

Full Length Research Paper

Routing in wireless sensor network based on soft computing technique

Samira Kalantari¹, Mohsen Amiri besheli², Zeinab Sam Daliri³, Shahaboddin Shamshirband^{3*} and Liang Shing Ng⁴

¹Young Researcher Club, Islamic Azad University, Sari Branch, Iran.

²Central Department, Andisheh Pars Institute of Research and Education, Iran.

³Department of Basic Science, Islamic Azad University, Chalous branch, Iran.

⁴Department of Artificial Intelligence, Faculty of Computer Science and Information Technology, University of Malaya, Malaysia.

Accepted 3 June, 2011

Wireless sensor networks consist of a large number of sensor nodes having limited power. The most important feature of these networks is the presence of dynamic topology which will lead to the mobility of the nodes. This mobility requires a routing capable of adapting to these changes. Despite the power restriction in these networks, the purpose in routing algorithm is not to find the shortest route; rather it is the power of each node which constitutes one of the most important issues. In this article, we have used soft computing techniques for routing in the network. The results obtained show that the combined method based on soft computing as compared to previous methods some what improves the minimalization of used power.

Key words: Fuzzy logic, multi agent system, Q learning, reinforcement learning, wireless sensor network.

INTRODUCTION

A wireless sensor network consists of a large number of sensor nodes scattered in an environment to collect information concerning the environment. Among the main features of such networks are that they do not have fixed structures and that they are used without any fixed stations or any wire connections to exchange information and to manage the networks. Moreover, the nodes present in such networks work in cooperation with each other (Mauri et al., 2005). To have this cooperation and coordination, there must be communication among which send information. Each sensor node has a specific sensory range and hence, in order to send the information package to the destination, it needs to locate neighboring nodes and to communicate with them so that package is guided to the destination (Jalali, 2009). Information packages are sent by nodes through the use of routing algorithms. Furthermore, in the routing process in sensor-type networks, the hardware imposes restrictions on the network. The suitable routing algorithm

must process the following characteristics: accurate operation, simplicity, stability, equity and optimality. The routing algorithm must select the link that overcomes the restrictions imposed on this type of network (Ghadimi, 2008). Since sensors have limited processing capability and power, the information obtained from the nodes is transmitted to a node which is strange in these characteristics. This node functions as the central node (sink) and processes complete knowledge of the network (Stefano, 2008). To transmit this information, the nodes must have a lot of power and many solutions have been offered to provide for this. For example, the LEACH protocol, which deals with the clustering problem in networks, has the ability to aggregate data in order to reduce the energy used by sensors, it can facilitate the process of collecting information from the sensor network and it is capable of forming the suitable structure for expandable routing. To do this, the nodes present in a cluster transmit their information to a node titled the cluster head, and this node aggregates the data and sends it to a sink (Rabiner et al., 2000). In this way, the number of transmissions is reduced.

The PEGASIS protocol like the LEACH protocol also

*Corresponding author. E-mail: Shamshirband@iauc.ac.ir.

Table 1. The situation table.

Node ID	Pos X	Pos Y	Power	Busy
1	2	4	8	No
3	3	5	20	Yes
5	6	4	50	No
9	7	1	41	Yes

Table 2. The neighbors table.

Neighbor ID	Neighbor status	Link status	power	Pos x	Pos y
1	Member	Bidirectional	1	3	2
3	Member	Unilateral	0	2	8
5	Cluster-Head	Bidirectional	4	1	6
9	Member	Bidirectional	1	4	4

reduces the number of transmissions through aggregation of the data but the difference is that in this protocol a chain of sensor nodes is formed in which each node can receive information from and send it to a neighbor is closer to it (Chow-Sing et al., 2008). The knowledge contained in each node is stored in two tables as follows (Pour and Kord, 2009):

The situation table

In this table the latest information contained in the agent about itself is stored including its physical position which is determined through the use of GPS, the power used, the state of being active or inactive, etc (Table 1).

The neighbor's table

In this table, information is stored about the neighbor's nodes such as the destination node, the position of the neighbor's nodes, the amount of power used by the neighbor's node, etc. when the node decides to send the package to a specific destination, it first searches this table to find the best neighbor for negotiations because the last time a package was sent this neighbor had the best conditions (Table 2). Knowledge of the nodes is updated through negotiations during the transmission of information packages (Jeremy et al., 2010). Cooperation among nodes is needed in order to determine the best node for routing in a WSN. This cooperation and coordination is possible through the use of multi-agent systems so that each node functions as an intelligent agent in the network and has a specific task. These agents have the task of transmitting information packages from a source to a destination. Multi-agent systems which are formed when these agents are connected to each other, improve the performance of the network (Jong-Myoung et al., 2008). Like other artificial

intelligence techniques (such as machine learning, neural networks, genetic algorithms, artificial immune systems, etc.) and together with them, fuzzy logic is used in manufacturing intelligent machines. Since fuzzy systems are employed in decision making, implementing them in the agents is very useful (Raghavendra et al., 2011). The CHEF algorithm is a kind of clustering algorithm used in choosing cluster-heads. This algorithm is implemented by using fuzzy logic compared to other similar algorithms, reduces overhead and increases the lifetime of the network. Fuzzy if-then rules employed in CHEF (Seyed et al., 2010). However, the thing which seems to be ultimately necessary is to select a node for routing which has the required capability and which is also closest to the destination. To solve the mentioned problems in network routing based on exploratory algorithms, we decided to design an agent which could use machine learning techniques and the popular algorithm Q-learning to perform network routing in the best possible way and in the shortest time and, at the same time, increase the life-time of the network. The new capability added to this agent is in rewarding agents through the use of a fuzzy technique (Sutton, 1998).

This article is organized as follows: subsequently, it deals with the definition of the multi agent system after which the Q-learning problem and fuzzy logic are defined and explained respectively. Furthermore, the study's proposed method is presented before the results obtained from the simulation are given which include solving the problem of finding the shortest route through the use of previous algorithms (Dijkstra). Finally, future research and the conclusions drawn from this study are presented.

REINFORCEMENT LEARNING AND Q-LEARNING

Reinforcement learning methods are a class of artificial intelligence problems in which an agent interacts with its

environment through trial and error. In this method, there are a set of states (S) and a set of actions (A) the agent can perform. There is also a set of reinforcement signals (R) which are given to the agent. When an input from the environment reaches the agent, the agent, depending the state it is in ($s \in S$) performs an action ($a_t \in A(s_t)$); the

$A(s_t)$ being all the actions allowed in the state s_t . The effect of this action cannot be supported. This action causes the agent to change to state s' ($s' = s_{t+1}$). Moreover, a reward which is a reinforcement signal given as an evaluation of the performed action is received by the agent from the environment and the agent tries to maximize this reward (Sutton, 1998). The main components of an agent in a learning system are divided into the following four groups:

Policy

This is the decision-making function which defines what action should be chosen for the current state; in other words, the mapping of state to action, which forms a lookup table is called policy that is, in our system the agents (the nodes) infer the action to be performed from this table. This table is a condition between state and action.

Reward function

This function determines which actions have been good and which have been loaded. It gives a number to each pair (state/action). The reward function in our proposed system is obtained through the use of fuzzy logic as follows: a matrix is formed which lines and columns are the nodes present in the network; and this matrix, depending on the connections among nodes decides on the values of the reward which are the elements of the matrix. This function is discussed in detail in our propose system.

Environment

The thing the agent interacts with is called the environment. The agents (nodes) in our proposed system based on the communication they can have with each other; form a group on the basis of which movements are carried out.

Value function

This function assigns a number to each state. In other words, the value function specifies the states which the agent, by starting from them expects to obtain a total

reward in the future which is equivalent to the numbers assigned to the states. The Q-learning algorithm is one of the reinforced learning techniques which does not require knowledge of the environment and which operates by estimating value for pairs (state/action). The QL algorithm is the simplest method compared to all other methods of reinforcement learning to understand and to implement because it does not presuppose anything regarding the dynamism of the environment. The QL algorithm assigns a value to each state-action. An action is performed by the evaluation function with the help of the memory of the agent to respond to the current situation. This action is chosen such that there will be a greater probability of receiving a reward. After the action has been performed, the reinforcement function prepares a reinforcement value, (+1, 0, -1); and this reward is increased if there is an increase in the knowledge and efficiency of the agent and will be reduced if there is a decrease in the efficiency and knowledge of the agent. This reward will cause the value of the state/action pair to be updated. The agent uses the exploration strategy which guides the behavior of the agent; and the actions must be carried out in the direction of receiving positive signals towards the inclination of achieving the purpose (Yoav, 2002). The Q matrix which infact shows the extent the agent learns from the environment is based on Formula 1 and is given a value at every episode of the algorithm (Sutton, 1998).

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \quad (1)$$

Where α is the rate of learning and $0 < \gamma < 1$ is the attenuation coefficient.

Finally, the purpose is for the agent to be able to receive rewards from the environment in order to maximize the total amount of rewards received and to become convergent to the degree of error present in the problem we are concerned with (Shoham Yoav, 2002). The degree of convergence in our proposed system is stated in convergence in Q-learning. The wireless sensor network acquires this learning due to the cooperation which exists among nodes. This is the basis of a multi-agent system and is explained below.

MULTI-AGENT SYSTEM

Multi-agent systems or MAS are a new strategy in designing, analyzing and implementing company systems. In multi-agent systems, there are several agents each of which takes parts in interactions and possesses mechanisms which are used to coordinate the behavior of independent agents (Shoham Yoav, 2002). These intelligent agents, as a subset of distributed artificial intelligence focus on systems which consist of several independent entities and which mutually affect each other in a domain. The way the agent operates is

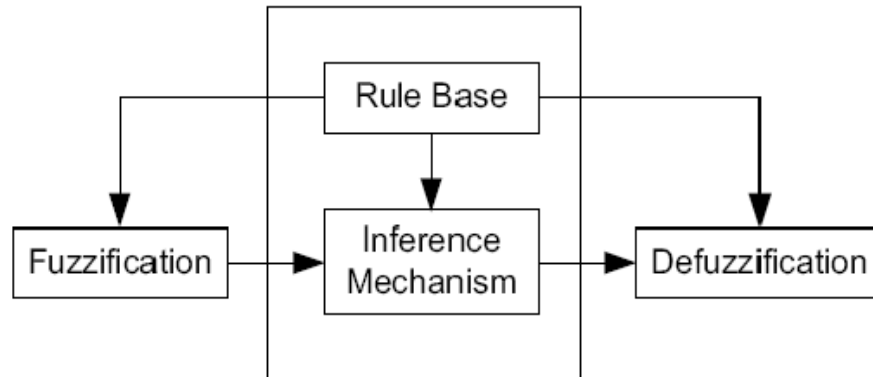


Figure 1. The structure of a fuzzy system.

considered to be its behavior. The main characteristics of agents include autonomy, carrying out mutual actions, responsiveness, reliability, mobility, intelligence, etc. some of these characteristics must be present in an agent, but some can be excluded depending on the behavior of the agent (Koblenz, 2005). Multi-agent systems have many advantages over single-agent and ordinary systems; for example, decisions in an MAS are made in a distributed manner and due to the cooperation and coordination among agents, parallel processing is easily executed in an MAS; and this reduces the volume of heavy processing and increases the speed of operation (Hellmanna, 2002). Multi-agent systems employed in teams of providing help for victims and rescuing them through the use of robots, increase the efficiency of these teams. In these systems, the agents (robots assisting the victims) trained by reinforcement learning carried out in their systems are dispersed in the area where the accident has happened with the task of supervising help operations and rescuing injured people. In these help and rescue operations, it has been proved that multi-agent systems are effective in help operations and compared to other systems rescue a greater number of injured people (Sutton, 1998). The agents in our proposed system are the nodes present in the network. The most important task of these nodes is deciding on neighboring agents to establish communication with it. This decision is made by using learning algorithms which every agent possesses.

FUZZY LOGIC

The concept of fuzzy logic was put forward by Dr. Lotfizadeh, an Iranian professor at the University of California in Berkeley as not only a control methodology but also as a way of processing data on the basis of authorizing membership in small groups rather than membership in cluster groups. This logic is the mathematical representation of the formation of human concepts and of reasoning concerning human concepts

(Lotfizadeh, 1965). Operations employed in using fuzzy logic are as follows (Shahabodin et al., 2010):

- 1) Determining the input and the output of the system.
- 2) Selecting the shape and boundaries of input membership functions (MF).
- 3) Converting input numerical variables into fuzzy variables.
- 4) Selecting the shape and boundaries of output membership functions (MF).
- 5) Determining suitable rules and applying them on the input.
- 6) Converting fuzzy answers to numerical values as the output.

A system based on fuzzy logic is given in Figure 1. Fuzzy logic is a simple rule on the basis of: If x and y , then z (Lotfizadeh, 1965). In our proposed method, the sender and the receiver (the possible positions for these nodes) are used as the input and the reward function is used as the output. Monitoring of gas pipes has been carried out by using fuzzy logic; and the natural gas consumption pattern has been improved by measuring gas pressure and consumption so that gas pressure will not drop in a specific area and will be balanced in all gas pipes (Javad, 2005).

PROPOSED METHOD

In this study, our purpose has been to perform routing in WSN network to increase the life-time of the network and to transmit information packages in the shortest possible time. Increase in network life-life is possible when the nodes have sufficient power to continue the process of routing in the network; and sending information in the shortest possible time can only be achieved by covering the shortest distance to the destination. In most cases, we cannot find the best route if we consider only one of these two parameters, because there is this probability that the node which is at the shortest distance from the destination may not have sufficient power to transmit the information package. Therefore the routing problem is discussed while simultaneously considering energy and distance. In our article, we train every node in the network which is

Table 3. The subintervals of the parameters power and distance.

Parameter	Linguistic variable	Interval
Power	Low, medium, high	(0,30), (30,70), (70,100)
Distance	Short, middle, long	(0, 10), (10,30), (30,50)

Table 4. The fuzzy linguistic variables in proposed system.

Parameters	type	The linguistic variable
Sender	Input	S1-S2-S3-S4-S5-S6-S7-S8-S9
Receiver	Input	S1-S2-S3-S4-S5-S6-S7-S8-S9
Reward	Output	V-bad, bad, v- poor, poor, mid, good, v-good, excellent, v-excellent

considered and agent by using the Q-learning algorithm so that each agent has the necessary knowledge about the moves it makes and is able to make decisions. In the Q-learning algorithm, the agent must receive a reward for every action it performs. The rewards in our study are determined by using fuzzy logic.

Use of fuzzy logic in determining reward

As was stated before, transmission of information from one node to another must be carried out under the best circumstances that is, the node chosen should have enough power and also be at an appropriate distance from the destination. We can use the following formula to calculate. The ratio of power to distance which determines the state of each of the nodes:

$$state_i = s_i = [pow]/[dis] \tag{2}$$

For example, given a sensor range of 50 m for transmission information packages and 100 w nodes, these states are specified:

Power: (0, 30), (30, 70), (70, 100)
 Distance: (0, 10), (10, 30), (30, 50)

In this instance, the power and the distance are divided into three intervals, hence we have nine states. If a node is in the power interval of 0, 30 and in the distance interval of 0, 10, it will have the following states (Table 3):

$$state_i = s_i = pow/dis =$$

$$s_1 = [0-30]/[0-10] = low/short$$

$$s_2 = [0-30]/[10-30] = low/middle$$

$$s_3 = [0-30]/[30-50] = low/long :$$

$$s_9 = [70-100]/[30-50] = high/long$$

We can see that s_7 has the highest power and the shortest distance to the destination, while s_3 has the lowest power and the longest

distance to the destination. That is a node in state s_7 is the best choice for routing while a node in state s_3 is the worst choice for that purpose. Therefore, a node wanting to go to state s_7 receives the highest reward while a node wanting to go to state s_3 receives the lowest reward. If we want to compare the priorities of the states for the purpose of determining the rewards, Formula 3 will be suitable:

$$S_{priority} = \frac{the\ mean\ of\ the\ pow}{the\ mean\ of\ the\ dis} \tag{3}$$

This formula is implemented for all the states and we find the following value:

$S_1 = 3, S_2 = 0.75, S_3 = 0.35, S_4 = 10, S_5 = 2.5, S_6 = 1.25, S_7 = 17, S_8 = 4.25, S_9 = 2.12.$

Therefore, we have the following priorities:

$$S_7 \rangle S_4 \rangle S_8 \rangle S_1 \rangle S_5 \rangle S_9 \rangle S_6 \rangle S_2 \rangle S_3$$

Thus, each node has a specific state in the network. In routing between two nodes, we go from one state to another and going to a more suitable state must enjoy a greater reward. These rules are implemented by using fuzzy logic. The states of the two senders and receiver nodes are taken as the input of the fuzzy system and the reward is considered as the output. We have considered nine MF_s with the names s_1 (state₁)... s_9 for the inputs. Table 4 contains some of decisions made in the fuzzy system. A number of these rules are also presented below (Table 5):

- 1) If (sender is S1) and (receiver is S1) then (reward is good)
- 2) If (sender is S1) and (receiver is S2) then (reward is bad)
- 3) If (sender is S1) and (receiver is S4) then (reward is excellent)
- 4) If (sender is S3) and (receiver is S7) then (reward is v-excellent)
- 5) If (sender is S7) and (receiver is S3) then (reward is v-bad)
- 6) If (sender is S7) and (receiver is S6) then (reward is v-poor)

For example, the fourth rule states that if the source node (sender) is in state s_3 (that is, it has the lowest power and is farthest away from the destination) and intends to send an information package to a node in state s_7 (that is, a node with the highest power and the shortest distance to the destination), it will receive the highest

Table 5. Rules determining the rewards in proposed system.

Sender state	Receiver state	Reward
S1	S3	0.1
S1	S7	0.9
S2	S6	0.3
S3	S4	0.8
S4	S5	0.5
S5	S1	0.6
S6	S2	0.2
S7	S3	0.1
:	:	:

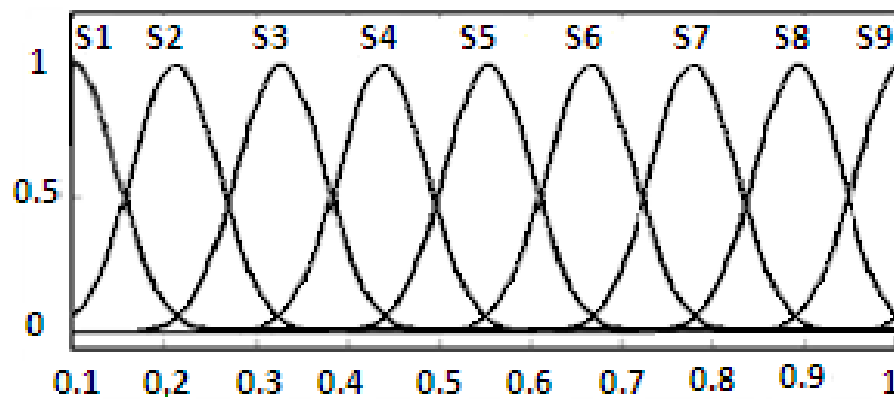


Figure 2. The input membership function of the sender and receiver.

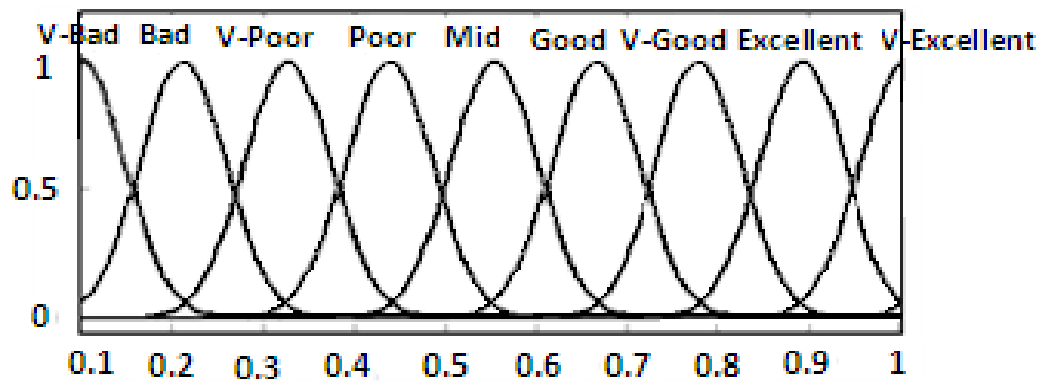


Figure 3. The output membership functions of the reward.

reward (v-excellent or 0.9). Figures 2 and 3 shows the input membership function of the sender and the receiver and the output membership function of the reward, respectively. Figure 4 is a surface graph showing rewards received in relation to the inputs. Figure 5 shows the reward which is related to the input. The matrix of the learning algorithm can be completed by using the diagram in Figure 5.

SIMULATION

Implementation

This project has been simulated in the software visual studio.net and a generating function has been used to

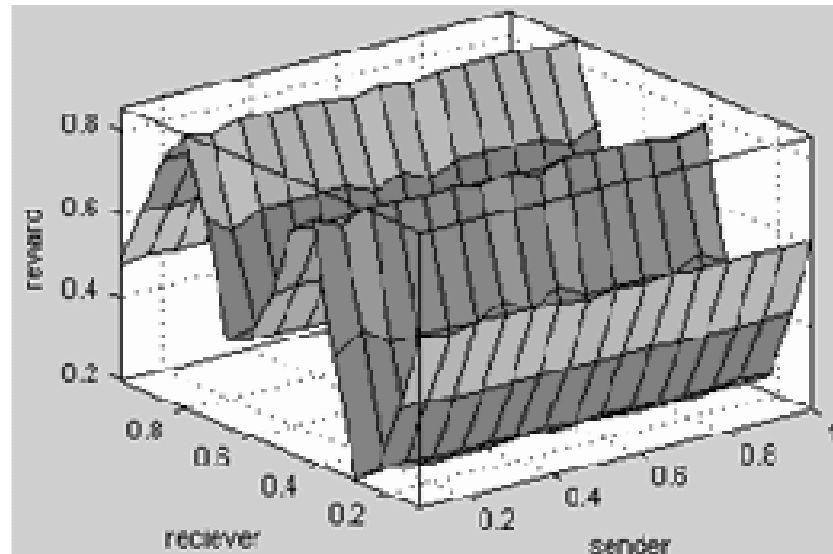


Figure 4. Rewards received in relation to the states of the nodes.

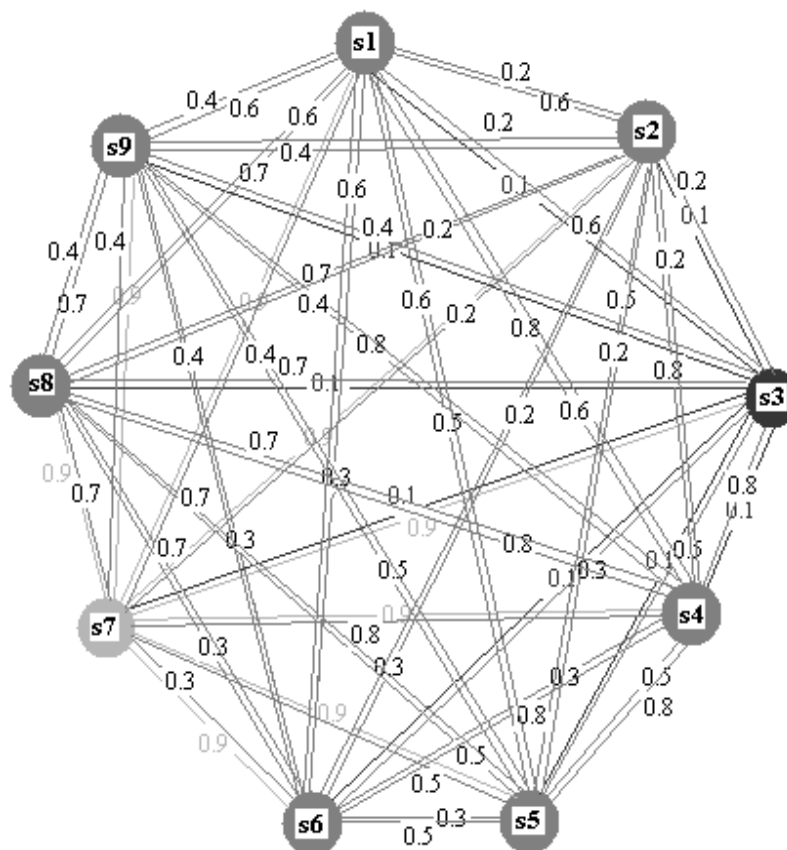


Figure 5. An example of a network graph model.

generate the nodes in the network. This function is a Poisson distribution function (Cormen et al., 2001). These nodes form a graph in the network. This graph is dynamic

and its structure changes at every instant according to the node generating function and the node mobility in the network (Figure 8). We have compared routing in our

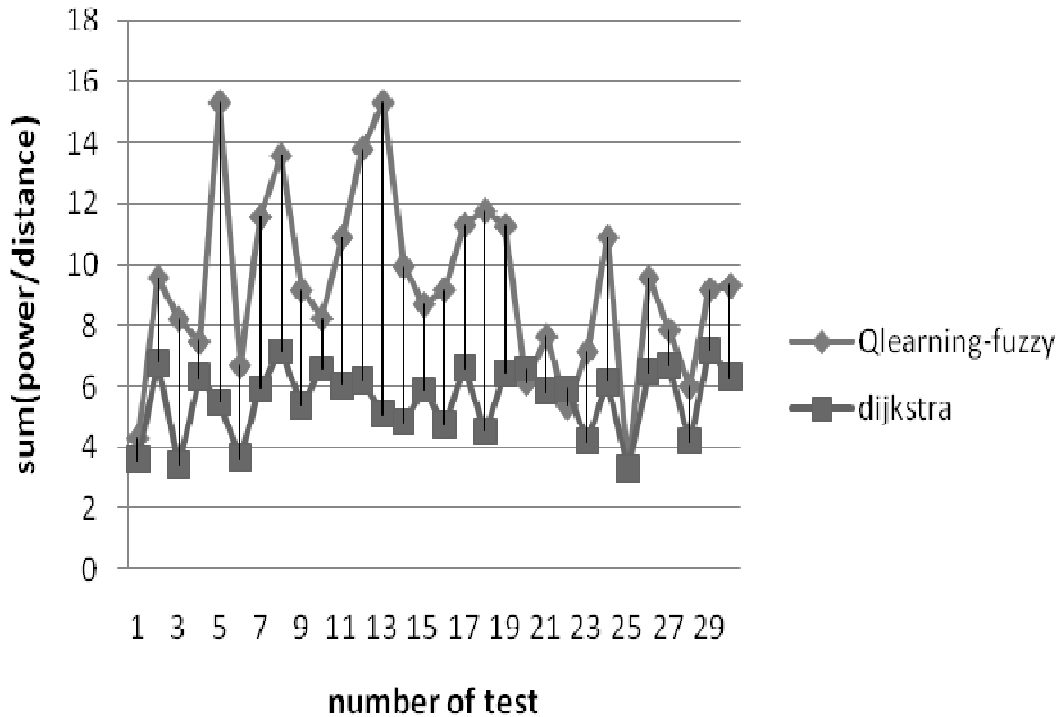


Figure 6. A comparison between the power level in relation to distance for Q learning-fuzzy and Dijkstra algorithm.

proposed method in which this graph is used with routing in which Dijkstra is employed. Dijkstra is an algorithm for finding the shortest path between two nodes. This algorithm, like the Prim algorithm is of the greedy type and is used for the minimum overlapping tree problem. Dijkstra is an algorithm of the order of $\theta(n)$. However, what we should address in setting up wireless sensory networks is the problem of the life-time of the network; this problem is ignored in many routing algorithms and only the shortest distance is considered. In our proposed Q learning-fuzzy algorithm, the shortest distance is chosen only if the life-time of the network is not reduced. In Figure 6 a comparison is made between the Q learning-fuzzy and the Dijkstra algorithms. These two algorithms were used to route 15 nodes out of a total of 50 nodes present in the network between two different points. The $\sum(pow/dis)$ ratio was also measured in each routing. We know that the greater the power and the less distance, the better the routing operation will be performed because the shortest distance is chosen while maintaining as much of the power of the network as possible. This means that when comparing methods, the one in which the fraction $\sum(pow/dis)$ is bigger will lead to a longer life-time of the network.

In our comparison, the $\sum(pow/dis)$ in our proposed method was bigger than that for Dijkstra. This comparison was carried out in 30 iterations. While the shortest possible path is not chosen in our proposed

method, the shortest path is chosen which does not cause a reduction in the life-time of the network.

Proving convergence in Q-learning

To prove convergence, we prove that the error in the input having the highest error in table Q is reduced by γ every time a visit is made. Since all states/actions will be repeated infinitely, if we consider distances which have been repeated at least once, then the error at the n th time values in the table change will be equal to:

$$\hat{Q}(S, a) \leftarrow r(S, a) + \gamma \max \hat{Q}(S', a)$$

The error at the $(n + 1)$ th time the values in the table change will be equal to:

$$\begin{aligned} & | \hat{Q}_{n+1}(S, a) - Q(S, a) | = | (r + \gamma \max \hat{Q}(S', a)) - (r + \gamma \max Q(S', a)) | \\ & = \gamma | \max \hat{Q}(S', a) - \max Q(S', a) | \\ & \leq \gamma \max | \hat{Q}(S', a) - Q(S', a) | \\ & \leq \gamma \Delta_n \end{aligned}$$

In Figure 7, the extent of convergence in our proposed method for routing between two nodes is shown. The extent of the learning of the agents (the nodes present in

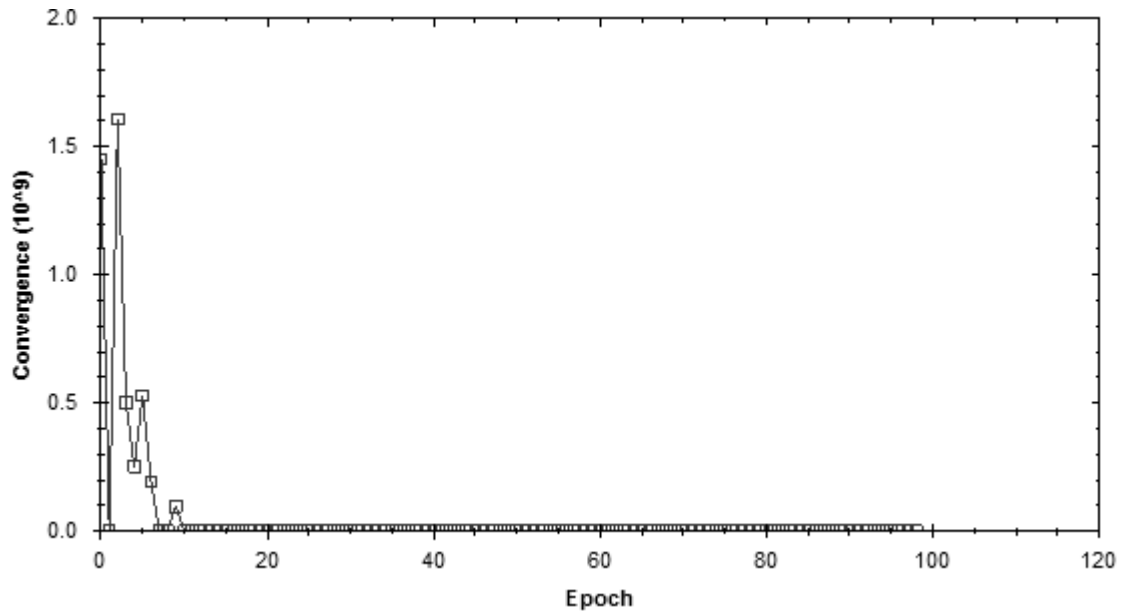


Figure 7. The extent of convergence in our proposed method.

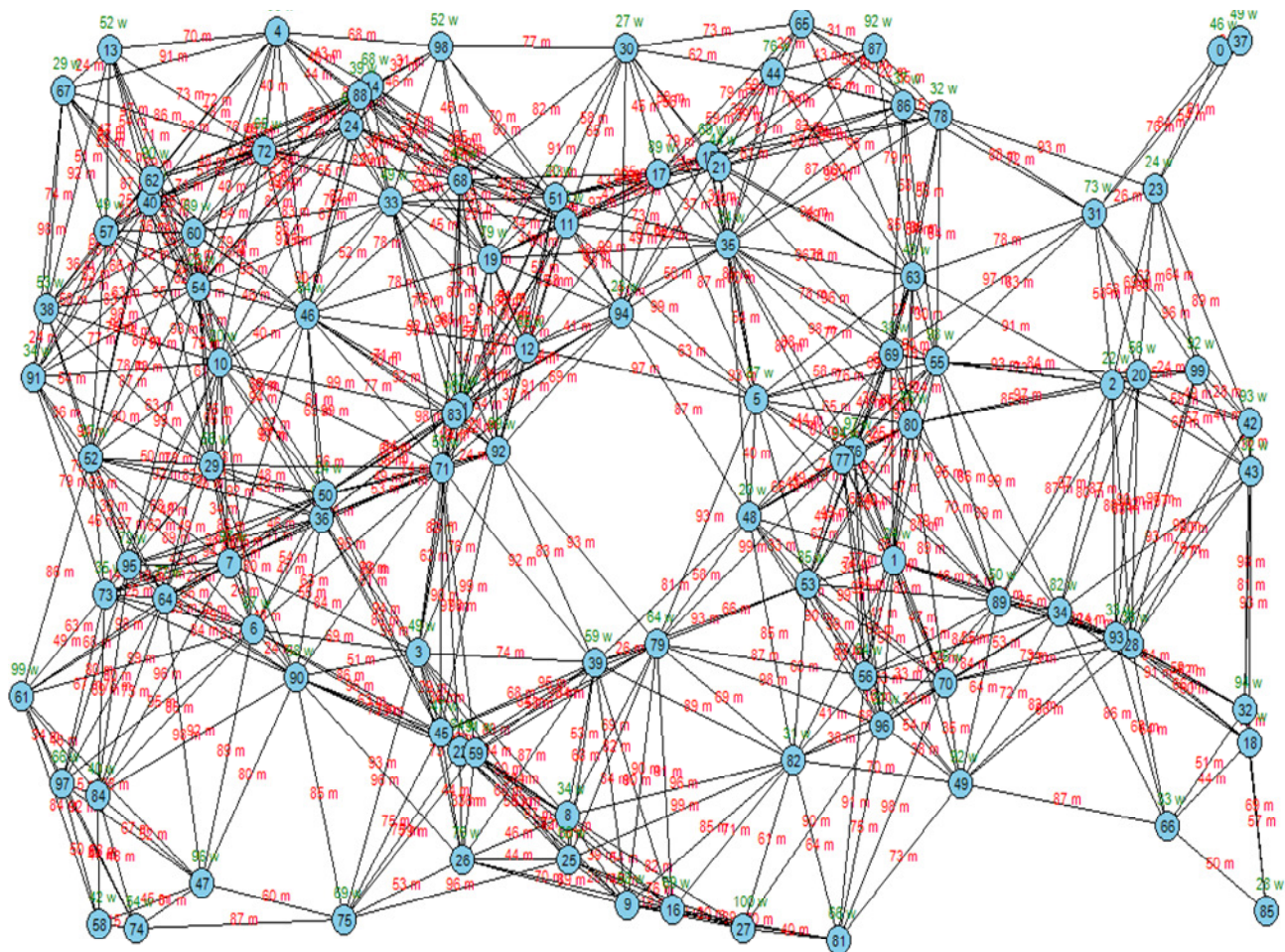


Figure 8. WSN in simulation environment.

the network) is shown after several iterations. The number of iterations chosen at this instance was 50.

CONCLUSION

This article was written on the subject of routing in wireless sensor networks. We used the parameters present in these networks such as power and distance determined the possible states for each node through the use of these parameters and by employing fuzzy logic and obtained a reward which Q learning could use to learn the agent we had in mind (which was to find the most suitable route as for distance and power are concerned).

ACKNOWLEDGEMENT

This paper was essentially supported by research grant from Islamic Azad University, Chalous branch.

REFERENCES

- Chow-Sing L, Chia-Nan H, Rong-Jyue F (2008). "A power-efficient data gathering scheme on grid sensor networks", Proceedings of the 8th WSEAS International Conference on Multimedia systems and signal processing. Pages, pp. 142-147.
- Cormen T, Leiserson C, Rivest R, Stein C (CLRS) (2001). "Introduction to Algorithms", Second Edition,
- Ghadimi A (2008). "Review how extraction in wireless sensor networks", Tehran University, Jan, pp. 125-130.
- Hellmanna M (2002). fuzzy logic introduction "laboratoire antennes radar telecom, F.R.E CNRS 2272, EQUIPE RADAR POLARIMETRIE
- jalali MN (2009). "Validate protocol network members moving", National Conference Software Engineering Applications, 3-5 march Javad behboodan, "introductory statistics and probability", third edition, 2005.
- Jeremy L, Damien P, Abdellatif M, Lagorse J (2010). "A multi-agent system for energy management of distributed power sources", Renewable Energy, 35: 174-182.
- Jong-Myoung K, Seon-Ho P, Young-Ju H, Tai-Myoung C (2008). "CHEF:Clustering Head Election mechanism using Fuzzy logic in wireless sensor network", Feb.17-20, 2008 iCACT
- Koblenz (2005). Multi-agent system technologies "september 11 - 13, germany university of koblenz-landau incorporating the 9th international workshop on cooperative information agents (CIA 2005).
- Mauri K, Marko H, Timo DH (2005). "A Survey of Application Distribution in Wireless Sensor Networks", EURASIP J. Wireless Commun. Networking, pp. 774-788.
- Yoav S (2002). Introduction to Multi-Agent Systems (Written with Trond Grenager)", April, Book.
- Pour K, kord A (2009). "Collapse of data a new method for wireless sensor networks ", 1'th National Conference on Software Engineering Applications, 3-5 march
- Rabiner HW, Chandrakasan A, Balakrishnan H (2000). "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," Proceedings of the 33rd International Conference on System Sciences (HICSS '00),
- Raghavendra A, kulkarni AF, Ganesh K (2011). "Computational intelligence in wireless sensor network: A Survey", to appear in IEEE commun. surv. Tutorials, 13(13)
- Seyed SS, Hossein S, Saeed S (2010). "Design of rescue multi agent system based on soft computing techniques", Adv. Elect. Comput. Eng., 10(1).
- Shahabodin S, samira K, zeinab K (2010). "Design a smart multi-agent system based on fuzzy logic to improve the gas consumption pattern", ISSN 1992-2248 ©2010 Acad. J. Sci. Res. Essays., 5(6), pp. 592-605, 18 March,
- Stefano B, Alessio C, Emanuel M (2008). "Controlled sink mobility for prolonging wireless sensor networks lifetime "journal wireless network, Springer Netherlands, 14: 6.
- Sutton RS (1998). Learning to predict by the methods of temporal differences, Mach. Learn. 3: 9-44.
- Sutton RS, Ed (2002). Machine learning: Special issue on reinforcement learning. Cambridge, MA: MIT Press, 8: 3-4.