Balanced and Safe Weighted Clustering Algorithm for Mobile Wireless Sensor Networks

Amine Dahane, Nasr-Eddine Berrached, Abdelhamid Loukil

Balanced and Safe Weighted Clustering Algorithm for Mobile Wireless Sensor Networks

Dahane Amine*, Berrached Nasr-Eddine, Loukil Abdelhamid

Intelligent Systems Research Laboratory (LARESI)
University of Sciences and Technology of Oran
P.O. Box 1505, Oran, Algeria
*amineusto.laresi@gmail.com

Abstract. The main concern of clustering approaches for mobile wireless sensor networks (WSNs) is to prolong the battery life of the individual sensors and the network lifetime. In this paper, we propose a balanced and safe weighted clustering algorithm which is an extended version of our previous algorithm (ES-WCA) for mobile WSNs using a combination of five metrics. Among these metrics lie the behavioral level metric which promotes a safe choice of a cluster head in the sense where this last one will never be a malicious node. The goals of the proposed algorithm are: offer better performance in terms of the number of re-affiliations which enables to generate a reduced number of balanced and homogeneous clusters, this algorithm, coupled with suitable routing protocols, aims to maintain stable clustering structure. We implemented and tested a simulation of the proposed algorithm to demonstrate its performance.

Keywords: WSNs, Clustering, Homogenous Clusters, Energy Efficiency, Security.

1 Introduction

After the success of theoretical research contributions in previous decade, wireless sensor networks (WSNs) [1,2] have become now a reality. Their deployment in many societal, environmental and industrial applications makes them very useful in practice. These networks consist of a large number of small size nodes which sense ubiquitously some physical phenomenon (temperature, humidity, acceleration, noise, light intensity, wind speed, etc.) and report the collected data to the sink station by using multi-hop wireless communications. The clustering concept, that means grouping nodes which are close to each other, has been studied largely in ad-hoc networks [2,3,4,5,6,7,8] and recently in WSNs [9,10,11,12,13] where the purpose in general is to reduce useful energy consumption and routing overhead, however, cluster-heads must be selected carefully and diligently. Recent research studies recognize that organizing mobile WSNs, in the sense defined above, into clusters by using a clustering mechanism is a challenging task [9,10]. This is due to the fact that cluster heads (CHs) carry out extra work, and consequently consume more energy compared with cluster members (CMs) during the network operations and this will lead to untimely death causing network partition and therefore failure in communication link.

For this reason, one of the frequently encountered problems in this mechanism is to search for the best way to elect CH for each cluster. Indeed, a CH can be selected by computing quality of nodes, which may depend on several metrics: connectivity degree, mobility, residual energy and distance of a node from its neighbors. Significant improvement in performance of this quality can be achieved by combining these metrics [2,3,8,9,13,14].

In this paper, we propose balanced and safe weighed clustering algorithm for mobile WSNs (BS-WCA) using a combination of the above metrics with the behavioral level metric which we have added. Our approach enables to generate a reduced number of balanced and homogeneous clusters in order to minimize the energy consumption of the entire network and prolong sensors lifetime. In the other sense, the behavioral level is decisive and allows the proposed clustering algorithm to avoid any malicious node in the neighborhood to become a CH, even if the remaining metrics are in its favor. The election of CHs is carrying out using weights of neighboring nodes which are computed based on selected metrics. So, this strategy ensures the election of legitimate and trustworthy CHs with high weights. The Node-Weight heuristic assigns node-weights based on the suitability of nodes acting as cluster heads and the election of the cluster head is done on the basis of the largest weight among its neighbors. This means that a node decides to become a cluster head or stay as an ordinary node depending on the weights of its one hop neighbors [2].

The preliminary results obtained through simulation study demonstrate the effectiveness of our algorithm in terms of number of equilibrate clusters, number of re-affiliations, by comparing it with WCA [2], DWCA [14] and SDCA [11].

These results also reveal that our approach is very suitable if we plan to use in network layer reactive routing protocols instead of proactive ones after the clustering mechanism was launched. The contribution of our paper is as follows:

- Maintaining stable clustering structure and offering better performance in terms of the number of re-affiliations using the proposed algorithm BS-WCA.

The remaining part of this paper is organized as follows: We first, in Section 2, discuss the existing studies. The details of our approach are described in section 3. Section 4 introduces and explains the selected metrics for the proposed approach of clustering. A special attention was reserved for this last aspect in this research. More details on the proposed algorithm are provided in section 5. Section 6 presents the simulation tool developed for the evaluation and provides simulation results to show the effectiveness of the proposed algorithm. Section 7 concludes the paper.

2 Related Works

In this section, we outline some approaches of clustering used in Ad-hoc networks and WSNs. Abbasi *et al*. [15] presented taxonomy and classification of typical clustering schemes, and then summarized different clustering algorithms for WSNs based on classification of variable convergence and constant convergence time protocols. They also highlighted objectives, features, and algorithms complexity. Research studies on clustering in Ad-hoc networks evolve surveyed works on clustering algorithms [16] and cluster head election algorithms [3,10]. For the single metric based on clus-

tering, as in paper [17], the node with the least stability value is elected as CH among its neighbors, however the choice of CH which has a lower energy level, could quickly become a bottleneck of its cluster. Safa *et al*. [4] designed and implemented a dynamic energy efficient clustering algorithm (DEECA) for mobile Ad-hoc networks (MANETs) that increases the network lifetime, however, the cluster formation in this scheme is not based on connectivity so the formed clusters are not well connected; this induces an increase of re-affiliation rate and re-clustering situations. Other proposals use strategy based on weights computing in order to elect CHs [2,3,8,14].The main strategy of these algorithms is based mainly on adding more metrics such as connectivity degree, mobility, residual energy and distance of a node from its neighbors, corresponding to some performance in the process of electing CHs. Although, the algorithms using this strategy allow to ensure the election of a better CHs based only on their high weights computed from the considered metrics, but unfortunately they does not ensure that the elected CHs are legitimated nodes, which is to say if the election process of CHs is safe or not. Safa *et al.* [5] propose a novel cluster-based trust-aware routing protocol (CBTRP) for MANETs to protect forwarded packets from intermediary malicious nodes. The proposed protocol ensures the passage of packets through trusted routes only by making nodes monitor the behavior of each other and update their trust tables accordingly. However, in CBTRP all nodes monitor the network which lead rapid drainage of node energy and therefore minimize the lifetime of the network. Khalil *et al.* [18] proposed a protocol called DICAS, which uses local monitoring and mitigates the attacks against control traffic by detecting, diagnosing and isolating the malicious nodes. Hsin *et al.* [19] proposed a self-monitoring mechanism that pays more attention to the system-level fault diagnosis of the network, especially for detecting node failures. However, they did not deal with malicious behaviors. Little effort has been made in introducing security aspect in clustering mechanism. Yu *et al.*[7] tried to secure clustering mechanism against wormhole attack in ad-hoc networks (communication between CHs) but after forming clusters, not during the election procedure of CHs. Hai *et al.*[21] propose a light-weight intrusion detection framework integrated for clustered sensor networks by using an over-hearing mechanism to reduce the sending alert packets. Elhdhili *et al.* [6] propose a reputation based clustering algorithm (RECA) that aims to elect trust-worthy, stable and high energy cluster heads but during the election procedure, not after forming clusters. Benahmed *et al.* [11] used clustering mechanism based on weighted computing as an efficient solution to detect misbehavior nodes during distributed monitoring process in WSNs. However, they focused only on the misbehavior of malicious nodes and not on the nature of attacks, the formed clusters are not homogeneous, the proposed secured distributed clustering algorithm (SDCA) is not coupled with routing protocols and doesn't give much importance to energy consumption.

In the context of these surveyed research works about clustering in both ad-hoc networks and WSNs, we classified our contribution among approaches based on the computing of the weight of each node in the network, this approach focuses around strategy of distributed resolution which enables to generate a reduced number of balanced and homogeneous clusters in order to minimize the energy consumption of the entire network and prolong sensors lifetime. Moreover, we introduced a new metric

(the behavioral level metric) which promotes a safe choice of a cluster head in the sense where this last one will never be a malicious node.

3 Our Approach

In the literature, no research has thought to use energy efficiency and monitoring mechanism using the same cluster-based architecture. Our first objective is to make the network able to self-organize in order to achieve its tasks with a least cost. In this context, we must determine the parameters for generating a reduced number of stable and balanced clusters. Our second objective is to propose a mechanism that assures the distributed monitoring of WSNs security reasons. This mechanism uses a cluster-based architecture, as well as new set of metrics and rules for diagnosing the state of the sensors. The advantages of this solution are that it reduces the flow of communication and provides stable surveillance environment. This approach gives more importance to the election criteria of nodes responsible for monitoring the network. The details of this approach are illustrated in our proposed algorithm BS-WCA.

4 Metrics for CHs Election

This section introduces the different metrics used for cluster-head election. In our earlier work [9], we insisted in Mobility (Mi), connectivity (Ci), residual energy (Eri) and distance of node ni (Di) to its neighbors. In this paper, we focus our study on behavior level metric.

- **The behavior level of a node n_i (BLi)**

The behavioral level of a node n_i is a key metric in our contribution. Initially, each node is assigned an equal static behavior level "BL_i=1".
However, this level can be decreased by the anomaly detection algorithm if a node is misbehavior as illustrated by Fig 1.

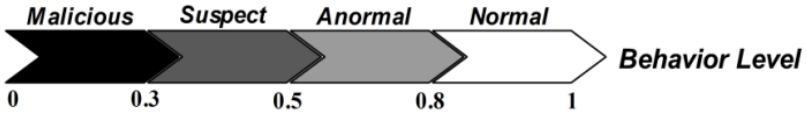

Fig.1. The behavior level (BL_i)

For computing the behavior level of each node, nodes with a behavior level less than threshold behavior will not be accepted as CH candidates even if they have the other interesting characteristics such as high energy, high degree of connectivity or low mobility. Nevertheless, abnormal node and suspect node can always belong to a cluster as a CM but never as a CH. So, we define the behavior level of each sensor

node n_i, noted BL_i, in any neighborhood of the network as presented in Fig.1. BL_i is classified by the following mapping function ($Mp(BL_i)$):

$$Mp(BL_i) = \begin{Bmatrix} Normal\ node: & 0.8 \leq BL_i \leq 1 \\ Abnormal\ node: & 0.5 \leq BL_i < 0.8 \\ Suspect\ node: & 0.3 \leq BL_i < 0.5 \\ Malicious\ node: & 0 \leq BL_i < 0.3 \end{Bmatrix} \quad (1)$$

The values in the formula (1) are chosen on the basis of several reputed models of WSNs adopted by numerous researchers like Shaikh *et al*. [20] and Hai *et al*. [21].
For each node, we must calculate its weight P_i, according to the equation:

$$P_i = w_1 * BL_i + w_2 * Er_i + w_3 * M_i + w_4 * C_i + w_5 * D_i \quad (2)$$

Where $w_1, w_2, w_3, w_4,$ and w_5 are the coefficients corresponding to the system criteria, so that:

$$w_1 + w_2 + w_3 + w_4 + w_5 = 1 \quad (3)$$

We propose to generate homogeneous clusters whose size lies between two thresholds:$Thresh_{Upper}$ and $Thresh_{Lower}$. These thresholds are arbitrarily selected or depend on the topology of the network. Thus, if their values depend on the topology of the network, they are calculated as follows according to [12]:

$$Thresh_{Upper} = \frac{1}{2}(\delta_{12}(u) + AVG) \quad (4)$$

$$Thresh_{Lower} = \frac{1}{2}(\delta_{12}(v) + AVG) \quad (5)$$

With:

$$\delta_{12}(u) = \max(\delta_{12}(u_i):\ u_i \in U) \quad (6)$$

$$\delta_{12}(v) = \min(\delta_{12}(v_i):\ v_i \in U) \quad (7)$$

$$AVG = \frac{\sum_{i=1}^{n} \delta_{12}(u_i)}{N} \quad (8)$$

Where:

- u represents the node that has the maximum number of neighbors with one jump;
- v represents the node that has the minimum number of neighbors with one jump;
- AVG denotes the average cardinal of the groups with one jump of all the nodes of the network;
- N is the number of nodes in the network.

The weight P_i calculated for each sensor is based on the above parameters ($BL_i, M_i, D_i,\ Er_i$ and C_i). It means for our case the trust level of each node in the network. The values of coefficients w_i should be chosen depending on the importance of each metric in considered WSNs applications. For instance, we can assign a greater value to the metric BL_i compared to other metrics if we promote the safety aspect in the clustering mechanism. We can also assign a same value for each coefficient $w_i$in case when all metrics are considered having the same importance. An approach based on these weights will enable us to build a self-organizing algorithm able to form small

number of homogenous clusters in size and radius by grouping geographically close nodes. The resulting weighted clustering algorithm reduces energy consumption and guaranty the choice of legitimate CHs.

5 Weighted Clustering Algorithm (BS-WCA)

In this section, we first give assumptions of the proposed algorithm: Balanced and Safe Weighted Clustering Algorithm (BS-WCA).

Then we present, in detail, an extended version of ES-WCA [9] followed by an illustrative example.

5.1 Assumptions

Before heading into the technical details of our algorithm, this paper is based on the same assumptions as in [9]. We add the fact that a malicious node can use its own ability to move freely in the space area. The behavior of the malicious node by moving frequently inside a same cluster or from a cluster to another is a normal behavior to not attract attention of the neighborhood and therefore to be detected .

5.2 Re-affiliation phase

During the first phase, it may not be possible for all clusters to reach the $Thresh_{Upper}$ threshold. Moreover, it is possible that clusters whose size is lower than $Thresh_{Lower}$ may be created, since there is no constraint relating to the generation of these types of clusters. BS-WCA uses four types of messages in the Re-affiliation phase. The message RE_AFF_CH, that is sent in the network by the CH which the cluster size is less than $Thresh_{Upper}$. The second one is the REQ_RE_AFF message that is sent by the neighbors of CH if it wants to join this cluster. Finally a CH must send a response ACCEPT_RE_AFF message or DROP_AFF message as illustrated by Fig. 2. Hence, in this second phase, we tried to reduce the number of clusters formed and reorganize them in order to obtain balanced and homogeneous clusters. For that, we propose to re-affiliate the sensor nodes belonging to clusters that have not attained the cluster size$Thresh_{Lower}$to those that did not reach $Thresh_{Upper}$.

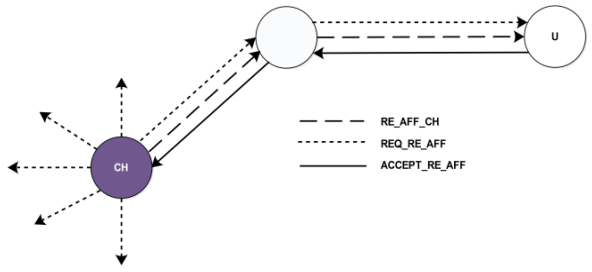

Fig.2. Procedure of Re-affiliation of node 'U' to a cluster

We demonstrate our set up phase algorithm and re-affiliation phase with the help of four figures (Fig. 3, Fig. 4, Fig. 5 and Fig. 6).

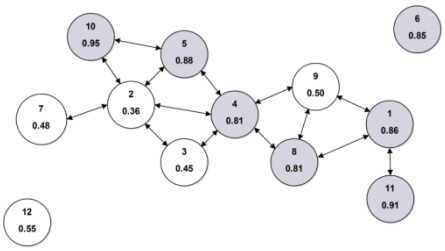

Fig.3. Topology of the network

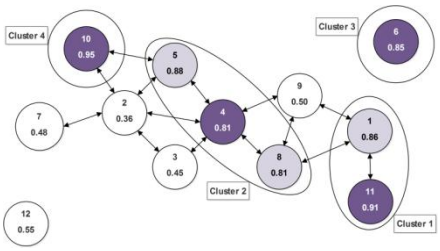

Fig.4. Identification of clusters nodes

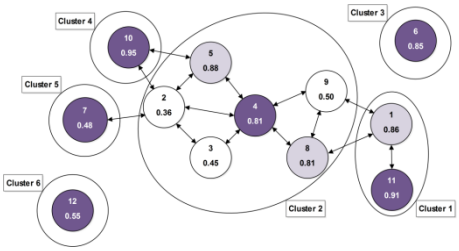

Fig.5. The final identification of clusters

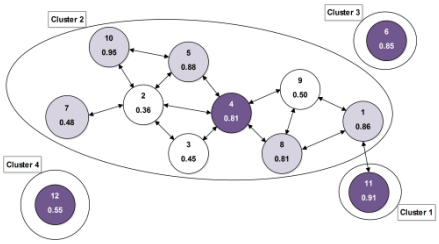

Fig.6. The final identification of clusters

Algorithm: Re-affiliation Phase Algorithm

Inputs: $Thresh_{Upper}$, $Thresh_{Lower}$;
Outputs: set of clusters
Begin
For num_cl = 1 to Count (Cluster) **Do**
  If (Size (Cluster [num_cl]) < $Thresh_{Upper}$)
    Then
    CH sends a message "RE_AFF_CH" to its neighbours (N(CH));
    J = Count (N(CH));
  EndIf
For I = 1 to J **Do**
  If ($n_i \in N$(CH) receives the message)
    && ($n_i \in$ (Size(Cluster[num_cl]) <$Thresh_{Lower}$)
    Then
    n_i sends a Select message "REQ_RE_AFF" to the CH;
  If (Size (Cluster [num_cl]) < $Thresh_{Upper}$)
    Then
    CH sends a message "ACCEPT_RE_AFF" to n_i ;
    CH updates its state vector;
    CH → CH → Size = Size + 1;
    n_i updates its state vector;
    n_i → CH → ID = ID;
    Else CH sends a "FIN_ AFF" message to n_i ;
    Go to 2;
  EndIF
  Else n_i sends a "DROP_AFF" message to CH;
EndIf
End For
End For
End.

Table I shows the values of the different criteria for the nodes that have behavior level BL_i > 0.8 (Normal nodes). Table II shows the weights P_i of neighbors for each node that have behavior level BL_i > 0.8.

Table 1. Values of the various criteria of normal nodes

Ids	BL_i	E_{ri}	C_i	D_i	M_i	P_i
1	0.86	3842.12	3	1.15	1.20	769.632
4	0.81	4832.54	5	2.30	0.30	968.133
5	0.88	4053.25	3	1.30	0.55	811.829
6	0.85	4620.43	0	0.00	0.20	924.361
8	0.81	4816.80	4	1.05	1.40	964.753
10	0.95	3650.25	2	0.55	0.10	730.805
11	0.91	4816.80	4	0.75	2.20	964.753

Table 2. Weight P_i of neighbors

Ids	1	4	5	6	8	10	11
1	769.632	-	-	-	964.753	-	964.753
4	-	968.133	811.829	-	964.753	-	-
5	-	968.133	811.829	-	-	730.805	-
6	-		-	924.361	-	-	-
8	769.632		-	-	964.753	-	-
10	-	968.133	811.829	-	-	730.805	-
11	769.632		-	-	-	-	964.753

Nodes in Fig.3 are presented by circles containing their identity Ids at the top and the levels of behavior at the bottom. According to table 2, node 1 has a choice between CH11 and CH8 (they have the same weight), but the behavior level of node 11 is greater than the node 8 (BL_{11}>BL_8), so node 1 will be attached to CH11. For the other nodes, we have various conditions. Node 4 declares itself as a CH. Node 5 will be attached to CH4. Node 6 declares itself as a CH, because it is an isolated node. Node 8 will be attached to CH4. Node 10 is connected with CH5, but node 5 is attached to CH4; thus, node 10 declares itself as a CH. Node 11 declares itself as a CH. These results give us the representation shown in Fig.4. Node 2 is connected with CH4 and CH10. Node 2 will be attached to CH4, because CH4 has the maximum weight (968.133). Node 3 is connected with CH4, which implies that node 3 will be attached to CH4. Node 7 is not connected with any CH, so node 7 declares itself as CH. Node 9 is connected with CH4, and then node 9 will be attached to CH4. Node 12 is not connected with any CH, which implies that node 12 declares itself as a CH. These results give us the representation shown in Fig.5. We propose to generate homogeneous clusters whose size lies between two thresholds: $Thresh_{Upper}$= 9 and $Thresh_{Lower}$= 6. For that, we suggest to re-affiliate the sensor nodes belonging to the clusters that have not attained the cluster size $Thresh_{Lower}$ to those that did not reach $Thresh_{Upper}$. Node 4 have the highest weight and his size is less than $Thresh_{Upper}$. Nodes 1, 7 and 10 are neighbors of the node 4 with 2 hops and belong to the clusters that have not attained the cluster size $Thresh_{Lower}$, so these nodes get merged to cluster 2. Clusters 1, 3, and 4 will be homogeneous with cluster 1 when the network becomes densely. At the end of this example, we obtain a network of four clusters (as shown in Fig. 6).

There are five situations that require the maintenance of clusters:

- Battery depletion of a node.
- Behavior level of a node less than or equal 0.3.
- Adding, moving or deleting a node.

In all of these cases, if a node n_i is CH then the set-up phase will be repeated.

6 Implementation Results

In this section, we present our simulator 'Mercury' and the results of our work. To determine and evaluate the results of the execution of algorithms that are introduced previously, the number of sensors (N) to deploy must be less than or equal to 1000. There are two types of sensor node deployment on the sensoring field: random and manual. "Mercury" offers users the ability to select a sensor type from 5 predefined types. Each one has its characteristics (radius, energy, etc.). The user can also introduce his own characteristics. The unity of the used energy is the nano joules (1 Joule = 10^9 NJ).

6.1 Discussion and Results

In all experiments, N varies between 10 and 100 sensor nodes, the transmission range (R) varies between 10 and 70 meters (m) and the used energy (E) equal to 50000 NJ. By default, for each set of simulation, we conduct 100 runs with different node generations and report the average. The sensor nodes are randomly distributed in a "570m × 555m" space area by the following function:

```
for (int n = 0; n < node_tobe_deployed; n++)
{
    X_ = rand( )  %  image_Field_Of_Collecting -> width;
    Y_ = rand( )  %  image_Field_Of_Collecting -> Height;
}
```

To measure the performance of BS-WCA algorithm, we considered the following four metrics:

a. The number of clusters;
b. The number of re-affiliations;

The values of weighting factors used for simulation were:

$$w_1 = 0.3,\ w_2 = 0.2,\ w_3 = 0.2,\ w_4 = 0.2 \text{ and } w_5 = 0.1.$$

Note that these values are arbitrary at this time and should be adjusted according to the system requirements. To evaluate the performance of the BS-WCA algorithm with other algorithms, we studied the effect of the density of the networks (number of sensor nodes in a given area) and the transmission range on the average number of formed clusters. Then we compare it with a DWCA (Distributed Weighted Clustering Algorithm) proposed in [14], WCA (A weighted Clustering Algorithm for Mobile Ad-hoc Networks) proposed in [2] and SDCA (secured distributed clustering algorithm) proposed in [11]. We omit presenting all results and the monitoring phase due to the space limitation. The highlight of our work is summarized in a comprehensive

strategy for monitoring the network that will be presented in our future works. The goal is to detect and remove the malicious nodes

Fig.7 depicts the average number of clusters that are formed with respect to the total number of nodes in the network. The communication range used in this experience is 200m. As we can see in Fig. 7, the proposed algorithm produced the same number of clusters than DWCA when the node number is equal to 20 nodes. If the node density has increased, BS-WCA would have produced constantly less clusters than SDCA and DWCA regardless of node number. The result of BS-WCA is so unstable between 60 and 90 because we use a random deployment so if the distance between the nodes increases, the number of clusters increases too. When there were 100 nodes in the network, the proposed algorithm produced about 61.91% fewer clusters than DWCA [14] and about 38.46% than SDCA [11]. As a result, our algorithm gave better performance in terms of the number of clusters when the node density in the network is high, because BS-WCA generates a reduced number of balanced and homogeneous clusters, whose size lies between two thresholds: $Thresh_{Upper}$ and $Thresh_{Lower}$ (Re-affiliation Phase) in order to minimize the energy consumption of the entire network and prolong sensors lifetime.

Fig.8 shows the variation of the average number of clusters with respect to the transmission range. The results are shown for varying N. We observe that the average number of clusters decreases with the increase in the transmission range. As we can see in Fig.10, the proposed algorithm produced 16% to 35% fewer clusters than WCA when the transmission range of nodes was 10m. If the node density increased, BS-WCA produced constantly fewer clusters than WCA regardless of node number. When there were 70 nodes in the network, the proposed algorithm produced about 47% to 73% fewer clusters than WCA. According to the result, our algorithm gave better performance in terms of the number of clusters when the node density and transmission range in the network are high.

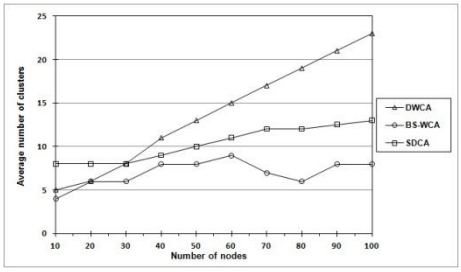

Fig.7. Average number of clusters vs number of nodes (N) for BS-WCA, DWCA and SDCA

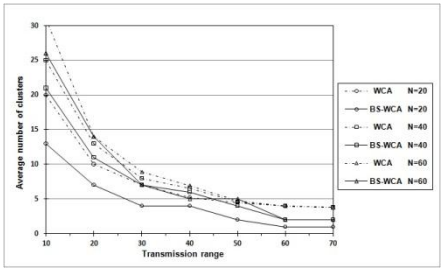

Fig.8. Average number of clusters vs transmission range BS-WCA and WCA

Fig.9 depicts the average number of re-affiliations that are formed with respect to the total number of nodes in the network. We propose to generate homogeneous clusters whose size lies between two thresholds: $Thresh_{Upper} = 18$ and $Thresh_{Lower} = 9$. The number of re-affiliations increased linearly if there were 30 or more nodes in the network for both WCA and DWCA, but for our algorithm the number of re-affiliations increased starting from 50 nodes. According to the results, our algorithm gave better performance in terms of number of re-affiliations. The main reason is that the frequency of invoking the clustering algorithm is lower in BS-WCA, thus resulting in longer duration of stability of the topology. The benefit of decreasing the number of

re-affiliations mainly comes from the localized re-affiliation phase in our algorithm. From Figure 10 it is observed that the sensor nodes 3 and 19 are malicious and have a behavior level less than 0.3. We also note that the sensor 11 is suspicious so if it continues to move frequently it's behavior will gradually be decreased until it reaches the malicious state in this case this node will be deleted from the neighborhood and finally it will be added to the black list. The behavior level of these nodes decreased by 0.001 units when it moves one meter away from its original location but this malicious node does nothing just mobility so in our future works, we will detect the internal misbehavior nodes during distributed monitoring process in WSNs by the follow-up of the messages exchanged between the nodes.

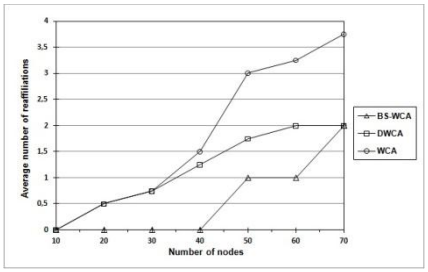

Fig.9. Average number of re-affiliations

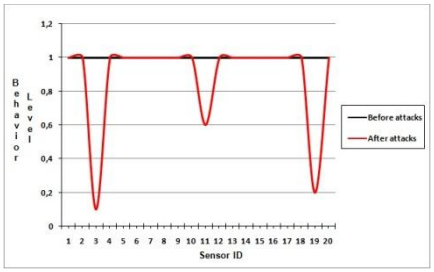

Fig.10. Behavior level of some sensors before and after attacks

7 Conclusions

In this paper, we have presented a new algorithm called "BS-WCA" for the self-organization of mobile sensor networks. Obtained results from simulations prove that our algorithm outperforms WCA, DWCA and SDCA. It yields a low number of clusters and preserves network structure better than WCA and DWCA by reducing the number of re-affiliations. The proposed algorithm chooses the most robust and safe CHs with the responsibility of monitoring the nodes in their clusters and maintaining clusters locally. As a result of this work, we plan to add a monitoring phase which analyses and detects specific misbehavior in the WSNs by the follow-up of the messages exchanged between the nodes.

Acknowledgements

The authors are grateful to the anonymous referees and Professor Bouhadiba F. for their insightful comments and valuable suggestions, which greatly improved the quality of the paper.

References

1. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "Wireless sensor networks: A Survey", Computer Networks (Elsevier), Vol. 38, No.2, pp. 393-422 (2002)
2. M. Chatterjee, S. Das, and D. Turgut, "WCA: a weighted clustering algorithm for mobile ad hoc networks", Journal of Cluster Computing (Special Issue on Mobile Ad-hoc Networks), Vol. 5, pp. 193-204 (2002)
3. A. Zabian, A. Ibrahim, F. Al-Kalani, "Dynamic Head Cluster Election Algorithm for clustered Ad-hoc Networks", Journal of Computer Science, Vol. 4, No. 1 (2008)

4. H. Safa, H. Artail, D. Tabet, “A cluster-based trust-aware routing protocol for mobile ad-hoc networks”, Wireless Networks (Springer), Vol.16, N°4, pp.969-984 (2010)
5. H. Safa, O. Mirza, H. Artail, “A Dynamic Energy Efficient Clustering Algorithm for MANETs”, IEEE International Conference on Wireless & Mobile Computing, Networking & Communication, pp.51-56 (2008)
6. M. Elhdhili, L. Azzouz, F. Kamoun, “Reputation based clustering algorithm for security management in ad hoc networks with liars”, International Journal of Information and Computer Security, Vol. 3, N°3, pp.228 -244 (2009)
7. Y. Yu, L. Zhang, “A Secure Clustering Algorithm in Mobile Ad-hoc Networks”, 2012 IACSIT Hong Kong Conferences, Vol. 29, pp.73-77 (2012)
8. R. Agarwal, R. Gupta, M. Motwani, “Review of Weighted Clustering Algorithms for Mobile Ad-hoc Networks”, Computer Science and Telecommunications, Vol. 33, No 1, pp.71-78 (2012)
9. A. Dahane, N. Berrached, B. Kechar “Energy Efficient and Safe Weighted Clustering Algorithm for Mobile Wireless Sensor Networks”, The 9th International Conference on Future Networks and Communications, Procedia Computer Science (Elsevier) , Vol. 34, August 17 -20, Niagara Falls, Ontario, Canada, pp.63-70 (2014)
10. S. Soro and W.B. Heinzelman, “Cluster head election techniques for coverage preservation in wireless sensor networks”, Ad-Hoc Networks Journal (Elsevier), Vol. 7, No. 5, pp. 955-972 (2009)
11. K. Benahmed, M. Merabti, H. Haffaf, “*Distributed monitoring for misbehavior detection in wireless sensor networks”,* Security and Communication Networks (Wiley), Vol. 6, No. 4, pp.388-400 (2013)
12. M. Lehsaini, H. Guyennet, M. Feham, “An efficient cluster-based self-organization algorithm for wireless sensor networks”, Int. Journal. Sensor Networks, Vol. 7, No. 1-2, pp.85-94 (2010)
13. K.A. Darabkh, SS. Ismail, M.Al-Shurman, “Performance evaluation of selective and adaptive heads clustering algorithms over wireless sensor networks”, Journal of Network and Computer Applications, Vol.35, No.6, pp.2068-2080 (2012)
14. W. Choi, M. Woo, “A Distributed Weighted Clustering Algorithm for Mobile Ad Hoc Networks”, Proc. of the IEEE Advanced International Conference on Telecommunications and International Conference on Internet and Web Applications and Services (AICT/ICIW 2006), pp.73 (2006)
15. A. Abbassi, M. Younis, “A Survey on Clustering Algorithms for Wireless Sensor Networks”, Computer Communications Journal (Elsevier), Vol.30, No 14-15, pp.2826-2841 (2007)
16. M. Chawla, J. Singhai, J. L. Rana, “Clustering in Mobile Ad- hoc Networks: A Review”, International Journal of Computer Science and Information Security, (IJCSIS), Vol. 8, No. 2, pp.293-301 (2010)
17. I. I. Er, W. K. G. Seah, “Mobility-based d-Hop Clustering Algorithm for Mobile Ad Hoc Networks”, In IEEE Wireless Communications and Networking Conference (WCNC"2004), pp.2359 -2364 (2004)
18. I. Khalil, S. Bagchi, N.B. Shroff, “LITEWORP: a lightweight Counter measure for the wormhole attack in multihop wireless networks”, International Conference on Dependable Systems and Networks, pp612-621 (2005)
19. Hsin, M. Liu, “Self-monitoring of wireless sensor networks”, Computer Communications (Elsevier), Vol. 29, No. 4, pp. 462-476 (2006)
20. R.A. Shaikh, H. Jameel, S. Lee, Y.J. et al., “Trust management problem in distributed wireless sensor networks”. Proceedings of the 12th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA), pp.411-414 (2006)
21. T. H. Hai, E.N. Huhi, M. Jo, “A lightweight intrusion detection framework for wireless sensor networks”, Wireless Communications and Mobile Computing (Wiley), Vol. 10, No. 4, pp.559-572 (2010)