International Journal of Computational Intelligence in Control
Vol. 13 no. 3 (December, 2021)

Received: 12th April 2021 *Revised: 28th June 2021* *Accepted: 29th July 2021*

Topology Management Using Fuzzy Logic for Mobile Ad-hoc Networks: A Semi-Distributed Approach

Jishan Mehedi
Department of Electronics and Communication Engineering, National Institute of Technology, Silchar, Assam
E-mail: jmehedi2007@yahoo.co.in

Mrinal Kanti Naskar
Advanced Digital and Embedded Systems Lab, Department of Electronics and Telecommunication Engineering
Jadavpur University, Kolkata, *E-mail: mrinalnaskar@yahoo.co.in*

Abstract: *In this paper, we propose an algorithm to maintain connected topology of Mobile Ad-hoc Networks (MANETs) by suitably selecting 'Leader' among the nodes constituting the MANET. Once selected, the 'Leader' is entrusted with the responsibility to broadcast its positional information periodically while the other nodes individually decide the logic they need to follow in order to maintain the topology of the network, thereby eliminating the need for routing. The algorithm ensures that the entire network moves in one direction while each node can move freely. Both the process of 'Leader' election and the movement of nodes are based on fuzzy logic. The results obtained through the simulation show the effectiveness of the algorithm.*

Keywords: *MANET, Topology Management, Semi-distributed Approach, Fuzzy Logic*

1. INTRODUCTION

Ad-hoc networking is an emerging domain in wireless communications for mobile hosts (nodes) with no fixed infrastructure such as base stations or mobile switching centers [1, 2]. Mobile nodes that are within each other's radio range communicate directly via wireless links, while those that are far apart rely on other nodes to relay messages as routers. MANET has received tremendous research interests in recent years. It is very useful for sharing information in areas lacking of communication infrastructures such as disaster prone areas, combat zones, or unexplored territories (e.g., deep space or deep sea). In these scenarios, mobile hosts are scattered around to observe various parts of a large unfamiliar territory and record their experiences in their local storage.

An experience is a location and observer dependent data that may be in the form of text, images, audio, or videos. Each experience is associated with a recording time, location, and information about the host that creates the experience. Node mobility in an ad hoc network causes frequent changes of the networks topology. Thus routing is needed to find the path between source and destination and to forward the packets appropriately. Hence, in case of ad-hoc networks the nodes not only behave as usual trans-receivers but also as routers taking part in route discovery and maintenance [3]. Mobile ad-hoc networks are extensively used to retain connectivity of nodes in inhospitable terrains. It is also projected to play significant roles in network maintenance in case of search-and-rescue operations, unplanned meetings, spontaneous interpersonal communication etc. where preconceived infrastructure is absent and sudden data acquisition is necessary [1,4].

Random node movement makes routing an essential requirement for MANET. Due to frequent node movement it may so happen that when the source node wants to transmit packets, the destination node may be out of range of the source node. Further, transmission losses occurring due to different natural phenomena may be another cause of frequent network disruption. Hence, the current focus of many researchers is to find out an efficient routing protocol which ensures node connectivity whenever required without much delay and unnecessary overhead. But none of the routing protocol gives the guaranteed connectivity during the movement of the network. Hence, the topology management schemes came into picture. The two widely accepted approaches for topology management are centralized and distributed [5]. However, in the centralized algorithms considered so far, the workload on the coordinator is immense. The distributed approaches on the other hand are very expensive. The centralized topology management schemes in [1, 5, and 6] discuss a self-adaptive movement control algorithm for topology management.

In this paper, we present a fuzzy based semi-distributed algorithm for maintaining connectivity of nodes assuming unidirectional movement of the topology but the nodes are free to move in any direction. The proposed fuzzy based semi-distributed algorithm ensures that all the nodes of the network will remain connected during movement. The rest of the paper is organized as follows: related works are discussed in the next section. Problem definition and system modeling are given in section 3. Section 4 presents the algorithm for 'Leader' election. Section 5 deals with the movement algorithm. Section 6 reports the effectiveness of our algorithm with simulation results. A comparative study with a previous algorithm is done in section 7. Finally, Section 8 concludes the paper.

2. RELATED WORKS

There are many existing routing protocols for MANET namely proactive, reactive and hybrid [3]. The frequently employed routing schemes include Destination Sequence Distance Vector (DSDV), Wireless Routing Protocol (WRP), Optimized Link State Routing (OLSR), Cluster Switch Gateway Routing (CSGR) under proactive schemes, Ad hoc On-Demand Distance Vector Routing (AODV), Dynamic Source Routing (DSR), Associativity Based Routing (ABR), Temporally Ordered Routing (TORA) etc. under reactive schemes and Landmark Ad hoc Routing Protocol (LANMAR), Zone Routing Protocol (ZRP), Preemptive Routing (PR) etc. under hybrid schemes. All these protocols and their comparative study have been discussed in [2, 5, 7, 8]. Besides flooding and dynamic cluster based routing are also prevalent [9].

Some initial work has been done to maintain a topology with a connected network in the MANET. R. Ramnathan and R. Rosales-Hain [10] proposed an algorithm to adjust the node transmission power to maintain network connectivity. C. Bettstetter [11] models the neighboring node distribution by nearest neighbor methods known from analysis of spatial data. Their work provides little evidence to show how randomly distributed nodes can be modeled using such a distribution. Most of the work deals with randomly distributed static nodes. None of the above schemes guarantees the connectivity.

S. S. Basu and A. Chaudhari [3] proposed a topology control scheme using the idea of sending HELLO message. They described an algorithm for movement controlling using START, STOP, RUSH control words. The algorithm selects their coordinator based only on positional information. So, there is a chance of failure of the coordinator. Moreover, in this algorithm sometimes the whole network becomes static. Soumya Sankar Basu, Atal Chaudhari [1] modifies their previous algorithm by incorporating Global Positioning System [12]. It increases the mobility of network. S. Samanta, S. S. Ray, S. Sen Gupta, M. K. Naskar [6] discussed the probabilistic approach for maintaining connectivity in a centralized manner. The idea of sharing location dependent information in a distributed manner is given by Kihwan Kim, Ying Cai and Wallapak Tavanapong [13]. Jinshan Liu, Francoise Sailhan [14] presented the group management technique for Mobile Ad Hoc Networks. A. Seetharam, A. Bhattacharyya and M. K. Naskar [15] have implemented their centralized algorithm to test the performance in real environment. However, in the centralized algorithms considered so far, the workload on the coordinator is immense. Abhishek Bhattacharyya, Anand Seetharam and M. K. Naskar [16] proposed one centralized algorithm for maintaining connected topology through multiple cooperating coordinators to reduce the work load of the coordinator for the so called centralized shemes.

Avik Ray, Kanad Basu, Samir Biswas, Mrinal K. Naskar [17] proposed a distributed connectivity maintenance algorithm. Another distributed scheme was proposed by Jishan Mehedi, Surendra S. Dalu, and M. K. Naskar in [4]. These schemes completely eliminate the concept of the coordinator and the work load of the network is distributed to all the nodes. An important problem of these schemes is that sometimes in emergencies there are rapid velocity changes in the nodes.

Currently, the field of fuzzy logic [18] is largely overlooked by the Ad-hoc and Wireless Sensor Network (WSN) community. However, fuzzy logic has several properties that qualify it as an effective tool for MANET. Firstly, it can be implemented on limited hardware and is computationally fast. Secondly, it handles unreliable and imprecise information, offering a robust solution to decision fusion under uncertainty. Thirdly, fuzzy-based methodology substantially reduces the design and development time in control systems. Finally, fuzzy controllers handle non-linear systems better when compared to conventional approaches. In [19] we have tried to incorporate fuzzy logic in our algorithm considering pursue mobility model. T. Camp, J. Boleng, and V. Davies [20] described different mobility models for MANET. These models are based on either the mobility of a single node or a group of nodes. In group mobility model the decision regarding movement of a particular node is dependent on the movement of other nodes in the group, thus it needs a topology management scheme. In this paper, we propose a semi distributed algorithm for the group mobility model.

3. PROBLEM DEFINITION AND SYSTEM MODELING

3.1. Problem

In MANET all the nodes are mobile, so the problem is to keep all the nodes within the range of communication during movement through the topology management.

3.2. System Modeling

First, we define a few parameters and next mention some of the important assumptions for modeling the system.

3.2.1. Parameter Definitions

Maximum Communication Range (R_{max}): The maximum range of distance over which two nodes can communicate among themselves is called the maximum communication range and it is denoted as 'R_{max}'

Safe Distance (R_{safe}): Each node will try to maintain a distance, lesser than a particular distance called safe distance. This distance is denoted as 'R_{safe}' where maximum communication range 'R_{max}' is greater than the safe distance, 'R_{safe}'.

Beacon Interval ('T'): It is defined as the timing interval after which the nodes transmit their information periodically. The beacon interval is denoted as 'T'. It is chosen as $(R_{max} - R_{safe}) / 2\ V_{max}$ where V_{max} is the maximum velocity of the node. The lemma and its proof for choosing the beacon interval are given in section 3.2.3.

3.2.2. Assumptions

1. Initially all the nodes can communicate with one another.
2. Each node is enabled with a GPS receiver.
3. Every node has a predefined maximum velocity, V_{max}
4. The acceleration and deceleration of the nodes are instantaneous.
5. The network moves in only one direction, (e.g. the x-direction) while each node can move in any direction.
6. Each node has the facility to read its energy level.
7. Each node has its own identification number.

3.2.3. Lemma

Lemma: If we choose the beacon interval as $T \le (R_{max} - R_{safe}) / 2\ V_{max}$, where, the maximum communication range is 'R_{max}' , the safe distance is 'R_{safe}' and the maximum preferred velocity is 'V_{max}', then there is no chance for the nodes to go out of the communication range.

Proof: Maximum preferred velocity of a node is V_{max}. So the maximum possible relative velocity between two nodes is $2\ .V_{max}$, when they are in opposite direction. So the maximum relative distance traveled in a beacon interval is $2T\ V_{max}$. Since initially maximum separation between two nodes may be 'R_{safe}', so a neighbor node cannot become a non-connected node

$$\text{If, } 2.T.V_{max} \le (R_{max} - R_{safe})$$

$$\text{Or, } T \le (R_{max} - R_{safe}) / 2\ V_{max} \qquad (1)$$

3.2.4. Network Model

In this paper, we propose a fuzzy based semi-distributed algorithm for maintaining a connected network throughout the time of movement of the nodes. In this approach, the 'Leader' will be elected first. The node which is closer to the centre of the network and the node which has higher energy level will be selected as 'Leader'. After 'Leader' election all the other nodes will follow the 'Leader' to be connected with the 'Leader'. The network model is shown in the figure 1.

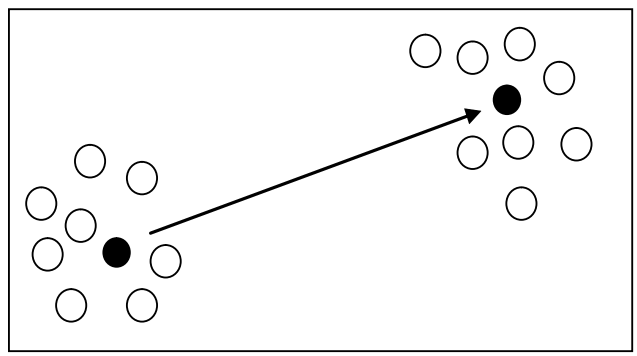

Figure 1: Showing 'Leader' Marked as Black and the Movement of the Topology

4. LEADER ELECTION ALGORITHM

As stated in section 4, all the nodes forward their initial positional data to every other node in the topology. Among the set of all x-coordinates $\{x_1, x_2\ldots, x_{n-1}, x_n\}$ and the set of all y-coordinates $\{y_1, y_2, \ldots, y_{n-1}, y_n\}$ received by a node, the minimum and maximum x-coordinate and the minimum and maximum y-coordinate are selected, denoted by x_{min}, x_{max}, y_{min}, y_{max} respectively i.e.

$$x_{min} = \min \{x_1, x_2\ldots, x_{n-1}, x_n\}$$

$$x_{max} = \max \{x_1, x_2\ldots, x_{n-1}, x_n\}$$

$$y_{min} = \min \{y_1, y_2, \ldots, y_{n-1}, y_n\}$$

$$y_{max} = \max \{y_1, y_2, \ldots, y_{n-1}, y_n\}$$

The centre of the network is chosen as (x_c, y_c), where $x_c = (x_{max} + x_{min}) / 2$ and $y_c = (y_{max} + y_{min}) / 2$. Now every node will calculate its distance from the centre.

4.1. Fuzzy Logic Control

The model of fuzzy logic control consists of a fuzzifier, fuzzy rules, fuzzy inference engine, and a defuzzifier. We have used the most commonly used fuzzy inference technique called Mamdani method due to its simplicity. The process is performed in four steps:

- Fuzzification of the input variables energy level and the distance from the centre - taking the crisp inputs from each of these and determining the degree to which these inputs belong to each of the appropriate fuzzy sets.
- Rule evaluation - taking the fuzzified inputs, and applying them to the antecedents of the fuzzy rules. It is then applied to the consequent membership function (Table 1).
- Aggregation of the rule outputs - the process of unification of the outputs of all rules.
- Defuzzification - the input for the defuzzification process is the aggregate output fuzzy set chance and the output is a single crisp number. During defuzzification, we have adapted Max-membership principle method. This method is also known as height method. The scheme is limited to peaked output function.

4.2. Expert Knowledge Representation

Expert knowledge is represented based on the following two descriptors:

- Node Energy Level - energy level available in each node, designated by the fuzzy variable energy level,
- Distance from the centre - the value is calculated from the centre of network as discussed in the earlier section, designated by the fuzzy variable distance.

The linguistic variable used to represent the node energy level is divided into three levels: low, medium and high and distance is divided into three levels: small, medium and large, respectively. The outcome representing the chance of a node being the 'Leader' is divided into five levels: very small, small, medium, large, and very large.

The fuzzy rule base currently includes rules like the following: if the energy is high and the distance is small then the chance of the node to be elected as 'Leader' is very large. Thus we used $3^2 = 9$ rules for the fuzzy rule base. The rule base is shown in Table1. The membership functions developed and their corresponding linguistic states are represented in Figure 2 through 4.

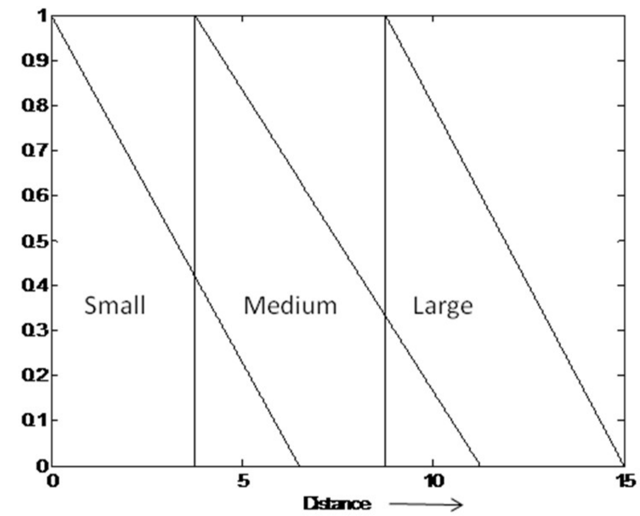

Figure 2: Fuzzy Set for Fuzzy Variable,' Distance'

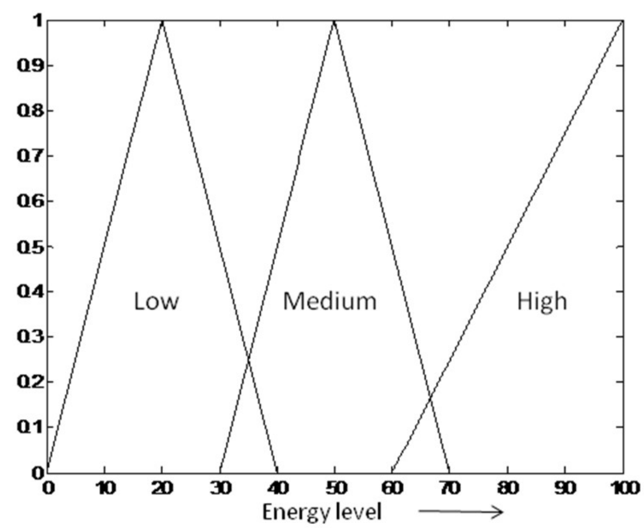

Figure 3: Fuzzy Set for Fuzzy Variable, 'Energy Level'

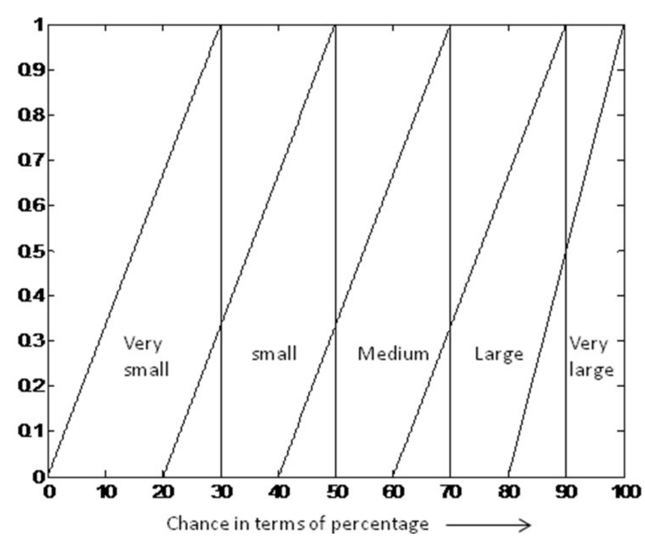

Figure 4: Fuzzy Set for Fuzzy Variable 'Chance for Leader'

Table 1
Fuzzy Rule Base

Fuzzy Rule	*Distance*	*Energy Level*	*Chance for 'Leader'*
RULE1	Small	Low	Small
RULE2	Small	Medium	Large
RULE3	Small	High	Very large
RULE4	Medium	Low	Small
RULE5	Medium	Medium	Medium
RULE6	Medium	High	Large
RULE7	Large	Low	Very small
RULE8	Large	Medium	Small
RULE9	Large	High	Medium

All the nodes are compared on the basis of the chances and the node with the maximum chance is then elected as the 'Leader'. If more than one node gets the maximum chance for being a 'Leader', the node having lowest identification number among them will be elected as 'Leader'.

4.3. Formal Representation of the Leader Election Algorithm

Notations used:

1. x_c is the X-coordinate of centre of the topology
2. y_c is the Y-coordinate of centre of the topology
3. min{X} is the minimum of set 'X'
4. max{X} is the maximum of set'X'

Begin

Step 1: *Broadcasting the positional information obtained through GPS by all the nodes*

Step 2: *Centre of the topology is calculated by all the nodes using*

$x_c = (x_{max} + x_{min}) / 2$ *and* $y_c = (y_{max} + y_{min})$

where

$x_{min} = \min \{x_1, x_2 \ldots, x_{n-1}, x_n\}$

$x_{max} = \max \{x_1, x_2 \ldots, x_{n-1}, x_n\}$

$y_{min} = \min \{y_1, y_2, \ldots, y_{n-1}, y_n\}$

$y_{max} = \max \{y_1, y_2, \ldots, y_{n-1}, y_n\}$

for i= 1 to N

{

Step 3: *Crisp value of distance between node-i and the centre of the topology is calculated and function for fuzzification is called to obtain fuzzy 'distance'. Energy level of the node-'i' is read and function for fuzzification is called to obtain fuzzy 'Energy level'.*

Step 4: *Fuzzy rule base is called with fuzzy 'distance' and 'Energy level' to obtain fuzzy output of chance of being a 'Leader' for node-i.*

Step 5: *Defuzification method is called to obtain crisp value of chance for being a leader in terms of percentage for node-i.*

}

Step 6: *Chance of being a 'Leader' is again broadcasted by all the nodes.*

Step 7: *Chance of being a 'Leader' is stored in a queue according to their identification number.*

for j = 1:N

Step 8: *if chance (j)=max(chance), then node j will be declared as 'Leader'.*

end

End

5. MOVEMENT ALGORITHM

Once the 'Leader' is elected, all other nodes will move maintaining connectivity with the 'Leader'. Since 'Leader' is not controlling the movements of other nodes rather other nodes are taking their own decision for movement, this scheme is not totally centralized. Movement algorithm is also fuzzy logic based. The algorithm is illustrated in the following sections.

5.1. Fuzzy Logic Control

We propose a fuzzy based algorithm for movement of the nodes. The model of fuzzy logic control consists of a fuzzifier, fuzzy rules, fuzzy inference engine, and a defuzzifier. The process is performed in four steps:

- Fuzzification of the input variables the distance from the 'Leader', velocity of the 'Leader' and the position of the node - taking the crisp inputs from each of these and determining the degree to which these inputs belong to each of the appropriate fuzzy sets.
- Rule evaluation - taking the fuzzified inputs, and applying them to the antecedents of the fuzzy rules. It is then applied to the consequent membership function (Table 2).
- Aggregation of the rule outputs - the process of unification of the outputs of all rules.
- Defuzzification - the input for the defuzzification process is the aggregate output fuzzy set modulated velocity and the output is a single crisp number. During defuzzification, we have adapted Weighted Average method. The weighted average method is formed by

weighting each membership function in the output by its respective maximum membership value. The algebraic expression is given in equation 2.

$$\bar{x} = \frac{\sum_i (x), c_i}{\sum_i \mu_i (x)} \quad (2)$$

where $\mu_i(x)$ is the membership value for firing the 'i' th rule, c_i is the weight for the 'i' th rule and $\bar{x}$ is the defuzzified crisp value and 'i' is the rule number fired.

5.2. Expert Knowledge Representation

Expert knowledge is represented based on the following three descriptors:

- Distance from the 'Leader' - the value is calculated from the centre of the network as discussed in the earlier section, designated by the fuzzy variable 'distance'.
- Velocity of the 'Leader' - the value is directly received through GPS, designated by the fuzzy variable 'velocity'.
- Position of the node - it is calculated from the coordinates of the node and the Leader, designated by the fuzzy variable 'position'.

The linguistic variable used to represent the 'distance' from the 'Leader' has three levels: low, medium and high, 'velocity' is divided into three levels: small, medium and large, and 'position' of the node is divided into three levels: behind, aligned and front respectively. The outcome representing the velocity for the next beacon interval is divided into three levels: small, medium and large.

Thus, we used $3^3 = 27$ if-then rules for the fuzzy rule base. The rule base is shown in Table 2. The membership functions developed and their corresponding linguistic states are represented in Figures 5 through 7.

Table 2
Fuzzy Rule Base for Movement Algorithm

Fuzzy Rule	*Distance*	*Velocity*	*Position*	*Velocity for next beacon interval*
RULE 1	Small	Small	Behind	Small
RULE 2	Small	Small	Aligned	Small
RULE 3	Small	Small	Front	Small
RULE 4	Small	Medium	Behind	Small
RULE 5	Small	Medium	Aligned	Small
RULE 6	Small	Medium	Front	Medium
RULE 7	Small	Large	Behind	Medium
RULE 8	Small	Large	Aligned	Medium
RULE 9	Small	Large	Front	Large
RULE 10	Medium	Small	Behind	Medium
RULE 11	Medium	Small	Aligned	Medium
RULE 12	Medium	Small	Front	Small
RULE 13	Medium	Medium	Behind	Medium
RULE 14	Medium	Medium	Aligned	Medium
RULE 15	Medium	Medium	Front	Medium
RULE 16	Medium	Large	Behind	Large
RULE 17	Medium	Large	Aligned	Large
RULE 18	Medium	Large	Front	Medium
RULE 19	Large	Small	Behind	Large
RULE 20	Large	Small	Aligned	Large
RULE 21	Large	Small	Front	Small
RULE 22	Large	Medium	Behind	Large
RULE 23	Large	Medium	Aligned	Large
RULE 24	Large	Medium	Front	Small
RULE 25	Large	Large	Behind	Large
RULE 26	Large	Large	Aligned	Large
RULE 27	Large	Large	Front	Medium

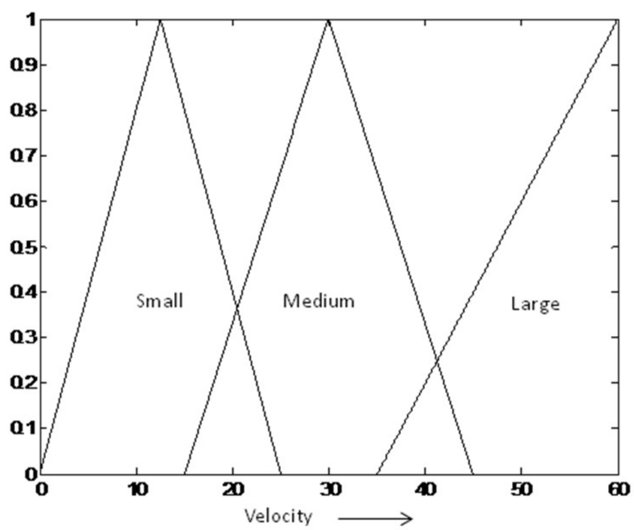

Figure 5: Fuzzy Set for Fuzzy Variable, 'Velocity'

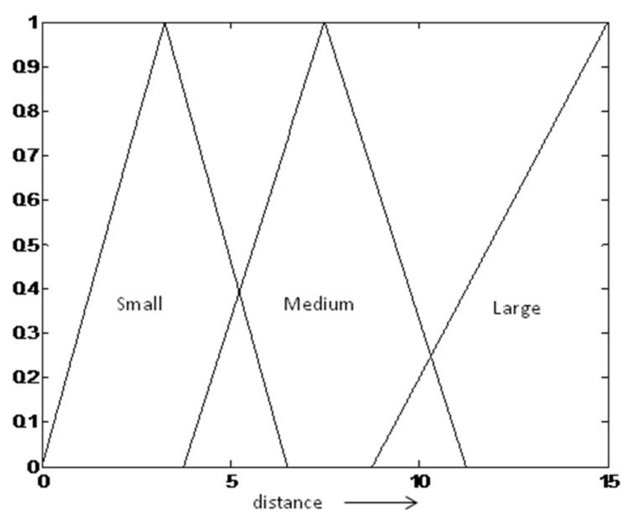

Figure 6: Fuzzy Set for Fuzzy Variable, 'Distance'

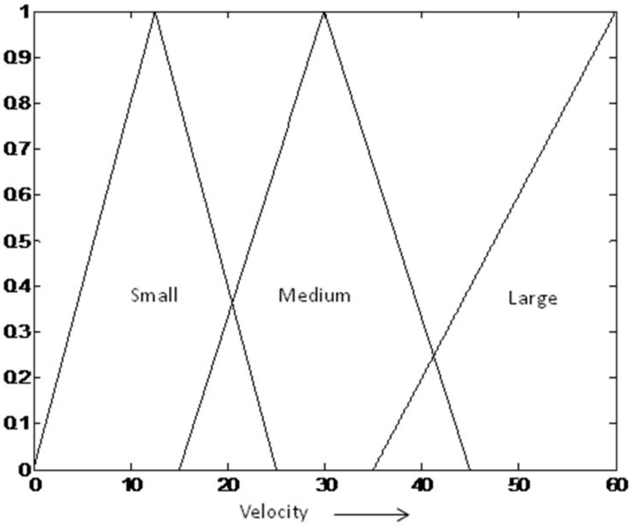

Figure 7: Fuzzy Set for Fuzzy Variable, 'Velocity for the Next Beacon Interval'

5.3. Formal Representation of the Movement Algorithm

Begin

> ***Step 1:*** *Broadcasting of position and velocity by the 'Leader'.*
>
> *for i= 1 to N-1*
>
> *{*
>
> ***Step 2:*** *Information is received from 'Leader.'*
>
> ***Step 3:*** *Crisp value of distance between node-i and 'Leader', velocity of 'Leader' and position of the node-i with respect to 'Leader' are calculated and function for fuzzification is called to obtain fuzzy variable.*
>
> ***Step 4:*** *Fuzzy rule base is called with fuzzy distance, velocity and position.*
>
> ***Step 5:*** *Defuzification method is called to obtain crisp velocity of the node-i.*
>
> *}*
>
> ***Step 6:*** *Choose velocity of the 'Leader' by any random value within the* V_{max}.
>
> ***Step 7:*** *Allow all nodes to move for one beacon interval.*
>
> ***Step 8:*** *Go to step 1.*

End

6. SIMULATION RESULTS

To test and analyze the algorithm, experimental studies were performed. The simulator was programmed using MATLAB in Windows environment. We modeled the energy level of the nodes by taking random values within the range of 0 to 100. Simulator is designed for any number of nodes and we have studied simulation of our algorithm for different number of nodes and for different simulation time ranging from 1 hour to 20 hour. The result obtained through simulation is encouraging. But for the simplicity and clarity, simulation results with twenty and twenty one nodes for five hour are shown graphically for two different sample networks. For both the sample networks, simulation run of 'Leader' election and the movement algorithm is carried out. For simulation of these two sample networks we considered the system parameters as, maximum communication range 'R_{max}' = 15KM, maximum allowable preferred velocity 'V_{max}' = 60KM / hr, safe distance 'R_{th}'=10KM. So, the beacon interval is (15-10) / 2 x 60 hr. i.e. 2.5 minutes. Based on the energy level and the distance from the centre of the network a particular node is elected as leader. Graphically that node is shown along with the other nodes of the network. During movement of the network, distances of all other nodes from the leader are also shown graphically.

6.1. Sample Network 1

The network is of twenty nodes and their initial coordinates are (2, 1), (1.5, 0), (-2, 2), (3, 0), (1, 0), (-1, -1), (2, 2), (-2, -2), (2, 3), (4.5, 1), (2, -1), (3.2, 1), (0, 2), (3, 2), (3, -1), (2, 4), (-2, 1.5), (1.3, 2), (-1, 2.5) and (3, -3) respectively. The energy levels of the nodes are shown in the Figure 8.

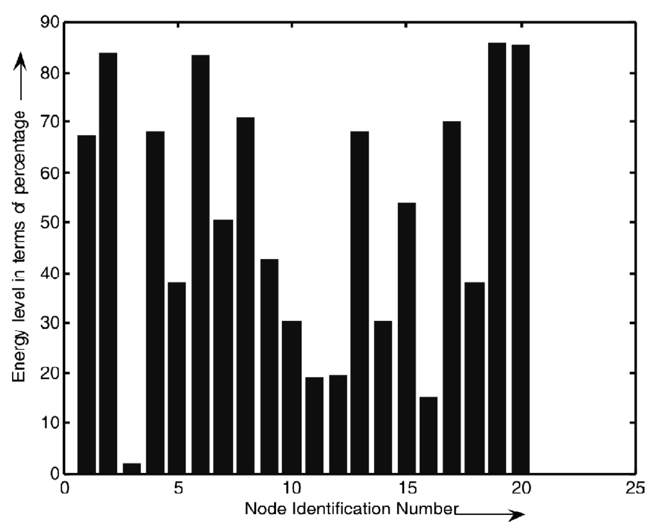

Figure 8: Showing Energy Level of Twenty Nodes

6.1.1. Leader Election

According to our 'Leader' election algorithm and based on the above information, node number 2 is selected as 'Leader'. In Figure 9, position of all the nodes and the 'Leader' is shown.

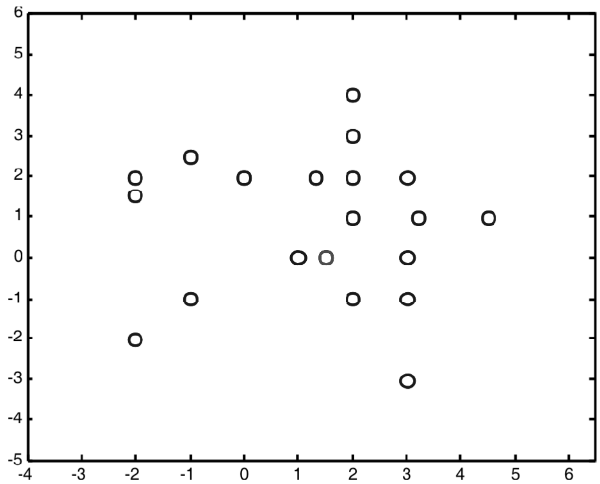

Figure 9: Showing the "Leader', Marked as Red and all other Nodes for Sample Network -1

6.1.2. Movement Algorithm

During movement, distances from the 'Leader' for all other nodes are shown graphically in Figure 10 to Figure 13 for the sample network-1.

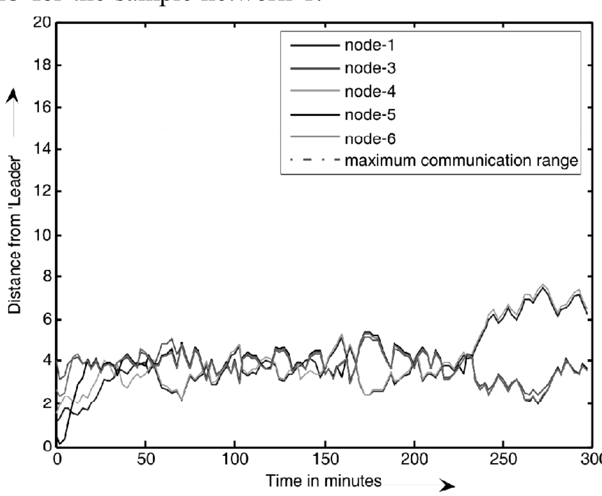

Figure 10: Showing the Distance from the 'Leader' for node-1, 3, 4, 5 and 6 for five hours, for sample-1

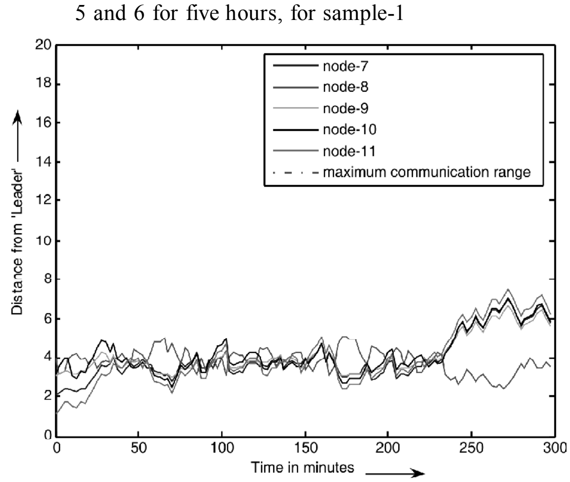

Figure 11: Showing the Distance from the 'Leader' for node-7, 8, 9, 10 and 11 for five hours, for sample-1

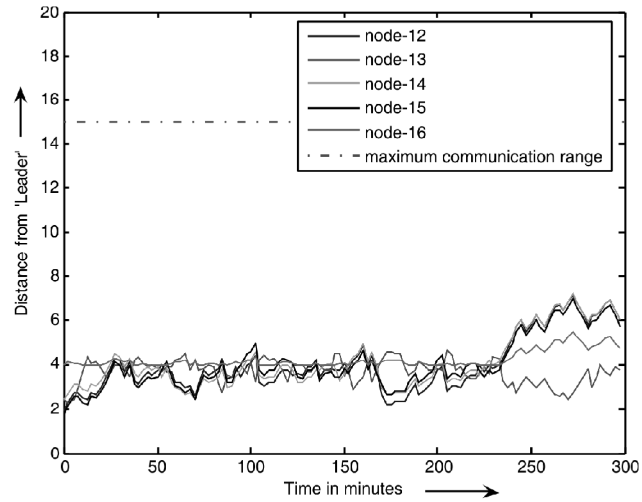

Figure 12: Showing the Distance from the 'Leader' for node-12, 13, 14, 15 and 16 for five hours, for sample-1

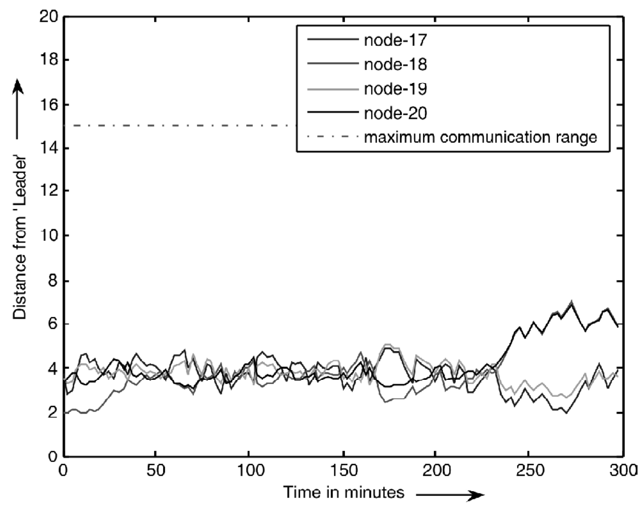

Figure 13: Showing the Distance from the 'Leader' for node-17, 18, 19 and 20 for five hours, for sample-1

6.2. Sample Network 2

The network is of twenty one nodes and their initial coordinates are (3.4, 1.2), (0, 5.6), (-2, 3.1), (3, 6), (1, 0), (-1, -1), (2, 5.1), (-2, -1.4), (5, 3), (1, 1.5), (2, -1.8), (-3,-1), (0, 2), (3, 2), (3, 5), (2, 4), (-2, -1.5), (3, 1. 2), (-1, -2.5), (0,0) and (3.2, -3) respectively. The energy levels of the nodes are shown in the Figure 14.

6.2.1. Leader Election

According to our 'Leader' election algorithm and based on the above information, node number-14 is selected as 'Leader'. In Figure 15, position of all the nodes and the 'Leader' is shown.

6.2.2. Movement Algorithm

Distance from the 'Leader' for all the remaining nodes are shown graphically in Figure 16 to19 for the sample network-2.

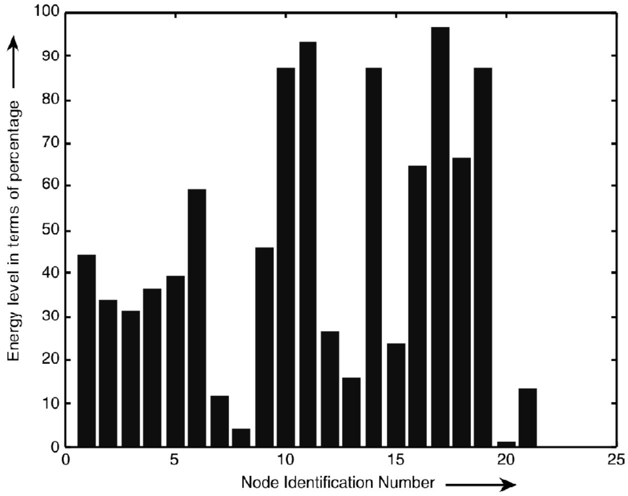

Figure 14: Showing the Energy Level for 21 Nodes of Sample Network-2

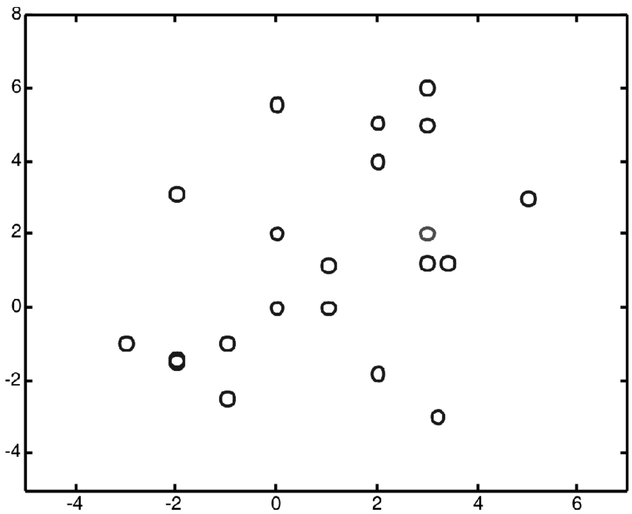

Figure 15: Showing the "Leader", Marked as Red and all other Nodes for Sample Network –2

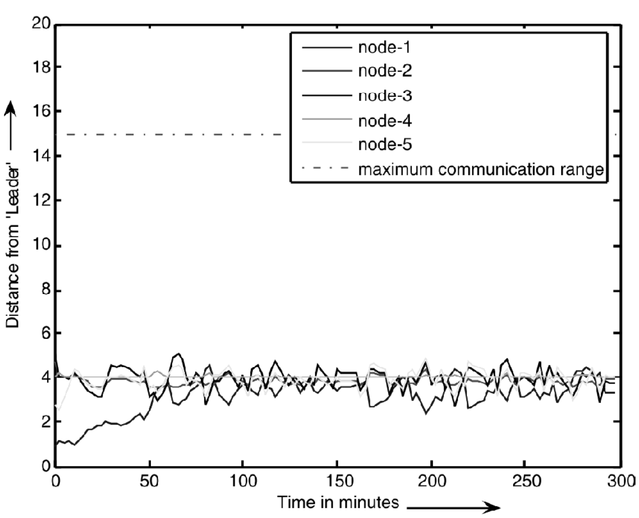

Figure 16: Showing the Distance from the 'Leader' for Node-1, 2, 3, 4 and 5 for Five Hours, for Sample-2

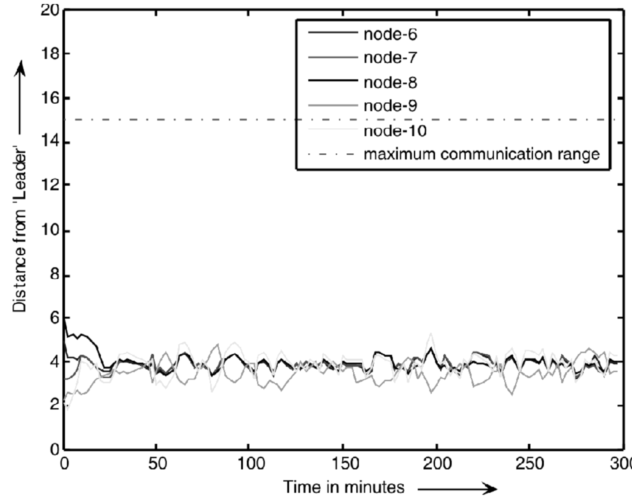

Figure 17: Showing the Distance from the 'Leader' for node-6, 7, 8, 9 and 10 for Five Hours, for Sample-2

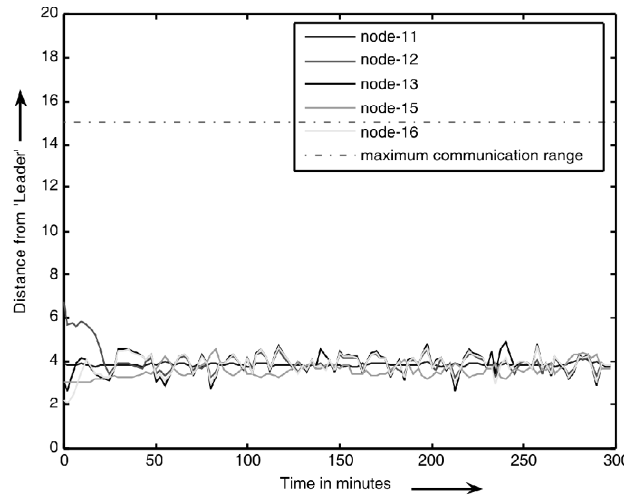

Figure 18: Showing the Distance from the 'Leader' for node-11, 12, 13, 15 and 16 for Five Hours, for Sample-2

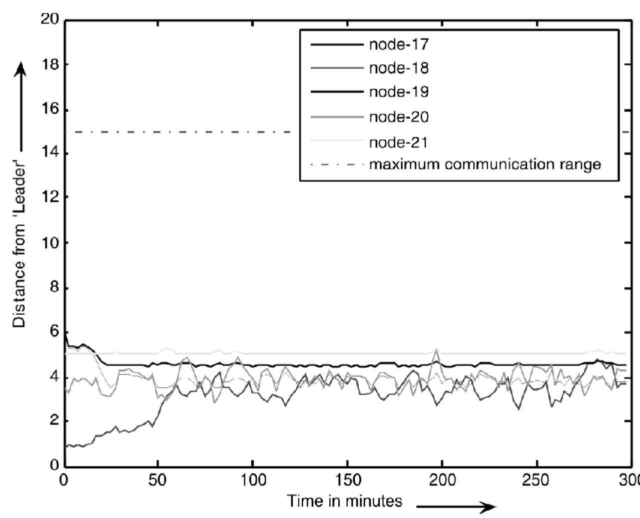

Figure 19: Showing the Distance from the 'Leader' for Node-17, 18, 19, 20 and 21 for Five Hours, for Sample-2

Through simulation we observe the following points:

(i) (a) For sample network-1 node 19 has the highest energy level but this node is not the central node. On the other hand node-5 is the central node but its energy level is below 40 %. Our algorithm is selecting node-2 as the leader as shown in figure 9. Now if we observe the energy level and the centrality. This is the best choice depending on the energy level and proximity from the centre.

(b) During movement, the distances from the leader are shown graphically in Figure 10, 11, 12 and 13. From these figures it is clear that all the nodes are not only maintaining maximum communication range but also maintaining safe distances during movement of the network.

(ii) (a) Again for the sample network-2 it is observed from the Figure 14 that the node number 10 is located at the center of the network and its energy level is also high. Node number 17 has the highest energy level but it is not close to the center as shown in the Figure 14 and 15. But our algorithm is neither selecting node number 10 nor node 17 as the leader. Now, we observe that node 14 has an energy level slightly greater than that of node 10 but in centrality point of view node 10 is better. But our algorithm is selecting node 14 as the leader. So our algorithm is giving more importance to the energy level for selecting the leader just to avoid the drainage of battery of the leader.

(b) Figure 16, 17, 18 and 19 show the distances of all the nodes from the leader and it is observed that the change in distances are very smooth during movement of the nodes. It is also observed that all the nodes are always maintaining the safe distance from the leader.

From the above simulation result it is now clear that our semi-distributed scheme is capable of maintaining topology of the network by suitably selecting a leader which is not only the central node but also the node with moderate energy level.

7. PERFORMANCE STUDY

The performance of the algorithm is evaluated by comparing it with the algorithm proposed by S.S. Basu and A. Chaudhuri in [1]. In [1], an algorithm to elect coordinator and a separate algorithm for node movement have been proposed. A comparative study between the two algorithms regarding coordinator / leader election is given in table 3.

Table 3
Performance Evaluation of the Proposed Algorithm

Sl. No.	*Algorithm proposed in [1]*	*Algorithm proposed in this paper*
1.	Coordinator is elected based on only the positional information.	'Leader' is elected not only based on positional information but also based on the energy level of the nodes.
2.	Coordinator election algorithm is conventional one.	'Leader' election algorithm is fuzzy logic based.
3.	Coordinator is selected to give the full responsibility to control the movement of all other nodes in the network. So there is a huge control overhead on the part of the coordinator.	'Leader' is selected not to give it the responsibility to control other nodes; rather other nodes will take the responsibility to maintain the connectivity with the 'Leader' node. So the control overhead is distributed over all the nodes.

Now a comparison is made with respect to the movement algorithm. The connectivity of the network is maintained by velocity modulation in both the cases during movement of the network. These algorithms are giving the guaranteed connectivity of the network. But at the same time stability of the network should be ensured. If a particular node has to change its velocity with a huge amount within one beacon interval, then the stability of the node will be degraded and thereby the stability of the network will be hampered. It is proved from the simulation results that both the algorithms are capable to maintain connectivity of the network during movement. In [1], entire responsibility for movement control is with the coordinator and hence the coordinator is overloaded but the same responsibility is distributed among all the nodes in our proposed algorithm. To evaluate the performance of both the algorithms with respect to stability, simulation run of the algorithms with the same initial network configuration is carried out. It is clear from the above simulation run with different network scenarios that the algorithm proposed in this paper is better compared to the algorithm proposed in [1]. For example, simulation run of both the algorithms for only one network scenario are given next.

Test Network Sample: The network consists of five nodes with initial coordinates (-4, -2), (2, 7), (5, 2), (-2, 4) and (3, 5). The energy levels of the nodes are given in the Figure 20.

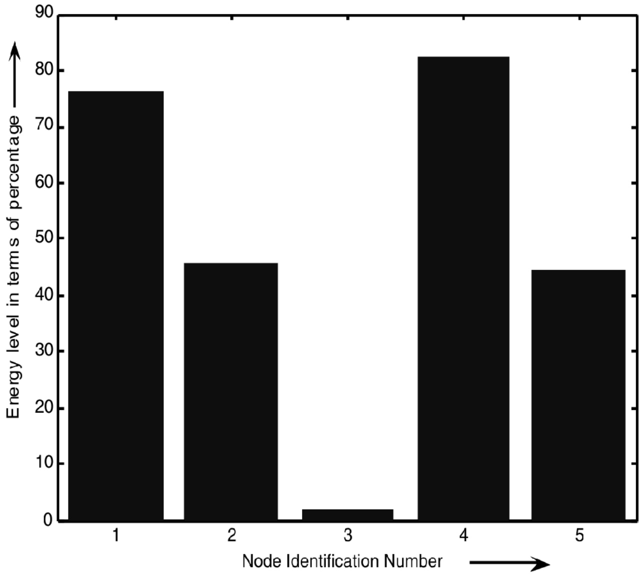

Figure 20: Energy Level of the Nodes

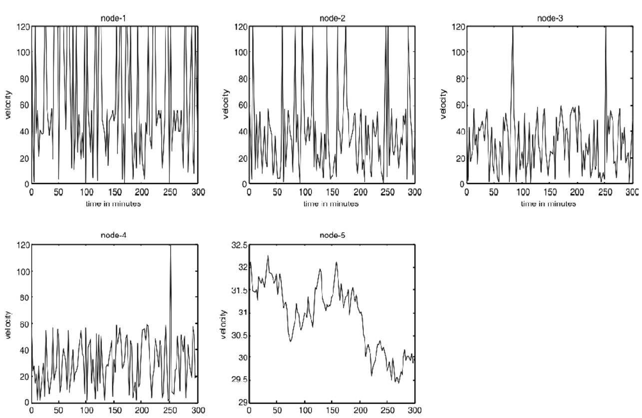

Figure 21: Showing Velocity Changes of the Nodes for the Test Network Sample for Algorithm Proposed in [1]

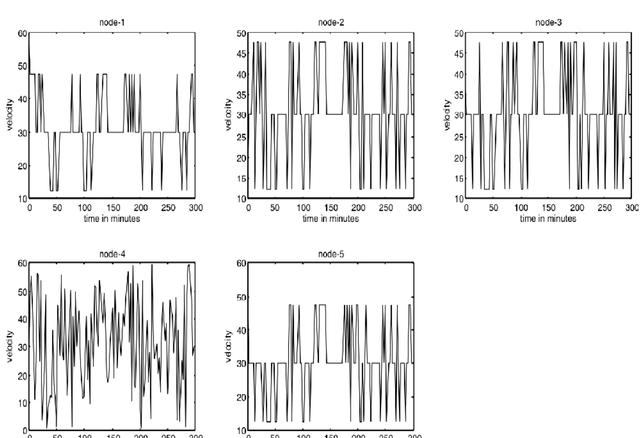

Figure 22: Showing Velocity Changes of the Nodes for the Test Network Sample for Proposed Algorithm

According to the algorithm in [1] node 5 becomes the coordinator for the above test network sample but if we consider energy level of the node also then the election of the coordinator is not a good one. Figure 21 shows the change of velocity of different nodes due to the movement algorithm proposed in [1].

The algorithm proposed in this paper select node 4 as a 'Leader'. Obviously it is a good choice compared to the algorithm in [1] since its energy level is also good. Velocity changes of different nodes during movement due to the proposed fuzzy based movement control algorithm are plotted in Figure 22. From the results it is now clear that velocity change is lesser with significant amount in the proposed algorithm compared to the algorithm proposed in [1]. Again the algorithm proposed in [1] is an improvement with respect to the algorithm proposed in [3]. So the performance of the algorithm proposed in this paper is better than the algorithm proposed in [1, 3].

8. CONCLUSION

As proposed earlier, we have been able to develop a novel algorithm in which the nodes of the network always maintain the topology. Furthermore we observe that although the network moves in one direction each individual node can move in any preferred direction thereby removing some of the problems present in [1, 3 and 6]. Since we have introduced fuzzy logic in both the 'Leader' election and the movement algorithm, MANET nodes are becoming more intelligent. In this algorithm 'Leader' will not control the other nodes rather each individual node will take its own decision for movement keeping its connectivity with the 'Leader'. Since there is a 'Leader' it seems a centralized approach but 'Leader' is not responsible for controlling other nodes rather it just decides the direction of movement of topology. So, each individual node is independent enough. In this sense the approach is distributed. So, we can say our scheme is semi-distributed. Due to the semi-distributed scheme the topology is not vulnerable if one of the nodes becomes non-functional except the 'Leader'. The proposed algorithm is electing 'Leader' not only based on positional information but also on its energy level just to reduce the chance of failure of the 'Leader'. Apart from this, no node ever diverges out of communication range during movement. The system never becomes static as a whole and hence no time is wasted in maintaining the network topology thereby ensuring greater efficiency.

REFERENCES

[1] Soumya Sankar Basu, Atal Chaudhari, "Self-Adaptive MANET: A Centralized Approach", *Foundations of Computing and Decision Sciences*, **29**, 2004.

[2] Elizabeth M. Royer, and Chai -Keong Toh , "A Review of Current Routing Protocols for Ad-hoc Mobile Wireless Networks", *IEEE Personal Communications*, April 1999.

[3] S. S. Basu and A. Chaudhari, "Self-adaptive Topology Management for Mobile Ad-Hoc Network", IE (I) Journal-ET, **84**, 2003.

[4] Jishan Mehedi, Surendra S. Dalu, and M. K. Naskar, "A Distributed Approach to Maintain Connectivity of Nodes in Mobile Ad-Hoc Networks", International Conference on Intelligent Systems & Networks IISN-2007, Haryana, February 23-25, 2007, 353-358.

[5] David B. Johnson and David A. Maltz "Dynamic Source Routing in Ad-Hoc Wireless Networks", Computer Science Department, Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15, 213 –3891.

[6] S. Samanta, S. S. Ray, S. SenGupta, M. K. Naskar, "A Novel Algorithm for Managing Network Configuration", Asian International Mobile Computing Conference 2006, Kolkata, Jan 04-07, 51-58.

[7] Jishan Mehedi, Surendra S. Dalu, and M. K. Naskar, "Performance Comparison of Some Routing Protocols for Mobile Ad-Hoc Networks" National Conference, INDIACom2007, February 23-24, 2007, 205-208, 212.

[8] J. Broch, D. A. Maltz, D. B. Johnson, Y. C. Hu, and J. Jetcheva. A Performance Comparison of Multi-hop Wireless Ad-hoc Network Routing Protocols. In Proceedings of Mobile Computing and Networking, 85–97, 1998.

[9] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector routing (DSDV) for Mobile Computers", ACM SIGCOMM, **24**(4), 1994.

[10] R. Ramnathan and R. Rosales-Hain, "Topology Control of Multihop Radio Networks using Transmit Power Adjustment" in Proceedings of IEEE Infocom, 404-403, 2002.

[11] C. Bettstetter, "On the Minimum Node Degree and Connectivity of a Wireless Multihop Network", in Proceedings of ACM Mobihoc, 80-91, 2002.

[12] Elliot D. Kaplan, "Understanding GPS: Principles and Applications", Arleets House, March 1996.

[13] Kihwan Kim, Ying Cai and Wallapak Tavanapong "Sharing Location Dependent Experiences in MANET" In Proceedings of the 7th International Conference on Mobile Data Management (MDM'06).

[14] Jinshan Liu, Francoise Sailhan, "Group Management for Mobile Ad Hoc Networks: Design, Implementaion and Experiment" in proc. of MDM 2005, Ayia Napa, Cyprus.

[15] A. Seetharam, A. Bhattacharyya and M. K. Naskar, "Implementing a Topology Management Algorithm for Mobile Ad-Hoc Networks" *Journal of Applied Computer Science*, Issue No-4, 2008, 44-50.

[16] Abhishek Bhattacharyya, Anand Seetharam and M.K.Naskar, "A Centralized Algorithm for Topology Management in Mobile Ad-hoc Networks through Multiple Coordinators" In proceedings of ICDCN2008, 581-586.

[17] Avik Ray, Kanad Basu, Samir Biswas, Mrinal K. Naskar, "A Novel Distributed Algorithm for Topology Management in Mobile Ad-hoc Networks" , International Conference on Computers and Devices for Communication 2006 (CODEC 06), 99-102, 2006, Kolkata, India.

[18] A. Konar, "Computational Intelligence: Principles, Techniques and Applications", Springer.

[19] Jishan Mehedi, M. K. Naskar, "A Fuzzy Based Distributed Approach to Maintain Connectivity of Nodes in Mobile Ad-Hoc Networks Considering Pursue Mobility Model", *International Journal of Computational Intelligence: Theory and Practice*, **4**(2), 2009.

[20] T. Camp, J. Boleng, and V. Davies. A Survey of Mobility Models for Ad-hoc Network Research. Wireless Communications & Mobile Computing (WCMC): Special issue on Mobile Ad HocNetworking: Research, Trends and Applications, **2**(5): 483–502, 2002.