Mathware & Soft Computing 16 (2006), 23-36

A Model of Route Lifetime Optimization with Linguistic Knowledge in Wireless Ad-hoc Networks

E. Natsheh, S. Khatun and A. Jantan
Dept of Computer & Communication Systems Engineering,
Univ. Putra Malaysia, P.O. Box 341, 43400 UPM, Serdang, Malaysia
dr_natsheh@hotmail.com, *sabira@eng.upm.edu.my*,
adznan@eng.upm.edu.my

Abstract

Ad-hoc On-Demand Distance Vector (AODV) routing protocol has been and continues to be a very active and fruitful research protocol since its introduction in the wireless ad-hoc networks. AODV uses a static value for its route lifetime parameter called Active Route Timeout (ART) which indicates the time that the route can stay active in the routing table. Route lifetime may be more accurately determined dynamically via measurement, instead of static value. To accomplish this, the fuzzy logic system is used to obtain adaptive values for ART depending on the situation of the transmitter and intermediate nodes. To study the effect of various parameters on ART, three design methods for fuzzy ART are proposed here, namely: fuzzy-SKP, fuzzy-Power, and fuzzy-Comb. Analysis shows that the proposed design methods are quite efficient and superior to the conventional design method with respect to routing overhead (25%) and average end-to-end delay (41%). Hence, the proposed fuzzy system is able to optimize ART efficiently.

Key words: Ad-hoc networks, AODV, adaptive route timeout, fuzzy route lifetime.

I. Introduction

Mobile multi-hop wireless networks, called Ad-hoc networks, are networks with no infrastructure such as access points or base stations. A node communicates directly with the other nodes within adequate radio propagation and indirectly through multi-hope routing with all others. To allow such on-the-fly formation of networks, numerous routing protocols have been developed.

The route lifetime value is one of the most important parameters for the design of an on-demand ad-hoc routing protocol. This parameter determines the duration of an active path/route in the routing table to transmit the packets reliably. This is to ensure that the routing table does not attempt to discover a new route and/or delete an existing active route within its lifetime. So, too long route lifetime may lead to retardation in updating the routing table even though some paths are broken. This results large routing delay and control overhead from attempts to transmit across paths that do not exist. On the other hand: too short route lifetime may remove some active paths from the routing table. This leads the routing protocol running the discovery process for those paths again, resulting large routing delay and traffic overhead due to the new path search. In essence, this means that, the protocol designer has to choose the value of route lifetime carefully to represent the real availability of source-destination paths.

Ad-hoc On-Demand Distance Vector (AODV) routing protocol has been designed to be used in ad-hoc mobile networks [1–3]. It allows users to find and maintain routes for other users in the network, whenever needed (on-demand). Since the production of this protocol by Perkins [3], static route lifetime values have been used, called Active Route Timeout (ART) which state the time that the route stays active in the routing table. However, the unpredictability and the randomness of the node movement make the adaptive determination of route lifetime value better than a static approach. Due to the complexity of this determination, very few network researchers attempted to use adaptive route lifetime values. Advanced mathematical tools are used to predict the adaptive route lifetime, which are very complicated and difficult to understand. These mathematical models results nonlinearity and some degree of errors for estimate nodes mobility.

In this study, adaptive route lifetime determination through a fuzzy logic system is proposed. Fuzzy logic is chosen due to the uncertainty associated with node mobility estimation and drawbacks of mathematical models. Definition of fuzzy sets (membership functions) and a set of rules (rule-base) have been proposed to design the new method, called *fuzzy ART*. This new method is evaluated with the AODV routing protocol, we believe it can be generalized for other ad-hoc routing protocols as well.

The rest of this paper is organized as follows. Section II summarizes related work on optimum route lifetime and using fuzzy logic in routing protocols. Followed by the implementation of AODV using the fuzzy ART method, performance analyses of the proposed method, and finally the conclusion.

II. Related work

In this section, we summarize literature on optimum route lifetime. Existing surveys on using fuzzy logic in routing protocols are also discussed here.

A. Route Lifetime Optimization

In designing on-demand ad-hoc routing protocols, four values are used for route lifetime. These are:

1. Route lifetime is equal to 0. This means the route is founded when a packet is ready to be transmitted, and kept active during transmission, and deleted at the end of transmission. An example of such a protocol is Associatively Based Routing (ABR) [4]. ABR measures the lifetime of a link using hello messages which are periodically broadcast.
2. Route lifetime is equal to infinity. This means that from the time the route is discovered, it is kept active until the broken link is discovered. Examples of such

protocols are Dynamic Source Routing (DSR) [5] and Temporally Ordered Routing Algorithm (TORA) [6].

3. Route lifetime is equal to a predetermined static value. This means that from the time the route is discovered, it is kept active up to predetermined amount of time. An example of such a protocols is AODV [3]. In this protocol, ART is set to 3 milliseconds.
4. Route lifetime is equal to an adaptive value. This category is subdivided to two subcategories:
   a. Restricted adaptive lifetime: Paul *et al.* [7] introduce a parameter – *affinity* – which characterizes the strength and stability of a relationship between two nodes. The path with minimum affinity will be used to transmit data between those two nodes. This path will be saved in the routing table as long as the affinity is greater than a certain threshold.
   b. Un-restricted adaptive lifetime: The route lifetime is adaptively calculated according to network situation and kept active as long as the route not breaks. Examples of such protocols are those proposed by Ben *et al.* [8], Agarwal *et al.* [9] and Tseng *et al.* [10].

   Protocols that used the adaptive route lifetime method found interesting results in minimizing routing delay and traffic overhead. Researchers who designed these protocols used advanced mathematical tools to determine the values of adaptive route lifetime. In this paper, we attempt to simplify these protocols by using the fuzzy logic system.

B. Using Fuzzy Logic in Routing Protocols

Ghosh *et al.* [11] presented a survey on the use of fuzzy logic in telecommunication networks. Sekercioglu *et al.* [12] and Bonde *et al.* [13] reported a similar survey on the use of fuzzy logic in ATM networks.

Using fuzzy numbers to represent uncertainty in the delay values, Pithani *et al.* [14] have developed fuzzy comparison criteria using this uncertainty in making routing path decisions. Aboelela *et al.* [15] define a fuzzy cost to reflect the crisp values of the different metrics that possibly can be used in the network links. The fuzzy system is then integrated into a complete routing system. Pasupuleti *et al.* [16] propose an adaptive routing algorithm in which the link cost is dynamically assigned using a fuzzy system. The traffic in the network is re-routed to nodes which are less congested, or have spare capacity.

A few studies have also been undertaken using fuzzy logic in ad-hoc routing protocols. Wong *et al.* [17] presented a fuzzy-decision-based protocol, developed on DSR protocol with the support of QoS parameters.

III. AODV with fuzzy art

In this section, the concept and rules for fuzzy ART that will be used with AODV are introduced and the method to design its membership functions is presented.

A. Effect of path length on ART

In mobile ad-hoc networks, node mobility causes paths between nodes to break frequently. Although using more hops may reduce the distance between paths, the increasing number of hops also introduces greater risk of route breakage. When the number of hops between the source and destination (*HopCount*) is high, the probability that the path will break because of node movement is also high. The probability of a path break p_b can be calculated as [18]:

$$p_b = 1 - (1 - p_l)^k \tag{1}$$

where p_l is the probability of a link break and k is a path length. Figure 1 shows p_b versus *HopCount* when p_l is equal to 0.1, 0.3 and 0.5. It is clear that the probability of a path break increases as the path length increases, terminating the lifetime of the routes containing those paths (the ART time). Based on previous studies, we can state that when *HopCount* is high, the route lifetime must be low and vice versa. Consequently the following rules are proposed:
R1: If *HopCount* is high then ART must be low
R2: If *HopCount* is medium then ART must be medium
R3: If *HopCount* is low then ART must be high

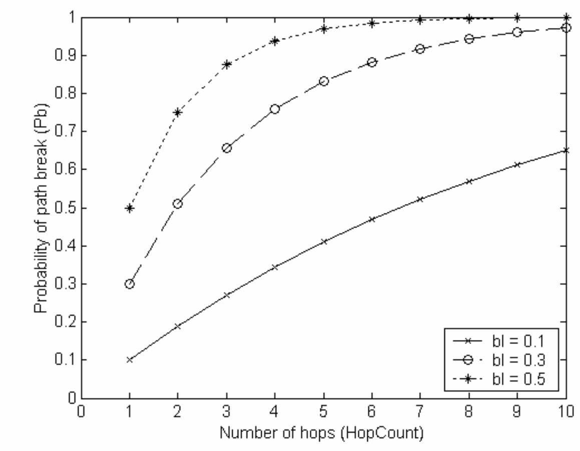


Fig. 1. Probability of path breaks versus *HopCount*

B. Effect of node mobility on ART

Ad-hoc networks experience dynamic changes in network topology because of the unrestricted mobility of the nodes in the network. If the end nodes (source and destination) move frequently, then it is highly probable that their path will break. The node movement can be measured by the number of sent control packets (*SentCtrlPkt*) between two sampling intervals. *SentCtrlPkt* is any message of the following type: RREQ, RREP, RERR and RREP_ACK. The description of these messages is shown in Table I. A high number of *SentCtrlPkt* transmissions occur either due to the movement of the intermediate nodes in the path or to the movement of end nodes results high probability to loose some of the current links in the path and creating new ones. In general, a rule can be defined: when *SentCtrlPkt* is high, the route lifetime must be low and vice versa. Consequently the following rules are proposed:
R4: If *SentCtrlPkt* is high then ART must be low
R5: If *SentCtrlPkt* is medium then ART must be medium
R6: If *SentCtrlPkt* is low then ART must be high

TABLE 1
MESSAGES USED BY AODV

Message	*Description*
RREQ	a Route Request message
RREP	a Route Reply message
RERR	a Route Error containing a list of the invalid destinations
RREP_ACK	a *RREP* acknowledgment message

C. Effect of node transmission power on ART

The routes lifetime used by nodes of ad-hoc network is highly sensitive to the transmission power of those nodes. Transmission power (*TransPower*) is the strength with which the signal is transmitted.

In our system, signal power degradation is modeled by the *free space propagation model* [19] which states that the received signal strength is:

$$P_r(d) = \frac{P_t G_t G_r l^2}{(4p)^2 d^2 L} \tag{2}$$

where P_r and P_t are the receive and transmit powers (in Watts), G_t and G_r are the transmit and receive antenna gains, *d* is the transmitter-receiver separation distance, *L* is a system loss factor (*L* = 1 in our simulations which indicates no loss in the system hardware), and *?* is the carrier wavelength (in meters) which related to the carrier frequency by:

$$l = \frac{c}{f_c} \tag{3}$$

where f_c is the carrier frequency (in Hertz) and *c* is the speed of light (3×10^8 m/s). Assuming a unity gain antenna with a 900 MHz carrier frequency, Figure 2 shows the relation between the transmission range and the transmission power of a node for different values of the receiver power.

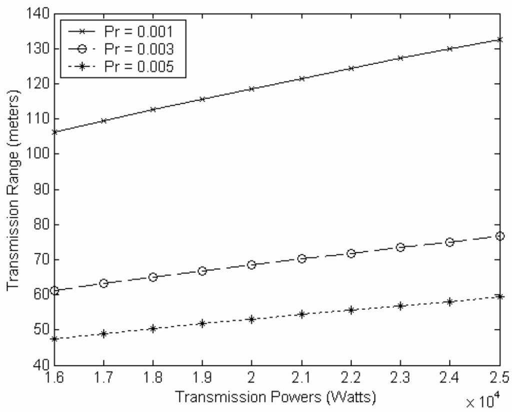


Fig. 2. Transmission range versus transmission power

Increased transmission power means larger transmission range. If the transmission power of a node is too low, then its signal will reach to few neighbors only and its links with those neighbors may have very weak and easy to break. High transmission power of a node will lead to high average number of its neighbors and hence increase the lifetime of its routes. Consequently the following rules are proposed:
R7: If *TransPower* is high then ART must be high
R8: If *TransPower* is medium then ART must be medium
R9: If *TransPower* is low then ART must be low

D. The rule- base for fuzzy ART

To compare between different parameters that effect on ART, we have proposed three methods to design the fuzzy ART:

1. *Fuzzy-SKP*: in this method the effect of path length and node mobility are considered. To implement this method, the first six previous rules (R1 to R6) can be combined with one 2-dimensional rule-base for controlling the ART adaptively as presented in Table II.
2. *Fuzzy-Power*: in this method the effect of path length (rules R1 to R3) and transmission power (rules R7 to R9) are combined to design a rule-base shown in Table III.
3. *Fuzzy-Comb*: in this method, previous two methods are combined. So, ART is calculated by tacking the average of ARTs produced by fuzzy-SKP and fuzzy-Power methods.

TABLE 2
RULE-BASE FOR FUZZY-SKP

HopCount	*SentCtrlPkt*		
	Low	Medium	High
Low	High	High	Medium
Medium	High	Medium	Low
High	Medium	Low	Low

TABLE 3
RULE-BASE FOR FUZZY-POWER

HopCount	*TransPower*		
	Low	Medium	High
Low	Medium	High	High
Medium	Low	Medium	High
High	Low	Low	Medium

E. Membership functions for the fuzzy variables

After having defined the fuzzy linguistic ‘if-then’ rules the membership functions corresponding to each element in the linguistic set (*HopCount*, *SentCtrlPkt*, *TransPower*, and *ART*) must be defined. For example, if the *HopCount* equal to 4, conventionally, we may say that the *HopCount* is either ‘low’ or ‘medium’ but not both. In fuzzy logic, however, the concept of membership functions allows us to say the *HopCount* is ‘low’ with 20% membership degree and it is ‘medium’ with 80% membership degree.

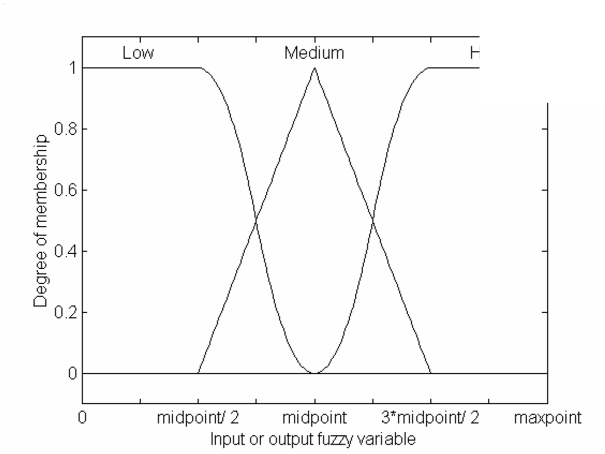


Fig. 3. Membership functions used in fuzzy AODV

We propose to use the membership functions shown in Figure 3 because the parametric, functional descriptions of these membership functions are most economic. In these membership functions, the designer needs only to define two parameters; *midpoint* and *maxpoint*. These membership functions contain mainly the *triangular* shaped membership function. It has been proven that *triangular* membership functions can approximate any other membership function [20]. This function is specified by three parameters (a, b, c) as follows:

$$triangle(x;a,b,c)=\begin{cases}(x-a)/(b-a) & for\ a\le x\le b\\ (c-x)/(c-b) & for\ b\le x\le c\\ 0 & \text{elsewhere}\end{cases} \tag{4}$$

where a = *midpoint*/2, b = *midpoint*, c = 3 × *midpoint*/2 and x is the input to the fuzzy system. The remaining membership functions are as follows: Z-shaped membership to represent the whole set of low values and S-shaped membership to represent the whole set of high values.

Midpoint is the value of the fuzzy variable, which can be chosen from the real network, simulation and analysis or from the default values of protocol specification as follows.

Tseng *et al.* [10] compared route breakage probability distribution obtained from random simulation and analysis on route length equal to 3 links, 6 links, 9 links, and 12 links. The results showed that the practical sizes of ad-hoc networks would range around 5 nodes. Hence, for *HopCount* membership function, *midpoint* should be equivalent to 5 nodes.

The value of *SentCtrlPkt* depends on the number of nodes in the network. So, the *midpoint* can be calculated as:

midpoint = number of nodes × 10.

This value has been observed during a run of ad-hoc network simulator (described in section IV) with different sizes of the network.

Midpoint of transmission power membership function can be the average transmission power of the mobile nodes in the network. For example, if the transmission powers of the nodes are in between 18 kW and 24 kW then *midpoint* is 21 kW.

AODV protocol specification [3] stats that the static value of *ART* is 3 milliseconds. Hence, for the *ART* membership function, *midpoint* should be equivalent to 3 milliseconds.

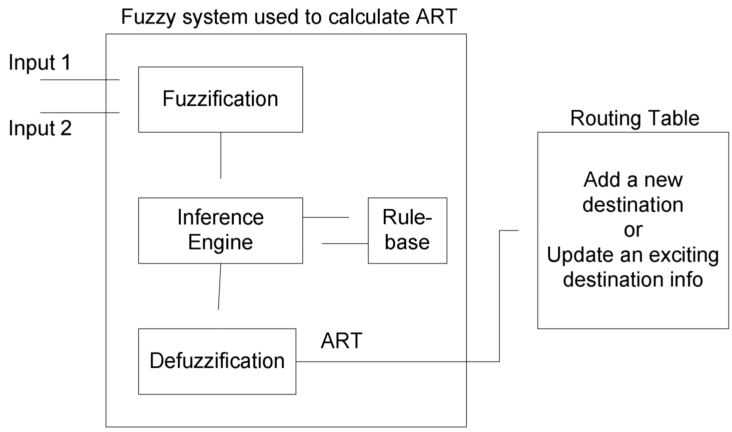
Since the values of input variables (*HopCount*, *SentCtrlPkt*, and *TransPower*) occur during the simulation run, exact knowledge of their values cannot be determined. The range of values (*maxpoint*) for these variables must be quite large. Hence, *Maxpoint* can be defined as follows:

For input variables: *maxpoint* = 3 × *midpoint*.
For output variable: *maxpoint* = 2 × *midpoint*.

F. Fuzzification, Inference and Defuzzification

The elementary basic diagram of the fuzzy system is presented in Figure 4. Fuzzification is a process where crisp input values are transformed into membership values of the fuzzy sets (as described in section E). After the process of fuzzification, the inference engine calculates the fuzzy output using fuzzy rules described in Table II (fuzzy-SKP method) or Table III (fuzzy-Power method). Defuzzification is a mathematical process used to convert the fuzzy output to a crisp value. This crisp output is the ART value.

The fuzzy logic system has been simulated using C++ programming language. There are a variety of choices in the fuzzy inference engine and the defuzzification method. Based on these choices, a number of different fuzzy systems can be constructed. In this study, we choose the most commonly used fuzzy system [21].



$$t_i = A_{i1}(x_1) \wedge A_{i2}(x_2) \tag{6}$$

where t_i is called the *degree of firing* of the i^{th} rule with respect to the input values *HopCount* = x_1 and *SentCtrlPkt* = x_2. The next step is the determination of the individual rule output F_i (fuzzy set) which obtained by:

$$F_i(y) = t_i \wedge B_i(y) \tag{7}$$

The third step is the aggregation of the rules outputs to obtain the overall system output F (fuzzy set), where Max ($\vee$) operator was chosen as OR connective between the individual rules:

$$F(y) = \vee_i F_i(y) = \vee_i (t_i \wedge B_i(y)) \tag{8}$$

For use in the ad-hoc networks environment a fourth step must be added. We need a crisp single value for ART. This process is called *defuzzification*. Center of area (COA) was chosen as the defuzzification method given in the following:

$$ART = \frac{\sum_{j=1}^{m} F(y_j) \times y_j}{\sum_{j=1}^{m} F(y_j)} \tag{9}$$

here y_j is a sampling point in a output F discrete universe, and $F(y_j)$ is its membership degree in the membership function.

IV. Performance analysis of the proposed fuzzy art

A. Simulation Environment

Simulating of the proposed AODV design method was done using *OMNeT++* version 2.3 with *Ad-Hoc simulator* version 1.0 developed by *Nicola Concer* [22]. *OMNeT++* is a powerful object-oriented modular discrete event simulator tool. Each mobile host is a compound module which encapsulates the following simple modules: a physical layer, a MAC layer, a routing layer, an application layer, and a mobility layer. Each host has defined transmission power was chosen from a uniformly distributed number between 18 kW and 24 kW. Two different network sizes are modeled: 700m×700m map size with 25 nodes and 850m×850m map size with 35 nodes. Each simulation run takes 300 simulated seconds. Multiple runs were conducted for each scenario and collected data was averaged over those runs.

The *random waypoint* model was adopted for the mobility model. In this mobility model, a node randomly selects a destination. On reaching the destination, another random destination is targeted after 3 seconds *pause time*. The speed of movement of individual nodes range from 0 to 10 m/s. The direction and magnitude of movement was chosen from a uniformly distributed random number. Other simulation parameters are reported in Table IV.

TABLE IV
SIMULATION PARAMETERS

Application Layer		**Physical Layer**	
Message packet size	512 byte	Channel Bandwidth	11 Mb/s IEEE 802.11a
Burst length	64 packets	Channel Delay	10 *m* s
Send Packet Rate	3/sec	Channel Error probability	1 bit on 10^6

B. Performance Metrics

Two metrics were used for measuring performance:

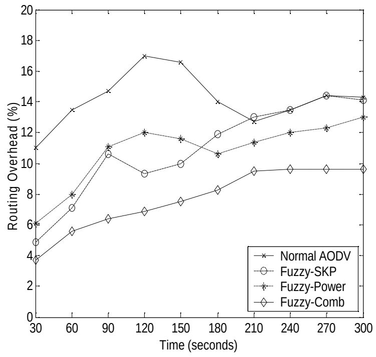
- Routing Overhead:

$$Overhead = \frac{\sum_{i=1}^{n} Number\ of\ \ SentCtrlPkt\ by\ source}{\sum_{i=1}^{n} Number\ of\ \ received\ data\ by\ destination} \tag{10}$$

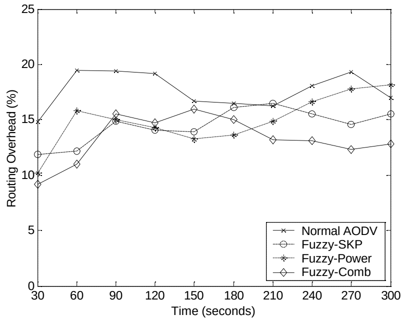
  where n is number of nodes in the network. This metric can be employed to estimate how many transmitted control packets are used for one successful data packet delivery to determine the efficiency and scalability of the protocol.
- *Average End-to-End Delay:*
  Average packet delivery time from a source to a destination. First, for each source-destination pair, average delay for packet delivery is computed. Then the whole average delay is computed from each paired average delay. End-to-end delay includes the delay in the send buffer, the delay in the interface queue, the bandwidth contention delay at the MAC, and the propagation delay.

C. Simulation Results and Evaluations

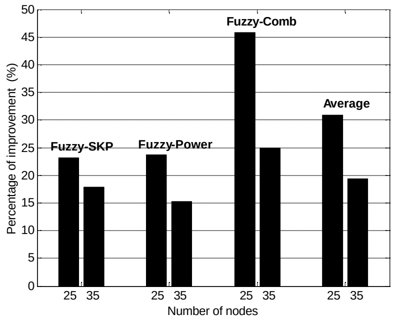
Comparison between routing overhead of normal AODV and the proposed fuzzy design methods are shown in Figure 5. Using normal AODV as a base system, the results show that the proposed fuzzy methods decrease routing overhead with average 25.2% than the normal AODV. This decrement in the routing overhead is due to the decrease in the number of SentCtrlPkt that were used to maintain and recover the connection, as well as minimum data loss through broken paths, hence increased the number of received data by destination. Fuzzy AODV methods have less route recoveries, and hence less SentCtrlPkt. It therefore improves the efficiency and scalability of the protocol. It is interesting to note that fuzzy-Comb method has shown significant enhancement than the normal AODV (and to a lesser extent non-combined fuzzy methods). This is due to combining the three parameter (path length, node mobility, and transmission power) to choose a reliable value for ART. In this method, many paths are given a very short ART due to the inability to maintain a route. Hence, with fewer paths being maintained, fewer route recoveries are necessary.



(a) 25 nodes

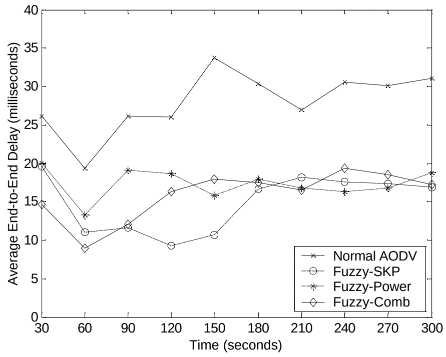


(b) 35 nodes

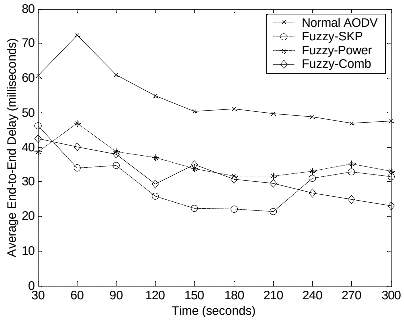


(c) Percentage of improvement than the normal AODV

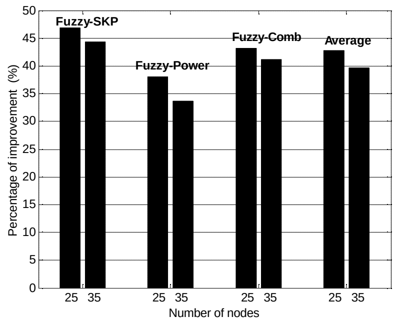
Fig. 5. Routing overhead comparison



(a) 25 nodes



(b) 35 nodes



(c) Percentage of improvement than the normal AODV

Fig. 6. Average end-to-end delay comparison

Figure 6 indicates that the proposed fuzzy AODV methods have lower average end-to-end delay compared to normal AODV with average 41.2%. The normal AODV needs more routing delay to recover from broken paths and discover new ones. To recover a broken path, a RERR message must first be initiated from the intermediate node to inform their end nodes (i.e., source and destination nodes) about the link break. The end nodes delete the corresponding entries from their routing table. The RREQ must then be broadcast from the source to the destination, and a RREP consequently has to be transmitted back to the source. Data packets are buffered at the source node during this process and the duration of their buffering adds to the end-to-end delay. Fuzzy AODV methods, on the other hands, have reliable routes that minimize the need to this recovery process. As expected, node mobility parameter used by fuzzy-SKP method had more effect on route reliability than transmission power parameter used by fuzzy-Power method.

This enhancement for the fuzzy methods is a result of choosing the reliable adaptive route lifetime to update the paths in the routing table. The worse result of normal AODV is due its specification stating that a route lifetime for a path has to be shifted in the future each time a data message is sent using that path. This is a very bad role played by the AODV as it makes the paths request for much more time than they actually needed. Work toward developing techniques for quickly re-establishing valid routes is likely to be of the highest importance for improving the AODV protocol.

In the normal AODV, Active Route Timeout (ART) always take a static value of 3 milliseconds, Figure 7 shows the values used by the proposed fuzzy ART for randomly chosen nodes in our simulated network. It is shown that the fuzzy ART uses a variety of values of between 1 millisecond and 4.5 milliseconds. This value of fuzzy ART is used by one node in our 25 nodes simulated scenario. Every node in the network has its own values of ART for every path in the routing table.

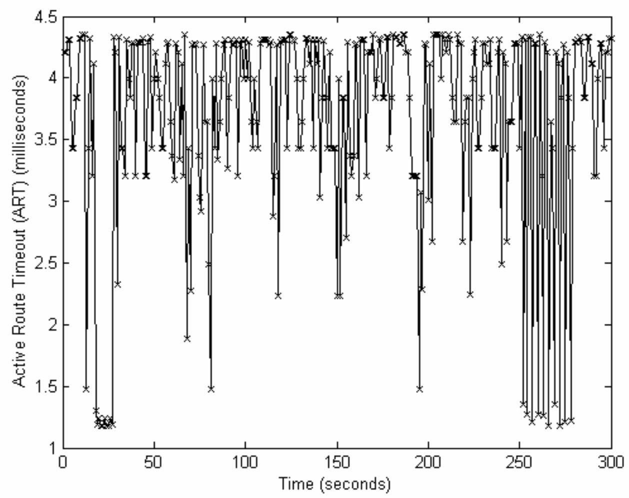


Fig. 7. Fuzzy ART values used by a node

V. Conclusions

The paper proposes the use of a fuzzy mechanism for generating adaptive values for optimum route lifetimes in the AODV routing protocol. Three approaches utilizing the path length, the node mobility, and the transmission power have been used to create a 2-dimensional rule-bases to control the timeout delay adaptively. The performance of the proposed models has been compared with the performance of the original AODV. The performance analysis showed that the proposed fuzzy models have a better routing overhead and average end-to-end delay than the original method. Hence fuzzy logic AODV has shown advancement than the original AODV and is expected to perform better in wireless ad-hoc networks.

References

[1] C. E. Perkins and E. M. Royer, "The Ad-hoc On-Demand Distance Vector Protocol," in: "*Ad Hoc Networking*", edited by C. E. Perkins, Ch. 6, pp. 173-220, Addison-Wesley, 2001.

[2] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," *Proc. of the 2nd IEEE workshop on Mobile Computing Systems and Applications*, New Orleans, LA, pp. 90-100, Feb. 1999.

[3] C. Perkins, "*Ad Hoc On Demand Distance Vector (AODV) routing*", Internet-Draft, draft-ietf-manet-aodv-00.txt (Work in progress), Nov. 1997.

[4] C-K. Toh, *Wireless ATM and Ad Hoc Networks: Protocols and Architecture*s, Kluwer Academic Press, 1997.

[5] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," in: *Mobile Computin*g, edited by T. Imielinski and H. Korth, Ch. 5, pp. 153-181, Kluwer Academic Publishers, 1996.

[6] V. Park and M. S. Corson, MANET Internet Draft "draft-ietf-manet-tora-spec-03.txt," Nov. 2000.

[7] K. Paul, S. Bandyopadhyay, A. Mukherjee, and D. Saha, "Communication-Aware Mobile Hosts in Ad-Hoc Wireless Network," *Proc. Int'l Conf. Personal Wireless Comm.*, pp. 83-87, 1999.

[8] B. Liang and Z. J. Haas, "Optimizing Route-Cache Lifetime in Ad Hoc Networks," *Proc. of the 22th IEEE INFOCOM*, San Francisco, CA, April 1-3, 2003.

[9] S. Agarwal, A. Ahuja, J. Singh, and R. Shorey, "Route-Lifetime Assessment Based Routing (RABR) Protocol for Mobile Ad-Hoc Networks," *IEEE Int'l Conf. Comm.*, pp. 1697-1701, 2000.

[10] Y. Tseng, Y. Li and Y. Chang, "On Route Lifetime in Multihop Mobile Ad Hoc Networks", *IEEE Trans. on Mobile Computing,*, Vol. 2, No. 4, pp.366-376, Oct. 2003.

[11] S. Ghosh, Q. Razouqi, H. J. Schmacher, and A. Celmins, "A survey of recent advances in fuzzy logic in telecomunications networks and new challenges," *IEEE Trans. on Fuzzy System*s, vol. 6, pp. 443–447, Aug. 1998.

[12] A. Sekercioglu, A. Pitsillides, A. Vasilakos, "Computational intelligence in management of ATM networks: A survey of current state of research", *Soft Computing Journal*, vol. 5, pp.257-263, Sep. 2001.

[13] A.R. Bonde and A. Ghosh, "A comparative Study of Fuzzy Versus 'Fixed' Threshold for Robust Queue Management in Cell-Switching Networks", *IEEE/ACM Trans. on Networking*, Vol. 2, pp.337-344, Aug. 1994.

[14] S. Pithani and A. S. Sethi, "A fuzzy set delay representation for computer network routing algorithms", *Proc. of the Second Int. Symposium on Uncertainty Modeling and Analysis*, College Park, pp. 286-293, April 1993.

[15] E. Aboelela and C. Douligeris, "Routing in Multimetric Networks Using A Fuzzy Link Cost", *The 2nd IEEE Symposium on Computers and Communications*, Alexandria, Egypt, pp. 397-401, July 1997
[16] A. Pasupuleti, A.V. Mathews, N Shenoy and A. Dianat, "A fuzzy system for Adaptive Network Routing", *Proc. of SPIE*, Vol. 4740, Orlando, USA. April 2002
[17] Y. F. Wong and W. C. Wong, "A Fuzzy-Decision-Based Routing Protocol for Mobile Ad Hoc Networks," *Proc. of the 10th IEEE Int. Conf. on Networks (Icon 2002)*, pp. 317-322, Aug. 2002.
[18] C. S. Murthy and B. S. Manoj, "*Ad Hoc Wireless Networks: Architectures and Protocols*", pp. 457-459, Prentice Hall, 2004.
[19] T. S. Rappaport, "*Wireless Communications: Principles & Practice*," ch.3, pp. 70- 74, Prentice Hall, 1996.
[20] W. Pedrycz, "Why triangular membership functions?," *Fuzzy Sets Syst*., vol. 64, pp. 21–30, 1994.
[21] R. R. Yager and D. P. Filev, "*Essentials of Fuzzy Modeling and Control*", pp. 109-153, John Wiley & Sons, 1994.
[22] Available at: http://www.omnetpp.org/