.

2.2 The security aspect in Wireless Sensor Networks

WSNs are vulnerable networks because of their nature. Indeed, being ad hoc networks, they are easily attackable. That is why it is important to ensure the security of the data that transit while setting up communication mechanisms for these networks. Unfortunately, to our knowledge, very little works focused on this security aspect, this results in the vulnerability of information that circulate in the network.

Note at this level that the attacks faced by WSNs are of various kinds [28, 29] :

— Denial of services : this type of attack comes through the help of external sensors. These sensors use the interference radio frequencies from network's sensors to flood the WSN with useless messages.

— Node Tampering : in this case, an adversary gets hold of a sensor node physically and gains access to all data information and important cryptographic material. When a node is tampered, it is compromised physically and it can be used to listen to communications, interrupt them, intercept, modify and fabricate messages.

— Node Compromise : an adversary can exploit a hole in the system software of a sensor node to gain control of the node. After gaining control of the sensor node, the adversary can access all the data and information stored on the sensor node. Cryptographic keying material are also lost. Compromised node can listen to the communication between other nodes, interupt communications, intercept, modify and fabricate messages.

— Selective Forwarding : a false or compromised node is used to create a black hole in the target sensor network. False or compromised node deliberately drops data packets to disrupt network operations. This occurs in the network level.

— Sinkhole attacks : this is similar to selective forwading except that it is not a passive attack. In this case, traffic is attracted towards the compromised or false node.

— Sybil attacks : malicious node presents multiple identities to the sensors of the network, either by creating them or by stealing the identities of normal nodes.

— Wormhole attacks : two distant malicious nodes are used to create a wormhole in the target sensor network. Both malicious nodes have an out-of-band communication channel. One node is placed near the sensor nodes. It advertises shortest path to the sink node through the other one, which is placed near the sink node. This creates sinkholes and routing confusions in the target sensor network.

— Hello flood attacks : a malicious node plays or replays a hello packet with a high signal strength in the target sensor network. High signal strength makes all normal nodes think that the malicious node is their neighbour. It then creates a wormhole. Also, other normal sensor nodes loose their energy in replying to the hello packet.

In [30], $Perrig$ et $al.$ present $\mu Tesla$ protocol, a variant of $Tesla$ protocol adapted for WSNs. The latter uses a cryptographic system based on digital signature, which requires enormous calculations, and therefore, if a sensor wants to decrypt an encrypted message with $Tesla$, it must make a huge effort in computing time, and this may considerably decrease its energy level. $\mu Tesla$ requires the base station and sensor nodes to be synchronized. To broadcast, the base station calculates a MAC on the item with a secret key at this time. When the sensor receives the item, it can verify that the encryption key has not yet been broadcast by the base station. As long as the station is convinced that the key is only known by the base station, it stores the item in a buffer of its memory while waiting for the base station to broadcast the encryption key to all the receivers. Upon receipt of the key disclosed by the base station, the sensors check the correctness and then use the key to decrypt the item.

$Oliveira$ et $al.$ [10] propose SecLEACH, one of the first protocols that use the random predistribution of encryption keys to secure the clustering of a WSN. In their protocol, sensors are pre-assigned some keys for authentication before their deployment. The secure protocol is based on LEACH protocol. It cannot prevents internals attacks as Selective Forwading or Sinkhole attacks.

$Wang$ et $al.$ in [31] propose a new method to detect sybil attacks in a WSN. Their method is based both on the power of the signal that a sensor receives at a given moment, and the information concerning it that the CH knows.

Recently, $Mansour$ at $al.$ in [6] designed several authentication protocols in WSN. They propose different multi-hop node authentication protocols and a mechanism to detect possible attack based on their evaluation results. The originality of their work resides in the fact that it suites large scale multi-hop WSN, which is due to the limited number of cryptographic operations regardless of the number of hops separating the communicating sensor nodes.

$Selvi$ et $al.$ [13] propose a security based clustering algorithm in order to increase the lifetime of network and to protect data. They use a special node called Mobile Data Collector (MDC) to collect and aggregates the data from cluster heads and forwards these data to the Base Station. The data are protected from intruders by authenticates the cluster head using a shared secret key and the digital signature.

The specificity of the protocol that we set up in this article is that it is an instantaneous secure clustering, in other words, the clusterring is carried out in constant time and integrates the security management. Since our protocol performs a clustering, it is essential to be reassured that the sensors of the obtained clusters are authenticated, otherwise we risk having malicious intrusions in the network and therefore the data provided by the sensors may not provide the expected results . Based on the use of a base station to allow a sensor to join the network, we give solution to the followings attacks in our protocol : Sinkhole attacks, Selective forwading, Sybil attacks and Hello flood attacks.

3 An Instantaneous and Secure Clustering protocol for WSNs

The aim in this section is to present our energy-efficient and secure clustering protocol in a WSN. The proposed protocol will help to facilitate the multi-hop communications in the network.

Figure 1 present the architecture that we are going to set up.

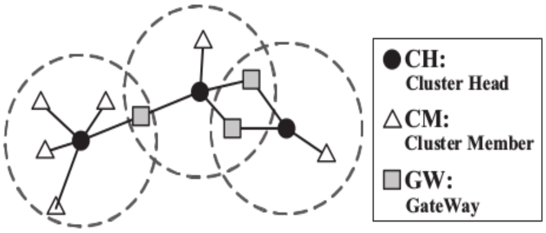

FIGURE 1: Example of cluster topology in one hop communication [7]

Thus, we must establish a connection between all the sensors of the network. This connection establishment will have to respect some fundamentals :

1. The clustering procedure should have a very fast speed of execution to avoid the sensors to spend too much energy.
2. At the end of the clustering, all the sensor nodes should be connected to the others : it must not have an isolated sensor.
3. Malicious intrusions should be avoided, in other words, the clustering process should be secured.

3.1 Prerequisites

— The WSN is populated by n sensor nodes.
— Each sensor node knows the identity of all the others in the network.
— $KS(S, D)$ is the shared session key between sensors S (source) and D (destination).
— n_i is a HELLO message initiated by the node of $ID = i$.
— To avoid collisions in the network, the CSMA protocol is used to manage access to the medium.
— G is a public generator point of the elliptic curve[6, 32].
— We will use the hybrid cryptographic system to secure communications in the network. This system uses both the symmetric cryptography and the asymmetric cryptography. Its use in our protocol is presented as follows :

1. **The asymmetric cryptographic system** : this system is a quite robust system based on the use of private keys of each sensor in the network. However, it consumes a lot of material resources (important processing time, use huge amount of sensor's memory space, etc.). That is why we will only use it when sensors want to join the network. The operation is as follows :

   At the pre-deployment, the base station B assigns a unique asymetric key pair generated from elliptic curve cryptography (ECC) to each of the network's sensor.

For example, sensor A will have a unique key pair (P_A, K_A) (where K_A is randomly chosen in an interval $[1, m]$ with m a parameter of the elliptic curve used). K_A is then considered as the private key of A while P_A is its public key obtained by the scalar multiplication $P_A(x_A, y_A) = K_A * G(x, y)$ [33]. B also assigns its public key P_B (generated using the same principle) to all the sensors. It is using P_B that the sensors will be able to establish a secure communication with B using Diffie-Hellman without interaction (DHWI) [33]. No message is exchanged at this level.

After deployment, the association of sensors to the network is the next step. For this, each sensor must send a request to the base station to ask its association with the existing network. This request is encrypted by the public key of the base station, that is, only the base station will be able to decrypt the message sent to it, namely using its private key. Once the base station has identified the sensor transmitting the message, it creates a session key that will now allow communication between it and the sensor, and then sends that session key to the requesting sensor by always encrypting the message with the public key of the said sensor. Upon receipt of this message from the base station, the sensor uses its private key to decrypt the message sent by the base station. After that, all the communications between the base station and the requesting sensor must be done using the recently shared session key. This principle of association of a sensor to a network is more explained in [33]

2. **The symmetric cryptographic system** : we use this system to allow direct communication between network's sensors. This is mainly advantageous for these sensors since they must now use the symmetric keys generated when they join the network, and therefore they will not longer spent their energy to compute the encryption key as it is done in the asymmetric cryptographic system. Note that to join the network, two scenarios are presented to the sensors : either the base station is in the sensor's communication range and it makes a direct request to the base station, or the the base station is not in its communication range. In the last case, the request will be forwarded from sensor to sensor until it arrives to the base station. When a sensor D receives a message for the first time from the sensor S, it sends a request to the base station to reassure itself that S is authorized to communicate in the network. The response to this braodcast is accompanied by the session key $KS(S, D)$ if S is authorized and in this case, this key is also sent to S. This way, S and D can now communicate using their session key.

3.2 Sequence of clustering

Base on the clustering protocol presented in [7], we present a new clustering protocol by introducing the security aspect. Indeed, $L.\ Kong$ et $al.$[7] present a fast and energy efficient clustering protocol for WSNs. This protocol aims to address three main challenges :

The first challenge is to determine the CH without exchanging a message. Thereby, using certain information known in advance by sensors, (i.e the total number of network's sensor, the transmission radius of each sensor, and the surface covered by the network), $L.\ Kong$ et $al.$ deduce the maximum number of CHs that the network needs, this is done using a stochastic distribution. Then, they introduce a redundancy coefficient of the CH to face the unknown positions of the sensors during deployment and finally they assign a probability to all sensors in the network. This probability is calculated with n and the redundancy coefficient. It is using the previous probability that a sensor can locally determine if it can be CH or not.

The second challenge is to be ensured that the obtained CH have successfully informed the Cluster Members (CM) and Gateways (GW) attached to them. To do this, authors introduce a period to allow the CH to broadcast their IDs to their members. This period includes the minimum number of slots adequate for a CH to transmit its ID without a risk of collision.

The third challenge is to set up a very light protocol to be easily implemented practically. Results presented in [7] show that this challenge has been raised.

In addition to these challenges, authors identify another challenge. Indeed, if the CH are close to the same geographical region, there is a risk of finding some CM without CH to which they are associated. That is why as soon as a station is presupposed to be CH, it checks if it is not close to another CH, in which case it associates itself directly to the latter, either as CM or as GW. Moreover, a mechanism to verify that there are no isolated sensors is presented.

Below, we present our secure protocol :

We assign the probability $P_{CH} = \beta m/n$ to become CH in each sensor, where :

— m is the number of CH estimates to cover the entire network, $m = C(logn)\frac{a^2}{r2}$ with :
  - — a^2, the surface covered by the network ;
  - — r, the transmission radius of each sensor : in our study, we consider that the sensors transmission radius is perfect, in order to facilitate our simulations, but we are aware that in practice the transmission range cannot be perfect.
  - — $\frac{1}{4logn} \leq C \leq \frac{1}{logn}$, is a random parameter.

— β is a constant coefficient considers as a number of CHs in a cell of the figure 2

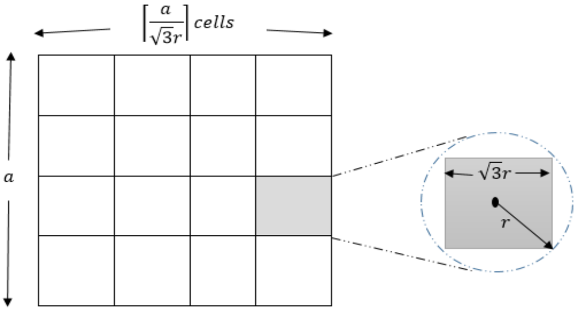

FIGURE 2: Mathematical description of the perception zone[7]

After the deployment, each sensor executes the following steps :

1. A sensor becomes a candidate CH with the probability P_{CH} and executes the algorithm 3.1 ;

   In this case, it generates a random number k, between 1 and T, and it listens for the network communications up to the slot number $k-1$. If during the listening phase it does not receive a message from its neighbors, it becomes CH and it informs its neighbors about its new status by a message encrypted at the k^{th} slot. The encryption key is the one assigned by the base station at the deployment time.

   If a message encrypted by $K_{DH}(I, S)$ coming from a sensor I arrives to sensor D, the latter uses the key assigned on it at the pre-deployment to check if I is allowed to communicate in the network, in which case it becomes a CM associated with I by a secure notification to I.

If several secure messages arrived to sensor D, it checks if the received message come from the authenticated sensors in the network, and for those of them that are authenticated, D becomes GW.

If between the slots $k + 1$ and $T - 1$ several secure messages arrive to the sensors D, the latter executes the previous procedure to become GW serving the authenticated sensors that sent him messages.

2. A sensor becomes CM/GW with the probability $1-P_{CH}$ and it executes the algorithm 3.2. This algorithm is quite similar to algorithm 3.1, except that the sensors which execute this one can only be either GW or CM. In this algorithm, the sensor listen from the first to the T^{th} slot. If a sensor receives a single secure message, it becomes CM for the sensor (GW) that sent it the message. If it reveives several messages, it is the procedure to become GW of algorithm 3.1 which is executed. On the other side, it can happen that it does not receive any message, in this case, the sensor executes the compensation mechanism which consists to become CH and to execute algorithm 3.1. This case occurs when the sensor is isolated.

Where :

— T is the minimum number of slots that can allow any CM/GW to be successfully connected to a CH

— Δ is the number of slots needed to run the compensation mechanism (which mechanism allows sensor to rerun the protocol until it connects to the network)

— $\omega()$ is a distribution function determined by the present status of the sensor (its residual energy, its previous roles, its position, the number of its neighbors, etc.).

The choice of the preceding parameters is justified in [7]. Bellow, D is the sensor that executes the algorithm and I is the sender of message.

Algorithme 3.1 : Secure_ICP_CH_Algorithm(T, Δ)

```
1  Begin
2      Set $k$ to be a random number $[1, T]$ with seed $\omega()$;
3      /* become a $CH$ after $k$ or abdicate within $k$ time slots*/;
4      Listening from $1^{st}$ to $(k-1)^{th}$ time slots;
5      Begin
6          If hear no message during these $(k-1)$ slots Then
7              Be $CH$, send the message $\{n_I, I, S\}_{K_{DH}(I,S)}$ at $k^{th}$ slot, quit
                ALgorithm3.1;
8              /* Where $I$ is the $ID$ of the sensor that broadcasts and $S$ is the $ID$ of the
                BS*/;
9          End If
10         If hear the message $\{n_I, I, S\}_{K_{DH}(I,S)}$ from $I$ Then
11             Send the message $\{\{n_I, I, S\}_{K_{DH}(I,S)}\}_{K_{DH}(D,S)}$ to $S$.;
12             If $I$ is allowed to communicate in the network (BS will have sent $KS(D, I)$
                to $D$ and $I$ repectively) Then
13                 Return the message $\{n_I, D\}_{KS(D,I)}$ to $I$, be a $CM$ of the cluster where
                    $I$ is $CH$;
14             End If
15         End If
16         If hear multiple messages $\{n_{I_l}, I_l, S\}_{K_{DH}(I_l,S)}$ from $CHs(I_1, I_2, etc.)$ Then
17             For $l = 1; l < n; l = l + 1$ Do
18                 Send the message $\{\{n_{I_l}, I_l, S\}_{K_{DH}(I_l,S)}\}_{K_{DH}(D,S)}$ to $S$.;
19                 If $I_l$ is allowed to communicate in the network Then
20                     Return the message $\{n_{I_l}, D\}_{KS(D,I_l)}$ to $I_l$ and become $GW$ for $I_l$;
21                 End If
22             End For
23         End If
24         /* if not, serve as $CM$ or $GW$ in the rest time slots*/;
25     End
26     Listening from $(k+1)^{th}$ to $T^{th}$ time slot;
27     Begin
28         If hear more messages $\{n_{I_l}, I_l, S\}_{K_{DH}(I_l,S)}$, $1 \leq l \leq n$ Then
29             For $l = 1; l < n; l = l + 1$ Do
30                 Send the message $\{\{n_{I_l}, I_l, S\}_{K_{DH}(I_l,S)}\}_{K_{DH}(D,S)}$ to $S$.;
31                 If $I_l$ is allowed to communicate in the network Then
32                     Return the message $\{n_{I_l}, D\}_{KS(D,I_l)}$ to $I_l$ and become $GW$ for $I_l$ ;
33                 End If
34             End For
35         End If
36         /* Keep listening during the compensation period*/;
37     End
38     Listening from $(T+1)^{th}$ to $(T+\Delta)^{th}$ time slots;
39     Begin
40         Execute line 28 to 35
41     End
42 End
```

Algorithme 3.2 : Secure_ICP_CM/GW_Algorithm(T, Δ)

```
Begin
    /* Serve as $CM/GW$ within $T$ time slots */;
    Listening from the $1^{st}$ to $T^{th}$ slots;
    Begin
        If hear the message $\{n_I, I, S\}_{K_{DH}(I,S))}$ Then
            Send the message $\{\{n_I, I, S\}_{K_{DH}(I,S)}\}_{K_{DH}(D,S)}$ to $S$.;
            If $I$ is allowed to communicate in the network Then
                Return the message $\{n_I, D\}_{KS(D,I)}$ to $I$, be $CM$ of the cluster of $I$;
            End If
        End If
        If hear more messages $\{n_{I_l}, I_l, S\}_{K_{DH}(I_l,S))}$ Then
            For $l = 1; l < n; l = l + 1$ Do
                Send the message $\{\{n_{I_l}, I_l, S\}_{K_{DH}(I_l,S)}\}_{K_{DH}(D,S)}$ to $S$.;
                If $I_l$ is allowed to communicate in the network Then
                    Return the message $\{n_{I_l}, D\}_{KS(D,I_l)}$ to $I_l$, be $GW$ for $I_l$;
                End If
            End For
        End If
    End
    If hear no message until $T^{th}$ slot Then
        Become a $CH$, do lines 2-35 of algorithm 3.1;
    End If
End
```

Lemma 1 ***Finally, our clustering protocol takes a constant time $2T$ slots to complete while each station broadcasts not more than m messages. In addition, all stations remain awake for at most $2T$ slots.***

Proof 1 *The demonstration of this lemma is simple. Indeed, we have three types of sensor in the network : CM, CH and GW. Depending on the status of the sensor, we face the following situations :*

— *If the sensor is CM, it only braodcasts a single message during the clustering (the message to join its CH). It also means that it receives a single message during the whole clustering. On the other hand, if it is bode to be CM, it remains awake during the whole protocol ($T + \Delta$ slots) because it can receive other messages at any moment to become a GW.*

— *If the sensor is GW, in the worst case it has to receive m messages (i.e it serves all CHs of the network) and it has broadcast at most m messages (messages to inform the CH to which it is linked that it is their GW). Since, it is going to receive messages at any time from the CH, it is therefore obliged to stay awake during the whole phase of clustering, that is $T + \Delta$ time slots.*

— *If the sensor is CH, then it has broadcast exactly one message (to inform its CM and GW). This also means that it has received $n - 1$ messages in the worst case (if the network has a single cluster, then all $n - 1$ other sensors in the network are connected* *to the single CH). Since a sensor bode to be CH can receive several messages at any moment to become GW, the CHs stay awake during the whole clustering, that is $T + \Delta$ time slots.*

From the above, we firstly note that the sensors remain awake during $T+\Delta$ slots to complete the clustering. Secondly, we note that the maximum number of diffusion made by a sensor during the clustering is m messages.

The presentation of results obtained in [7] shows that unlike LEACH[23] for example where the time taken by the protocol to complete depends on the number of sensors in the network, ICP protocol to which we add the security aspect ends in constant time $T = 160ms$ when sensors varying from 2000 to $10,000$. In addition, these results also show that $\Delta = T$. This allows us to conclude that the sensors remain awake for $2T$ slots while executing our protocol, which completes our demonstration.

4 Comparative study of our protocol with some others protocols existing in literature

In this section, we make a comparative study between our protocol and five other protocols. LEACH[23] and PEGASIS[24] were chosen for comparison because they are reference protocols in WSNs, but they are not secure. The protocol of Sun et $al.$[26] is chosen here because it is a secure clustering protocol just like the protocol we propose. ICP[7] is used because it is the one that we secure.

The table bellow shows a comparison between our protocol and the previous protocols.

Table 1. Comparison between our protocol and some others

Protocols	Security Management	Distribution of CHs in the perception zone	Clustering time
LEACH [23]	NO	Not equitable	Depend on the number of sensors
PEGASIS [24]	NO	Not equitable	Depend on the number of sensors
by SUN[26]	YES ($\mu Tesla$ and PKI)	Equitable	Depend on the number of sensors
by Selvi[13]	YES ($Digital$ $Signature$)	Not Equitable	Depend on the number of sensors
ICP[7]	NO	Equitable	Constant
Our protocol	YES (ECC)	Equitable	Constant

From the previous table, we can do the following observations :

— Many protocols do not integrate the security aspect while setting up the clustering protocol, it is the case in [7, 23, 24]. However, to integrate this aspect, several posibilities are offered : the Public key Infrastructure PKI, $\mu Tesla$, ECC, etc. We opted to use ECC in our protocol.

— The distribution of CHs in the perception zone is not equitable in most of the proposed protocols. This distribution is too important since that the sensors often use the CH to transmit items to the BS. That is, more the CH is away from a sensor, more the later spends energy to transfer items to the CH. Unlike protocols presented in [23, 24, 13], our protocol and the ones presented in [7, 26] assure a good distribution of CHs in the perception zone.

— Finally, we note that the time taken by our protocol is constant, just as the one presented in [7]. Contrary to the last two protocols, all the others in the table have the execution time which depend on the number of sensors in the network.

Another memorable point not presented in the previous comparative table is the management of communication media. Indeed, media management is an important factor in WSNs because the network is often made up of an important number of sensors. So, it is important to manage the channel access used by sensors, otherwise, there is a risk of collision in the network, and that situation could lead to the loss of sensitive information. As LEACH, we manage the channels in our protocol by using CSMA. Note that this management of channels access makes it possible to reconcile our results to the reality of sensors after they are deployed in a real life application. Unfortunately, this aspect is not managed in the clustering protocols presented in [7, 24, 26] and it is why we can conclude that our protocol is more realistic than these ones.

In the next section, we present the results of the simulations obtained by comparing our clustering protocol to some others.

5 Simulation results

We present the results of our simulations in this section. These simulations were performed in a laptop (Dual Core, 4 Go RAM, Ubuntu 14.04) with the network simulator NS 2.35. We assume that once the sensors are deployed, they are static and they have the same transmission radius. The experiment takes place in an area square with $L = 100$ as side. Each curve is an average of 100 experiments. A sensor is close to another if it can listen to a HELLO message sent by this one. In our implementation, access to the medium is managed such that a node can only receive one message at a given moment.

In our simulations, we highlight the instantaneous aspect of our protocol by presenting the energy conservation due to this effect (section 5.1), in which we highlight the cost of security during the clustering. Then, in section 5.2 we make a comparative study between our protocol and that of sun et al. In the latter case we are no longer referring to LEACH because at the launch of LEACH we have to set the number of clusters we want to get, so its inclusion in the comparison in this phase would be no longer interesting. On the other hand, in section 5.2, we are no longer referring to ICP because it gives the same results as ISCP.

5.1 Evolution of the residual energy of sensors

The energetic model we use is similar to the one used in [23],

$$E = ET + ER = N * n(e_t + e_{amp} * d^2) + e_r * n$$

where ET and ER are respectively the energy used for the transmission and the reception in the network. The energy dissipated by the transmitter, the amplifier and the receiver are respectively expressed by e_t, e_{amp} and e_r. Moreover, d is the euclidian distance between nodes, N is the mitigation parameter $(2 \leq N \leq 4)$ and n is the number of items. Thus, based on this model, we value ET to $0.02J$ and ER to $0.01J$ with initial energy of sensors to $10J$ and then, we have the figure 3. This figure presents the evolution of residual energy of sensors during the clustering according to the number of sensors in the network. Some observations can be made there. Namely, note that the residual energy of sensors decreases considerably while the

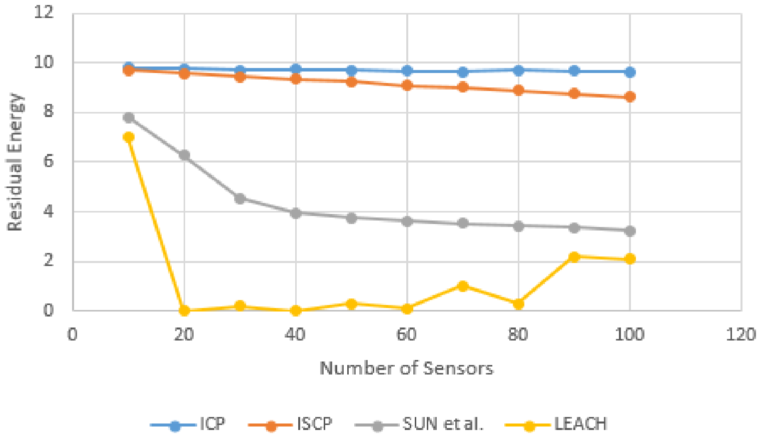

FIGURE 3: Evolution of the residual energy of sensors.

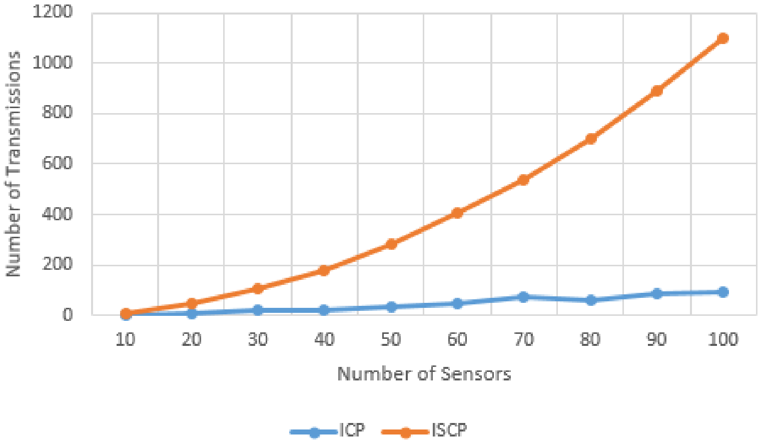

FIGURE 4: Comparison of the number of Transmissions between ICP and $ISCP$.

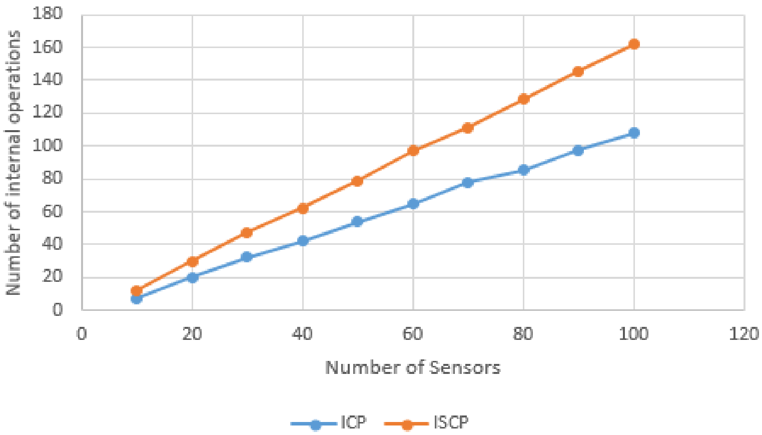

FIGURE 5: Comparison of the number of internal operations between ICP and $ISCP$.

number of sensors grows in the protocol of Sun et $al.$. This remark is also made for LEACH [23] protocol. We also note that the residual energy of sensors that execute ICP and $ISCP$ is too high compared to the first two protocols mentioned before. However, $ISCP$ consumes more than ICP. The reason is simple, although $ISCP$ uses the same principle as ICP, it includes the security aspect which consumes too. We can remark the consumption due to the security aspect in the figure 4 where the number of transmission caused by that aspect increases the number of thansmissions in $ISCP$ compared to the one of ICP.

Another important effect of security management in our protocol can be seen in Figure 5. In fact, this figure shows the number of internal operations performed by ICP and $ISCP$, we note that the difference between the number of operations performed by these two protocols is in favor of ICP as the number of sensors increases. Thus, the figures 4 and 5 clearly justify the fact that the residual energy of the ICP sensors is greater than that of the $ISCP$ sensors in the figure 3. However, the residual energy of sensors running ICP and $ISCP$ is larger than the one of sensors that ran [23] or [26]. This leads us to conclude that our protocol is more energy-efficient than these of Sun [26] and the reference protocol $LEACH$[23].

5.2 *Evolution of the number of clusters according to the sensors*

Figure 6 is the curve showing the evolution of the average number of clusters while the number of sensors increases. This curve compares $ISCP$ to the protocol of Sun et $al.$. We note that the number of Cluster obtained while running $ISCP$ is lower than to that obtained using the approach presented in [26]. This is advantageous in that there will be fewer sensors that will play the role of CH and therefore, fewer sensors will spend more energy in our protocol. This for sure is a real asset for the extension of the lifetime of the network.

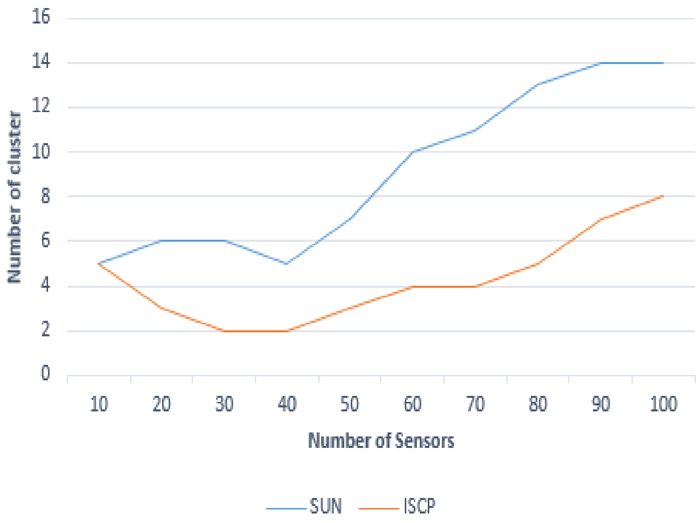

FIGURE 6: Evolution of the number of clusters according to the sensors

6 Conclusion and future works

Wireless sensor networks (WSNs) are ad-hoc networks. They are often used in danger zones (earthquake, volvano, etc.). these networks are subject to energy constraints. Since that the lifetime of the network depends on that of the sensors, it is important to make sure that sensors do not spent their energy uselessly, this is done when setting up routing protocols for WSN. A way used to minimize the energy consumption of sensors is to partition the network in clusters so that, sensors do not make communications over long distances. These networks are also vulnerable due to the absence of a pre-defined infrastructure. That is the reason for which it must be reassured that communications are secured. In this article, we present an Instantaneous (i.e energy-efficient) and secure clustering algorithm. Simulation results show that our protocol consumes less energy than other clustering protocols in the literature (the reference protocol presents by $Heinzelman$ et $al.$[23] (LEACH) and the protocol introduced by Sun et al[26]. In addition, although our protocol is less energy efficient than ICP[7]), it is secure ICP contrary to this later. The additional energy consumption is the overhead caused by the security mechanism put in place.

Although we perform an instantaneous and secure clustering which is energy-efficient, our protocol still has some limitations. Indeed, if a CH breaks down or generally if there is displacement of a sensor, no mechanism describes how the re-election of the CH is done or there is no mechanism that ensures the displaced sensor to be in a cluster. Another limit is that of the lack of a hierarchy between the CHs for the inter-cluster communication. It would be equally interesting to improve the security mechanism so that it can also be able to respond to problems of denial of service, node compromise, node tampering and wormhole attacks.

References

[1] M. Tewari and K. S. Vaisla, "Adec-energy proficient four-level deterministic hierarchical clustering protocol for wireless sensor network," *International Journal of Future Generation Communication and Networking*, vol. 9, no. 5, pp. 231–242, 2016, http://dx.doi.org/10.14257/ijfgcn.2016.9.5.23.

[2] A. Kaur and A. D. Bharti, "Energy efficient clustering scheme for elongating the life time of wireless sensor networks," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 5, no. 10, Oct. 2015.

[3] N. K. Chaubey and D. H. Patel, "Energy efficient clustering algorithm for decreasing energy consumption and delay in wireless sensor networks (wsn)," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 4, no. 5, May 2016.

[4] V. Kumar, S. B. Dhok, R. Tripathi, and S. Tiwari, "A review study of hierarchical clustering algorithms for wireless sensor networks," *International Journal of Computer Science*, vol. 11, pp. 92–101, 2014.

[5] R. Kalaiprasath, R. Kalaipriya, and N. Arulkumaran, "Efficient assessment on hierarchical clustering algorithms in wireless sensor networks," *International Journal of Science and Research*, vol. 5, pp. 1422–1425, Feb. 2016.

[6] I. Mansour, G. Chalhoub, and P. Lafourcade, "Evaluation of secure multi-hop node authentication and key establishment mechanisms for wireless sensor networks," *Journal of Sensor and Actuator Networks*, vol. 3, pp. 224–244, 2014, http://dx.doi.org/10.3390/jsan3030224.

[7] L. Kong, Q. Xiang, X. Liu, X.-Y. Liu, X. Gao, G. Chen, and M.-Y. Wu, "Icp : Instantaneous clustering protocol for wireless sensor networks," *Computer Networks*, pp. 144–157, Jan. 2016, http://dx.doi.org/10.1016/j.comnet.2015.12.021.

[8] A. P. Abidoye, N. A. Azeez, A. O. Adesina, and K. K. Agbele, "Ancaee : A novel clustering algorithm for energy efficiency in wireless sensor networks," *Wireless Sensor Network*, vol. 3, pp. 307–312, Sept. 2011, http://dx.doi.org/10.4236/wsn.2011.39032.

[9] S. K. Singh, M. P. Singh, and D. K. Singh, "Energy efficient homogenous clustering algorithm for wireless sensor networks," *International Journal of Wireless and Mobile Networks (IJWMN)*, vol. 2, no. 3, 2010, http://dx.doi.org/10.5121/ijwmn.2010.2304.

[10] L. B. Oliveira, H. C. Wong, M. Bern, R. Dahab, and A. A. F. Loureiro, "Secleach - a random key distribution solution for securing clustered sensor networks," *Network Computing and Applications*, 2006, http://dx.doi.org/10.1109/NCA.2006.48.

[11] A. C. Ferreira, M. A. Vilaa, L. B. Oliveira, E. Habib, H. C. Wong, and A. A. Loureiro, "On the security of cluster-based communication protocols for wireless sensor networks," *In Proc. 4th IEEE International Conference on Networking (ICN05)*, pp. 449–458, 2005.

[12] P. Schaffer, K. Farkas, A. Horváth, T. Holczer, and L. Buttyán, "Secure and reliable clustering in wireless sensor networks : a critical survey," *Computer Networks*, vol. 56, pp. 2726–2741, July 2012.

[13] G. V. Selvi and R. Balasubramanian, "Secure based clustering algorithm for wireless sensor networks," *International Journal of Computer Applications*, vol. 117, no. 1, May 2015.

[14] S. Banerjee and S. Khuller, "A clustering scheme for hierarchical control in multi-hop wireless networks," *Proceedings of the 20th IEEE International Conference on Computer Communications*, vol. 3, pp. 1028–1037, 2001.

[15] A. B. Bomgni, E. T. Fute, M. L. F. Sindjoung, and C. T. Djamegni, "A tree-based distributed permutation routing protocol in multi-hop wireless sensors network," *Wireless Sensor Network*, vol. 8, pp. 93–105, 2016, http://dx.doi.org/10.4236/wsn.2016.86010.

[16] A. B. Bomgni and J. F. Myoupo, "A deterministic protocol for permutation routing in dense multi-hop sensor networks," *Wireless Sensor Network*, vol. 2, pp. 293–299, 2010, http://dx.doi.org/10.4236/wsn.2010.24040.

[17] G. Raghunandan and B. Lakshmi, "A comparative analysis of routing techniques for wireless sensor networks," *National Conference on Innovations in Emerging Technology*, pp. 17–22, Feb. 2011.

[18] S. Basagni, "Distributed clustering for multi hop wireless network," *Proceedings of the IEEE International Symposium on Wireless Communications*, pp. 41–42, June 1999.

[19] A. McDonald and A. Zanati, "A mobility-based framework for adaptive clustering in wireless ad hoc networks," *IEEE Journal on Selected Areas in Communications*, vol. 17, pp. 1466–1487, 1999, http://dx.doi.org/10.1109/49.780353.

[20] A. Amis, R. Prakash, T. Vuong, and D. Huynh, "Max-min d-cluster formation in wireless ad hoc networks," *INFOCOM*, vol. 1, pp. 32–41, 1999.

[21] D. Baker, A. Ephremides, and J. Flynn, "The design and simulation of a mobile radio network with distributed control," *IEEE Journal on Selected Areas in Communications*, vol. 2, pp. 226–237, 1984, http://dx.doi.org/10.1109/JSAC.1984.1146043.

[22] O. Younis and S. Fahmy, "Distributed clustering in ad-hoc sensor networks : A hybrid, energy-efficient approach," *INFOCOM*, vol. 1, 2004, http://dx.doi.org/10.1109/INFCOM.2004.1354534.

[23] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," *Proceedings of the 33rd Hawaii International Conference on System Science (HICSS '00)*, pp. 1–10, Jan. 2000.

[24] S. Lindsey and C. S. Raghavendra, "Pegasis : Power-efficient gathering in sensor information systems," *IEEE Aerospace Conference Proceedings*, vol. 3, pp. 1125–1130, 2002.

[25] A. Wadaa, S. Olariu, L. Wilson, M. Eltoweissy, and K. Jones, "Training a wireless sensor network," *Mobile Networks and Applications*, vol. 10, pp. 151–168, 2005.

[26] K. Sun, P. Peng, P. Ning, and C. Wang, "Secure distributed cluster formation in wireless sensor networks," *22nd Annual Computer Security Applications Conference,Las Vegas*, vol. 10, pp. 131–140, 2006.

[27] W. JERBI, A. GUERMAZI, and H. TRABELSI, "A novel clustering algorithm for coverage a large scale in wireless sensor networks," *International Journal on Computational Science and Applications (IJCSA)*, vol. 6, no. 2, April 2016, http://dx.doi.org/10.5121/ijcsa.2016.6201.

[28] S. M. K. Raazi and S. Lee, "A survey on key management strategies for different application of wireless sensor networks," *Journal of Computing Science and Engineering*, vol. 4, pp. 23–51, 2010.

[29] K. Chatterjee, A. De, and D. Gupta, "A secure and efficient authentication protocol in wireless sensor network," *Wireless Pers Commun*, vol. 81, pp. 17–37, 2015, http://dx.doi.org/10.1007/s11277-014-2115-2.

[30] A. Perrig, R. Szewczyk, V. Wen, and D. C. J. D. Tygar, "Spins : Security protocols for sensor networks," *Mobile Computing and Networking,Rome, Italy*, 2001.

[31] J. Wang, G. Yang, Y. Sun, and S. Chen, "Sybil attack detection based on rssi for wireless sensor network," *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 2684–2687, 2007.

[32] M. T. Wankhede-Barsagade and S. A. Meshram, "Use of elliptic curves in cryptography : An overview," *IJMS*, vol. 11, no. 3–4, pp. 289–296, Dec. 2012.

[33] I. Mansour, "Contribution á la sécurité des communications des réseaux de capteurs sans fil," Ph.D. dissertation, Université Blaise Pascal, June 2013.

Copyright Disclaimer