Zahra REZAEI, Javad Akbari TORKESTANI

Department of Computer Engineering, Arak Branch, Islamic Azad University, Arak, Iran

An Energy-Efficient MCDS-based Routing Algorithm for Wireless Sensor Networks: Learning Automata Approach

Abstract. *Wireless sensor networks which are used to monitor and control a specific area, are made of many small sensor nodes; they are compressed and spread in an environment. One of the most challenges in these typesof networks is Energy limitation which has direct influence on sensor network lifetime. Unicast routing in wireless sensor networks is a way for data delivery to a receiver. In this paper we are intended to present a unicast routing algorithm in wireless sensor networks, and we make a virtual backbone out of Minimum Connected Dominating Set (MCDS). This virtual backbone is initiated according to Energy level, Neighboring, and distance from Sink node. So, to this end we use an algorithm based on learning automata named UMCDS-LA deal with the unicast routing problem. Finally, we simulate proposed method by ns2 simulator. Thus the results show high performance of the proposed algorithm.*

Streszczenie*. W artykule zaprezentowano algorytm routowania typu unicast w sieci bezprzewodowej (ang. Wireless Sensor Network). Szkielet sieci oparty został na strukturze typu MCDS (ang. Minimum Connected Dominating Set). Do realizacji wykorzystano algorytm oparty na automacie uczącym się UMCDS-LA. Metodę przebadano na symulatorze ns2. (****Energowydajny algorytm routowania w sieciach bezprzewodowych oparty na strukturze MCDS – automaty uczące się****)*

Keywords. Wireless Sensor Network; Minimum Connected Dominating Set; Backbone formation; distributed learning automata
Słowa kluczowe: Bezprzewodowa Sieć Sensorów, Minimum Connected Dominating Set MCDS, szkielet sieci, rozproszony automat uczący się

Introduction

Wireless sensor network is made of many sensor nodes. They are randomly distributed in the environment, and their role is to gather information from such environment. Then they process this information and at last send it to the base station. They have restrictions in energy, computational capacity and memory [1,2]. Considering energy limitation in sensor nodes and requiring decrease transmission of excessive information to prevent energy dissipation, existence of algorithms to be distributed and work with local information is very essential [3]. Assumed that we use a graph $G = (V, E)$ to represent a WSN. In this graph, V is the set of nodes in network and E is the set of edges that shows all links in the network. If two nodes are located together in one transmission range, there is an edge between them. It means they could be related to each other, if all the nodes have the same transmission ranges, graph G will be known as a Unit Disk Graph (UDG), otherwise G is a general graph. Unit Disk Graph or UDG is shown below in figure1.

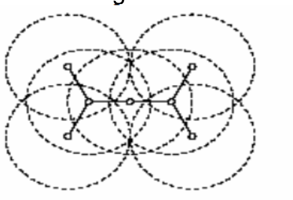

Fig.1.Unit Disk Graphs (UDG)

Independent set (IS) is a subset of network nodes or graphs with no two neighboring members [4,5].Maximal Independent Set (MIS) is an independent set with maximum of possible members in a graph. In the other word it is an independent set that adding every network vertex to it leads to the loss of its independence property. So every nodes which is not a member of MIS is at least adjacent to one of MIS's vertexes that is shown in figure 2 [6].

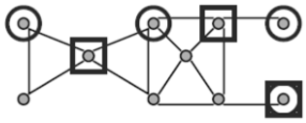

Fig. 2.Maximal Independent Set (MIS)

Dominating set (DS) is a subset of network nodes, like V; in a way that every node belong to V-DS is adjaceant to at least one of the nodes in DS. So, MIS is a DS itself but not vice versa. DS is often used to relate nodes to the cluster heads in clustering of wireless networks. Finding a dominating set with the least size or MDS is a NP-Hard problem. Connected Dominating Set (CDS) is a dominating set which its inductive sub-graph is connected. Minimum Connected Dominating Set (MCDS) is a CDS with the least possible nodes in the network. Determining MCDS is also a NP-Hard problem and is a matter of concern in flooding massage in wireless networks [7,8]. In such networks there is no prescribed and fix infrastructure since there is no physical backbone, it is possible to consider a virtual backbone. Regarding definition, CDS is a subset of network nodes in a way that every nodes either belongs to CDS subset or at least adjacent to a node in CDS. Since this structure is connected, it can be used to initiate a virtual backbone to find rout and flooding packets. The proposed method aims at minimizing the number of nodes which are responsible for relaying the unicast packets. In MCDS-based unicast routing protocols, a subset of the nodes is chosen as dominators to construct a route from the unicast source to a receiver; that is the minimum connected dominating set forms a virtual backbone that some members are dominated, while the number of them are responsible for broadcasting reduced to the number of nods in the backbone. Researches on learning automata and characteristics of sensor networks show that learning automata is a very convenient to use in sensor networks since it has some features as low computational overhead, ability to use in distributed environment with inaccurate information, and adaptation to the changes in environment [9,10]. In this paper, a DLA-based backbone formation is proposed algorithm to form a virtual backbone for the wireless sensor networks by finding a near optimal solution to the MCDS problem. To implement this approach, a network of the learning automata, isomorphic to the unit disk graph of the wireless sensor network, is initially formed by equipping each node to a learning automaton. Then, at each stage, the learning automata choose one of their actions in such a way that a solution to the MCDS problem could have been found. The action probability vectors of the learning automata are then updated depending on the response received from the environment [11,12]. The rest of the paper is organized as follows. In section2 reviews the related work. Section 3 describes the learning automata and distributed learning automata. In Section 4, the proposed algorithm is presented. In Section 5, the performance of the proposed algorithm is evaluated through the simulation experiments and comparison with the CDS-based algorithms and unicast routing algorithm and in Sections 6 is the conclusion.

Related Work

Until now several rout finding algorithm has been developed which are classified as following: data oriented protocols, hierarchal based protocols, location based protocols and based on QoS (Quality Of Service) [13]. Each of routing protocols purpose some criteria, such as short distance and minimum energy consumption in order to increase the lifetime of networks [14, 15]. Flooding and gossiping [16] are two classical mechanisms to relay data in sensor networks without the need for any routing algorithms and topology maintenance. In flooding, each sensor receives a data packet broadcasts to all of its neighbors and this process continues until the packet arrives at the destination or the maximum number of hops for the packet is reached. On the other hand, gossiping is a slightly enhanced version of flooding where the receiving node sends the packet to a randomly selected neighbor, which picks another random neighbor to forward the packet to and so on. Although, flooding is feasible to implement, it has several draw backs , such as implosion; caused by duplicated messages sent to same node, overlap; Gossiping avoids the problem of implosion by just selecting a random node to send the packet rather than broadcasting. However, this cause delays in propagation of data through the nodes. There has been many research works done on constructing a CDS-based backbone in WSN. Based on the information required to build a CDS, we could divide the existing algorithms into two broad categories: Centralized, and Distributed. Furthermore, if we consider the certainty of a CDS construction, algorithms fall into either: Probabilistic, or Deterministic. As centralized CDS construction algorithms require global information on the network, these approaches are not well suited for large networks due to a massive number of nodes and dynamic networks due to a variable number of nodes. However, distributed algorithms only need n-hop neighborhood information (for small n). These algorithms have low construction cost and show fast convergence. Li et al, [17] designed a distributed localized algorithm (r-CDS) to construct a 1-CDS. At first this algorithm chooses a set of dominators for the backbone, and then, as the CDS construction progresses, some connector nodes get selected to connect the initial dominators. Wu et al, [18] proposed both centralized and distributed algorithms to construct k-connected m-dominating set (km-CDS) backbones for WSNs. Their distributed km-CDS construction is based on Li et al.'s r-CDS algorithm. Earlier works to build CDS concentrated on special cases like k=1, 2 or k=m but Wu et al, designed algorithms for general k and m. Sajid and Mubashsharul [19] construct a graph model for a dense Wireless Sensor Network (WSN) and investigate energy efficient routing path for network communication. They used Connected Dominating Set (CDS), because it provides smaller network backbone compared to Minimum Spanning Tree (MST) and designed a distributed CDS construction algorithm.1-CDS construction is divided into two phases: 1) neighborhood discovery and2) shortest cost path tree construction with coloring, this is the main phase where each node gets its color, therefore designed a distributed CDS construction algorithm to create a virtual backbone in order to provide energy efficient communication.

Learning Automata

Learning Automata are adaptive decision-making devices that operate on unknown random environments. A learning Automaton has a finite set of actions to choose from and at each stage, its choice (action) depends upon its action probability vector. For each action chosen by the automaton, the environment gives a reinforcement signal with fixed unknown probability distribution. The automaton then updates its action probability vector depending upon the reinforcement signal at that stage, and evolves to some final desired behavior. A class of learning automata is called variable structure learning automata and are represented by quadruple$\{\alpha, \beta, p, T\}$in which $\alpha \equiv \{\alpha_1, \alpha_2, \ldots, \alpha_r\}$represents the action set of the automata,$\beta \equiv \{\beta_1, \beta_2, \ldots, \beta_m\}$ represents the input set,$\{p_1, p_2, \ldots, p_r\}$represents the action probability set, and finally $p(n+1)=T[\alpha(n),\beta(n),p(n)]$represents the learning algorithm. Let$\alpha_i$be the action chosen at time n, then the recurrence equation for updating p is defined as

(1)

$$P_i(n+1) = P_i(n) + a[1 - P_i(n)]$$
$$P_j(n+1) = (1-a)P_j(n) \forall j;\ j \neq i$$

for favorable responses, and

(2)

$$P_i(n+1) = (1-b)P_i(n)$$
$$P_j(n+1) = \frac{b}{r-1} + (1-b)P_j(n) \forall j;\ j \neq i$$

for unfavorable ones. In these equations , a and b are reward and penalty parameters respectively. If $a=b$, learning algorithm is called($LR-P$), if $a<<b$, it is called($LR-\varepsilon P$),and if $b=0$, it is called($LR-I$).For more information about learning automata the reader may refer to [9].

Distributed learning automata

A learning automaton is by design a simple unit by which simple things can be done. The full potential of the learning automata will be realized when a cooperative effort is made by a set of interconnected learning automata to achieve the group synergy. A Distributed learning automata (DLA) [12] is a network of interconnected learning automata which collectively cooperate to solve a particular problem. Formally, a DLA can be defined by a quadruple$<A, E, T, A_0>$, where $A = \{A_1, \ldots, A_n\}$is the set of learning automata, $E \subset A \times A$is the set of the edges in which edge $e(i,j)$ corresponds to the action αj of the automaton A_i , $T$is the set of learning schemes with which the learning automata update their action probability vectors, and A_0 is the root automaton of DLA from which the automaton activation is started. The operation of a DLA can be described as follows: initially, the root automaton randomly chooses one of its outgoing edges (actions) according to its action probabilities and activates the learning automaton at the other end of the selected edge. The activated automaton also randomly selects an action which results in activation of another automaton.

Proposed algorithm

In this section is created of a virtual backbone in wireless sensor networks in order to prevent direct data transmission to the destination node. As you know, our purpose to present a proposed algorithm is to decrease energy consumption in nodes and successively prolonging the lifetime of network.Since energy consumption in a sensor network directly relates to the square of the distance, so one-hop connections with long distance, consume more energy than multi hopes connections. Initiating a MCDS, only the members of dominating group are responsible for transmitting data to the destination node. In such way, direct transmission of data is prevented and we could take advantage of learning automata technique to choose appropriate nodes. A detailed description of algorithm follows.

Table 1. Pseudo code of The Proposed Algorithm

Proposed Algorithm: finding MCDS withmaximumof life time by LA

Input : A Stochastic Graph $G = (V, E, W)$
Output : Minimum Connected Dominated Set (MCDS) as a Backbone for unicast routing
Assumptions:
Let Door_list_determined is assumed for saving determined Dominators set and initially is {0};
Let Door_list_labeled is assumed for saving undetermined Dominators set and initially is {0};
LetDee_listisassumedforsavingdominateessetandinitially is{0} ;
Let V_i be selected node ith;
Let W_i be the remaining energy of V_i ;
Let D_i be the distance of sink of V_i ;
Let E(n) be the vector for saving the remaining_energy of selected node for backboneand initially is{0} ;
Let Action Vector be the vector for saving 1-hop neighbors of V_i;
Let P_{MCDS} is the threshold for ending Learning Automata iteration ;
Let C be counter of Dominators members and initially is {0} ;
Begin Algorithm
Repeat
Step1- <construct virtual backbone phase>
Let SN be Sink Node;
Let V_S be the source node of unicast routing ;
Start from V_S to create a backbone for unicast routing ;
Door_list_labeled = Door_list_determined= V_S ;
Dee_list = Neighbors (V_S) ;
E[1]= W_S ;
C=1;
while (Sink node did not Dominator yet) **do**// begin of while
Select oneoftheneighborsfromDee_list with maximum neighbor of degree; // as V_i
Door_list_labeled = Door_list_labeled$\cup$ V_i ;
Dee_list = Dee_list$\cup$ Neighbors (V_i) ;
C= C+1 ;
E[c] = Wi ;
end while
//find one hop neighbors with max neighbor of degree and placedinDoor_list_labeled
Step2-<Learning phase>
while (Sink node did not Dominator yet) **do** // to determine Door_list_ determined
Select one if the elements Door_list_labeled;
If (E_i>E_{avg}) and (D_i<D_{avg}) **then**
reward to selected action and update Action Vector;
else panelize selected action update Action Vector;
If (number of neighbors of V_i>average number of its neighbor nods neighbors) **then**
reward to selected action and update Action Vector;
else panelize selected action update Action Vector;
Door_list_ determined = Door_list_labeled$\cup$ V_i ;//based on max probability Action Vector
Dee_list = Dee_list$\cup$ Neighbors (V_i) ;
C= C+1 ;
E[c] = Wi ;
end while
if (V_i is Dee and hasn't neighbor of Door) **then**
V_i will be one of elements Door_list_ determined;
until (Probability of selection of new MCDS > P_{MCDS}) ;
send the unicast message through the created route to sink node;
End Algorithm.

Assumption of proposed algorithm

- We consider our network as a Unit Disk Graph (UDG). It means that transmission ranges of all nodes in the graph are same.
- Also, it is assumed that nodes are randomly distributed in network graph. Graph is defined as a classified triple $G = <V, E, W>$. V is set of vertex; E is the set of graph edges and W is energy attributed to the set of weights of graph's vertex.
- In this method we assumed that all the nodes in sensor network are similar and each node is able to stand two states. Nodes in MCDS are dominator nodes and dominate nodes. Dominator nodes are responsible to receive information from other nodes in the network and send it to the next dominator, so come to the sink node. The amount of energy consumption in dominator state is higher in comparison with dominate state. Nodes in dominate state gather data which are produced in their sensor range and deliver them to the dominator node.

In this proposed algorithm, main criteria are:
1) The amount of energy consumption:
Energy consumption in wireless sensor networks is one of the main factors. It has great influence on efficiency, reliability and network lifetime. Energy consumption in network is investigated from two aspects: a) The amount of energy consumption in each node (nodes with higher energy are more likely to become dominators.), b) Total energy consumption in network: it results in prolonging lifetime of the network. It applies when dominators are minimized and also they have the least distance to the sink node since finding unicast routing in sensor network.
2) Connected network:
It is also a critical matter in sensor network. To have a connected senor network two issues must be kept in our mind: First; Dominating set (DS) is a subset of network nodes, like V, and each node of V –DS is at least adjacent to one of the nodes in DS. Second; Connected Dominating Set (CDS) is a dominating set that its inductive sub-graph is connected. Proposed algorithm consists of two phases: Construction phase of primitive virtual backbone and learning phase is shown in Table 1.In this method sensor nodes choose one of dominator or dominate (ordinary) states using learning automata technique. Corresponding automata can choose between Door and Dee according to its probability vectors. Choose of Door is equivalent to become a dominator node and Dee is equivalent to become an ordinary node. In this way a virtual backbone forms and each ordinary node recognize its dominator based on Distance parameter and then learning phase begins. In learning phase, ordinary nodes deliver aggregated data to the dominator. Since our proposed method applies in Event driven networks, ordinary nodes immediately after witnessing an event, deliver their data to the sink node through the same virtual backbone. Moreover, in learning phase whenever the amounts of energy in Dominator node become less than a certain threshold, it chooses another node with more energy from its connected set as dominator based on probability vector. Description of various phases in algorithm follows. As mentioned earlier, construction of primitive virtual backbone has to be formed, and dominator nodes must be defined. Assigning dominator nodes take place in this phase. Illustrations of these two phases are detailed below.

Construction of initial virtual backbone phase

One of the most important parameters in construction of a virtual back bone is minimizing the number of dominator nodes and also our purpose is to minimize them as far as it is possible. If nodes with a large number of neighbors choose as dominators, totally the numbers of dominators will decreases. In this phase, we should be very careful to choose a dominator that keeps connectivity of the nodes. So, we achieve our first goal which is constructing a Minimal backbone. In construction of virtual backbone phase, each node informs number of its neighbors to adjacent nodes in a packet. Then each node which has the most neighbors in comparison with its adjacent nodes choose itself as dominator and informs this to the other neighbors as well. In this phase, nodes chosen as dominator are not fixed. It means now they only have the criterion of having the most neighbors, but maybe they are not in an appropriate situation from average energy and distance point of view. In other words, it is possible for a node to have the most neighbors but less energy and after data delivery from ordinary nodes to this node, depletion of energy and death waits for them. On the other hand, it is also possible for a node to have the most neighbors but more distance to sink node in comparison with its neighbors, so it consumes more energy. Therefore in learning phase, these criteria have to be checked before delivering data from Door to Dee and the best nodes are chosen as Door so they are fixed in the next phase.

Learning phase

Since dominating nodes were chosen in the previous phase, now we are trying choosing the best nodes as dominator until repetition of this phase result in the construction of a converge optimal virtual backbone. We determine the state of nodes with use of Distributed Learning Automata (DLA) technique. To make profit of this technique, we consider a learning automata equivalent to each sensor that each node can stand one of Door or Dee state. In learning phase each node recognize its neighbors and gathers some information about their energy and distance from sink node, then calculate remaining energy average of neighboring nodes (E_{avg}) and Distance average of neighboring nodes from sink node (D_{avg}) to make use of it later in learning phase. Each choice has influence on the Probability of becoming a Door for a node and it decreases or increases based on different parameters. These parameters are considered in construction of virtual backbone as the number of node's neighbors: Since the amount of energy consumption in dominators is high, we must minimize their number as far as it is possible. Therefore, nodes with the most number of neighbors must be chosen as dominators; so when a node chooses as dominator, if its neighbors are more than the average number of neighbors in its neighbor is rewarded. Moreover, in this phase, it takes into consideration that a set is minimal or not. Besides, number of neighbor dominators: One of the very important criteria in construction of virtual backbone is connectivity. It means that each node can send its information to the sink node. So each node either must be a dominator or one of its neighbors. An ordinary node with no dominator neighbors will penalize. Thirdly the amount of node's energy: dominator node is responsible to gather information from ordinary nodes and send it to the sink node. So energy consumption in dominator node is higher; therefore we try to choose those nodes as dominators that have more energy. Thus we use node's energy difference criterion with its neighbor's energy average. Eventually distance from sink node: Regarding this issue that energy consumption has a direct relation with distance square; it is selected a node as dominator which has the less distance to the sink node in comparison with the average of its neighbors distance. In this phase whenever a node observes an event, a data is delivered to the sink node and it must be done through virtual backbone. But prior to sending, it must be specified that in the previous phase this node is chosen as a member of virtual backbone accurately or not and done with the use of distributed learning automata. Each node is equipped with learning automata that its actions are the same as the number of node's one hop neighbors. In this phase, the probability of node's neighbors to be chosen as dominator is more than ordinary nodes. But, by passing of time this probability vector changes and updates in a way that whenever a node observes an event it choose its dominator to send data. So compares remaining energy of dominator with E_{avg} and the distance of dominator node with D_{avg}. In this case we can consider four states to choose from for delivering data to the backbone: If $E_i >= E_{avg}$ and $D_i <= D_{avg}$; the chosen action is rewarded. It means that if the energy of node is selected as dominator is grader than energy average of neighbors and its distance is less than distance average of neighbors, the node is selected correctly and awarded. In this method, three key factors in the construction of a virtual backbone are considered; remaining energy, the least distance to the sink node, and the number of node's neighbors. So, in construction of virtual backbone, those nodes are chosen as dominated which have more neighbors, more energy and less distance from sink node. After passing a certain time, learning phase is repeated to let nodes update their information about energy and neighbor node's distance.

Simulation Results

In this section, compare proposed routing protocol with Gossiping and CDS algorithm (LCPT)[16,19]. All of these protocols are implemented in ns2.The simulation experiments conducted in this section are concerned with investigating the efficiency to distributed algorithms, proposed for solving the minimum CDS problem. In all experiments presented in this paper, the reinforcement scheme used for updating the action probability vectors is *L R−I*, and the learning rate is 0.1. To generate the random graphs, a number of vertices are uniformly distributed in a two-dimensional simulation area of size 1*00* m×1*00m* at random. Then each algorithm is tested only on the connected graphs and the reported results are averaged over 100 runs. So each algorithm is terminated when the probability of choosing the MCDS approaches 0.95; number of nodes are between 25 to 150, initial energy level of each node is 200 unit, radio transmit power is approximately double that of radio receive power, MAC protocol is 802.11, the data packet size is 512 bytes. the performance measure of interest in this study are:(a) impact of number of nodes to lifetime, (b) impact of neighbor degree to lifetime and (c) number of member virtual backbone as a dominator.

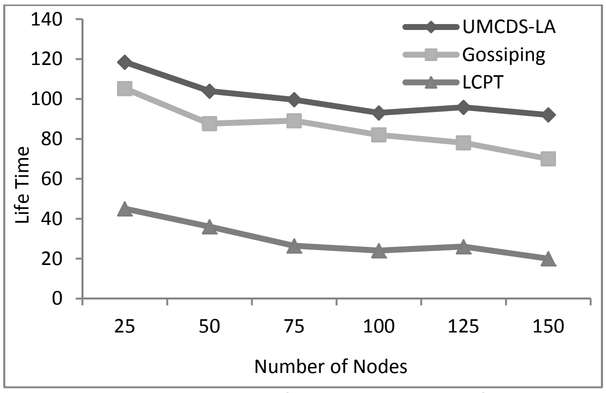

Fig. 4.Relationship between lifetime and number of nodes

Figure 4 shows the network lifetime for the proposed algorithm is higher than that for other algorithms. As we can be seen from figure 4, when the number of nodes increases in the network, the consumed energy will increase, and as a result the remaining energy will decrease. According to the lifetime of network definition, is the time when the energy of first node is elapsed. Hence in this diagram, in fact lifetime displays the remaining energy of nodes in the network. With the incrassation of the number of nodes, its lifetime would decrease. The proposed algorithm compared to other algorithms, has more remaining energy in the virtual back bone. Initial energy is 200 unit that means equivalent to 2 Joule.

The main goal of this experiment is to compare the algorithms when the neighboring degree of the network is changed while number of the nodes is fixed (75nodes) and in all of the simulation time. Given that these algorithms work based on information from neighbors, it is vital to measure their performance with different neighbor degrees ,as can be witnessed in figure 5through increasing neighboring degrees ,it decrease the life time, but UMCDS-LA algorithm has been less than decrees in comparison whit other algorithms, because neighbor of degree one of the important parameters in formation backbone. As we tried to form a network with a graph, so the adjacent nodes are those which have direct edge with each other. In our algorithm the initial factor to select a dominator is neighboring degree therefore it has a greater lifetime compared to other algorithm. Whatever number of neighbors is more, in reorganization of neighbor phase, Will generate more control packets. Therefore, will consume more energy and decrease lifetime. Then, to increase the number of nodes decrease lifetime.

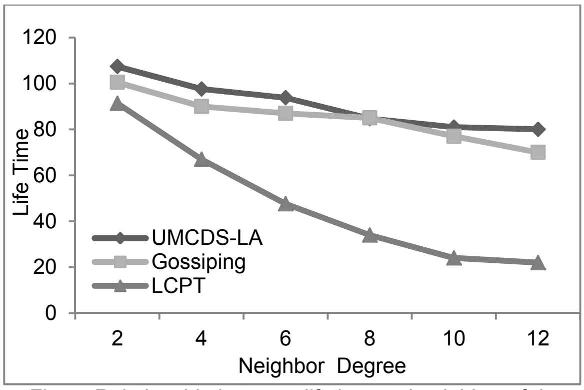

Fig. 5. Relationship between lifetime and neighbor of degree

Figure6shows the changes in the size of CDS with incrassation of deployed nodes in corresponding area.(by Neighbor Degree k=6) that is by increasing the number of nodes, the members of backbone increases as dominator, but in UMCDS-LA algorithm minimum member will be dominator. Because our goal is minimum connected dominating set, and we utilizes factors like greater degree neighbors, least distance and highest energy node in contracting a virtual backbone. Therefore compare to other method it has list nodes in contracting a virtual backbone.

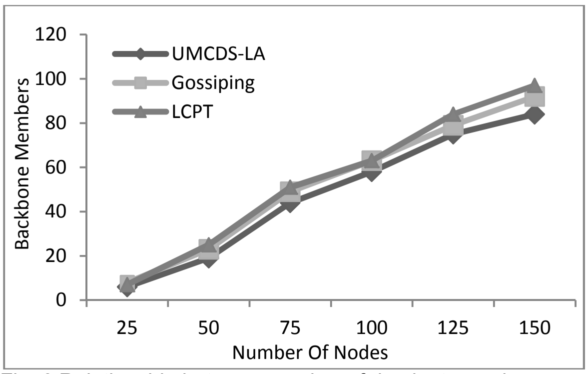

Fig. 6.Relationship between number of dominator and number of nodes

Conclusion

We designed a distributed CDS construction algorithm to create a virtual backbone in order to provide energy efficient communications. This algorithm designed for unicast routing by learning automata that increase network lifetime. We considered different scenarios such as the life time by 2 aspects: (1) increasing number of nodes, (2) increasing neighbor degrees, and at the end, we investigate our algorithm from the perspective of number of members' dominators. In all the cases, the proposed algorithm improves the result the simulation as compared to others algorithm.

REFERENCES

[1] Sterbenz J. P. G., Krishnan R., Hain R. R., Jackson A. W., Levin D., Ramanathan R. and Zao J., Survivable Mobile Wireless Networks: Issues, Challenges, and Research Directions, ACM Workshop on Wireless Security,(2002), 31–40
[2] AkyildizI. F., Su W., Sankarasubramaniam Y. and Cayircl E., A survey on sensor networks, in: Proceedings of the IEEE Communication Magazine., 40(2002)
[3] Ilyas M. and Mahgoub I., Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems, in: Proceedings of the CRC Press,(2005)
[4] Wu J.,Dai F.,Gao M., Stojmenovic I., On calculating power-aware connected dominating sets for efficient routing in ad-hoc wireless networks, Journal of Communications and Networks, 4(2002), No.1, 1–12
[5] Han B., Zone-based virtual backbone formation in wireless ad-hoc networks, Ad-Hoc Networks,7(2009)
[6] Alzoubi K. M., Wan P-J. and Frieder O., Maximal independent set, weakly connected dominating set, and induced spanners for mobile ad-hoc networks, International Journal of Foundations of Computer Science, 14(2003)No. 2, 287–303
[7] Clark B. N., Colbourn C. J. andJohnson D. S., Unit Disk Graphs. Discrete Mathematics, 86(1990), 165–177
[8] Marathe M. V., Breu H., Hunt H. B., Ravi S. S. and Rosenkrantz D. J., Simple Heuristics for Unit Disk Graphs Networks,25(1995),59–68
[9] NarendraK. S. and ThathacharM. A. L. ,Learning automata: An introduction, Printice-Hall,(1989)
[10] NarendraK.S.,Thathachar M. A. L., Learning automata a survey, IEEE transactions on Systems, 4(1974), No.4
[11] Akbari Torkestani J. and MeybodiM. R., Approximating the minimum connected dominating set in stochastic graphs based on learning automata, international conference on information management and engineering,(2009)
[12] Akbari Torkestani J. and Meybodi M. R., Clustering the wireless ad-hoc networks: A distributed learning automata approach, Journal of Parallel and Distributed Computing, Elsevier Publishing Company,(2010)
[13] Akyildiz I. F, Su W., Sankarasubramaniam Y., Cayirci E., Wireless sensor networks: a survey, Computer Networks, 38(2002), 393-422
[14] Chang J. H. and Tassiulas L., Routing for maximum system lifetime in wireless ad-hoc networks, in Proc. of 37th Annual Allerton Conf. on Communication, Control, and Computing,(1999)
[15] Chang J. H. and Tassiulas L., Maximum lifetime routing in wireless sensor networks, IEEE/ACM Trans. on Networking, 12(2004), No. 4,609-619
[16] Hedetniemi S. and Liestman A., A survey of gossiping and broadcasting in communication networks, Networks 18 (1988), 319–349
[17] Li Y., Zhu S., Thai M. and Du D-Z., Localized construction of connected dominating set in wireless networks, in Proceedings of NSF International Workshop on Theoretical Aspects of Wireless Ad Hoc, Sensor and Peer-to-Peer Networks ,(2004)
[18] Wu Y., Wang F., Thai M. and Li Y., Constructing k-connected m dominating sets in wireless sensor networks, in Proceedings of the IEEE Military Communications Conference ,(2007),1– 7
[19] Hussain S. and MubashsharulI. S., Energy Efficient Routing in Wireless Sensor Networks through Virtual Backbone, IEEE,(2009)

Authors
Zahra Rezaei,z.rezaei2010@gmail.com,
JavadAkbariTorkestani, j-akbari@iau-arak.ac.ir,
Department of Computer Engineering, ArakBranch, Islamic Azad University, Arak, Iran