LQI-DCPSec: Secure Distributed d-Cluster Formation in Wireless Sensor Networks

Cherif Diallo(✉) and Maimouna Tedy Sow

Laboratoire Algèbre, Cryptographie, Codes et Applications (ACCA),
UFR Sciences appliquées et de Technologies (UFR SAT),
Université Gaston Berger, BP 234, Saint-Louis, Senegal
cherif.diallo@ugb.edu.sn

Abstract. In wireless sensor networks (WSN), grouping of small group nodes, called clusters, is an effective technique for facilitating scalability, self-organization, energy saving, access channels, routing, data aggregation, etc. Several clustering protocols have been proposed for WSN. Depending on the order in which the formation of clusters and the election of cluster leaders are performed, these protocols can be subdivided into two categories: the Leader-First (LF) and the Cluster-First (CF) approaches. In LF approaches, clusterheads are first elected according to certain metrics (e.g., degree of connectivity, remaining energy), and then agree to assign the other nodes to different clusters. While in CF techniques, all sensors first form clusters, and then each cluster chooses its leader. In previous work, we proposed LQI-DCP, which is a non-secure distributed protocol for multihop clustering with an LF approach. During the clustering process, one or more malicious nodes could attempt to disrupt the cluster leader election by electing illegitimate nodes. In this paper we propose LQI-DCPSec which is a secure version of LQI-DCP that we will compare, in terms of energy consumption, to SEFA which is another secure clustering algorithm using an LF approach. Simulation results show that LQI-DCPSec is more energy-efficient than SEFA.

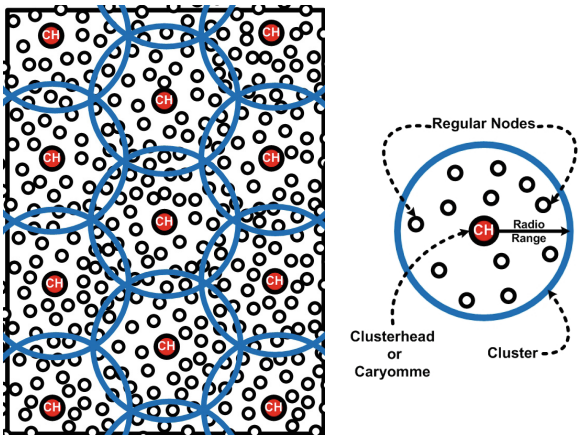
Keywords: WSN · Clustering · LQI-DCP · Security · LQI-DCPSec

1 Introduction

A WSN application often requires a clustering protocol to partition (Fig. 1) the network in order to guarantee scalability and better performance [1]. When cluster leaders are required, the nodes of each cluster will be able to execute an election protocol to determine their cluster leader [1–3].

Many clustering protocols have been proposed. However, most of them, such as LQI-DCP [1], LCA, LCA2, MaxMin [3], LEACH [4], LSCA [5] and SOS [6], assume healthy networks and are not resistant to attacks from malicious participants in hostile environments. In Leader-First approaches, malicious nodes

C. Li and S. Mao (Eds.): WiCON 2017, LNICST 230, pp. 210–220, 2018.
https://doi.org/10.1007/978-3-319-90802-1_18



(a) *WSN Clustering* (b) *Cluster*

Fig. 1. Clusters formation [2,3]

may lie about their metrics (for example, increasing the criterion value to be elected as clusterhead, when the node sends it to its respective neighbors in the first step of the LQI-DCP process [1]). As a result, such kind of nodes could control all nodes in their clusters. Similarly, none of the Cluster-First protocols can guarantee a consistent view of group memberships when malicious nodes send erroneous information.

In [7], Vasudevan et al. have proposed two secure leader election algorithms using a trusted authority to certify the metrics of each node used in the leader election process. However, these algorithms assume that all participating nodes are reliable and that no messages are lost or delayed, which can not be guaranteed in case of malicious nodes.

In [1], we have proposed a distributed multihop clustering protocol, called LQI-DCP, which is based on the Link Quality Indicator (LQI). Intended to dense wireless sensor networks, its basic idea is to enhance the network efficiency by selecting the best located sensors as clusterheads. It takes place in 2 rounds. The first round consists of information exchanges to initialize the algorithm and to preselect preferable nodes as clusterheads. In this round, the undesirable nodes with respect to the preselected ones are identified. In the second round, caryommes are elected among the remaining non-clusterized nodes and then among the previous identified undesirable nodes in last resort, while avoiding having single-node clusters [8] in the network.

This previous work [1] have shown that LQI-DCP helps in producing clusters of which each clusterhead has a better positioning regarding the locations of other clusterheads. Therefore in cluster formation, LQI-DCP has shown that it is quite important to sufficiently outspread the clusterheads (or caryommes) in order to improve the network efficiency. However, producing a secure clustering protocol was not part of our objectives in [1]. This is precisely what we are

trying to achieve in this paper by proposing LQI-DCPSec which is a secure version of LQI-DCP. Because like most WSN protocols, LQI-DCP clustering protocol should meet security requirements. However, sensors are constrained by limited energy resources and the need to manipulate simple data for the cluster-head election process poses major challenges in implementation of security mechanisms and data processing.

To carry out our work, this paper is organized as follows: in Sect. 2, we will consider an attacker's model. Then, we present LQI-DCPSec security mechanisms in Sect. 3. Finally, LQI-DCPSec is compared with SEFA protocol [7] regarding to network performance in Sect. 4.

2 Attacker's Model

We consider an attacker who is in the vicinity of the nodes to listen to traffic during the clustering process. In other words, in this attacker model, the attacker snoops the network communications and then records the messages sent by the legitimate nodes during the clustering process. Then, he could produce adequate messages that would allow him to be illegitimately elected as clusterhead (with a better metric, for example). The attacker could also try to insert one or more malicious nodes in the network in order to disrupt the cluster-head election process.

With this simple model, an attacker could perform, during the LQI-DCP clustering process, the most of common WSN attacks described in [9,10] such as: passive listening or eavesdropping attack, traffic analysis attack, man-in-the-middle attack, message alteration attack, sybil attack, black hole attack, selective forwarding attack and many others kind of attacks. Therefore, the LQI-DCPSec protocol is intended to prevent such attacks in the WSN by adding confidentiality and integrity features to secure the messages exchanged during the clustering process.

3 LQI-DCPSec Security Mechanisms

LQI-DCP is described in detail in [1,3]. Further results concerning LQI-DCP are published in [11]. Here, we pursue the objective of ensuring the integrity and confidentiality of the messages exchanged (sending of the criterion value in the first step of LQI-DCP, PN-INFORM-MSG, WBN-SELECTION-MSG and CH-INFORM-MSG) during the cluster formation process by LQI-DCP. This is that updated version of the LQI-DCP protocol that we have called LQI-DCPSec.

3.1 Initialization Phase

Prior to the deployment of the network, the base station assigns to each node u a unique identifier Id_u and a key pair (private key and public key). In addition, each node u knows the public key of all its neighbors. In addition, the base station generates a unique secret key shared by all the sensors in the network.

3.2 Hash and Encryption Functions

To secure LQI-DCP, we will use the hash and encryption functions of the PRESENT protocol [12]. PRESENT is an ultralight block encryption algorithm that runs on 31 rounds [12]. This algorithm works on blocks of 64 bits and uses keys of 80 or 128 bits length. These two versions are called PRESENT-80 and PRESENT-128 depending on whether the key size is 80 or 128 bits. For a better application, the authors of [12] propose the use of 80-bit keys. Therefore, we will use PRESENT-80 for messages encryption. We will also use the DM-PRESENT-80 hash function to calculate the messages digest for each packet: sending of the criterion value in the first step of LQI-DCP, PN-INFORM-MSG, WBN-SELECTION-MSG and CH-INFORM-MSG. Such as PRESENT-80, DM-PRESENT-80 uses 80-bit keys inputs and returns a message digest of 64-bit.

3.3 Signing and Encrypting a Message from Node A to Node B

In this section we use the following notations:

- $K_{Priv}(A)$: The private key of a node A
- $K_{Pub}(A)$: The public Key of a node A
- $K_{WSN}(t)$: The shared key, at time t, by all the nodes of the network
- F_{Hash}: The hashing function, Hash algorithm.

Signing a Message from Node A to Node B

- **Message digest computation:** The main purpose of the message digest computation is to ensure the integrity of the message. $\mathbf{Digest} = [\mathbf{Message}]F_{Hash}$
- **Signature of the digest:** The sender (node A in this case) generates a signature digest by using its private key. Moreover, the name of the hashing algorithm used by the transmitter is sent with the generated signature. With this information, anyone could decrypt and verify the signature using the sender public key and the hash algorithm. Given the properties of public key encryption and hashing algorithms, the recipient has the proof that:
  1. The digest was encrypted using the transmitter private key.
  2. The message is protected against any alteration.
  3. At this step, one has the initial message and the computed digest which is signed with the sender private key: $\mathbf{Message} + [\mathbf{Digest}]K_{Priv}(A)$.

Message Encryption

- We use a single encryption/decryption key because cryptographic algorithms that use asymmetric keys are too slow and symmetric key algorithms are more energy efficient in WSNs.
- Then, the entire message (the initial message and its digital signature) is encrypted using the network shared key $K_{WSN}(t)$.
- At this step, one has the entire message which is signed with the WSN shared key: $[\mathbf{Message} + [\mathbf{Digest}]K_{Priv}(A)]K_{WSN}(t)$

3.4 Deciphering and Verifying the Signature of the Received Message

Message Decryption. The message (which includes the message itself and the digital signature) is then decrypted using the network shared key $K_{WSN}(t)$.

Signature Verification. The signature verification includes the three following steps:

- Deciphering the message digest. The digest was encrypted using the private key of the sender (node A). The digest is now decrypted using the transmitter's public key.
- Evaluation of the digest. Since hashing is a one-way process, in other words, it is impossible to retrieve the original message from the digest, the recipient must re-evaluate the digest using exactly the same hashing algorithm as the sender.
- Comparison of condensed matter. The deciphered condensate and the evaluated condensate are compared. If they agree, the signature is verified and the recipient can then be sure that the message has been sent by the sender and has not been altered. If they do not agree, it is possible that:
  1. The message has not been signed by the issuer or it was corrupted.
  2. In either case, the message must be rejected.

3.5 Generating the Unique Key

As in [13], we propose a solution which consists of using a generation key. For each period or generation t, the base station sends a new key $K_{WSN}(t)$ to the entire network. This key serves as a certificate for each node to prove its membership in the network. If a malicious node tries to enter the WSN and does not have this generation key, it could not be accepted within it. This technique also makes it possible to limit the substitution attacks of a sensor and its reprogramming in order to be reinserted into the network. If a node is stolen at time 0 with the generation key $K_{WSN}(0)$, the time an attacker reprograms it to put it back in the WSN, it will have elapsed a time x. When the sensor is reinserted in the network, the new generation key would then be $K_{WSN}(x)$. The malicious node will ask its neighbors to enter the network with the generation key $K_{WSN}(0)$ and not $K_{WSN}(x)$ because it could not having receive the new generation key. As $K_{WSN}(0)$ is different from $K_{WSN}(x)$, neighbors will not accept its request and the malicious node will not then be able to enter the network.

4 LQI-DCPSec Performance Evaluation

To evaluate the performance of LQI-DCPSec, we compare it to the SEFA protocol. Some clustering protocols are more suitable for some deployment strategies than others [11]. So, in this section performance evaluation are done by using

in results (Figs. 2 and 3) a random deployment type without constraint [14], in results (Figs. 4 and 5) a deployment type with a remoteness constraint λ between nodes as defined in [14], and a grid topology deployment in the result of Fig. 6.

4.1 SEFA Algorithm

In [7], authors consider the problem of secure leader election and propose Secure Extrema Finding Algorithm (SEFA) as a cheat-proof election algorithm. SEFA assumes a synchronous distributed system in which the various rounds of election proceed in a lock-step fashion. SEFA assumes that all elector-nodes share a single common evaluation function that returns the same value at any elector-node when applied to a given candidate-node. The details of SEFA protocol are described in [7].

4.2 Simulation Parameters

In the simulation model N sensors are deployed over an area of length L, and width l. The transmission range of each sensor (including the Base Station) is R = 20 m. As in [1–3], we use the same energy consumption model. Let $E_{Tx}(k, d)$ the energy consumed to transmit a *kbits* message over a distance d [4]:

$$E_{Tx}(k, d) = E_{elec} * k + \varepsilon_{amp} * k * d^2 \tag{1}$$

Let E_{Rx} the energy consumed to receive a *kbits* message:

$$E_{Rx}(k, d) = E_{Rx-elec}(k) = E_{elec} * k \tag{2}$$

$$E_{elec} = 50\,\text{nJ/bit}\ and\ \varepsilon = 100\,\text{pJ/bit/m}^2 \tag{3}$$

The main parameters are summarized in the Table 1.

Table 1. Simulation parameters used for each result

Parameters	Figure 2	Figure 3	Figure 4	Figure 5	Figure 6
Radio range R	20 m	20 m	20 m	20 m	20 m
Area length L	100 m	100 m	100 m	100 m	100 m
Area width l	60 m	60 m	100 m	100 m	100 m
Number of sensors	50 to 200	20	50 to 200	50	50
λ			0.25	0.25	
Step					10

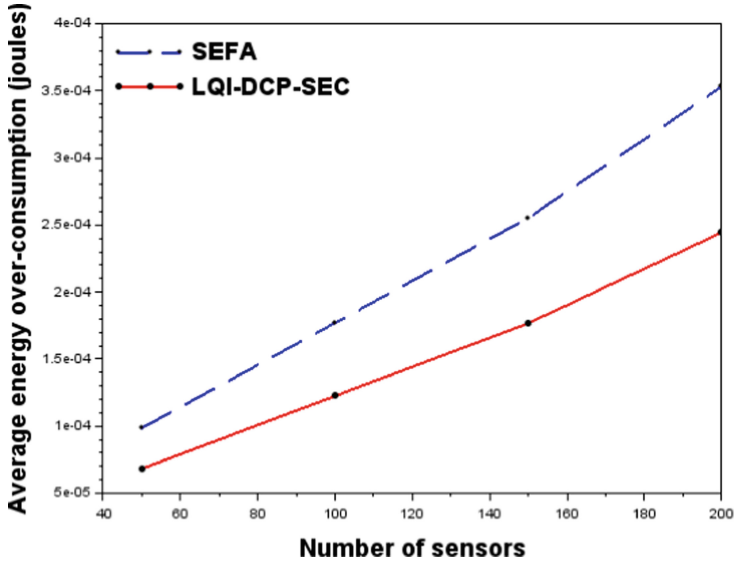


Fig. 2. Total average energy consumption as a function of the number of sensors deployed according to the random deployment without constraint (Color figure online)

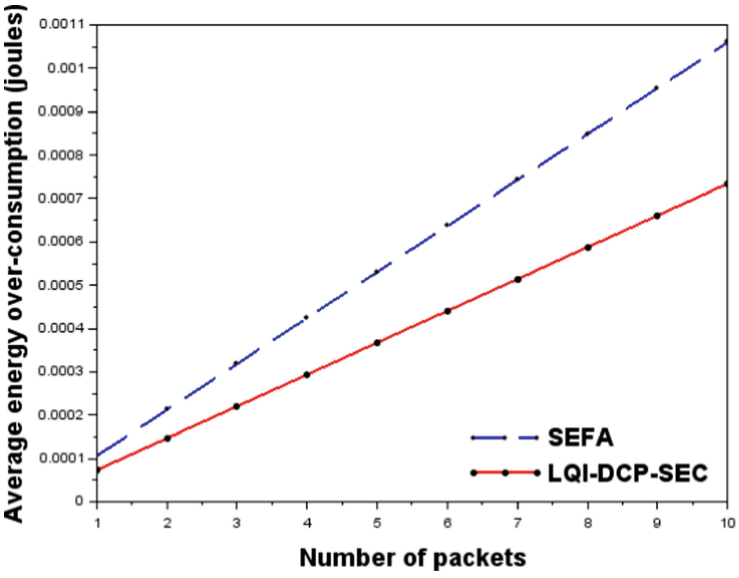


Fig. 3. Total average energy consumption as a function of the number of packets sent per sensor following random deployment without constraint (Color figure online)

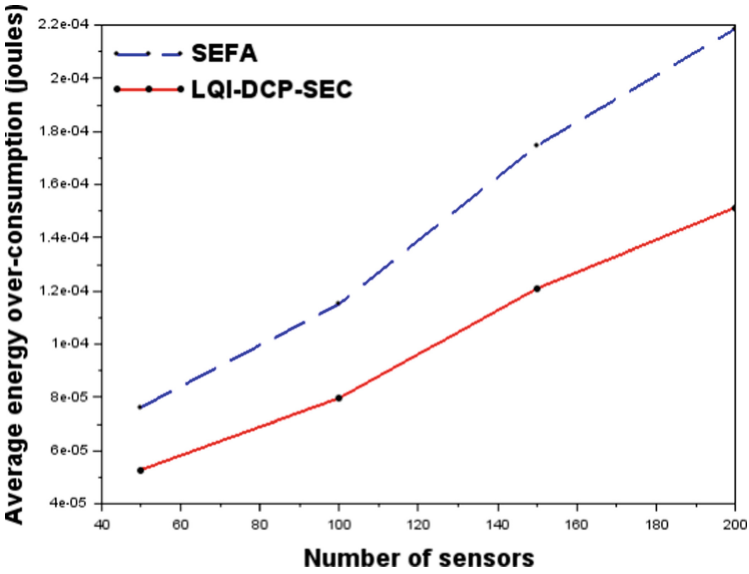


Fig. 4. Total average energy consumption as a function of the number of sensors deployed according to a deployment type with a remoteness constraint λ between nodes (Color figure online)

4.3 Simulation Results

In each Figs. 2, 3, 4, 5 and 6, we have two curves: the first one, in red and in continuous lines (LQI-DCPSec), outputs the total average energy consumption using the LQI-DCPSec protocol for forming clusters; and the second one in blue dotted lines (SEFA) is obtained by using the SEFA protocol.

To compare the energy cost of the LQI-DCPSec and SEFA clustering protocols, we calculated the total average energy over-consumption in the case where the number of packets sent per sensor varies from 1 to 10 in each case of random deployment without constraint (Fig. 3) and deployment type with a remoteness constraint λ (Fig. 5) as well as in the case of the grid deployment (Fig. 6). We also calculated the total average additional energy expenditure in the case where the number of sensors varies from 50 to 200 according to the LQI-DCPSec and SEFA clustering protocols in the case of random deployment (without constraint (Fig. 2) and with a remoteness constraint λ (Fig. 4)).

The results of Figs. 3, 5 and 6 show that the more the packet exchanges increase in the network, the greater the gap between energy over-consumption becomes greater between SEFA and LQI-DCPSec. SEFA consumes up to 57% more energy than LQI-DCPSec (Fig. 3). Similarly, this difference increases up to 55% in Fig. 5. On the other hand, in the case of the grid topology SEFA consumes up to 33% more energy compared to LQI-DCPSec (Fig. 6). This means that when the average number of neighbors per node decreases, the gap also decreases. It could be concluded that SEFA is more suitable for less dense

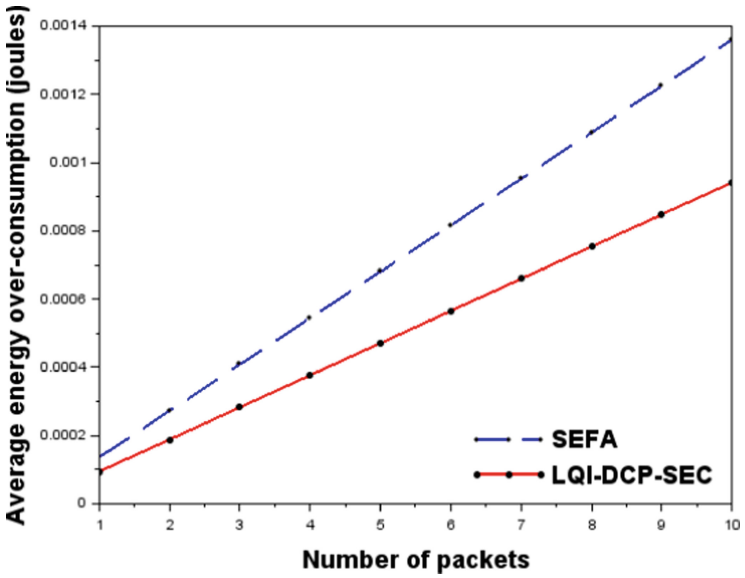


Fig. 5. Total average energy consumption as a function of the number of packets sent per sensor following a deployment type with a remoteness constraint λ between nodes (Color figure online)

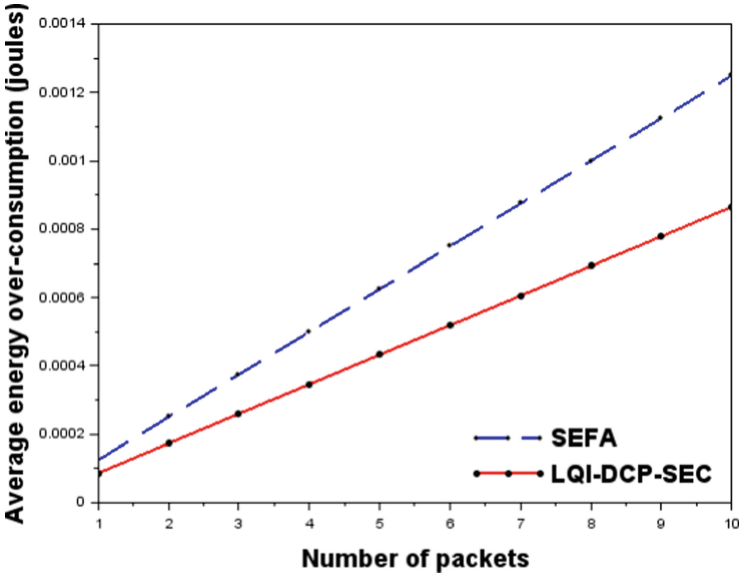


Fig. 6. Total average energy consumption as a function of the number of packets sent per sensor following the grid deployment (Color figure online)

networks even though in this case LQI-DCPSec still offers better performances. This is confirmed by the results of Figs. 2 and 4 where it should also be noted that when the density increases, the additional energy expenditure of SEFA with respect to LQI-DCPSec increases up to 45% for the result of Fig. 2, and up to 57% in the case of the result of Fig. 4.

We note that, in all cases, the total average energy consumption of the SEFA protocol exceeds that of LQI-DCPSec protocol. By observing the evolution of the curves, we note that protocol LQI-DCPSec succeeds, beyond the questions related to the security and the optimization of resources, in significantly reducing the overall average energy consumption of the network. Thus, LQI-DCPSec protocol is far better than SEFA from an energy expenditure point of view, which is a very important resource in the WSN. This does not mean that LQI-DCPSec is the best algorithm for secure cluster formation in terms of energy consumption, because we always have to think about decreasing the size of the used keys [14].

5 Conclusions and Future Works

As we can see, the LQI-DCPSec protocol guarantees the integrity of each message. This is facilitated by hashing the message content and signing the hash using the sender's private key. This allows the recipient of the message to verify the signature using the sender's public key that is known by everyone. Therefore, if a malicious node tries to corrupt a message from another node, it will be detected by the recipient. Moreover, beyond the integrity feature, authentication of the sender is also ensured with LQI-DCPSec. Finally, given the fact that all the messages are encrypted, which guarantees the confidentiality of the data, LQI-DCPSec protocol is also resistant to many attacks such as passive listening or eavesdropping, traffic analysis, man-in-the-middle, message alteration, sybil, black hole, selective forwarding, etc. Likewise, given the mechanism of unique network key generation, LQI-DCPSec prevents any malicious nodes insertion in the network. In addition, compared to SEFA, LQI-DCPSec offers better performance with much lower energy consumption.

However, from another point of view, since the messages exchanged during LQI-DCPSec clustering process are signed and encrypted, the packet size has increased. With an initial 64-bit packet size, one added the 64-bit digest obtained with the hash function DM-PRESENT-80, making a total of 128 bits. One adds to this the size of the PRESENT-80 encryption key which is 80 bits, which leads to a total packet size of 208 bits. That means to say that the package size has increased by 225%. This is high given our previous results [14]. Therefore, in our future work, we will try to propose our own lightweight integrity and privacy schemes lighter than the PRESENT algorithm in order to significantly reduce the size of exchanged packets and to gain much more performance.

Acknowledgment. This work is supported by CEA-MITIC (http://www.ceamitic.sn/), UFR SAT, Université Gaston Berger, Saint-Louis, Sénégal.

References

1. Diallo, C., Marot, M., Becker, M.: A distributed link quality based d-clustering protocol for dense ZigBee sensor networks. In: Proceedings of the Third IFIP/IEEE Wireless Days International Conference, WD 2010, Venice, Italy (2010)
2. Diallo, C., Marot, M., Becker, M.: Using LQI to improve clusterhead locations in dense ZigBee based wireless sensor networks. In: Proceedings of the 6th IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob 2010, Niagara Falls, Canada, October 2010, pp. 137–143 (2010)
3. Diallo, C.: Techniques d'amélioration du routage et de la formation des clusters multi-sauts dans les réseaux de capteurs sans fil. Ph.D. Télécom SudParis (2010)
4. Heinzelman, W.B., Chandrakasan, A., Balakrishnan, H.: An application-specific protocol architecture for wireless microsensor networks. IEEE Trans. Wirel. Commun. **1**(4), 660–670 (2002)
5. Gao, C., Jantti, R.: Link-state clustering based on IEEE 802.15.4 MAC for wireless ad-hoc/sensor networks. In: Proceedings of IEEE Wireless Communications and Networking Conference, WCNC 2006, Las Vegas, USA, vol. 1, pp. 499–504 (2006)
6. Shin, K., Abraham, A., Han, S.Y.: Self organizing sensor networks using intelligent clustering. In: Gavrilova, M.L., Gervasi, O., Kumar, V., Tan, C.J.K., Taniar, D., Laganá, A., Mun, Y., Choo, H. (eds.) ICCSA 2006. LNCS, vol. 3983, pp. 40–49. Springer, Heidelberg (2006). https://doi.org/10.1007/11751632_5
7. Vasudevan, S., DeCleene, B., Kurose, J., Towsley, D.: Secure leader election in wireless ad hoc networks. UMass Computer Science Technical report 01-50 (2010)
8. Diallo, C., Marot, M., Becker, M.: Single-node cluster reduction in WSN and energy-efficiency during cluster formation. In: Proceedings of the 9th IEEE/IFIP Annual Mediterranean Ad Hoc Networking Workshop, Med-Hoc-Net 2010, Juan-Les-Pins, France, June 2010. IEEE Communications Society (2010)
9. Diallo, C., Sawaré, A., Sow, M.T.: Security issues and solutions in wireless sensor networks. Int. J. Comput. Sci. Inf. Secur. IJCSIS **15**(3), 6 (2017). ISSN 1947-5500
10. Diallo, C.: Security issues and solutions related to data aggregation process in WSN. Int. J. Comput. Sci. Netw. Secur. IJCSNS **17**(4), 59–71 (2017). ISSN 1738–7905
11. Diallo, C.: Deployment strategies and clustering protocols efficiency. Sens. Transducers **213**(6), 9–23 (2017). International Journal ISSN 2306–8515, e-ISSN 1726–5479
12. Bogdanov, A., Knudsen, L.R., Leander, G., Paar, C., Poschmann, A., Robshaw, M.J.B., Seurin, Y., Vikkelsoe, C.: PRESENT: an ultra-lightweight block cipher. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 450–466. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74735-2_31
13. Bekara, C., Laurent-Maknavicius, M.: A new resilient key management protocol for wireless sensor networks. In: Sauveron, D., Markantonakis, K., Bilas, A., Quisquater, J.-J. (eds.) WISTP 2007. LNCS, vol. 4462, pp. 14–26. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72354-7_2
14. Sow, M.T., Diallo, C.: Energy over-consumption induced by securing network operations. In: Proceedings of 2nd IEEE International Conference on Frontiers of Sensors Technologies, IEEE-ICFST 2017, Shenzhen, China, April 2017, pp. 154–160. IEEE (2017). ISBN 978-1-5090-4858-8/17/