Coverage Improving with Energy Efficient in Wireless Sensor Networks

Amir Pakmehr
Department of computer engineering, Tabriz branch, Islamic azad university, Tabriz, Iran
Amir.Pakmehr@gmail.com
Ali Ghaffari*
Department of computer engineering, Tabriz branch, Islamic azad university, Tabriz, Iran
A.Ghaffari@iaut.ac.ir

Received: 17/Aug/2016 Revised: 07/Feb/2017 Accepted: 07/Mar/2017

Abstract

Wireless sensor networks (WSNs) are formed by numerous sensors nodes that are able to sense different environmental phenomena and to transfer the collected data to the sink. The coverage of a network is one of the main discussion and one of the parameters of service quality in WSNs. In most of the applications, the sensor nodes are scattered in the environment randomly that causes the density of the nodes to be high in some regions and low in some other regions. In this case, some regions are not covered with any nodes of the network that are called covering holes. Moreover, creating some regions with high density causes extra overlapping and consequently the consumption of energy increases in the network and life of the network decreases. The proposed approach causes an increase in life of the network and an increase in it through careful selection of the most appropriate approach as cluster head node and form clusters with a maximum length of two steps and selecting some nodes as redundancy nodes in order to cover the created holes in the network. The proposed scheme is simulated using MATLAB software. The function of the suggested approach will be compared with Learning Automata based Energy Efficient Coverage protocol (LAEEC) approach either. Simulation results shows that the function of the suggested approach is better than LAEEC considering the parameters such as average of the active nodes, average remaining energy in nodes, percent of network coverage and number of control packets.

Keywords: Wireless Sensor Networks; Clustering; Network Coverage; Covering Holes; Energy Efficient.

1. Introduction

Recent technological advancements in the realm of micro-electro-mechanical systems (MEMS), wireless telecommunications and digital electronics led to the production of sensors which are cheap, low-power and compact. These sensors can fulfill multi-functions such as monitoring and surveilling environments in unprotected areas and carry out telecommunication operations. Indeed, these sensors can accomplish different functions such as sensing, data processing and transmitting information which establish the rationale behind wireless sensor networks (WSNs) [1, 5, 7, 8, 24, 28]. WSNs consist of a high number of sensor nodes which are closely distributed with high density within a phenomenon. In general, there is no need for continuous monitoring of the location and position of sensor nodes; neither is it required to predetermine their locations [4, 6, 9-11]. Hence, these nodes can be randomly arranged and set in the network environment. This feature expands the application and use of WSNs to remote or dangerous areas where they can be used with minimum surveillance and monitoring. For achieving this purpose, protocols of WSNs should have the capability of self-organization. That is, the protocols and algorithms operated on WSNs are managed by the sensors themselves. Instead of transmitting all the raw data to the sink node, each sensor node has a processor which can process the data in a limited way; then, it can transmit the semi-processed data to the sink node. Since lots of sensor nodes with high-density are used in an environment, they might be very close to one another. Thus, it can be assumed that multi-hop communications can transmit data to the sink by using less power and energy than the single-hop communications. In contrast with the traditional wired networks, on the one hand, configuration and makeup costs are reduced in WSNS; on the other hand, instead of installing thousands of meters of wire, only tiny sensors the size of about a few cubic centimeters have to be distributed in the environment. By merely adding some nodes, these networks can be extended and there is no need for complex reconfiguration. One of the most important challenges in WSNs is phenomenon and area coverage [12-20]. Indeed, the method proposed in this paper is a cluster-based method for network. In this method, the maximum length of clusters was considered to be two hops which led to the reduction of hop length. Hence, the algorithm is executed within the cluster rather than being executed by all the hops. Consequently, the network can reach a steady and stable state earlier and its coverage begins earlier.

The rest of the paper is organized as follows. Section 2 briefly reviews related work on existing methods about Coverage and Energy issues. In Section 3, the proposed area

* Corresponding Author

coverage algorithm is presented. Section 4 shows the performance of the proposed algorithm through simulation experiments and comparison with the existing methods. Section 5 concludes the paper.

2. Related Work

Recently, many important related works in area coverage have proposed for WSNs [21-25]. The connected dominating set (CDS) [11] concept has recently emerged as a promising approach to the area coverage in WSN. However, the major problem affecting the performance of the existing CDS-based coverage protocols is that they aim at maximizing the number of sleep nodes to save more energy. This places a heavy load on the active sensors (dominators) for handling a large number of neighbors. The rapid exhaustion of the active sensors may disconnect the network topology and leave the area uncovered. Therefore, to make a good trade-off between the network connectivity, coverage, and lifetime, a proper number of sensors must be activated. Paper [11] presents a degree-constrained minimum-weight extension of the CDS problem called Degree-constrained CDS (DCDS) to model the area coverage in WSNs. The proper choice of the degree-constraint of DCDS balances the network load on the active sensors and significantly improves the network coverage and lifetime. A learning automata-based heuristic named as LAEEC [11] is proposed for finding a near optimal solution to the proxy equivalent DCDS problem in WSN. The computational complexity of the proposed algorithm to find a $\frac{1}{1-\epsilon}$ optimal solution of the area coverage problem is approximated.

In energy-limited wireless sensor networks [3], network clustering and sensor scheduling are two efficient techniques for minimizing node energy consumption and maximizing network coverage lifetime. When integrating the two techniques, the challenges include how to decide the most energy-efficient cluster size and how to select cluster heads and active nodes. [3] paper provide a computation method for the optimal cluster size to minimize the average energy consumption rate per unit area. In the proposed coverage-aware clustering protocol, define a cost metric that favors those nodes being more energy-redundantly covered as better candidates for cluster heads and select active nodes in a way that tries to emulate the most efficient tessellation for area coverage.

An energy-aware distributed unequal clustering protocol in multi-hop heterogeneous wireless sensor networks [23] is proposed. It elects cluster heads based on the ratio between the average residual energy of neighbor nodes and the residual energy of the node itself, and uses uneven competition ranges to construct clusters of uneven sizes. The cluster heads closer to the BS have smaller cluster sizes to preserve some energy for the inter cluster data forwarding, which can balance the energy consumption among cluster heads and prolong the network lifetime.

In [2], the authors provide a novel algorithm using trees and graph theory to detect and describe the existing holes in the region of interest. Simulation results show that the tree-based method can indicate the location, size, and shape of coverage holes accurately. Based on the results for hole detection, a tree-based healing method is also proposed. The method is divided into two phases, namely, hole dissection and optimal patch position determination. On [2] Results obtained from the experimental evaluation reveal that the proposed healing method can increase the coverage rate with only a few additional sensors compared to other related methods.

On connected target *k*-Coverage in heterogeneous wireless sensor networks in [26], focus on the connected target *k*-coverage (CTCk) problem in heterogeneous wireless sensor networks (HWSNs). A centralized connected target *k*-coverage algorithm (CCTCk) and a distributed connected target *k*-coverage algorithm (DCTCk) are proposed so as to generate connected cover sets for energy-efficient connectivity and coverage maintenance. To be specific, the proposed scheme in [26] aims at achieving minimum connected target *k*-coverage, where each target in the monitored region is covered by at least *k* active sensor nodes.

In [27], the authors proposed an integrated and energy-efficient protocol for Coverage, Connectivity and Communication (C3). C3 protocol runs in six steps. 1. *Formation of rings:* The C3 protocol divides the network into virtual concentric rings, based on the communication range (*Rc*), using received signal strength indicator (RSSI) distance estimator. 2. *Formation of clusters*: A cluster head is selected alternately from even or odd numbered rings, in a round. 3. *Formation of dings*: A ding is a subsection of ring with a cluster head. The cluster head identifies the nodes which are at the distance of $\sqrt{3}R_s$ to form the ding. Therefore, there might be multiple dings inside a ring. 4. I*dentification of redundant nodes*: C3 protocol uses triangular tessellation based on *Rs* inside the dings to identify redundant nodes. The redundant nodes can enter into sleep state for a time duration of T. 5. *Establish connectivity:* C3 protocol establishes connectivity between neighboring nodes and cluster head. 6. *Communication: Finally*, in C3 protocol data are transmitted to the sink node with the help of cluster heads.

3. Proposed Work

Before introducing the novel proposed method, the drawbacks of LAEEC [11] method are first outlined and some solutions for sorting out those shortcomings by means of the proposed method are mentioned. In the LAEEC [11] protocol, the whole network nodes are fixed and are randomly distributed in the network. Also, the network is not clustered in this protocol. The way of executing the LAEEC [11] algorithm in the network is as follows. At first, one node is randomly selected by the

sink node; then, the algorithm is executed by the selected node. Then, the node selected by the sink randomly chooses another node for executing the algorithm. This procedure is repeated until all the respective nodes of the network execute the algorithm. This operation will cause delays in algorithm execution in the network. In LAEEC algorithm, the failure or death of a node has the role of cut-vertex which leads to the division of the network into two distinctive parts. Consequently, firstly, the algorithm is executed in an infinite loop by the nodes of the first area of the network and it is never ended, unless a specific time is considered for the execution of the algorithm. Secondly, due to the lack of communication between the first and second areas, the nodes of the second area are never selected by the nodes of the first zone for executing the algorithm. as a result, the algorithm is executed incompletely. Cut-vertex is a vertex of the graph which will lead to an increase in the number of the connective variables of graph if it is removed. In case the graph is connective before cut-vertex is removed, then, it will be non-connective after it is removed. Cut-vertex is of high significance in computer networks.

In the LAEEC [11] algorithm, a node might not be located within the coverage range of the last node which executes the algorithm; hence, it is not selected as the node for executing the algorithm. the unselected node will result in the placement of the algorithm in an infinite loop. It should be noted that any particular predictions were not considered for such unselected nodes. However, in the proposed method, since the network is clustered, there is no need for gauging and scaling all the nodes for algorithm execution. In case a node does not receive any messages from its neighbors within a specific time, it will announce itself as the cluster head.

One more shortcoming of the LAEEC [11] approach is that environment impact was used for rewarding actions but the selection of action was done randomly. Nevertheless, in the proposed method, when a node is gone due to failure or energy exhaustion, the cluster head and redundant node make choices based on the parameters of the node degree and the remaining energy of the node. The respective parameters and assumptions of the proposed method are discussed below.

3.1 Parameters and Assumptions used in the Proposed Method

Table 1 shows the parameters and their descriptions.

Table 1. Parameters and their description

Parameter	description
S	The set of homogenous nodes which are distributed in the network
S_u, S_v	Two different nodes which are the subset of S
S_{CH}	The set of cluster heads
CH_k	Cluster head for the cluster number k
S_{H1}	The set of the single-hop members of cluster
S_{H2}	The set of the two-hop members of cluster
$S(ni)$	The set of nodes which overlap with the n_i node
K	The number of overlapping neighbors, in case a node has K overlapping neighbors, it can be regarded as a redundant node.
$C(ni)$	The set of nodes which have the minimum required coverage in comparison with n_i. The minimum required coverage refers to the ratio of the number of the positions of the neighboring nodes to the total positions of the respective node which is an optional number. In the proposed method, the minimum required coverage was regarded 60%.
Frame-number	The maximum number of the transmitted data frames throughout a cycle which was 100 frames in the present study.
Node-degree (S_v)	the number of neighbors of the S_v node so that: NodeDegree(S_v) = count({Su | dist(Sv, S_u)<Rc , Su ∈S, $S_u \neq S_v$})

The node S_u will be regarded as the neighbor of S_v node if node S_u is located within the radio range of S_v. In the following, the stages of the proposed method are described. It should be pointed out that the proposed method has the four stages of set-up, cluster head selection and cluster formation, identification of redundant nodes, gap detection and the movement of redundant nodes for covering the available gaps. The detailed description of each stage is given below.

3.2 Set-up

In this stage, a number of sensor nodes are randomly distributed in a two-dimensional space. These sensors have the capability of dynamicity. In the proposed method, it was assumed that each sensor node is aware of the geographical position by means of a Global Positioning System (GPS) machine or via other network positioning methods. It was assumed that all the sensor nodes are active in the network set-up stage. After sensors are distributed in the network, these sensors broadcast a hello message in the respective environment to identify all of their neighbors and transmit some data to them regarding their won state.

3.3 Selecting Cluster Head and Cluster Formation

In this stage, three messages are used which include the followings:

A. *State message:* using this message, each sensor transmits some information and data to their neighbors about their own state including node degree, the remaining energy, etc. for being selected as the cluster head.
B. *Cluster head (CH) message:* a node which considers itself as a cluster head candidate uses this message to announce its candidacy and some information about itself to neighbors.
C. *Join message:* each sensor node uses this message to introduce itself as the single-hop or two-hop member of a particular cluster head.

In this phase, an attempt is made to select cluster heads from the congested areas of the network. For achieving this objective, a variable called node degree was defined. Indeed, node degree refers to the number of single-hop neighbors of a node in the network. The way of measuring degree for each node is as follows. After network set-up stage and the deployment of the sensors in the intended environment, a set-up message is broadcast by the sink to all the network nodes. Upon receiving this message, each sensor node transmits its data to all of its neighbors within a random time interval from zero to T_{max}. By receiving this message, each of the neighbors can

determine their own degree by investigating the contents of the message. The reason for using a random time interval between zero and T_{max} is to reduce collision while transmitting hello messages. In the next step, each sensor node waits for a specific time (*Tni*) so that it can receive the message of cluster head candidacy from the closest cluster head candidate. In case it does not receive a message within a specific time interval, it will announce itself as the cluster head and will broadcast its cluster head message along with some other data about its state in the network. For selecting a node with the maximum degree as the cluster head, the T_{ni} time of this message in proportion to its degree should be measured based on Equation (1) [25].

$$T_{n_i} = \alpha . e^{\frac{1}{degree_{(ni)}}} \quad (1)$$

On equation (1) where α is a constant coefficient and factor which should be arranged and adjusted in a way that $0 < T_{n_i} \leq T_{max}$.

it might happen that two nodes within the same area not only have higher degrees than their neighbors, but also they have the identical degrees. Hence, in that area, two nodes which are close to one another will simultaneously announce cluster head candidacy. Thus, instead of becoming members of one cluster head, some nodes become members of one cluster head and some other nodes become members of the other cluster head. Consequently, the number of clusters will increase in the area which contradict the objective of minimizing the number of clusters in the network. For sorting out this problem and selecting the most efficient node as the cluster head in the area, the remaining energy of the candidates are taken into consideration. As a result, equation (1) changes into equation (2) and the node with the highest remaining energy is selected as the cluster head. Hence, there will be only one cluster rather than two clusters in one area.

$$T_{n_i} = \alpha . e^{k} \quad (2)$$

And k defined as Equation (3)

$$k = \frac{1}{\left(degree_{(ni)} + \left(\frac{E_{res(ni)}}{E_{ini_{(ni)}}} \right) \right)} \quad (3)$$

Where in Equation (3), $E_{res(ni)}$ is residual energy of node and $E_{ini(ni)}$ is the initial energy of node.

According to Equation (3), each sensor node should measure the valid threshold energy for cluster head before it announces itself as a candidate for cluster head. Hence, in case its remaining energy is less than the threshold value, it will consider T_{ni} as the maximum value so that it will not introduce itself as the cluster head as far as possible.

According to Equation (4), *L* stands for the length of data and *dist(BS,Sv)* denotes the distance between S_y and the sink [25].

$$E_{th}(S_v) = L \times ((E_{elec} + E_{DA}) \times (S_v + 1) + \in_{mp} \times d\,(BS, S_v)^4) \times F_n \quad (4)$$

In the next stage, upon receiving the message about cluster head candidacy from the closest candidate node, it transmits a message and introduces itself as the member of that cluster head. Also, as the cluster head receives this message, it stores the data of the member node in its database and, from that moment, it regards it as a single-hop member. Furthermore, as the neighboring nodes of that single-hop member receive the cluster head candidacy message, they introduce themselves as the two-hop members of that cluster head. In general, single-hop and two hop neighbors for one sensor are defined in the following way: if the distance between A and B sensors is less than *Rs* and *d(A, B)= Rs* is true, these two neighbors are single-hop neighbors of one another. In other words, these two nodes can directly communicate with each without intermediaries. If the distance between A and B nodes is in the way *Rs<dAB,=2Rs,* A and B nodes are regarded as the two-hop neighbors and they can communicate with one another via the help of one single-hop neighbor.

After cluster heads are selected and network clusters are formed, cluster heads specify a certain time for each of their members for transmitting data; hence, each member should transmit its data to the cluster head only within the specified time interval. The time duration for data transmission by each member is announced by means of message to each of the other members. Thus, scheduling and time planning among the members which is managed by the cluster heads prevents data collision while transmitting data to the cluster head. For maintaining a balance for network load, re-clustering is carried out at the beginning of each cycle.

3.4 Identifying Redundant Nodes

In this stage, the cluster head identifies redundant and unnecessary nodes with regard to covering an area and, if possible, inactivates them. The merit of this action is that in case some nodes are destroyed in an area and the respective area covered by them is destroyed, the redundant nodes can be transmitted to such areas so as to cover them. Different methods have been proposed for determining redundant nodes. As shown in Figure 1, in the proposed method, 3 n-polygons surrounded within a circle with r radius were used for detecting redundant nodes.

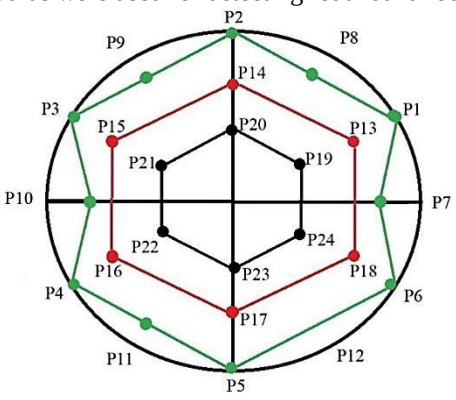

Fig. 1. 3 n-polygons surrounded within a circle with r radius were used for detecting redundant nodes.

Redundant nodes arte detected and identified throughout two stages in the following way:

A. Identifying single-hop redundant nodes by the cluster head:

1. In this step, the cluster head specifies certain nodes from among single-hop members which have *c(ni)* equal to or larger than k and stores their numbers in a temporary set $S(n_i)$. Then, their redundancies are determined through stages 2 and 3.
2. In this step, the cluster head selects a node like A from the $S(n_i)$ set with the lowest remaining energy as the redundant node.
3. After the selection of node as the redundant node, each active member of the *S(ni)* set with active members more than or equal to k will be also considered as a redundant node. this step will be repeated until no active node which is a member of *S(ni)* has the required conditions for being redundant. In case a redundant node is selected as the communication bridge between the cluster head and a node which is a member of a two-hop cluster, that node will be activated.

B. Identifying two-hop redundant nodes by the single-hop member nodes of the cluster

The steps for identifying two-hop redundant nodes are as follows:

1. In this step, each single-hop cluster member node specifies the nodes from among single-hop members with *C(ni)* sets which are greater or equal to k and stores them in the *S(ni)* temporary set. Then, it determines their redundancy through steps 2 and 3.
2. In this step, at first, single-hop member of the cluster head selects a node, namely A form the *S(ni)* set which has the lowest remaining energy as the redundant node.
3. Then, after the selection of *A* as the redundant node, if each active member of *S(ni)* has active *C(ni)* members which are greater than or equal to k, that node will also be selected as the redundant node. The third step is repeated until no active *S(ni)* member has the required conditions for being redundant.

After the cluster is produced, the cluster head uses the information obtained from the single-hop and two-hop members of the cluster to identify the available gaps in the cluster and covers them by using redundant nodes. In the following section, the way of identifying gaps and covering them are discussed.

3.5 Identifying Gaps and the Movement of the Redundant Nodes for Covering Available Gaps

In this stage, as shown in Figure 2, the cluster head surrounds a hexagon within the circle of its radio range and measures the coordinates of the specified points on the hexagon. Then, it investigates whether the obtained points are covered by the single-hop members or not? In case these points are not covered by the single-hop members, they will be considered as gaps. If *p* variable is assumed to denote the number of inactive single-hop members of the cluster head which have been selected as redundant nodes, then, the cluster head can cover $\frac{P}{2}$ of the gaps by means of the redundant inactive nodes. For instance, for covering two gaps, at least four redundant nodes should be used.

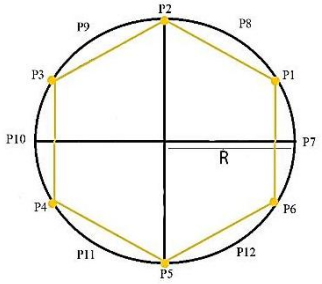

Fig. 2. the cluster head surrounds a hexagon within the circle of its radio range

3.6 Energy consumption model

Energy Consumption of the sensing device should be minimized and sensor nodes should be energy efficient since their limited energy resource determines their lifetime. The Energy consumption model should be measured based on equations (5), (6), (7)

$$E_{Tx} = k[E_{elec} + \varepsilon_{amp} d^2] \quad (5)$$

$$E_{Rx} = kE_{elec} \quad (6)$$

$$E_{Tot} = E_{Tx} + E_{Rx} = k\,[2E_{elec} + \varepsilon_{amp} d^2] \quad (7)$$

In above equations, *K* is packet size (bit), *d* is the distance between the sender and the receiver and E_{elec} is constant value.

4. Performance Evaluation

At first, a WSN in an environment the size of 100m×100m square meter including 50 to 250 sensor nodes which were randomly and steadily distributed with steady distribution was simulated by the software. Table 2 shown the parameters that used in the simulation.

Table 2. Parameters used in the simulation

Parameter	Value
Number of nodes	50-250
Reception range radius	10 m
Transmission range radius	20 m
Network area	100m×100m
The number of algorithm execution time	10
Sink location	50×50 m^2
	75×75 m^2
	25×25 m^2
α	0.18
K	2
E_{elec}	*50nj/bit*
E_{DA}	*5nj/bit/ message*
The size of data packets	4000 b
The size of control packets	200 b
ε_{fs}	100 Pj/bit/m^2
E_{elec}	0.0013 Pj/bit/m^4
initial energy	2 j

The efficiency of the proposed method was compared with that of the LAEEC [11] algorithm with respect to the following parameters: average number of active nodes,

average remaining energy of the sensors, the percentage of network coverage and the average number of control packets. The comparison of the evaluation parameters for the two algorithms are discussed below.

4.1 Comparing the Two Algorithms with Regard to the Average Number of Active Nodes

Regarding the average number of active nodes, the efficiency of the proposed method was compared with that of the LAEEC [11] algorithm under three different scenarios.

- **First scenario: sink in the [50 50] coordinate**

Figure 3 illustrates the average number of active nodes in relation to the number of network nodes within the range from 50 to 250 nodes. The proposed algorithm was executed within 1500 seconds. As shown in Figure 3, when the total number of network nodes was 50, the number of active nodes in the proposed method and the LAEEC protocol were about 38 and 43, respectively. In this way, when the total number of network nodes was 200, the number of active nodes in the proposed method and the LAEEC protocol were 113 and 142, respectively. As the average number of network nodes, active nodes of the network increases too which is illustrated in Figure 3. It should be noted that the degree of increase in the proposed method was less than that in the LAEEC protocol.

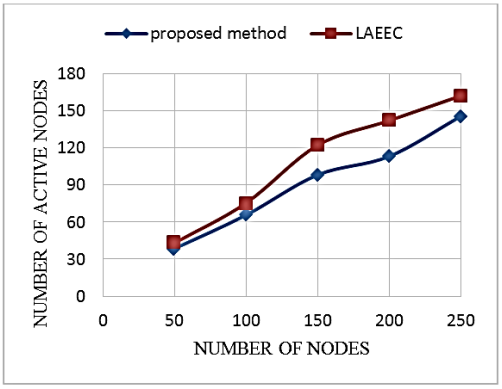

Fig. 3. The number of active nodes vs. the total number of network nodes.

- **Second scenario: sink in the [75 75] coordinate**

Figure 4 illustrates the average number of active nodes in relation to the number of network nodes within the range from 50 to 250 nodes. The proposed algorithm was executed within 1500 seconds. As shown in figure 4, when the total number of network nodes was 50, the number of active nodes in the proposed method and the LAEEC [11] protocol were about 37 and 44, respectively. In this way, when the total number of network nodes was 200, the number of active nodes in the proposed method and the LAEEC [11] protocol were 137 and 148, respectively. As the average number of network nodes, active nodes of the network increases too which is illustrated in Figure 4. It should be noted that the degree of increase in the proposed method was less than that in the LAEEC [11] protocol.

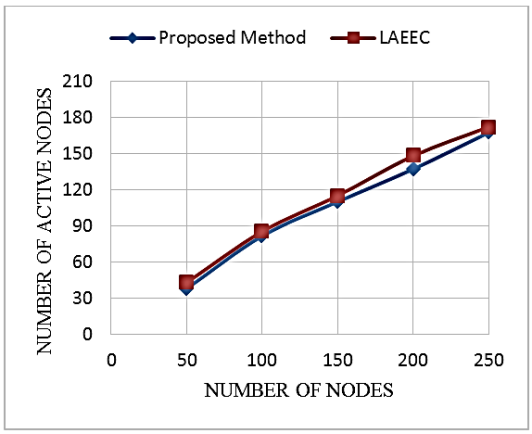

Fig. 4. The number of active nodes vs. the total number of network nodes.

4.2 Comparison of the Average Remaining Energy in Nodes

Regarding average remaining energy in the nodes, the efficiency of the proposed method was compared with the LAEEC method under three different scenarios.

- **First scenario: sink in the [50 50] coordinate**

Figure 5 shows the remaining energy in the nodes in relation to the number of network nodes within the interval from 50 to 250 nodes. After the execution of the algorithm for 1500 seconds, when the total number of network nodes is 50, the average remaining energy of the total nodes of the network in the proposed and LAEEC [11] methods were about 0.8 and 0.6 joules, respectively. In a similar vein, when the total number of network nodes was 200, the average remaining energy for the network nodes in the proposed and LAEEC [11] methods were about 1.3 and 1.2 joules, respectively. As shown in Figure 5, as the number of networks nodes increases, the average remaining energy of the network nodes in the proposed method was greater than that of the LAEEC [11]protocol. Consequently, it can be argued that network life time in the proposed method was greater than the network life time in the LAEEC [11] protocol.

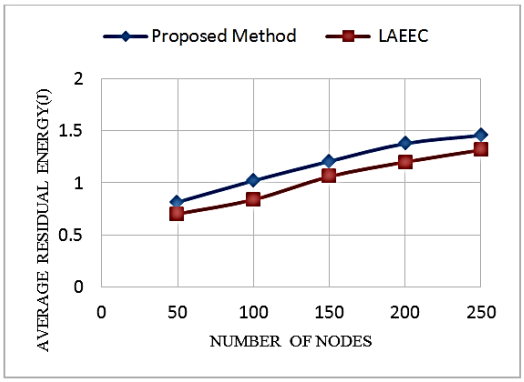

Fig. 5. The average residual energy of the active nodes as a function of the number of nodes.

- **Second scenario: sink in the [75 75] coordinate**

Figure 6 shows the remaining energy in the nodes in relation to the number of network nodes within the interval from 50 to 250 nodes. After the execution of the algorithm for 1500 seconds, when the total number of network nodes is 50, the average remaining energy of the total nodes of the network in the proposed and LAEEC [11] methods were about 0.6 and 0.8 joules, respectively. In a similar vein, when the total number of network nodes was 200, the average remaining energy for the network nodes in the proposed and LAEEC [11] methods were about 1.2 and 1.1 joules, respectively. As shown in Figure 6, as the number of networks nodes increases, the average remaining energy of the network nodes in the proposed method was greater than that of the LAEEC [11] protocol. Consequently, it can be argued that network life time in the proposed method was greater than the network life time in the LAEEC [11] protocol.

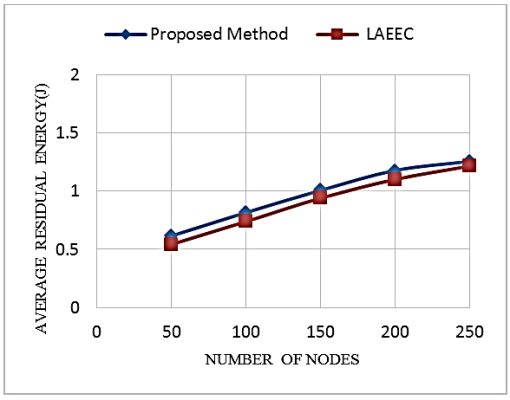

Fig. 6. The average residual energy of the active nodes as a function of the number of nodes.

4.3 Comparison of the Percentage of the Network Coverage

In this Section, the efficiency of the proposed method was compared with that of the LAEEC protocol in terms of network coverage under three different scenarios.

- **First scenario: sink in the [50 50] coordinate**

Figure 7 illustrates network coverage with respect to the number of network nodes within the range from 50 to250 nodes. Algorithm execution was completed after 1500 seconds. When the total number of network nodes is 50, 80% of the network was covered in the proposed but 71% of the network was covered in the LAEEC [11] protocol. Similarly, when the total number of network nodes was 200, 97% of the network in the proposed method and 93% of the network in the LAEEC [11] protocol were covered. In general, as the number of nodes increases, the percentage of network coverage increases too. Nevertheless, it should be highlighted that the percentage of network coverage in the proposed method was significantly greater than that of the LAEEC [11] method.

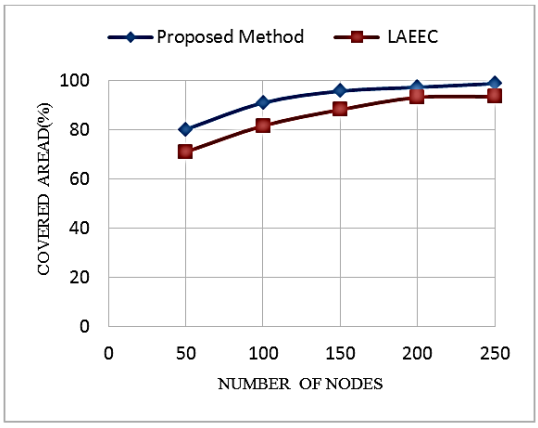

Fig. 7. The percentage of the covered area vs. the network size.

- **Second scenario: sink in the [75 75] coordinate**

The results of this scenario we saw no any difference with previous scenario on area coverage percent.

4.4 Comparison of the Average of Control Message Number

Figure 8 illustrates network average of control message number within the range from 50 to250 nodes. Algorithm execution was completed after 1500 seconds. When the total number of network nodes is 50, average of control message number in the proposed method and the LAEEC protocol were about 70 and 110, respectively. In this way, when the total number of network nodes was 200, average of control message number in the proposed method and the LAEEC protocol were 208 and 397, respectively. As the average number of network nodes, number of control message of the network increases too which is illustrated in Figure 8. It should be noted that the degree of increase in the proposed method was less than that in the LAEEC protocol.

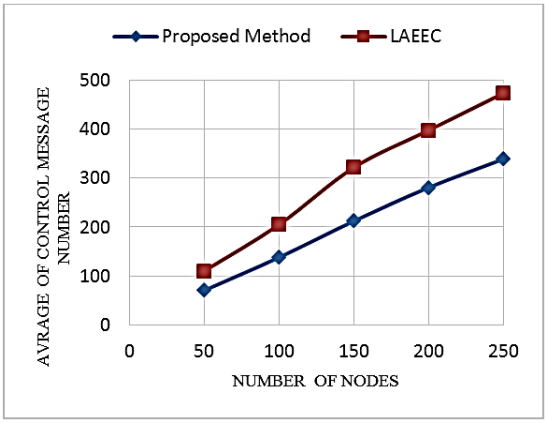

Fig. 8. The comparison of the average of control message number vs. the network size

- **Second scenario: sink in the [75 75] coordinate**

The results of this scenario we saw no any difference with previous scenario on average of control message number.

5. Conclusion and Future Work

The chief objective of the present study was to enhance network coverage by optimizing and improving clustering and reducing energy consumption of the nodes in WSNs. The protocols which were reviewed earlier in the paper were all aimed at improving significant parameters such as coverage and network life time. However, although many studies have been conducted in this domain, there are still many open problems and gaps. Hence, the present study was conducted to address these issues and improve the parameters. In this study, sensors with adaptable and dynamic reception range were used and an attempt was made to optimize clustering. Indeed, redundant nodes were selected to cover the probable gaps of the network, enhance network life time and better cover the network.

In this paper, the proposed method was compared with the LAEEC protocol. The results of the simulation indicated that the proposed method was better than the LAEEC protocol with respect to the following evaluative parameters: average number of active nodes, average remaining energy of the nodes, percentage of network coverage and the average number of control packets. In future, we will focus on providing new scheme based on soft computing. Also some new parameters must be considered to further improvement of this algorithm and extending the network lifetime.

References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," Computer networks, vol. 38, no. 4, pp. 393-422, 2002.

[2] W. Li and Y. Wu, "Tree-based coverage hole detection and healing method in wireless sensor networks," Computer Networks, vol. 103, pp. 33-43, 2016.

[3] B. Wang, H. B. Lim, and D. Ma, "A coverage-aware clustering protocol for wireless sensor networks," Computer Networks, vol. 56, no. 5, pp. 1599-1611, 2012.

[4] G. Wang, G. Cao, P. Berman, and T. F. La Porta, "Bidding protocols for deploying mobile sensors," IEEE Transactions on Mobile Computing, vol. 6, no. 5, pp. 563-576, 2007.

[5] B. Wang, H. B. Lim, and D. Ma, "A survey of movement strategies for improving network coverage in wireless sensor networks," Computer Communications, vol. 32, no. 13, pp. 1427-1436, 2009.

[6] A. Ghaffari, "Congestion control mechanisms in wireless sensor networks: A survey," Journal of network and computer applications, vol. 52, pp. 101-115, 2015.

[7] A. Ghaffari, "An energy efficient routing protocol for wireless sensor networks using A-star algorithm," Journal of applied research and technology, vol. 12, no. 4, pp. 815-822, 2014.

[8] A. Ghaffari and S. Nobahary, "FDMG: Fault detection method by using genetic algorithm in clustered wireless sensor networks," Journal of AI and Data Mining, vol. 3, no. 1, pp. 47-57, 2015.

[9] A. Ghaffari, "Designing a Wireless Sensor Network for ocean status notification system," Indian Journal of Science and Technology, vol. 7, no. 6, pp. 809-814, 2014.

[10] P. Cheng, X. Cao, J. Bai, and Y. Sun, "On optimizing sensing quality with guaranteed coverage in autonomous mobile sensor networks," Computer Communications, vol. 35, no. 9, pp. 1107-1114, 2012.Torkestani, J. A. (2013). An adaptive energy-efficient area coverage algorithm for wireless sensor networks. Ad hoc networks, 11(6), 1655-1666.

[11] G. Fan and S. Jin, "Coverage problem in wireless sensor network: A survey," JNW, vol. 5, no. 9, pp. 1033-1040, 2010 .

[12] A. Ghaffari, "Real-time routing algorithm for mobile ad hoc networks using reinforcement learning and heuristic algorithms," Wireless Networks, pp. 1-12, 2016.

[13] A. Ghosh, "Estimating coverage holes and enhancing coverage in mixed sensor networks," in Local Computer Networks, 2004. 29th Annual IEEE International Conference on, 2004, pp. 68-76: IEEE.

[14] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," IEEE Transactions on wireless communications, vol. 1, no. 4, pp. 660-670, 2002.

[15] H. Karl and A. Willig, Protocols and architectures for wireless sensor networks. John Wiley & Sons, 2005.

[16] J. Li, L. L. Andrew, C. H. Foh, M. Zukerman, and H.-H. Chen, "Connectivity, coverage and placement in wireless sensor networks," Sensors, vol. 9, no. 10, pp. 7664-7693, 2009.

[17] M. Liu, J.-N. Cao, G.-H. Chen, L.-J. Chen, X.-M. Wang, and H.-G. Gong, "EADEEG: An energy-aware data gathering protocol for wireless sensor networks," Ruan Jian Xue Bao/Journal of Software, 2007.

[18] R. Mulligan and H. M. Ammari, "Coverage in wireless sensor networks: A survey," Network Protocols and Algorithms, vol. 2, no. 2, pp. 27-53, 2010.

[19] O. Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," IEEE Transactions on mobile computing, vol. 3, no. 4, pp. 366-379, 2004.

[20] Z. Mottaghinia and A. Ghaffari, "A Unicast Tree-Based Data Gathering Protocol for Delay Tolerant Mobile Sensor Networks," Information Systems & Telecommunication, vol. 59, no. 25, pp. 1-12, 2016.

[21] G. Xing, X. Wang, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration for energy conservation in sensor networks," ACM Transactions on Sensor Networks (TOSN), vol. 1, no. 1, pp. 36-72, 2005.

[22] Z. Xinlian, W. Min, and X. Jianbo, "BPEC: an energy-aware distributed clustering algorithm in WSNs," J. Comput. Res. Dev, vol. 46, no. 5, pp. 723-730, 2009.

[23] R. Masoudi and A. Ghaffari, "Software defined networks: A survey," Journal of Network and Computer Applications, vol. 67, pp. 1-25, 2016.

[24] Z. Liu, Q. Zheng, L. Xue, and X. Guan, "A distributed energy-efficient clustering algorithm with improved

coverage in wireless sensor networks," Future Generation Computer Systems, vol. 28, no. 5, pp. 780-790, 2012.
[25] J. Yu, Y. Chen, L. Ma, B. Huang, and X. Cheng, "On connected target k-coverage in heterogeneous wireless sensor networks," Sensors, vol. 16, no. 1, p. 104, 2016.
[26] M. Akhlaq, T. R. Sheltami, and E. M. Shakshuki, "C3: an energy-efficient protocol for coverage, connectivity and communication in WSNs," Personal and Ubiquitous Computing, vol. 18, no. 5, pp. 1117-1133, 2014.
[27] N.-T. Le and Y. M. Jang, "Energy-efficient coverage guarantees scheduling and routing strategy for wireless sensor networks," International Journal of Distributed Sensor Networks, 2015.

Amir Pakmehr received the M.Sc degree in Computer Engineering from Tabriz Azad University, Iran in 2014, respectively. He is currently a Ph.D student of Computer Engineering in Qazvin Islamic Azad University (QIAU), he researches about Wireless Sensor Networks, His research interests include Wireless Networks, and Cover Energy Saving in Wireless Sensor Networks.

Ali Ghaffari received his B.Sc M.Sc and Ph.D degrees in computer engineering from the University of Tehran and IAUT (Islamic Azad University), Tehran, Iran in 1994, 2002 and 2011 respectively. As an assistant Professor of computer engineering at Islamic Azad University, Tabriz branch, IRAN, his research interests are mainly in the field of wired and wireless networks, Wireless Sensor Networks (WSNs), Mobile Ad Hoc Networks (MANETs), Vehicular Ad Hoc Networks (VANETs), Software Defined Networks (SDNs), networks security and Quality of Service (QoS). He has published more than 60 international conference and reviewed journal papers.