

# A Fuzzy Based Distributed Algorithm for Maintaining Connected Network Topology in Mobile Ad-Hoc Networks Considering Freeway Mobility Model

Jishan Mehedi<sup>1</sup> and Mrinal Kanti Naskar<sup>2</sup>

<sup>1</sup>Department of Electronics and Communication Engineering, National Institute of Technology, Silchar, Assam, India

<sup>2</sup>Advanced Digital and Embedded Systems Lab, Department of Electronics and Telecommunication Engineering, Jadavpur University, Kolkata, India

Mobile Ad-Hoc Networks (MANETs) present real-time embedded system that is being used in a wide variety of applications where traditional networking infrastructure is practically infeasible. The highly dynamic character of a Mobile Ad-Hoc Network (MANET) poses significant challenges on network communications. Previous work on MANET has resulted in numerous routing protocols aiming to maintain network connectivity among the active nodes. This paper presents a fuzzy-based distributed algorithm to maintain connected MANET considering freeway mobility model. According to the algorithm, each node will control itself in a way that it can maintain its connectivity with other nodes. In this approach each node is enabled with a Global Positioning System (GPS) receiver. Through GPS each and every node is getting its position and velocity. After getting the information all the nodes in a network transmit their position and velocity information periodically. Obtaining information from all other nodes, each node will decide its own velocity to maintain connectivity. Moreover, faults to a particular node have also been considered in this algorithm. Results obtained through simulation studies show the effectiveness of the proposed algorithm.

*Keywords:* ad-hoc networks, connectivity, fuzzy logic, freeway mobility model, distributed algorithm

## 1. Introduction

A Mobile Ad-Hoc Network (MANET) is a network of autonomous mobile nodes able to communicate over wireless links without the help of any fixed infrastructure [1, 2, 3, 12]. MANET is projected to play a vital role when there is little or no communication infrastructure or the existing infrastructure is expensive or inconvenient

to use. The set of applications for MANETs is diverse, ranging from large-scale, mobile, highly dynamic networks to small networks that are constrained by power sources. Examples of applications are like search-and-rescue operations, multi-platform battle deployment, mobile sensors or satellite networks for quick sharing and acquisition of data in inhospitable terrain.

In MANET, the nodes act as individual routers, besides being the regular transceivers. The current focus of the researchers is to prescribe a routing algorithm which is adaptive with the network mobility. The existing routing protocols [2, 4] are Destination Sequence Distance Vector (DSDV) [8, 9, 10], Wireless Routing Protocol (WRP) [11], Optimized Link State Routing (OLSR) [4, 11], Cluster Switch Gateway Routing (CSGR) [4] under proactive scheme, Ad-hoc On-Demand Distance Vector Routing (AODV) [8], Dynamic Source Routing (DSR) [8], Associativity Based Routing (ABR) [8], Temporally Ordered Routing (TORA) [8, 9], Zone Routing Protocol (ZRP) [11], etc. The main concern of all these types of routing protocols is to provide a reasonably stable route between two arbitrary nodes in the mobile network, with the reduction in time complexity and control message overhead. But, none of these existing protocols takes care to maintain constant connectivity between the nodes within the network. In the worst case, a node can even get disconnected from the whole network and hence the concept of connectivity management

came into picture. All the nodes are mobile in a MANET and movement pattern of the nodes may be different, depending on the application of a MANET. Researchers have developed different mobility models, depending on different movement patterns of the nodes, namely Pursue Mobility Model [17, 18], Nomadic Community Mobility Model [17, 18], Freeway Mobility Model [18], Group Mobility Model [17, 18], Manhattan Mobility Model [16, 18], Random Gauss-Markov Model [17] etc. It is a challenging task to maintain a connected network during the movement of a MANET. Since mobility pattern is different for different mobility models, connectivity maintenance algorithm will also be different for different mobility models.

Two widely accepted approaches for topology management are centralized and distributed [1, 3]. Few centralized connectivity management algorithms [1, 3, 5] tried to maintain the overall topology of the network and at the same time eliminated the risk of detachment of a node due to its mobility. But these algorithms were affected due to the failure of the Coordinator/Leader. Few distributed schemes [6, 7] were also proposed in the literature. But none of them considered the fault tolerance while maintaining connectivity.

In this paper, a fuzzy-based distributed fault tolerance algorithm for connectivity management is proposed considering freeway mobility model. 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. Fourthly, fuzzy logic control is capable of making real time decisions, even with incomplete information. Conventional control systems rely on an accurate representation of the environment which generally does not exist in reality. Fifthly, fuzzy logic systems can manipulate the linguistic rules in a natural way. Finally, fuzzy controllers handle non-linear systems better when compared to conventional approaches [25]. To get these advantages, fuzzy-based distributed approach is chosen for connectivity management.

This algorithm not only ensures the connectivity of the nodes when there is no fault, but also restores connectivity if it is broken due to faulty nodes during movement of the networks.

The remainder of the paper is organized as follows: Section 2 reviews related research works on MANET. Freeway Mobility Model is described in Section 3. In Section 4 network model used in the proposed algorithm is discussed. Section 5 presents the proposed fuzzy-based algorithm. Simulation results of the proposed algorithm and a comparative study are reported in Section 6 and Section 7 respectively. Finally, Section 8 concludes the paper.

## 2. Related Works

A large number of research efforts in various aspects of MANET have been made in recent years, ranging from routing, multicasting, media access protocols, distributed service discovery etc. Maximum of the research work is concerned with the routing. R. Wattenhofer, L. Li, P. Bahl and Y. M. Wang [13] described that a node increases its transmission power until it finds a neighbor node in every direction based on directional information. The scheme is using the idea of logarithmic change in power, depending on the number of neighbors. But the question to keep all the nodes within the maximum transmission range remains unanswered. They also proved that nodes chosen within cones having  $2\pi/3$  angle result in a minimum energy network for a static network topology. R. Ramanathan and R. Rosales-Hain [14] adjusted node transmission power to maintain network connectivity. They have also used the same idea as in [13]. Connectivity is not guaranteed in this case either. Betstetter [15] models the neighboring node distribution by nearest neighbor methods known from analysis of spatial data. His 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. F. Bai, N. Sadagopan and A. Helmy [16] discussed the impact of mobility on the performance of routing protocols. T. Camp, J. Boleng, and V. Davies [17] 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 models, the mobile nodes movement decision depends upon other mobile nodes in the group and needs topology management. Several mobility models for wireless nodes are discussed by J. Boudec J.-Y. L. Boudec and M. Vojnović in [18]. Maintaining a connected network during movement of the nodes is a challenging task, especially when the network topology is changing continuously. If the networks do not remain connected, then none of the routing protocols will be able to find a path for communication. Few centralized and distributed algorithms were proposed for maintaining connected network during movement. S. S. Basu and A. Chaudhari [1] discussed one scheme for maintaining connectivity of the MANET nodes. In this scheme, elected ‘Coordinator’ will control the movements of other nodes through ‘HELLO’ and ‘control’ message communications. They modified their previous algorithm incorporating nodes with GPS receiver in [3]. Due to the use of GPS receiver, exact location of a node is known to the coordinator, and, as a result, mobility of the network is improved in this scheme. S. Samanta, S. S. Ray, S. SenGupta, M. K. Naskar [5] discussed a probabilistic approach for maintaining connectivity in a centralized manner. Jinshan Liu and Françoise Sailhan [19] presented a group management technique for Mobile Ad-Hoc Networks. However, in the centralized algorithms considered so far, the workload on the coordinator is immense. If the coordinator fails, then the whole network will fail. Location dependent experiences sharing between the nodes in MANET are discussed by Kihwan Kim, Ying Cai and Wallapak Tavanapong [20]. This gives the idea of distributed movement control of the nodes for maintaining connected networks during movement. Avik Ray, Kanad Basu, Samir Biswas, Mrinal K. Naskar [7] proposed one distributed connectivity maintenance algorithm. Another distributed scheme is proposed in [6]. These schemes totally eliminate the concept of coordinator and the workload of the network is distributed among all the nodes. Main problem with the above schemes is that sometimes, in emergencies, there may be rapid velocity changes in the nodes. To overcome this problem, fuzzy logic is introduced for movement control. E. Natsheh, A. B. Jantan, S. Khatun and S. Subramaniam [21] have

discussed fuzzy-based approach for finding out the route for local connectivity of nodes, but in this approach there is no guarantee that the whole network will be always connected.

This paper proposes fuzzy-based distributed algorithm for Freeway mobility model. This algorithm guarantees connectivity of the network even if it is broken due to the failure of a node.

### 3. Freeway Mobility Model

Mobility models are necessary for studying the performance of a MANET. Most of the research in this area is mainly based on simulation. These simulations have several parameters related to mobility patterns and the communication traffic patterns. In a MANET, nodes should move in a coordinated manner depending on the application. There are a number of mobility models, each tries to focus on capturing a specific real world situation or a more general analysis of realistic events. Nevertheless, these models can be grouped into two categories, trace-based and synthetic models [17]. Trace-based modeling requires specific input of mobility traces of a user. This may be good and relevant for a specific, isolated case; however synthetic models’ goal is to represent the real world behavior of mobile users. There are several mobility models that attempt to capture movement and behavior of users. These models include the Random Waypoint Model, Random Walk Model [17], Random Trip Model [18], Reference Point Group Mobility Model [17] and Freeway Mobility Model [17]. Out of these models, the Freeway Mobility Model is selected for the present work. F. Bai, N. Sadagopan and A. Helmy [16] proposed this model to emulate the motion behavior of mobile nodes on a freeway [17, 22]. This model can be used in exchanging traffic status or tracking a vehicle on a freeway. Maps are used in this model. There are several freeways on the map and each freeway has lanes in both directions as shown in Figure 1. The velocity of mobile node is temporally dependent on its previous velocity. The differences between Random Waypoint Model (RWM) [23] and Freeway Mobility Model are: 1) each mobile node is restricted to its lane on the freeway, 2) the velocity of the mobile node is temporally dependent on its previous velocity and 3) if two mobile nodes on the

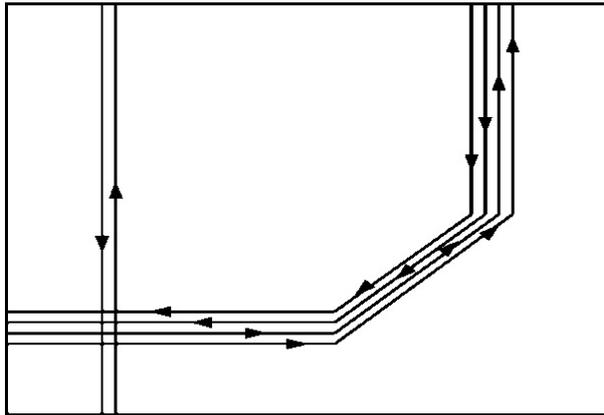


Figure 1. Maps for freeway mobility model.

same freeway are within the safe distance, the velocity of the following node cannot exceed the velocity of the preceding node. Numerous applications exist for this type of mobility model. This model emulates the motion behavior of mobile nodes on a freeway. This mobility pattern may have application in scanning and searching purposes e.g. anti-personal mines deactivation robots and can be used in exchanging traffic status or tracking a vehicle on a freeway and for transportation convoy. One particular useful application of a MANET is a Vehicular Ad-hoc Network (VANET), where vehicles represent mobile nodes that communicate using their onboard devices amongst ships where no infrastructure is available which helps to exchange information regarding pirates, weather, threat, emergency or critical condition.

#### 4. Network Formation

In this paper, MANET is modeled considering freeway mobility model for some specific applications like convoy of soldiers moving towards battlefield or rescue team in different vehicles moving towards same destination in the same road. In these cases, one vehicle can exchange its information through the onboard MANET nodes. To model the network for those applications, this paper defines some useful parameters along with some assumptions. A number of lemmas along with their proofs are given in the next section to choose the values of the parameters.

In this paper, a fuzzy-based distributed algorithm is proposed for maintaining a connected

network during the entire time of the movement of the nodes. Freeway mobility model has been considered assuming uni-directional movement of the nodes in this algorithm. The network model is shown in Figure 2. Here each node will try to maintain connectivity with its front node only and, as a result, the network will become connected. Here 'y' is the front node and 'u' is the rear node.

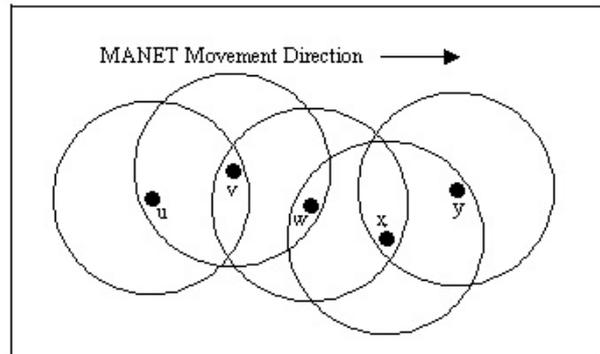


Figure 2. Network model.

#### 4.1. Parameter Definitions and Their Notations

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

*Threshold Distance ( $R_{th}$ ):* Initially, each node will have to maintain distance less than or equal to a specific distance from its front node. This distance is called threshold distance and it is denoted by ' $R_{th}$ ' where maximum communication range ' $R_{max}$ ' is greater than threshold distance, ' $R_{th}$ ' as shown in Figure 3.

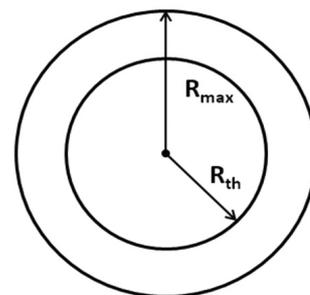


Figure 3. Demonstrating  $R_{max}$  and  $R_{th}$ .

**Beacon Interval ( $T$ ):** The time interval after which the nodes transmit their information periodically is called beacon interval which is denoted by ' $T$ '. Beacon interval ( $T$ ) is chosen  $(R_{max} - R_{th})/V_{max}$  where  $V_{max}$  is the maximum velocity of the node. The proof for choice of the beacon interval is shown in Lemma 4.3.1.

## 4.2. Network Setting

- i) Each node has a unique identification number.
- ii) All nodes are enabled with GPS trans-receivers which can furnish the current position and velocity of each node.
- iii) All the nodes have a predefined maximum velocity,  $V_{max}$ .
- iv) At the beginning, the network is connected and all the nodes can communicate with each other in single hop.
- v) Each node also acts as router and one routing table which keeps the record of all the single hop connected nodes with their positional information.

## 4.3. Lemmas

### 4.3.1. Lemma for choosing beacon interval

If maximum communication range is ' $R_{max}$ ' and threshold distance is ' $R_{th}$ ', where  $R_{max} > R_{th}$ , maximum preferred velocity ' $V_{max}$ ', then beacon interval must be less than or equal to  $(R_{max} - R_{th})/V_{max}$ , and 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  $V_{max}$ . Hence, the maximum relative distance traveled in a beacon interval is  $TV_{max}$ . Since initially maximum separation between two nodes may be ' $R_{th}$ ', a connected node cannot become disconnected if,

$$T \cdot V_{max} \leq (R_{max} - R_{th}) \quad (1)$$

or,

$$T \leq (R_{max} - R_{th})/V_{max}. \quad (2)$$

### 4.3.2. Lemma for choosing threshold distance, ' $R_{th}$ ', maximum communication range, ' $R_{max}$ ' and critical velocity, ' $V_c$ '

If maximum  $R_{th}$  amount of spacing between two nodes is allowed when there is no fault, and if critical velocity  $V_c = V_{max}$  and  $3R_{th} \leq 2R_{max}$  where  $R_{max}$  is the maximum communication range, then it is possible to recover connectivity within one beacon interval, even after connectivity is broken due to a faulty node.

**Proof:** The aim is to restore connectivity within one beacon interval. According to the algorithm, if there is no fault, all the nodes will maintain a distance less than or equal to  $R_{th}$  from their respective front nodes. So the worst case scenario before any fault to occur at node 3 is illustrated in Figure 4.

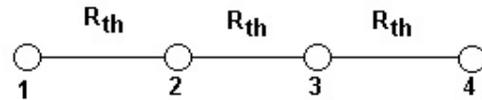


Figure 4. Showing the worst case separation.

It is clear from Figure 4 that in the worst case the distance between node 1 and node 2 is  $R_{th}$  and the distance between node 2 and node 4 is  $2R_{th}$ . In this case, according to the algorithm, node 2 will rush with critical velocity ( $V_c$ ) towards node 4 and node 4 will stop itself. So after one beacon interval distance between node 1 and node 2 will be  $(V_c \times T + R_{th})$  and the distance between node 2 and node 4 will be  $(2R_{th} - V_c \times T)$ . Hence, the restoration of connectivity is possible within one beacon interval if and only if

$$V_c \times T + R_{th} \leq R_{max} \quad (3)$$

and

$$(2R_{th} - V_c \times T) \leq R_{max}. \quad (4)$$

Now, if  $T$  is replaced in equation 3 using equation 2, then

$$V_c \leq V_{max}$$

Again, if  $T$  and  $V_c$  are replaced in equation 4,

$$2R_{th} - V_{max} \cdot (R_{max} - R_{th})/V_{max} \leq R_{max} \quad (5)$$

or,

$$3R_{th} \leq 2R_{max}. \quad (6)$$

## 5. Proposed Algorithm

In a centralized system, all the control overhead is with the central administrator or coordinator. But in a distributed system [7, 24], control overhead is shared by all the nodes. So the distributed system becomes faster than centralized one. Moreover, in the centralized system, if the administrator or coordinator fails, then the whole system collapses.

Global Positioning System (GPS) [25, 27] receiver is used to get the positional data of a location from satellite. A minimum of three satellite links are required for the module to get the valid data. Here the GPS module tracks up to 12 satellites. The module provides current time, date, speed, longitude, latitude, and altitude. It also provides travel direction. This paper considers that the module has  $\pm 3$  meter position accuracy. Clock synchronization is done using GPS time.

A fuzzy controller executes three basic steps: fuzzification, inference and defuzzification. During fuzzification, the numeric input values are mapped to fuzzy sets by applying the membership functions. Based on the fuzzified inputs, the controller infers through its IF – THEN rules and produces an aggregated fuzzy output. The final control action is derived by defuzzifying this aggregated fuzzy output.

### 5.1. Fuzzy Logic Control and Its Expert Knowledge Representation

The model of fuzzy logic control consists of a fuzzifier, fuzzy rules, fuzzy inference engine,

and a defuzzifier. There are specific components characteristic of a fuzzy controller to support a design procedure. The controller is between a preprocessing block and a post processing block, as shown in the block diagram in Figure 5.

Here, the objective is to control the movement of each node by itself. Since, this paper proposes an algorithm for maintaining connectivity considering freeway mobility model, it is reasonable to decompose the problem into three independent controllers corresponding to three different types nodes – front node, rear node and other nodes, with the benefit of simplifying the design and tuning. There are slight differences in the basic structure of the three controllers depending on the input variables. The fuzzy controllers follow the design methodology proposed by Mamdani [28]. The controller is designed with following considerations:

- Inputs: the distance from the front node ( $D_f$ ), distance from the behind node ( $D_b$ ), velocity of the front node ( $V_f$ ) and velocity of the behind node ( $V_b$ ) as input variables.
- Fuzzy sets: For an increased control granularity, three fuzzy sets are designed for distances, namely S (Small), M (Medium), L (Large) and Low (L), Medium (M) and High (H) for velocities.
- Membership functions. To leverage the computational effort, triangular membership functions are used.
- Rule-based inference: the max-min fuzzy inference method is used.
- Output: Output variable is the velocity of a node for the next beacon interval. The controller output is decided by defuzzifying the

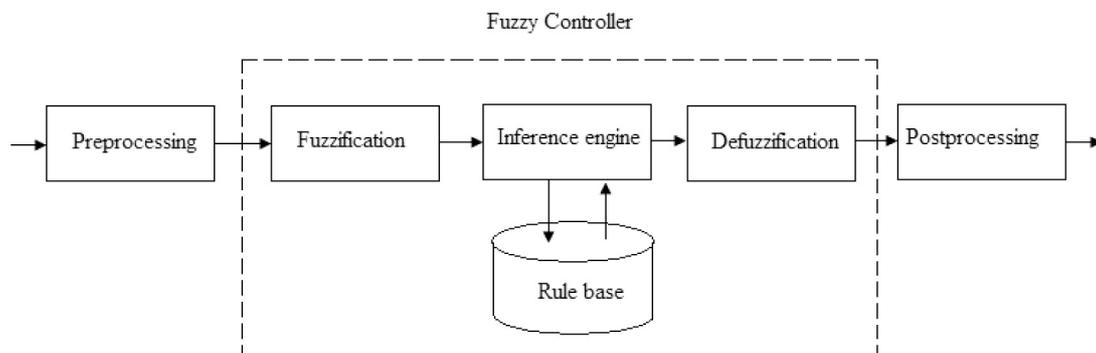


Figure 5. Blocks of a fuzzy controller.

aggregated inference result using weighted average method [28]. The weighted average method is formed by weighting each membership function in the output by its respective maximum membership value. The corresponding algebraic expression is given in equation 7. Where  $\mu_i(x)$  is the membership value,  $c_i$  is the weight and  $\bar{x}$  is the defuzzified crisp value.

$$\bar{x} = \frac{\sum_i \mu_i(x) \cdot c_i}{\sum_i \mu_i(x)} \quad (7)$$

The membership functions developed and their corresponding linguistic states are represented in Figure 6a and Figure 6b respectively.

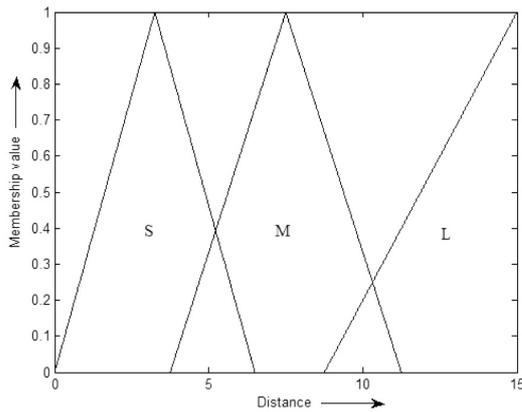


Figure 6a. Fuzzy set for fuzzy variable, distance.

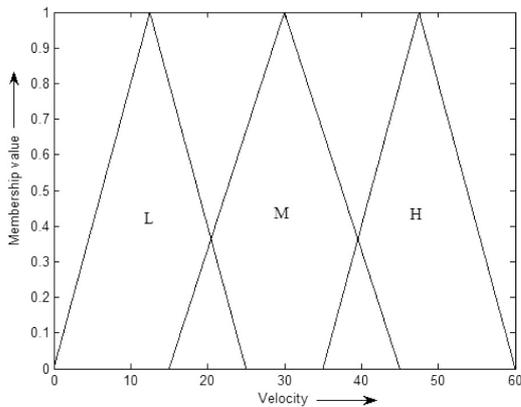


Figure 6b. Fuzzy set for fuzzy variable, velocity.

Distance ( $D_b$ ) \backslash Velocity ( $V_b$ )	S	M	L
	L	H	L
M	H	M	L
H	H	H	M

Table 1. Rule Base-1.

Distance ( $D_f$ ) \backslash Velocity ( $V_f$ )	S	M	L
	L	L	M
M	L	M	H
H	M	H	H

Table 2. Rule Base-2.

Distance ( $D_f$ ) \backslash Velocity ( $V_f$ )	S	M	L
	For distance ( $D_b$ ) = 'S'		
L	L	L	M
M	L	M	H
H	L	M	H
For distance ( $D_b$ ) = 'M'			
L	L	L	M
M	L	L	H
H	L	M	H
For distance ( $D_b$ ) = 'L'			
L	L	L	M
M	L	L	M
H	L	M	H

Table 3. Rule Base-3.

In this scheme, three different rule bases, depending on the index of the node, have been considered. Rule Base-1 is for front node, Rule Base-2 is for rear node and Rule Base-3 is for all other nodes.

**Fuzzification:** Fuzzy sets form a systematic basis for working with linguistic variables which are variables whose values are not numbers, but linguistic terms. Each term is characterized by a fuzzy set. Figure 6a gives an example of the linguistic variable Distance ( $D$ ), decomposed into three terms – Small, Medium and Large –

and the associated membership functions. The first step in fuzzy inference is to map the crisp inputs to the corresponding linguistic variables through the use of membership functions. This step is referred to as fuzzification. Considering the example in Figure 6a, a measured distance value 4 kilometers is fuzzified as 0.7692 Small and 0.0667 Medium. Often, in fuzzy decision and control systems, the current measured inputs are fuzzified and taken into account in the inference.

**Rule Base:** Fuzzy systems are fundamentally based on IF-THEN rules applied over the fuzzified inputs. Depending on the system to be designed, the rule base is constructed from expert knowledge, input-output data, or a combination of the two. The rules have the following format:

*IF  $x_1$  is  $A_1$  AND  $x_2$  is  $A_2$  AND . . .  $x_n$  is  $A_n$   
THEN  $y$  is  $B$ ,*

where  $x_i$  are input variables with their fuzzy sets  $A_i$  and  $y$  is the output variable with the corresponding fuzzy set  $B$ . Considering node movement algorithm, an example of such a rule would be:

*IF distance from the front node ( $D_f$ ) is Small  
AND distance from the behind node ( $D_b$ ) is  
Small AND velocity of the front node ( $V_f$ ) is  
Medium THEN velocity for the next beacon  
interval is Small*

**Inference:** In the inference step, each rule is interpreted as a fuzzy implication and the results of the rules are combined to produce an aggregate fuzzy output. Two of the most frequently used methods for implication and rule connection are max-min and sum-product [28]. The max-min fuzzy inference for the movement control of the front, discussed in the following section is considered. For example, the two input variables – distance from the behind node ( $D_b$ ), and velocity of the behind node ( $V_b$ ) – have measured values of 4 kilometers/hour and 27 kilometers/hour. The input distance from the behind node ( $D_b$ ) is fuzzified through three membership functions: Small, Medium and Large, using Figure 6a and the input velocity of the behind node ( $V_b$ ) is fuzzified through three membership functions: Small, Medium and Large, using Figure 6b. The rule base, as shown in Table 1, consists of 9 rules analyzing the fuzzified distance from the behind node

( $D_b$ ) and velocity of the behind node ( $V_b$ ) values. The measured distance from the behind node ( $D_b$ ) value, 4 kilometers/hour, is fuzzified as 0.7692 Small and 0.0667 Medium. Similarly, velocity of the behind node ( $V_b$ ) value, 27 kilometers/hour, is fuzzified as 0.8 Medium. So these fuzzy variables will fire two IF THEN rules. In each rule, the minimum membership value between the two variables is selected and used to trigger the level of the output variable. For example, when  $D_b$  is Small (0.7692) AND  $V_b$  is Medium (0.8), then velocity for the next beacon interval will be Large (0.7692). Here the minimum of the membership values (0.7692 and 0.8) is considered. Similarly, for the second case, when  $D_b$  is Medium (0.0667) AND  $V_b$  is Medium (0.8), then velocity for the next beacon interval will be Medium (0.0667). Now, when these two rules are combined, then maximum valued membership value will be considered. For the above example, inference engine will produce velocity for the next beacon interval as Large (0.7692).

**Defuzzification:** The final step is to defuzzify the aggregated output back to a crisp number that can be used for making decisions or taking control actions. Possible defuzzification methods include: centroid method, largest of maximum (LOM), mean of maximum (MOM), weighted average method etc [28]. For each of the defuzzification methods, crisp value of the output variable is calculated using the corresponding equations. 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 7. If the above example is considered and the weighted average method is used, then the crisp value of output velocity for the next beacon interval is 46.1043 kilometers/hour.

## 5.2. Formal Representation of the Algorithm

This paper presents connectivity maintenance algorithm for freeway mobility model considering fault tolerance. This algorithm is also distributed in the sense that all the nodes will take their own decision for movement to maintain network connectivity. This paper proposes

three different movement schemes: i) movement algorithm for the front node, ii) movement algorithm for the rear node and iii) movement algorithm for all the nodes other than front and rear nodes. In the following subsections these three schemes are presented formally, with necessary comments. It is assumed that all the nodes broadcast their velocity and positional information in every beacon interval.

Few notations used in the algorithms are:

- $N_f$ : front node
- $N_b$ : rear node
- $N_{f-i}$ :  $i$ -th behind node with respect to the front node
- $N_{b+i}$ :  $i$ -th front node with respect to the rear node
- $N_i$ :  $i$ -th Node

$I(N_{f-i})$ : position and velocity information of the  $i$ -th behind node with respect to front node.

$I(N_{b+i})$ : position and velocity information of the  $i$ -th front node with respect to the rear node.

$I(N_{i+1})$ : position and velocity information of the  $(i+1)$ -th node.

$I(N_{i-1})$ : position and velocity information of the  $(i-1)$ -th node.

$V_c$ : Critical Velocity

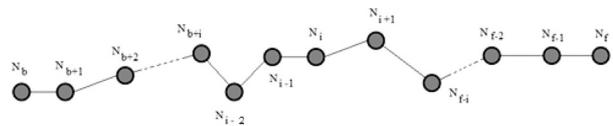


Figure 7. Network representation.

### 5.2.1. Movement algorithm for front node

Begin:

- Step1:** Broadcast the position and velocity information obtained through GPS.
- Step2:** Receive the position and velocity information from the nodes which are within the communication range of the front node.
- Step3:** Check whether  $I(N_{f-1})$  is received
  - if yes GOTO step7     $\backslash\backslash$ Normal movement $\backslash\backslash$
  - else                    $\backslash\backslash$ Fault occurs at  $N_{f-1}\backslash\backslash$

**Step4:** Check whether  $I(N_{f-2})$  is received

- if yes     $\backslash\backslash$ Fault occurs at  $N_{f-1}$  but  $N_{f-2}$  is within the communication range $\backslash\backslash$
- Update routing table and GOTO step7     $\backslash\backslash$ Normal movement $\backslash\backslash$
- else
- Step5:** stop for one beacon interval     $\backslash\backslash$ Fault occurs at  $N_{f-1}$  and  $N_{f-2}$
- is out of the communication
- range i.e. critical situation $\backslash\backslash$
- Step6:** Update routing table and GOTO step 1
- end

end                    $\backslash\backslash$ Normal movement $\backslash\backslash$

- Step7:** Crisp value of distance between  $N_f$  and  $N_{f-1}$  and velocity of  $N_{f-1}$  are calculated and functions for fuzzification are called to obtain fuzzy variables.
- Step8:** Fuzzy Rule Base-1 is called with fuzzy distance and velocity
- Step9:** Defuzification method is called to obtain crisp velocity of the  $N_f$
- Step10:** Allow the front node to move for one beacon interval

**Step11:** GOTO step 1

End

### 5.2.2. Movement algorithm for rear node

Begin:

**Step1:** Broadcast the position and velocity information obtained through GPS.  
**Step2:** Receive the position and velocity information from the nodes which are within the communication range of the rear node.  
**Step3:** Check whether  $I(N_{b+1})$  is received  
 if yes  
     GOTO step7      \\Normal movement\\  
 else  
     \\**Fault occurs at  $N_{b+1}$** \\

**Step4:** Check whether  $I(N_{b+2})$  is received  
 if yes      \\Fault occurs at  $N_{b+1}$  but  $N_{b+2}$  is within the communication range\\  
     Update routing table and GOTO step7  
 else  
     **Step5:** rush with  $V_c$  for one beacon interval  
         \\Fault occurs at  $N_{b+1}$  and  $N_{b+2}$  is out of the communication  
         range i.e. critical situation\\  
     **Step6:** Update routing table and GOTO step 1  
 end

end

\\**Normal movement**\\

**Step7:** Crisp value of distance between  $N_b$  and  $N_{b+1}$  and velocity of  $N_{b+1}$  are calculated and the functions for fuzzification are called to obtain fuzzy variables.  
**Step8:** Fuzzy Rule Base-2 is called with fuzzy distance and velocity  
**Step9:** Defuzzification method is called to obtain crisp velocity of the  $N_b$   
**Step10:** Allow the rear node to move for one beacon interval

**Step11:** GOTO step 1

End

### 5.2.3. Movement algorithm for the nodes other than front and rear nodes

Begin:

- Step1:** Broadcast the position and velocity information obtained through GPS.
- Step2:** Receive the position and velocity information from the nodes which are within the communication range of the node in consideration.
- Step3:** Check whether  $I(N_{i+1})$  and  $I(N_{i-1})$  are received  
 If both are yes  
     GOTO step13      \\Normal movement\\  
 else      \\Fault Detected\\  
 if  $I(N_{i-1})$  is not received but  $I(N_{i+1})$  is received  
     \\After fault detected at behind node\\

```

Step4: Check whether  $N_{i-1}$  is  $N_b$ 
if yes, follow the algorithm for rear node      \\fault occurs at rear node\\
else
    Step5: Check whether  $I(N_{i-2})$  is received
if yes
    update routing table and GOTO step13      \\Normal movement\\
else      \\Fault occurs at  $N_{i-1}$  and  $N_{i-2}$  is out of the
           communication range i.e. the critical situation\\
    Step6: Stop for one beacon interval
end
end
    
```

end

- if  $I(N_{i+1})$  is not received  
     \\After fault detected at front node\\

```

Step7: Check whether  $N_{i+1}$  is  $N_f$ 
if yes, follow the algorithm for front node      \\fault occurs at front node\\
else
    Step8: Check whether  $I(N_{i+2})$  is received
if  $I(N_{i+2})$  is received      \\Fault occurs at  $N_{i+1}$  but  $N_{i+2}$  is within
           the communication range\\
    Update routing table and GOTO step13      \\Normal movement\\
else
    Step9: rush with  $V_c$  for one beacon interval
           \\Fault occurs at  $N_{i+1}$  and  $N_{i+2}$  is out of the
           communication range i.e. critical situation\\
end
end
    
```

End

- Step10:** Update routing table
- Step11:** GOTO step1
- End

\\Normal movement\\

```

Step12: Crisp value of distance between  $N_i$  and  $N_{i-1}$ , distance between  $N_i$  and  $N_{i+1}$ , velocity of  $N_{i+1}$ 
are calculated and function for fuzzification is called to obtain fuzzy variable.
Step13: Fuzzy Rule Base-3 is called with fuzzy distances and velocity
Step14: Defuzification method is called to obtain crisp velocity of the  $N_i$ 
Step15: Allow the node to move for one beacon interval
    
```

- Step16:** GOTO step 1

End

### 6. Simulation Result

To evaluate the effectiveness of our proposed algorithm in maintaining the connectivity of a MANET, number of simulations using MATLAB in Windows environment have been performed. The simulator was designed for any number of nodes and the simulation of the algorithm for different number of nodes and simulation time ranging from 1 hr to 20 hr has been performed. For simulation of these three sample networks the following system parameters: maximum communication range ' $R_{max}$ ' = 15 Km, maximum allowable preferred velocity ' $V_{max}$ ' = 60 Km/hr, threshold distance ' $R_{th}$ ' = 10 Km are considered. So, the beacon interval is  $(15 - 10)/60$  i.e. 5 minutes.

#### 6.1. Sample Network I

The network is of ten nodes and their initial coordinates are  $(-3, 1)$ ,  $(0, 0)$ ,  $(10, 0)$ ,  $(15, 0)$ ,  $(23, 0)$ ,  $(25, 2)$ ,  $(27, -1)$ ,  $(31, 3)$ ,  $(33, -2)$  and  $(35, 0)$  respectively. Initial network is shown in Figure 8. For this network, simulation is carried out for 300 minutes. The distances between each pair of nodes are plotted in Figures 9 and 10 during the movement of the network.

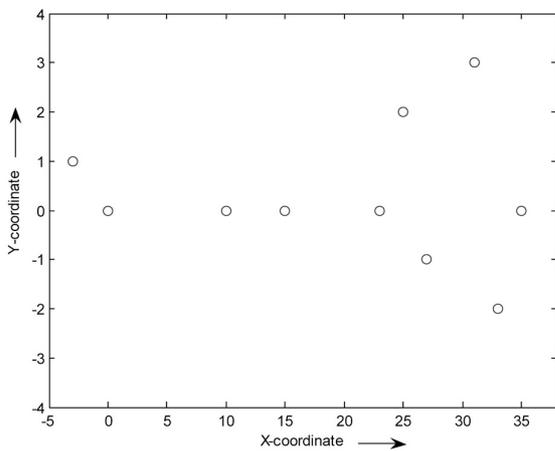


Figure 8. Showing initial network for sample I.

Distance between each pair of nodes is plotted in Figure 9 before the occurring fault and in Figure 10 after the fault. From the above plot it is observed that the fault occurred at node no. 8 at 5<sup>th</sup> beacon interval. But the connectivity is always maintained.

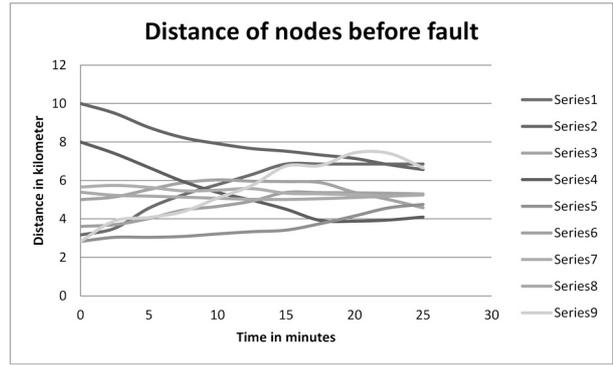


Figure 9. Showing the distance between each pair of nodes for sample network I before fault to node 8.

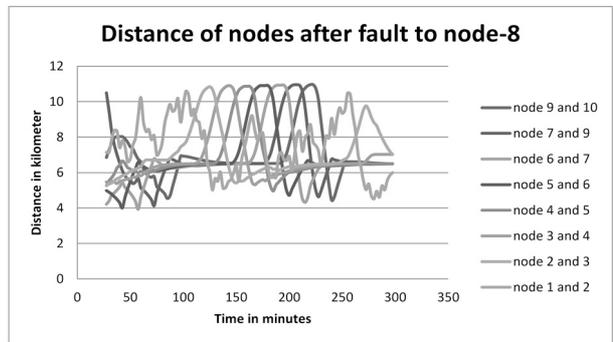


Figure 10. Showing the distance between each pair of nodes for sample network I after fault to node 8.

#### 6.2. Sample Network II

The network is of twelve nodes and their initial coordinates are  $(-1, 0)$ ,  $(4, 0)$ ,  $(10, 0)$ ,  $(17, 0)$ ,  $(19, 1)$ ,  $(25, 2)$ ,  $(30, 0)$ ,  $(34, 2)$ ,  $(36, 3)$ ,  $(39, 3)$ ,

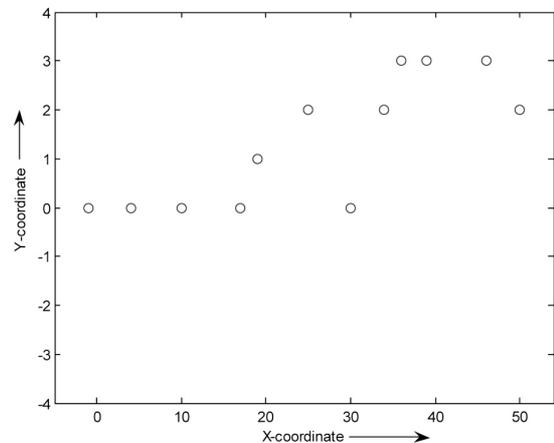


Figure 11. Showing the initial network for sample network II.

(46, 3), and (50, 2) respectively. Initial network is shown in Figure 11. In this network, two more nodes have been included and the simulation is again carried out for 300 minutes. During movement of the networks, separation between each pair of nodes is plotted in Figure 12.

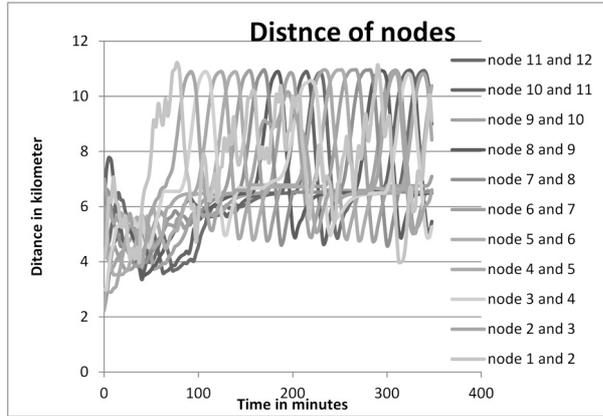


Figure 12. Showing the distance between each pair of nodes for sample network II.

From the above graph it is clear that connectivity of each pair of nodes is maintained for the entire time of the movement of the networks. In this simulation no fault occurred to a node.

### 6.3. Sample Network III

The network is of fifteen nodes and their initial coordinates are (0, 0), (10, 0), (20, 0), (30, 0) and (32, 1), (35, 2), (37, 0), (39, 1), (42, 2), (45, 2), (47, 3), (49, 2), (53, 1), (55, 0), (60, 1)

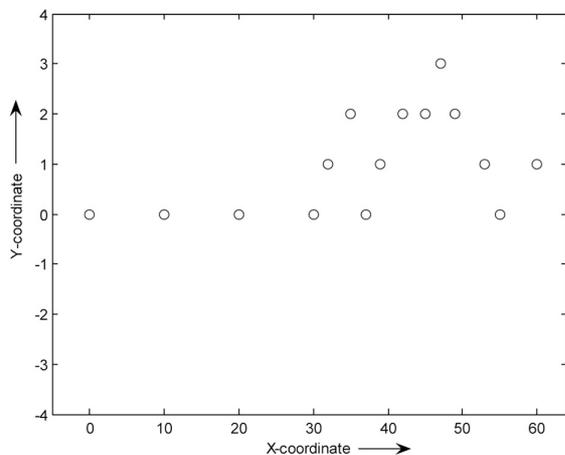


Figure 13. Showing the initial network for sample network III.

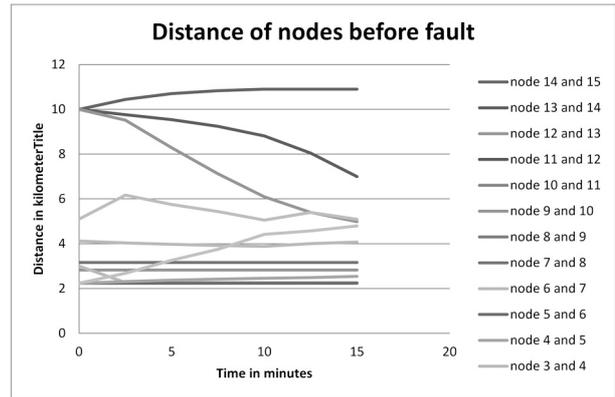


Figure 14. Showing the distance between each pair of nodes for sample network III before fault to node 14.

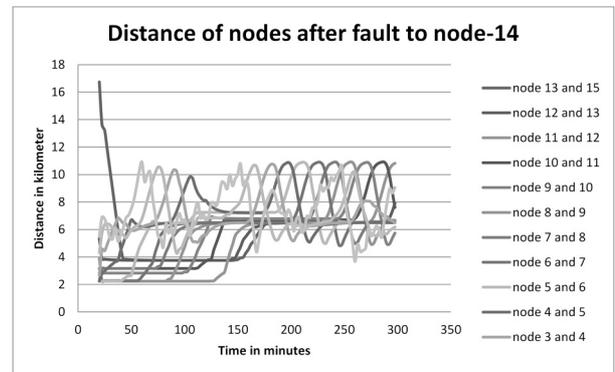


Figure 15. Showing the distance between each pair of nodes for sample network III after fault to node 14.

respectively, as shown in Figure 13. The distances between each pair of nodes are plotted in Figures 14 and 15 during the movement of the network.

Again three more nodes have been incorporated in this networks and the simulation is carried out for 350 minutes for this network. During simulation run it is seen that at 3<sup>rd</sup> beacon interval node 14 becomes faulty. At that beacon interval distance between nodes 13 and 15 is more than 15 kilometers. So, at this beacon interval connectivity is broken between nodes 13 and 15. But the connectivity is restored in the next beacon interval and this connectivity is maintained for the rest of the entire movement of the network. The distances between each pair of nodes, before fault to node14 and the distances of nodes after occurring fault are plotted in Figures 14 and 15 respectively.

## 7. Comparative Study

The performance of the proposed algorithm is compared with the algorithm proposed by S. S. Basu and A. Chaudhuri in [3] and non fuzzy algorithm of the authors [6]. In [3] an algorithm to elect coordinator and a separate algorithm for node movement have been proposed. This algorithm is the centralized one. The algorithm proposed in [6] is the distributed one, but the movement control algorithm is not fuzzy-based. To evaluate the performance of this fuzzy logic based distributed algorithm to maintain the connectivity of the network, a network of five nodes and their initial coordinates (1.5, 3), (0.5, 2), (2, 3), (3, 1), and (4, 1) have been considered. Velocity changes of the different nodes are shown graphically for the algorithm proposed in [3], [6] and this paper in Figure 16a and Figure 16b and Figure 16c respectively.

From the results presented in 16a, 16b and 16c, it is seen that the proposed algorithm is very effective in maintaining the connectivity, compared to others. This algorithm has the additional feature of capability of restoring connectivity if it is hampered due to a faulty node. This feature is not available in other algorithms. Stability of the nodes is hampered by the sudden change of velocity. But from the above comparison it is clear that the algorithm proposed in this paper is giving better stability, compared to others.

## 8. Conclusion

From the simulations it is clear that the network connectivity is maintained between two successive beacon intervals most of the time, hence the routing overhead is also reduced to a large extent. Due to the distributed nature of the scheme, control overhead is also distributed equally among all the nodes of the network. Nodes act as intelligent agents as fuzzy-based movement control scheme is introduced. Moreover, the algorithm has the capability to restore connectivity if it is broken due to a faulty node. In the future, hardware implementation with optimized power consumption of the algorithm may be proposed.

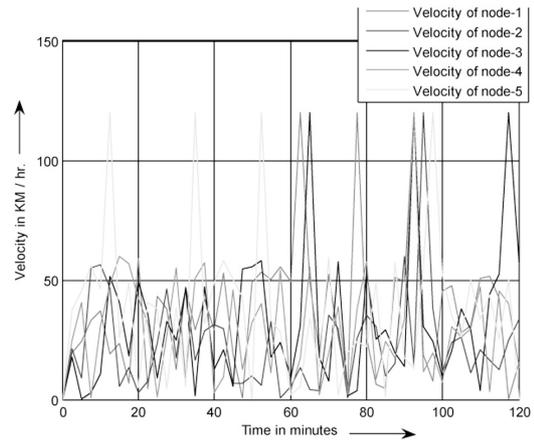


Figure 16a. Velocity changes of the nodes for the algorithm in [3].

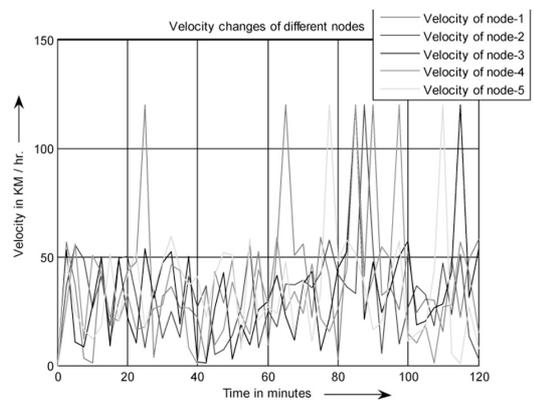


Figure 16b. Velocity changes of the nodes for the algorithm in [6].

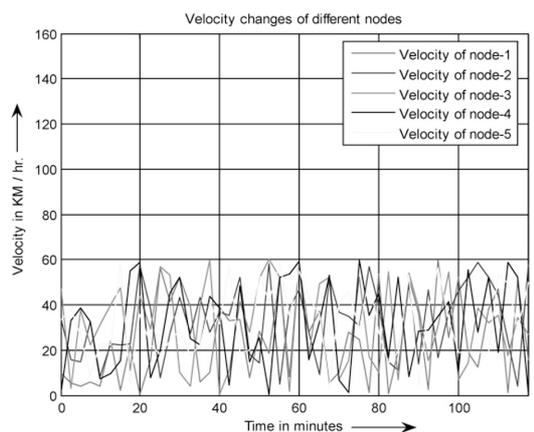


Figure 16c. Velocity changes of the nodes for proposed algorithm.

## References

- [1] S. S. BASU AND A. CHAUDHARI, Self-adaptive Topology Management for Mobile Ad-Hoc Network, *IE (I) Journal-ET*, Vol. 84, pp. 7–12, July 2003.
- [2] DAVID B. JOHNSON, DAVID A. MALTZ AND JOSH BROCH, *DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad-Hoc Networks*, Ad-Hoc Networking, edited by Charles E. Perkins, Chapter 5, pp. 139–172, Addison-Wesley, 2001.
- [3] SOUMYA SANKAR BASU, ATAL CHAUDHARI, Self-Adaptive MANET: A Centralized Approach, *Foundations of Computing and Decision Sciences*, Vol. 29, pp. 271–286 2004.
- [4] A. NASIPURI, R. CASTANEDA, AND S. R. DAS, Performance of Multipath Routing for On-demand Protocols in Mobile Ad-hoc Networks, *Mobile Network Applications*, Vol. 6, No. 4, pp. 339–349, 2001.
- [5] S. SAMANTA, S. S. RAY, S. SENGUPTA, M. K. NASKAR, A Novel Algorithm for Managing Network Configuration, in *Proc. Asian International Mobile Computing Conference (AMOC2007)*, pp. 51–58, January 2007, Kolkata.
- [6] JISHAN MEHEDI, SURENDRA S. DALU, AND M. K. NASKAR, A Distributed Approach to Maintain Connectivity of Nodes in Mobile Ad-Hoc Networks, in *Proc. International Conference on Intelligent Systems & Networks (IISN2007)*, pp. 353–358, February 2007, Haryana.
- [7] AVIK RAY, KANAD BASU, SAMIR BISWAS, MRINAL K. NASKAR, A Novel Distributed Algorithm for Topology Management in Mobile Ad-hoc Networks, in *Proc. International Conference on Computers and Devices for Communication 2006 (CODEC06)*, pp. 99–102, December 2006, Kolkata, India.
- [8] ELIZABETH M. ROYER AND CHAI -KEONG TOH, A Review of Current Routing Protocols for Ad-hoc Mobile Wireless Networks, *IEEE Personal Communications*, Vol. 6, No. 2, pp. 46–55, April 1999.
- [9] JISHAN MEHEDI, SURENDRA S. DALU, AND M. K. NASKAR, Performance Comparison of Some Routing Protocols for Mobile Ad-Hoc Networks, in *Proc. National Conference (INDIACom2007)*, pp. 205–208, 212, February 2007, Delhi.
- [10] C. E. PERKINS AND P. BHAGWAT, Highly Dynamic Destination Sequenced Distance Vector Routing (DSDV) for Mobile Computers, *ACM SIGCOMM*, Vol. 24, No. 4, pp. 234–244, October 1994.
- [11] 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 *Proc. Mobile Computing and Networking*, pp. 85–97, 1998.
- [12] T. KORAKIS, G. JAKLLARI, AND L. TASSIULAS, A MAC Protocol for Full Exploitation of Directional Antennas in Ad-hoc Wireless Networks, in *Proc. 4th ACM International Symposium on Mobile Ad-hoc Networking & Computing*, ACM Press, pp. 98–107, 2003.
- [13] R. WATTENHOFER, L. LI, P. BAHL AND Y. M. WANG, Distributed Topology Control for Power Efficient Operation in Multi-hop Wireless Ad-hoc Networks, in *Proc. IEEE INFOCOM*, April 2001.
- [14] R. RAMNATHAN AND R. ROSALES-HAIN, Topology Control of Multi-hop Radio Networks Using Transmit Power Adjustment, in *Proc. IEEE INFOCOM*, pp. 403–404, March 2002.
- [15] C. BETTSTETTER, On the Minimum Node Degree and Connectivity of a Wireless Multi-hop Network, in *Proc. ACM MOBIHOC*, pp. 80–91, June 2002.
- [16] F. BAI, N. SADAGOPAN, A. HELMY, Important: a Framework to Systematically Analyze the Impact of Mobility on Performance of Routing Protocols for Ad-hoc Networks, *INFOCOM 03*, Vol. 2, pp. 825–835, San Francisco, California, USA, March 2003.
- [17] 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-Hoc Networking: Research, Trends and Applications*, Vol. 2, No. 5, pp. 483–502, September 2002.
- [18] J. BOUDEC, J.-Y. L. BOUDEC AND M. VOJNOVIĆ, Perfect Simulation and Stationarity of a Class of Mobility Models, in *Proc. IEEE Information Communications Conference (INFOCOM05)*, pp. 72–79, 2005.
- [19] JINSHAN LIU, FRANCOISE SAILHAN, Group Management for Mobile Ad-Hoc Networks: Design, Implementation and Experiment, in *Proc. MDM 05*, pp. 192–199, Ayia Napa, Cyprus, 2005.
- [20] KIHWAN KIM, YING CAI AND WALLAPAK TAVANAPONG, Sharing Location Dependent Experiences in MANET, in *Proc. 7th International Conference on Mobile Data Management (MDM06)*, pp. 69.
- [21] E. NATSHEH, A. B. JANTAN, S. KHATUN AND S. SUBRAMANIAM, Fuzzy Reasoning Approach for Local Connectivity Management in Mobile Ad-hoc Networks, *International Journal of Business Data Communications and Networking*, Vol. 2, Issue 3, pp. 1–18, 2006.
- [22] FAN BAI, NARAYANAN SADAGOPAN, AHMED HELMY, The Important Framework for Analyzing the Impact of Mobility on Performance of Routing Protocols for Ad-hoc Networks, *Ad-Hoc Networks Journal – Elsevier Science*, Vol. 1, Issue No. 4, pp. 383–403, November 2003.
- [23] Random Waypoint Model, <http://www.netlab.tkk.fi/~esa/java/rwp/rwpmodel.html>.

- [24] 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*, Vol. 4, No. 2, pp. 79–84, 2009.
- [25] SCOTTIE BARNES AND LAFE LOW, Basic Essentials: Global Positioning System, *Globe Pequot. Pv.*, June 2000.
- [26] J. BIH, Paradigm Shift- An Introduction to Fuzzy Logic, *IEEE Potentials*, 25(1), pp. 6–21, 2006.
- [27] ELLIOT D. KAPLAN, *Understanding GPS: Principles & Applications*, Arleets House, March 1996.
- [28] AMIT KONAR, *Computational Intelligence: Principles, Techniques and Applications*, Springer, 2005.

Received: October, 2009  
Revised: June, 2012  
Accepted: June, 2012

*Contact addresses:*

Jishan Mehedi  
Department of Electronics  
and Communication Engineering  
National Institute of Technology  
Silchar, Assam-788010  
e-mail: jmehedi2007@yahoo.co.in

M. K. Naskar  
Advanced Digital and Embedded Systems Lab  
Department of Electronics  
and Telecommunication Engineering  
Jadavpur University  
Kolkata-700032  
e-mail: mrinalnaskar@yahoo.co.in

---

DR. JISHAN MEHEDI received the B. Tech degree in Electronics and Communication Engineering from the University of Kalyani, M. E. Tel. E and PhD degree in Electronics and Telecommunication Engineering from Jadavpur University. He worked as faculty member in Dumkal Institute of Engineering and Technology from 2002 to 2008. Currently, he is working as Assistant Professor in ECE Department at NIT, Silchar, Assam-788010, India. His research interests include mobile ad-hoc networks, wireless sensor networks and digital system design.

---

---

PROF. M. K. NASKAR received both the B. Tech and M. Tech degrees from E & ECE Department, IIT, Kharagpur and the PhD degree from Jadavpur University. He served as a faculty member in RIT, Jamshedpur and REC, Durgapur from 1991–1996 and 1996–1999 respectively. Prof. M. K. Naskar is currently working as Professor in the Department of Electronics and Telecommunication Engineering at Jadavpur University, Kolkata, India and is in-charge of the Advanced Digital and Embedded Systems Lab. His research interests include mobile ad-hoc networks, wireless sensor networks, optical networks and embedded systems.

---