

COMBINING FUZZY AND CELLULAR LEARNING AUTOMATA METHODS FOR CLUSTERING WIRELESS SENSOR NETWORK TO INCREASE LIFE OF THE NETWORK

Javad Aramideh¹, Hamed Jelodar²

¹ Department of Computer Engineering, Sari Branch, Islamic Azad University, Sari, Iran, e-mail: Aramideh.javad@gmail.com

² Department of Computer Engineering, Bushehr Branch, Islamic Azad University, Bushehr, Iran, e-mail: JelodarH@gmail.com

Received: 2014.09.25
Accepted: 2014.10.25
Published: 2014.12.01

ABSTRACT

Wireless sensor networks have attracted attention of researchers considering their abundant applications. One of the important issues in this network is limitation of energy consumption which is directly related to life of the network. One of the main works which have been done recently to confront with this problem is clustering. In this paper, an attempt has been made to present clustering method which performs clustering in two stages. In the first stage, it specifies candidate nodes for being head cluster with fuzzy method and in the next stage, the node of the head cluster is determined among the candidate nodes with cellular learning automata. Advantage of the clustering method is that clustering has been done based on three main parameters of the number of neighbors, energy level of nodes and distance between each node and sink node which results in selection of the best nodes as a candidate head of cluster nodes. Connectivity of network is also evaluated in the second part of head cluster determination. Therefore, more energy will be stored by determining suitable head clusters and creating balanced clusters in the network and consequently, life of the network increases.

Keywords: wireless sensor network, clustering, fuzzy logic, cellular learning automata.

INTRODUCTION

Wireless sensor networks which are used to collect information and study an environment multilaterally are composed of cheap sensors which have been dispersed in an environment and information collected with sensors should be transferred to a basic station. Since the sensor nodes have limited weight, size and cost which have direct effect on access to sources and considering that nodes have batteries, processing capability and limited communication and it is not proper to replace batteries in many applications, low power consumption is one of the essential needs in these networks and life of each sensor can be effectively increased by optimizing energy

consumption. In this paper, an attempt has been made to present a combined strategy for clustering of dispersed sensors in the environment to lead to effective and efficient storage resulting in reduction of the consumed energy and increase the life of the network. In this paper, we present some performed works and the proposed method and results of simulation.

RELATED WORKS

Application of clusters for transmission of information to a base station increases advantages of short transmission to most nodes by requiring only limited number of nodes for remote transmission to base station. Clustering concerns dividing

the network into some independent clusters, each with one head cluster which collects all the information from nodes inside its cluster. This head cluster sends information to the base center after data compression and Single-hop Communication directly or in Multi-hop Communication with lower number of hop and only with head cluster nodes. Clustering can reduce communication cost of most nodes considerably because they should only send information to the nearest head cluster instead of sending it directly to the base center which may be very far [Fahmy 2004].

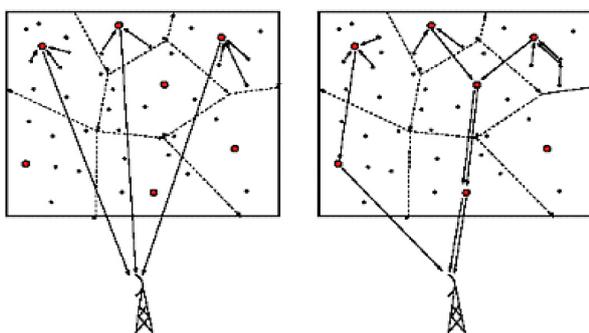


Fig. 1. Single-hop Communication and Multi-hop Communication with clustering

SOME WORKS WHICH HAVE BEEN DONE IN THE FIELD OF CLUSTERING

Xu et al. [2005] and Rosemark et al. [2005] have presented methods based on search. In these methods, a query has been created in a sink node and broadcasted in the network. Each node processes query by receiving it and sends it to its neighbors. After complete query was processed, the result will be returned to the sink node. In this scenario, some nodes process only queries and others broadcast them and obtain partial results. They aggregate results and return them to sink node.

In some works by Bontempi et al. [2005], Virrankoski et al. [2005] and Soro et al. [2005], network is first clustered and then head clusters aggregate the data received from each cluster separately.

Lotfi Nejad et al., [2004] tried to study effect of relatively dependent data on efficiency of the clustering methods in aggregation of data. They showed that dependence of data had a considerable effect on efficiency of clustering. It means that lower dependency of data lowers efficiency of clustering. They also showed that energy spent in a node is highly dependent on location

of that node in the network and the node which has longer distance from sink node consumes more energy and they also showed that life of the network had reverse relationship with energy consumption rate.

Other methods have been suggested for aggregation of data in the sensor network and Guestrin et al., tried to estimate data of a node in a time interval with a cubic function and send polynomial coefficients to sink node instead of sending data [Guestrin et al. 2004].

PROPOSED METHOD

As mentioned above, clustering is one of the considered methods for reduction of energy consumption in the wireless sensor networks. Therefore, in this paper, we decide to present a method for clustering of sensor network by combining two fuzzy method and cellular learning automata method to increase life of the network by reducing energy of the network. The proposed method has composed of several sections which are mentioned as follows.

Specifying candidate nodes for being head cluster

At the beginning of the proposed method, we will specify a candidate sensor nodes for being head cluster and for this purpose, we use fuzzy method. In this case, we use 3 factors effective on candidacy of a node for being head cluster in a network as input of fuzzy system. These factors include:

- 1) number of neighbor node (the nodes which are in communication range of each sensor),
- 2) remaining energy of sensor node,
- 3) distance between node and sink node.

Therefore, the used fuzzy system is composed of 3 inputs and 1 output and each of the inputs and also output of the system are divided into classes and we assign verbal words to each class and verbal words are classified and assigned to each input and output.

Two inputs of the number of neighboring node and distance to sink node were divided into 3 classes with low, medium and high verbal words.

The third input i.e. remaining energy of the sensor node was divided into 4 low, medium, high and very high classes.

Only output of the fuzzy system which specifies probability of candidacy of the sensor node for becoming head cluster was also divided into five very low, low, medium, high and very high classes.

In the above fuzzy system for AND and OR, Min and Max operators and Center of gravity defuzzifier. General scheme of fuzzy system, membership functions for input of energy level, output, fuzzy laws and a sample of test for determination of candidature of a sensor is shown and simulation of all of them has been done in the MATLAB software.

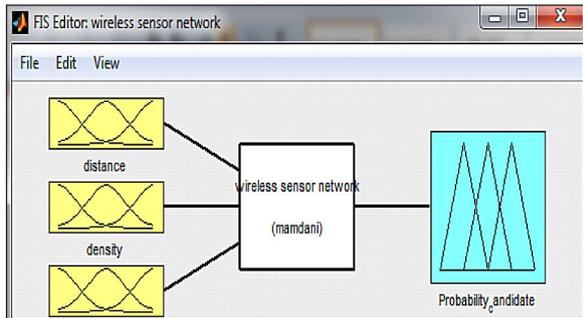


Fig. 2. General scheme of the presented fuzzy system

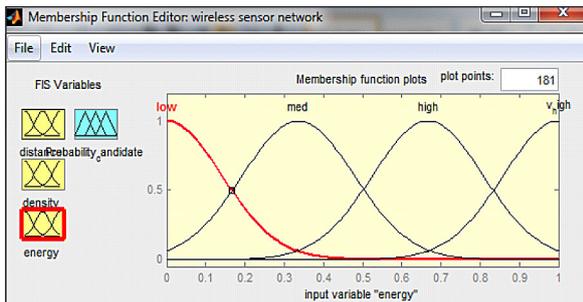


Fig. 3. Membership function relating to energy level

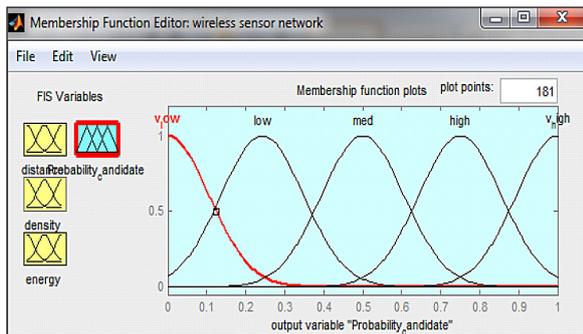


Fig. 4. Membership function relating to output

Set of the fuzzy laws applied in the discussed fuzzy system is as follows:

- If (distance is low) and (density is low) and (energy is low) then (Probability_candidate is v_low)
- If (distance is low) and (density is low) and (energy is med) then (Probability_candidate is med)

- If (distance is low) and (density is low) and (energy is high) then (Probability_candidate is med)
- If (distance is low) and (density is low) and (energy is v_high) then (Probability_candidate is high)
- If (distance is low) and (density is med) and (energy is low) then (Probability_candidate is low)
- If (distance is low) and (density is med) and (energy is med) then (Probability_candidate is med)
- If (distance is low) and (density is med) and (energy is v_high) then (Probability_candidate is v_high)
- If (distance is low) and (density is med) and (energy is high) then (Probability_candidate is high)
- If (distance is low) and (density is high) and (energy is low) then (Probability_candidate is low)
- If (distance is low) and (density is high) and (energy is med) then (Probability_candidate is high)
- If (distance is low) and (density is high) and (energy is high) then (Probability_candidate is v_high)
- If (distance is low) and (density is high) and (energy is v_high) then (Probability_candidate is v_high)
- If (distance is med) and (density is low) and (energy is low) then (Probability_candidate is v_low)
- If (distance is med) and (density is low) and (energy is high) then (Probability_candidate is med)
- If (distance is med) and (density is low) and (energy is v_high) then (Probability_candidate is high)
- If (distance is med) and (density is med) and (energy is low) then (Probability_candidate is v_low)
- If (distance is med) and (density is med) and (energy is med) then (Probability_candidate is low)
- If (distance is med) and (density is med) and (energy is high) then (Probability_candidate is med)
- If (distance is med) and (density is med) and (energy is v_high) then (Probability_candidate is high)
- If (distance is med) and (density is low) and (energy is med) then (Probability_candidate is v_low)
- If (distance is med) and (density is high) and (energy is low) then (Probability_candidate is low)
- If (distance is med) and (density is high) and (energy is med) then (Probability_candidate is med)
- If (distance is med) and (density is high) and (energy is high) then (Probability_candidate is high)
- If (distance is med) and (density is high) and (energy is v_high) then (Probability_candidate is v_high)
- If (distance is high) and (density is low) and (energy is low) then (Probability_candidate is v_low)
- If (distance is high) and (density is low) and (energy is med) then (Probability_candidate is low)

If (distance is high) and (density is low) and (energy is high) then (Probability_candidate is med)
 If (distance is high) and (density is low) and (energy is v_high) then (Probability_candidate is high)
 If (distance is high) and (density is med) and (energy is low) then (Probability_candidate is low)
 If (distance is high) and (density is med) and (energy is med) then (Probability_candidate is med)
 If (distance is high) and (density is med) and (energy is high) then (Probability_candidate is high)
 If (distance is high) and (density is med) and (energy is v_high) then (Probability_candidate is v_high)
 If (distance is high) and (density is high) and (energy is low) then (Probability_candidate is low)
 If (distance is high) and (density is high) and (energy is med) then (Probability_candidate is high)
 If (distance is high) and (density is high) and (energy is high) then (Probability_candidate is v_high)
 If (distance is high) and (density is high) and (energy is v_high) then (Probability_candidate is v_high)

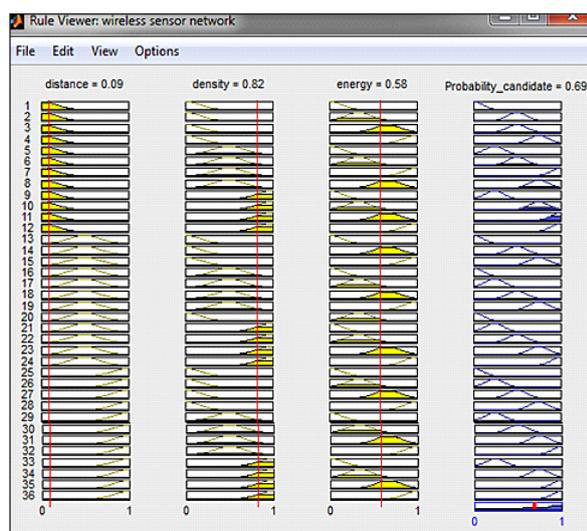


Fig. 5. A sample of a performed test for determination of candidate probability of a sensor\

It is necessary to note that the nodes are introduced as candidate node when the calculated probability in output of the fuzzy system is higher than the average rate.

Selection of head clusters among the candidate nodes

In this stage, we use cellular learning automata for determination of the certain condition (becoming or not becoming head cluster) of the nodes which have introduced them as node of its

candidate head cluster in the previous stage. Since each node of the sensor has two conditions, then each one of the candidate nodes can select one of two states of cluster head and common node. For this work, we consider a learning automata corresponding to each one of the candidate sensors and the corresponding automata can select one of two states of CH and CN based on their probability vector where CH indicates cluster head and CN indicates common node and probability of selecting each one of two actions is equal to 0.5. Probability of being cluster head based on different parameters decreases or increases based on different parameters with each selection. The parameters which we use for determination of cluster head among the candidate nodes include:

- **Energy rate.** Since cluster head node should collect information and send it to the sink node, it consumes more energy than other nodes. Therefore, attempt is made to select a node which has more energy than its neighbors. For this purpose, we use a difference between node energy and mean energy of its neighbors.
- **Number of neighbor nodes.** Since the energy consumption rate in cluster head node is high, an attempt should be made to select the lowest number of cluster head among the candidate nodes, therefore, the candidate nodes which should be selected for becoming a cluster head when the number of its neighbors is higher than the mean number of its neighbor nodes, otherwise, the selected candidate node is found for becoming cluster head.
- **Number of neighbor cluster head nodes.** One of the important criteria for clustering is connectivity. It means that each node should be able to send its information to sink node. Therefore, either each node should have a cluster head node or one of its neighbors should be cluster head. Hence, in case a candidate node is selected as common node so no cluster head node is its neighbor, it will be finally fined. Another important point is that since the number of cluster heads of the network should not be high, in case the candidate node which is selected as cluster head had neighbor cluster head node should be fined.
- **Capacity of cluster.** One of the connectivity criteria is maximum size of cluster. If the number of nodes of a cluster is high, the consumed energy of cluster head will be high for receiv-

ing and sending data and information may be lost on the other hand. Therefore, considering the capacity of the cluster, more balanced clusters will be created. Then, in case one of the candidate nodes is selected as common node and ratio of the number of its common neighbors to the number of its cluster head neighbors is higher than the capacity of cluster, it will be fined and we consider the capacity of cluster equal to $1.13N$, based on some performed works.

Considering the above-mentioned parameters, probability of performing that action will change by selecting each action with node automata. For rewarding and penalty, we use Relations (1) and (2), where a is a reward coefficient and b is a penalty coefficient.

$$p_i(n+1) = p_i(n) + a[1 - p_i(n)] \quad (1)$$

$$p_i(n+1) = 1 - p_i(n+1)$$

$$p_i(n+1) = (1 - b) p_i(n) \quad (2)$$

$$p_i(n+1) = 1 - p_i(n-1)$$

In each round of node, reinforcement signal i_β is calculated after selecting an action based on Relations (3) and (7). If i_β is equal to 1, the selective action is fined with Relation (2) and if it is equal to 0, it will be rewarded in Relation (1). If the selective action is head clustering, value of i_β is obtained from Relation (3):

$$\beta_i(n) = \begin{cases} 0 & : w_e \psi_i(n) + w_e (\lambda_i / 2 + \tau_i(n) / 2) > 0 \\ 1 & : \text{otherwise} \end{cases} \quad (3)$$

So that:

$$\psi_i(n) = \frac{|e_i(n) - e_\mu(n)|}{e_i(n) - e_\mu(n)}$$

where: $e(n)$ is the remaining energy in the n -th round and $e_\mu(n)$ is mean energy of node.

$$\lambda_i = \frac{|N_i - N_\mu|}{N_i - N_\mu} \quad (4)$$

where: N_i is the number of neighbor node and N_μ is the mean number of neighbors in the neighbor nodes.

$$\tau_i(n) = \begin{cases} 1 & : \sum_{j=0}^{N_i} \alpha_j(n) > 0 \\ -1 & : \text{otherwise} \end{cases} \quad (5)$$

where: $\alpha_j(n)$ is the selective action of node j .

Constant coefficient of w_e is the weight given parameters of energy in the clustering algorithm.

Constant coefficient of w_u is the weight given to parameters relating to quality of clustering infrastructure and these two coefficients are between 0 and 1 so that sum of coefficients becomes equal to 1. If the selective action is a selection of node as a common node, value of β_i will be obtained from Relation (6):

$$\beta_i(n) = \begin{cases} 0 & -w_e \psi_i(n) + w_e (-\lambda_i / 2 + \gamma_i(n) / 2) > 0 \\ 1 & \text{otherwise} \end{cases} \quad (6)$$

So that:

$$\gamma_i(n) = \begin{cases} 1 & \sum_{j=0}^{N_i} \alpha_j(n) > 0 \quad \text{and} \quad \frac{N_i}{\sum_{j=0}^{N_i} \alpha_j(n)} < MCS \\ -1 & \text{otherwise} \end{cases} \quad (7)$$

This stage is performed in several rounds. In each round, each sensor node selects one of the conditions of becoming or not becoming cluster head based on its probability vector and broadcasts a message to all of its neighbors and the message which contains node selective action, remaining energy and the number of its neighbors. After the specified time which all nodes received message of their neighbors, each node is rewarded or fined based on its selective action based on Relations (1) and (2) and increases or decreases probability of becoming or not becoming cluster head. In the next round, nodes select new state based on new probability and repeat operations.

Each node of which probability of becoming head cluster reaches zero or one selects a state based on probability vector and reaches stable condition. This algorithm continues until a clear percent of candidate nodes (95% in the performed tests) reach stable conditions. At this time, all nodes select their state and this stage ends. The status of the major part of the candidate nodes has been determined while status of few candidate nodes may not be determined. There are statuses where some nodes of which status has been determined as a common node are not in neighborhood of any cluster head node. Here, we add some candidate nodes to the cluster head nodes due to coverage of such common nodes and they are selected as cluster head node based on having abundant neighbors, compared with the adjacent nodes.

Formation of cluster

After specifying the status of the candidate nodes and determining cluster heads in the previous stages, it is time to create clusters. In this stage, the cluster head nodes broadcast a message

based on their geographical position to the adjacent nodes and then common nodes select the nearest node as cluster head among cluster head neighbors. To create cluster, each common node sends a packet to its cluster head and identifies members of the cluster by collecting these packets and form cluster and then broadcasts them to all nodes of the cluster based on time schedule. Time for submission of packet is specified by each one of the members.

Data transmission

In this stage, common nodes send their data to cluster head nodes alternatively and with specified time intervals and node of the cluster head aggregates packets and sends it as a single packet to the sink node after all nodes in a cluster have sent their data to cluster head. In this stage, each common node of which data was sent is activated based on time schedule and is inactivated after sending the packet to the sink node. The cluster head node will not be active in all time periods and is activated only when it wants to receive data of the member nodes.

Change of cluster head

It is evident that energy of cluster head nodes is consumed faster than other member nodes due to higher activity and the network is disrupted. Under such condition, we use change of cluster head node to prevent this problem such that the member nodes send some of their remaining energy to the cluster head node in each data submission period and after the cluster head node sent the aggregated packet to the sink node, the cluster head node would calculate mean energy of the member nodes and if this energy rate is higher than the threshold limit, the node which has more energy will become cluster head and in the new time period, each node which sends data to it responds with message of `chang_clustering` and introduces the new cluster head node [Saeedian at al. 2011].

Calculating energy consumption rate

In this paper, to calculate energy consumption rate of the sensors, it has been assumed that all sensors will send K bit data in each round. To calculate energy consumption of each sensor in each round, Relation (4) which was known as Hizelman's Relation will be used and this Relation is mentioned as follows:

$$TX = \xi_{elec} \cdot K + \xi_{amp} \cdot d^2 \cdot K \quad (8)$$

where: K – is the number of the transmitted bit in each round which is sent from each sensor node. K value is constant for all sensor nodes.

d – is distance between common sensor node and cluster head node. For the cluster head node, there is a distance between cluster head node and sink node.

Parameters ξ_{amp} and ξ_{elec} are regarded as energy which internal circuit of each sensor consumes at time of data transmission and is equal to constant values. ξ_{amp} is equal to $100 \text{ Pj/bit}\cdot\text{m}^2$ which is equivalent to $100 \cdot 10^{-12} \text{ j}$ and ξ_{elec} is equal to value of 50 nj/bit which is equivalent to value of $50 \cdot 10^{-9} \text{ j}$.

The point is that the cluster head nodes consume energy for receiving data and there are different methods to calculate energy of the cluster head at time of receiving data from member sensors. In the method which has been used in this paper, the cluster head node receives n -bit packets based on the number of member nodes and converts them into n -bit packet and sends them. The difference is that we consider the consumed energy for this conversion. This consumed energy is known as E_{DA} and its value is considered as $5 \cdot 10^{-9}$ which is added to the consumed energy value for submission of data to sink node [Al-Obaidy et al. 2008].

SIMULATION

In this paper, we evaluate the proposed method with 2 different scenarios and compare the above method with Leach's algorithm to show efficiency. In this research, attempt has been made to continue evaluation until the network has real value. For this reason, the network continues working until the number of the live nodes in the network is such that they create acceptable cover in the environment [Heinzelman et al. 2000].

In this paper, MATLAB software which is a suitable environment for simulation has been used for simulation and the constant parameters used in simulation are mentioned as follows:

- The sensor nodes are dispersed in the environment randomly and each node has range of sensor R_s and sensor environment of each node is a circle with radius of R_s .
- Base station (sink) is located in the middle of the environment.

- All nodes have equal energy and power at the beginning.
- Identification number of all nodes was recognized for sink node.
- The initial energy of all nodes is first 0.1 joules.
- In the performed tests, the network continues its activity until it has effective cover in the environment and when a node dies, its energy reaches below 0.05 joules.
- To determine effective cover of the entire plate as set of points that is we produce points 0·0 to L·L for a environment with dimensions of L·L. at the end of each round, we check data submission and some points are present in sensor range of the active nodes.

First scenario – studying death time of 10%, 20% and 30% of sensors. 100 sensors have been broadcasted in the environment with dimensions of 500·500 and sensor range of each sensor is equal to 20 and the diagram of number of live node relative to the number of executive round for the proposed method compared with LEACH algorithm (Figure 6, Table 1).

Second scenario – studying the number of effective executive round of the network. In this test, 30 sensors have been broadcasted in an environment with dimensions of 100·100 and the number of the executive round of the network has been compared with Leach algorithm until it creates effective cover in the network (here, we considered effective cover equal to cover above 59%).

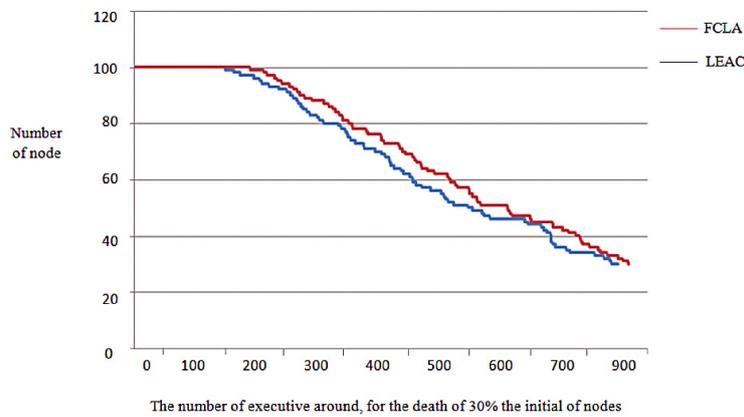


Fig. 6. Diagram of the number of live node relative to the first scenario time

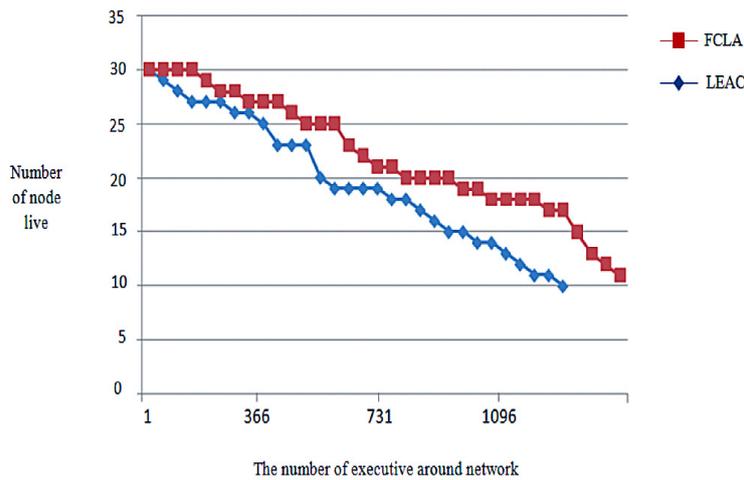


Fig. 7. Diagram of the number of executive round

Table 1. Comparing results of two algorithms in the first scenario

Method	Death time of the first node	Death time of the first 10%	Death time of the first 20%	Death time of the first 30%
FCLA	224	317	436	523
LEACH	172	275	363	457

CONCLUSION

Clustering is one of the main works for reduction of energy consumption and increase of life in the wireless sensor networks. For this reason, in this paper, attempt has been made to present a new method of clustering which is based on energy with combination of fuzzy and cellular learning automata (FCLA) and results of simulation prove that FCLA method shows clear preference over LEACH protocol in terms of increase of useful life of the network (suspension of the death time of the first node and initial death percent of nodes) and also the number of executive rounds.

REFERENCES

1. Fahmy Y.S.: Distributed clustering in ad-hoc sensor networks: A hybrid, energy-efficient approach. In: Proceedings of IEEE INFOCOM, Vol. 1, 2004, 629–640.
2. Xu Y., Lee W.C., Xu J., Mitchell G.: Processing window queries in wireless sensor networks. IEEE International Conference on Data Engineering, GA, April 2005.
3. Rosemark R., Lee W.C.: Decentralizing query processing in sensor networks. The Second International Conference on Mobile and Ubiquitous Systems. Networking and Service, CA, 2005, 270–280.
4. Bontempi G., Le Borgne Y.: An adaptive modular approach to the mining of sensor network data. Proceedings of the Workshop on Data Mining in Sensor Network, SLAM, SDM, CA, USA, April 2005.
5. Virrankoski R., Savvides A.: TASC: topology adaptive spatial clustering for sensor networks. In: IEEE International Conference on Mobile Adhoc and Sensor systems. DS, November, 2005.
6. Soro S., Heinzelman W.: Prolonging the lifetime of wireless sensor networks via Uneven clustering. Proceedings of the 5th International Workshop on Algorithms for Wirelees, Mobile, April 2005.
7. Lotfi Nezhad M., Liang B.: Effect of partially correlated data on clustering in wireless sensor networks. Proc. First IEEE Int'l Conf. Sensor and Ad Hoc Communications and Network, Santa Clara, California, October 2004.
8. Guestrin C., Bodik P., Thibaux R., Paskin M., Madden S.: Distributed regression: an efficient framework for modeling sensor network data. Intel corporation, 2004.
9. Al-Obaidy M., Ayesh A., Sheta A.F.: Optimizing the communication distance of an ad hoc wireless sensor networks by genetic algorithms. Artif. Intell. Rev. 29, 2008, 183–194.
10. Heinzelman W.R., Chandrakasan A., Balakrishnan H.: Energy-efficient communication protocol for wireless microsensor networks. IEEE, Proceedings of the 33rd Hawaii International Conference on System Sciences, 2000.
11. Saeedian E., Jalali M., Tajari M.M., Torshiz M.N., Tadayon G.H.: CFGA: Clustering wireless sensor network using fuzzy logic and genetic algorithm. IEEE, 7th International Conference, Wireless Communications, Networking and Mobile Computing (WiCOM), 2011, 1–4.