PAPER

VANET Broadcast Protocol Based on Fuzzy Logic and Lightweight Retransmission Mechanism

Celimuge WU[†a)], **Satoshi OHZAHATA**[†], **and Toshihiko KATO**[†], *Members*

SUMMARY   Vehicular ad hoc networks have been attracting the interest of both academic and industrial communities on account of their potential role in Intelligent Transportation Systems (ITS). However, due to vehicle movement and fading in wireless communications, providing a reliable and efficient multi-hop broadcast service in vehicular ad hoc networks is still an open research topic. In this paper, we propose FUZZBR (FUZZy BRoadcast), a fuzzy logic based multi-hop broadcast protocol for information dissemination in vehicular ad hoc networks. FUZZBR has low message overhead since it uses only a subset of neighbor nodes to relay data messages. In the relay node selection, FUZZBR jointly considers multiple metrics of inter-vehicle distance, node mobility and signal strength by employing the fuzzy logic. FUZZBR also uses a lightweight retransmission mechanism to retransmit a packet when a relay fails. We use computer simulations to evaluate the performance of FUZZBR.
key words: *vehicular ad hoc networks, broadcast protocol, fuzzy logic, retransmission*

1.   Introduction

A Vehicular Ad hoc Network (VANET) is a form of mobile ad hoc network providing communication between vehicles in close proximity, and between vehicles and nearby fixed roadside equipment. Vehicular ad hoc networks are expected to be able to significantly reduce the number of road accidents. Vehicles travel at a high speed on major roads, giving drivers very little time to react to the vehicle in front of them. In vehicular ad hoc networks, by using wireless communications, emergency information can be propagated along the road to notify drivers ahead of time so that the necessary action can be taken to avoid accidents. Vehicular ad hoc networks also can be used to disseminate traffic warning information and service information, making driving more efficient. To satisfy these demands, an efficient multi-hop broadcast protocol should be seriously considered.

Due to the following reasons, the design of multi-hop broadcast in vehicular ad hoc networks is very challenging. In vehicular ad hoc networks, vehicle densities change with the time and road. Therefore, a VANET broadcast protocol should work in various node densities. If vehicles are deployed in a dense manner, a simple broadcast scheme cannot work well because of redundant broadcasts which result in packet collisions and a low packet delivery ratio. However, it is difficult to reduce the number of redundant messages

while maintaining a high packet dissemination ratio. This is because wireless communications can be unreliable, especially when vehicles are moving at high speeds. In a sparse network, collisions also occur when the packet transmission rate is high.

There are many broadcast protocols [1]–[7] for VANETs. However, these protocols can not provide enough reliability and efficiency. In our previous work [8], we proposed a protocol in which only selected relay nodes rebroadcast a packet. This protocol also uses an acknowledgement method to detect whether all intended receivers have received a packet, and retransmits the packet when a packet loss occurs. However, since the received signal strength is not considered in the selection of relay nodes, the protocol's performance degrades in an environment where there is fading. Also, in the protocol, except for the specified relay nodes, receivers have to acknowledge the sender node explicitly. This increases the overhead of the protocol especially in a high density network. Therefore, a reliable and efficient relay node selection algorithm and a low cost retransmission mechanism are required.

In the relay node selection, multiple metrics of inter-vehicle distance, node mobility and signal strength should be considered jointly. However, it is difficult to establish a satisfactory relay node evaluation criterion for the following reasons. First, the network information (inter-vehicle distance, node mobility and signal strength) known by each node is inaccurate, incomplete and imprecise. Second, since these metrics may conflict with each other, it results in uncertainty. To deal with this imprecision and uncertainty, we propose FUZZBR, a multi-hop broadcast protocol which uses a fuzzy logic based method to select relay nodes. Based on the fuzzy logic, FUZZBR can select the best nodes to relay data messages by considering inter-vehicle distance, node mobility and signal strength.

The retransmission mechanism should not incur too much additional control messages which increase the packet collisions and end-to-end delay. FUZZBR uses a lightweight retransmission mechanism. Without additional control messages, a sender node retransmits a data message when packet losses occur at the specified relay nodes. By using the combination of the fuzzy logic based relay node selection method and the lightweight retransmission mechanism, FUZZBR can achieve a high packet dissemination ratio with a low overhead. We evaluate FUZZBR's performance using the network simulator ns-2 [9] and compare the protocol with other broadcast schemes.

The remainder of the paper is organized as follows. In Sect. 2, we give a brief outline of related work. In Sect. 3, we give a detailed description of the proposed protocol. Next, we present simulation results in Sect. 4. Finally, we present our conclusions and proposals for future work in Sect. 5.

2. Related Works of Multi-Hop Broadcast in Vehicular Ad Hoc Networks

The biggest challenge for a VANET broadcast protocol is to provide an reliable and efficient communications in a high density network environment. This is because when an accident happens, vehicles are usually deployed in a dense manner.

The simplest way to disseminate information is Flooding. In Flooding, each node rebroadcasts a packet upon its first reception. Obviously, in a high-density network, Flooding introduces too many redundant broadcasts and consequently incurs collisions and results in a low dissemination rate. There have been a lot of protocols to reduce the redundant broadcasts in a high-density network. Here, we classify these protocols into sender-oriented protocols and receiver-oriented protocols. In sender-oriented protocols, a sender node specifies the relay nodes of broadcast messages. In contrast, in receiver-based protocols, upon reception of a broadcast message, a receiver node determines its own action (whether to rebroadcast or not) in an autonomous manner.

Several receiver based broadcast protocols have been proposed [1]–[4]. However, in all the receiver-based protocols, each node determines whether to rebroadcast or not in an autonomous manner. As a result, redundant broadcasts cannot be eliminated entirely. Since receiver based broadcast protocols generally use a probabilistic method to rebroadcast a packet, it is difficult to provide enough reliability, especially in a sparse network. Therefore, receiver based broadcast protocols are not suitable for VANET applications that require a high reliability.

In the sender-oriented protocols, the selection of relay nodes is based on the information collected from an exchange of hello messages. Qayyum et al. [5] have proposed a multipoint relay (MPR) broadcast scheme (here we call this MPR Broadcast). Djedid et al. [6] have proposed a broadcast protocol which selects relay nodes based on a Connected Dominating Set. However, Ref. [5] and Ref. [6] do not consider node mobility as a factor in the relay node selection. As a result, the selection of relay nodes can become sub-optimal and can lead to the message being lost because of node movement. In our previous work, we have proposed a relay node selection method which considers increased radio range and node mobility (here we call this Enhanced MPR Broadcast) [8]. However, Enhanced MPR Broadcast does not consider the fading feature of wireless channels. In a wireless channel, a node can receive a hello message from a neighbor which is at a distance where stable communication is impossible. If the inappropriate neighbor node is selected as a relay node, the neighbor node might not receive the message with a high probability. Sahoo et al. [7] have proposed a protocol which uses the most distant node in the required direction to relay broadcast messages. However, in a fading channel, use of the most distant node may result in lost messages. Therefore, multiple metrics of inter-vehicle distance, mobility and signal strength should be considered in the relay node selection.

In a VANET broadcast protocol, a retransmission mechanism is required because of vehicle movement, packet collisions and random loss feature of a fading channel. In the Enhanced MPR Broadcast [8], a sender node retransmits a packet when there is a receiver node which does not receive the packet in a predefined time interval. To check whether a node does receive a packet or not, the Enhanced MPR Broadcast uses explicit ACK messages. This increases the protocol overhead especially in a dense environment. A high control message overhead incurs the packet collisions and increases the MAC layer contention time at each node. This results a high delay which makes a data message useless even if it can be received later. Therefore, a lightweight retransmission mechanism should be considered.

3. Proposed Protocol: FUZZBR

In order to reduce rebroadcast redundancy in high-density networks, FUZZBR uses only a subset of nodes in the network to relay broadcast packets. Before broadcasting a packet, a sender node attaches the addresses of the relay nodes to the packet. Upon reception of a packet, a node rebroadcasts the packet only if it is itself included in the relay node list. Vehicles exchange information through hello messages. Every vehicle inserts its own position information in hello messages. We assume every node knows its own position and road map information because it is possible to get this position information from GPS-like positioning services.

In the relay node selection, FUZZBR considers inter-vehicle distance, mobility and signal strength. These three metrics are contradictory. If we select the farthest node as a relay node, it will minimize the number of relays. However, the relay node might lose the packet because the signal is weak. Moreover, due to node movement, the relay node might move out of the transmission range of the sender node. These conflicts depend on the vehicle mobility, vehicle distribution and fading condition. Therefore, the mathematical model of the optimal relay problem is complex to derive and a solution based on it would be too expensive for practical application. Fortunately, fuzzy logic can handle imprecise and uncertain information. In FUZZBR, we use a fuzzy logic based method to identify those relay nodes that will give the best results.

FUZZBR also uses a lightweight retransmission mechanism to retransmit a packet when a relay fails. If a sender node does not receive the retransmitted packet from a relay node in a predefined *ack_delay_constraint* time, the sender node judges this as a packet loss and retransmits the packet.
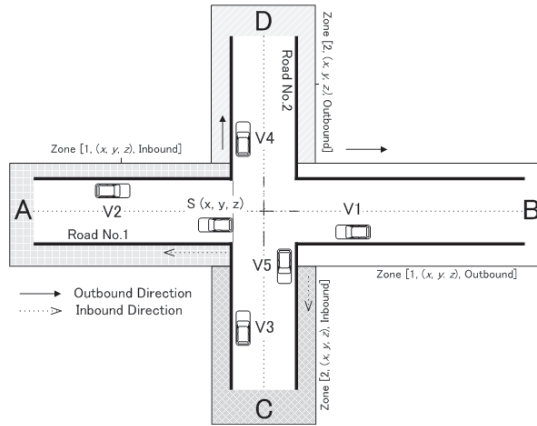
Fig. 1 An urban street topology.

Fig. 2 A mountain road topology.

3.1 Broadcast to All Intended Receivers with Only a Small Number of Rebroadcasts

In FUZZBR, the sender node specifies relay nodes. This poses the question of how many relay nodes should be selected and how to ensure that these relay nodes can reach all intended receivers. In FUZZBR, a sender node first groups neighbor vehicles according to [road_no, sender_pos, direction]. As shown in Fig. 1, "road_no" denotes the road number, "sender_pos" denotes the sender position and "direction" can be "outbound" or "inbound." We call a triad [road_no, sender_pos, direction] a "broadcast zone". For example, the triad [1, (x, y, z), outbound] shows the area which is on the road No.1 and in the "outbound" direction of position (x, y, z).

We note that "outbound" and "inbound" are predefined in each road. For a loopless road, since the start point and end point can be defined, we define the direction from the start point to the end point as "outbound," and define the direction from the end point to the start point as "inbound." For a loop road, we define the clockwise direction as "outbound" and the counter-clockwise direction as "inbound." As shown in Fig. 1, for road No.1, the direction from A to B is the outbound direction, and from B to A is the inbound direction. In here, "outbound" and "inbound" depend on the position of the vehicles but be independent to the driving directions of the vehicles. We say V1 is at the outbound direction of node V2. In contrast, V2 is at the inbound direction of node V1. In Fig. 2, R1, R2, R3, R4, R5 are at the outbound direction of S. Similarly, R4, R3, R2, R1 and S are at the inbound direction of R5.

Before broadcasting a data message, the source node specifies the intended area as a list of broadcast zones. The sender node selects one relay node in each of the specified broadcast zones. In the example in Fig. 1, to disseminate information in all directions, node S has to select 4 relay nodes.

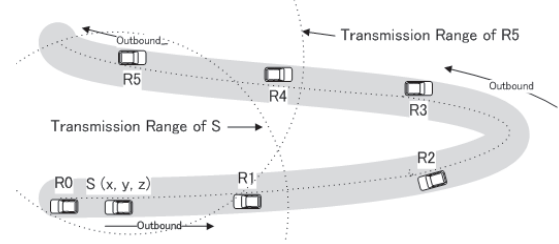In a large scale network, we do not need to let a data message traverse through the whole network. In this case

we can specify a border for each broadcast zone by specifying the most distant (from the sender node) position of the intended area. Another way is to define a life time for each message by specifying the hop count or TTL (Time To Live). In this paper, for simplicity, we consider all nodes in the network as the intended receivers. Without specific explanations, there is no border specified for a broadcast zone.

Before relaying a broadcast message, a node has to calculate the intended area using the position information of the source node and neighbor nodes. This is to deal with some special scenarios similar to that shown in Fig. 2. In this figure, the sender node S has to disseminate messages in the outbound direction. If the road number is 3 then the broadcast zone will be [3, (x, y, z), outbound]. FUZZBR selects a relay node from this broadcast zone. As shown in the figure, node R5 is also in this broadcast zone. As a result, node S may select node R5 as a relay node. In this case, node R5 has to disseminate the message in both backward and forward directions. Here, if node R5 only forwards the message in the outbound direction without re-calculating the intended area, node R3 will not receive the message. Therefore, R5 has to specify node R4 as an inbound relay node. Note that R5 also defines the position of the node S as the border for the inbound broadcast zone. For example, if a data message is transmitted to node R0 by a path 'R5→R4→R3→R2→R1→R0', node R0 should not rebroadcast the message furthermore. This is because node R0 is at the inbound direction of node S whose position is defined as the border of the broadcast zone. In this way, we can avoid a rebroadcast loop. In short, by defining broadcast zones and specifying relay nodes, we can ensure the message can be delivered to all intended receivers with only a small number of rebroadcasts.

3.2 Periodical Neighborhood Evaluation

In FUZZBR, upon receiving a hello message from a neighbor, a node evaluates the neighbor according to the inter-vehicle distance, mobility and signal strength. In this way, through exchanging hello messages, each node maintains an evaluation result for each neighbor. When selecting a relay node, these evaluation results are used. If a node does not receive any hello message from a neighbor in a time interval T which is defined as 3 times the hello interval (1 s), the node initializes the evaluation values (including all factors that will be given latter) of the neighbor. The neighbor also

will be deleted from the neighbor list.

The hello interval does affect the performance of the FUZZBR. If the hello interval is too long, the neighbor information can be inaccurate and useless. Contrarily, if the hello interval is too short, the increased message overhead can result performance degradation. The optimal hello interval depends on node density, node velocity and other factors. In FUZZBR, we use 1 s hello interval. We know this value is suitable for many scenarios from our experience [10].

3.2.1 Distance

Upon reception of a hello message from a neighbor X, a node calculates a Distance Factor (DF) as in Eq. (1). In Eq. (1), $d(X)$ is the distance between the current node and node X. R is the maximum distance over which stable communications can be provided. Here we assume that every node has the same transmission power and that the transmission power is constant.

$$DF(X) = \begin{cases} \frac{d(X)}{R}, & d(X) <= R \\ 1, & d(X) > R. \end{cases} \quad (1)$$

3.2.2 Mobility

Upon reception of a hello message from a neighbor X, a node calculates a Mobility Factor (MF) as in Eq. (2). MF indicates the mobility level of the neighbor node. The higher the value of MF, the more stable the neighbor node is. Here, $d_i(X)$ is the distance between the current node and the neighbor node at time i. α is a smoothing factor and its value is set to 0.7. MF is initialized to 0. In Eq. (2), we use an exponential moving average because we want to smooth out short-term errors. After a lot of experiments and analysis, we know that 0.7 is the most suitable value for α.

$$MF(X) \leftarrow (1 - \alpha) \times MF(X) \\ + \alpha \times \left(1 - \frac{|d_i(X) - d_{i-1}(X)|}{R}\right). \quad (2)$$

3.2.3 Signal Strength

Upon reception of a hello message from a neighbor X, a node calculates a Received Signal Strength Indication Factor (RSSIF) as in Eq. (3). In Eq. (3), RxPr is the received signal power, RXThresh is the reception threshold. The value of RXThresh is defined based on received power and a hello message cannot be received when the received power is lower than this value. RSSIF indicates the average signal strength of the neighbor node. Here RSSIF is initialized to 0.

$$RSSIF(X) \leftarrow (1 - \alpha) \times RSSIF(X) \\ + \alpha \times \left(1 - \frac{RXThresh}{RxPr}\right). \quad (3)$$

3.3 Relay Node Selection

3.3.1 Fuzzy Set Theory and Fuzzy Logic

A difference from classical set theory is that, in fuzzy set theory [11], elements have degrees of membership. By defining set membership as a possibility distribution, fuzzy set theory can represent incomplete or imprecise information. Based on fuzzy set theory, fuzzy logic deals with the concept of approximate rather than precise factors. For example, define a person's height as being 0.6 "high" and 0.4 "low," rather than "completely high" or "completely low." Since fuzzy logic can handle approximate reasoning which is similar to human reasoning, it has been widely accepted in industrial communities and used in many applications. In contrast to numerical values in mathematics, fuzzy logic uses non-numeric linguistic variables to express the facts. Fuzzy membership functions are used to represent the degrees of a numerical value belonging to linguistic variables.

Typically, a fuzzy logic based system consists of three steps: input, process and output steps. The input step converts input numerical values to linguistic variables. The process step collects logic rules which are defined in the form of IF-THEN statements and applies the rules to get the result in a linguistic format. The output step converts the linguistic result into a numerical value.

3.3.2 Procedure

As mentioned above, in FUZZBR each node evaluates its neighbors in term of distance, mobility and signal strength by exchanging hello messages. When a node has to send a packet, the node employs fuzzy logic to calculate an average relay fitness value for each neighbor based on the neighbor's distance, mobility and signal strength. The node then selects a relay node for each of the required broadcast zones. For each neighbor, the calculation steps are as follows.

- **Step1: Fuzzification** Use predefined linguistic variables and membership functions to convert the distance factor, mobility factor and RSSI factor to fuzzy values.
- **Step2: Mapping and combination of IF/THEN rules** Map the fuzzy values to predefined IF/THEN rules and combine the rules to get the rank of the neighbor as a fuzzy value.
- **Step3: Defuzzification** Use a predefined output membership function and defuzzification method to convert the fuzzy output value to a numerical value.

After calculating the relay fitness values for all neighbors, the sender node selects the node that has maximal fitness value to relay the packet to a particular zone.

3.3.3 Fuzzification

The process of converting a numerical value to a fuzzy value using a fuzzy membership function is called "fuzzification".
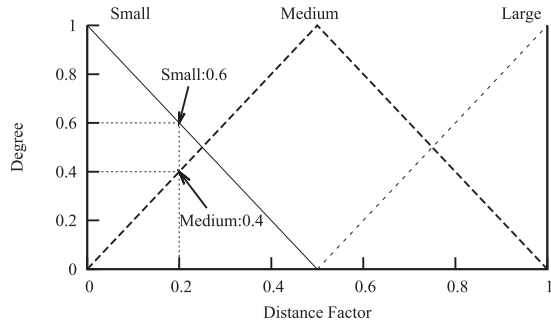
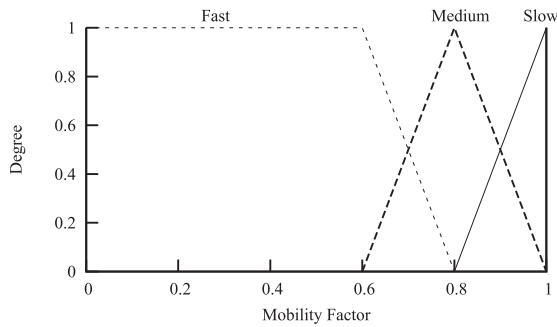**Fig. 3**   Distance membership function and an fuzzification example.

Fig. 5   Signal strength membership function.

Fig. 4   Mobility membership function.

Table 1   Rule base.

	Distance	Mobility	RSSI	Rank
Rule1	Large	Slow	Good	Perfect
Rule2	Large	Slow	Medium	Good
Rule3	Large	Slow	Bad	NotAcceptable
Rule4	Large	Medium	Good	Good
Rule5	Large	Medium	Medium	Acceptable
Rule6	Large	Medium	Bad	Bad
Rule7	Large	Fast	Good	NotAcceptable
Rule8	Large	Fast	Medium	Bad
Rule9	Large	Fast	Bad	VeryBad
Rule10	Medium	Slow	Good	Good
Rule11	Medium	Slow	Medium	Acceptable
Rule12	Medium	Slow	Bad	Bad
Rule13	Medium	Medium	Good	Acceptable
Rule14	Medium	Medium	Medium	NotAcceptable
Rule15	Medium	Medium	Bad	Bad
Rule16	Medium	Fast	Good	Bad
Rule17	Medium	Fast	Medium	Bad
Rule18	Medium	Fast	Bad	VeryBad
Rule19	Small	Slow	Good	NotAcceptable
Rule20	Small	Slow	Medium	Bad
Rule21	Small	Slow	Bad	VeryBad
Rule22	Small	Medium	Good	Bad
Rule23	Small	Medium	Medium	Bad
Rule24	Small	Medium	Bad	VeryBad
Rule25	Small	Fast	Good	Bad
Rule26	Small	Fast	Medium	VeryBad
Rule27	Small	Fast	Bad	VeryBad

The fuzzy membership function of the distance factor is defined as in Fig. 3. The sender node uses the membership function and the distance factor to calculate which degree the distance factor belongs to {Large, Medium, Small}. As shown in Fig. 3, when the distance factor is 0.2, we know that the vertical line showing this distance factor meets with "Small" and "Medium" at (0.2,0.6) and (0.2,0.4) respectively. Therefore, we can get the fuzzy value {Large:0, Medium:0.4, Small:0.6}. We note here that the distance membership function given above and the other membership functions which will be given later are all defined based on our simulation results.

The fuzzy membership function of the mobility factor is defined as in Fig. 4. The sender node uses the membership function and the mobility factor to calculate which degree the mobility factor belongs to {Slow, Medium, Fast}.

The fuzzy membership function of the RSSI factor is defined as in Fig. 5. The sender node uses the membership function and the RSSI factor to calculate which degree the RSSI factor belongs to {Good, Medium, Bad}.

3.3.4　Rule Base

Once the fuzzy values of distance factor, mobility factor and RSSI factor have been calculated, the sender node uses the IF/THEN rules (as defined in Table 1) to calculate the rank of the node. The linguistic variables of the rank are defined as {Perfect, Good, Acceptable, NotAcceptable, Bad, Very-Bad}. For example, in Table 1, Rule1 may be expressed as follows.

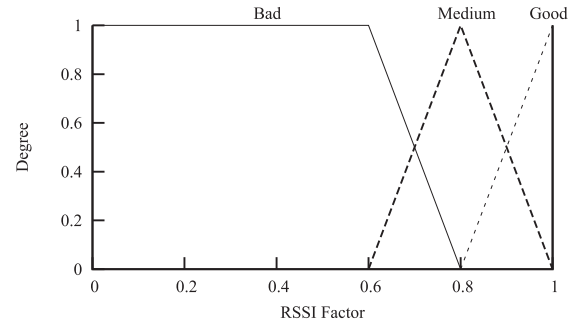**IF** *Distance* is Large, *Mobility* is Slow and *Signal*

Strength is Good **THEN** *Rank* is Perfect.

In a rule, the IF part is called the "antecedent" and the THEN part is called the "consequent." Since there are multiple rules applying at the same time, we have to combine their evaluation results. Here we use the Min-Max method. In the Min-Max method, for each rule, the minimal value of the antecedent is used as the final degree. When combining different rules, the maximal value of the consequents is used.

For example, as shown in Fig. 6, we assume a neighbor's distance, mobility and RSSI factors belong to the corresponding linguistic variables as {Large:1, Medium:0, Small:0}, {Slow:0.75, Medium:0.25, Fast:0}, {Good:0.5, Medium:0.5, Bad:0} respectively. In this case, these fuzzy sets would match Rule1, Rule2, Rule4 and Rule5. For Rule1, the degree for {Large} (Distance) is 1, the degree for {Slow} (Mobility) is 0.75 and the degree for {Good} (Signal Strength) is 0.5. In the Min-Max method, we take the min-
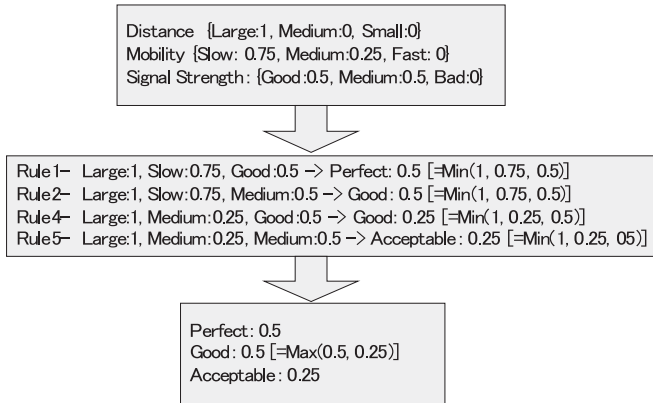
Distance {Large:1, Medium:0, Small:0}
Mobility {Slow: 0.75, Medium:0.25, Fast: 0}
Signal Strength : {Good:0.5, Medium:0.5, Bad:0}

Rule1– Large:1, Slow:0.75, Good:0.5 –> Perfect: 0.5 [=Min(1, 0.75, 0.5)]
Rule2– Large:1, Slow:0.75, Medium:0.5 –> Good: 0.5 [=Min(1, 0.75, 0.5)]
Rule4– Large:1, Medium:0.25, Good:0.5 –> Good: 0.25 [=Min(1, 0.25, 0.5)]
Rule5– Large:1, Medium:0.25, Medium:0.5 –> Acceptable: 0.25 [=Min(1, 0.25, 05)]

Perfect: 0.5
Good: 0.5 [=Max(0.5, 0.25)]
Acceptable : 0.25

Fig. 6　An example for fuzzy rule evaluations.
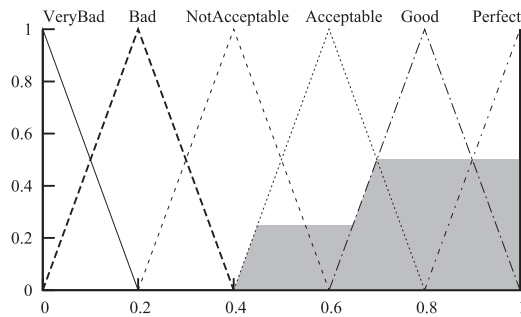
Fig. 7　Output membership function and an example of $\mu(x)$.

imal value of antecedent members and therefore the degree of the antecedent will be 0.5. Similarly, the degrees of the antecedents for Rule2, Rule4 and Rule5 will be 0.5, 0.25 and 0.25 respectively. As both Rule2 and Rule4 lead to the Rank Good, we take the maximal value of their consequents and therefore the degree of the Rank Good will be 0.5. In this way, all rules are combined to give a fuzzy result.

3.3.5　Defuzzification

Defuzzification is the process of producing a numeric result based on an output membership function and corresponding membership degrees. The output membership function is defined as in Fig. 7. Here we use the Center of Gravity (COG) method to defuzzify the fuzzy result. More specifically, we cut the output membership function (Fig. 7) with a straight horizontal line according to the corresponding degree, and remove the top portion. For the example given above, the degree for Rank {Acceptable} is 0.25, the degree for Rank {Good} is 0.5 and the degree for Rank {Perfect} is 0.5 and consequently the result function will form a shape as shown in Fig. 7. Then, we calculate the centroid of this shape. The x coordinate of the centroid will be the defuzzified value. If we use $\mu(x)$ to denote the result function and use x to denote the X-axis, the center of gravity will be

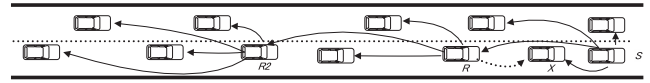$$COG = \frac{\int \mu(x)xdx}{\int \mu(x)dx}. \tag{4}$$

Fig. 8　Data dissemination in FUZZBR.

Here COG represents the fitness of the neighbor to be a relay node. The higher the value is, the better the neighbor node will be. In FUZZBR, for every broadcast zone, the sender node calculates a fitness value for each neighbor node and then selects the node which has the maximal fitness value as the relay node.

3.4　Retransmission Mechanism

In the IEEE 802.11 MAC layer, there is no ACK message for a broadcast message. However, in a fading channel, a reception check mechanism is required because a packet can be lost depending on the wireless link status. If each receiver sends back an ACK message to acknowledge the sender, the sender can know reception status at all intended receivers. This means that we can achieve a high reception ratio by using retransmissions. However, these ACK messages can incur packet collisions and increase the MAC layer contention time, which results a higher delay. To achieve a high packet dissemination ratio and a low delay, FUZZBR uses a lightweight retransmission mechanism.

In this retransmission mechanism, a sender node only retransmits a packet which is not received by the specified relay nodes. If a sender node does not receive a rebroadcasted packet from a relay node in a predefined *ack_delay_constraint* time, the sender node judges this as a packet loss. As shown in Fig. 8, node S retransmits a packet when the node does not receive the packet from node R in a *ack_delay_constraint* time period.

In Fig. 8, node X at least has two chances to successfully receive a packet. One is the transmission from node S, the other is from node R. That is, node X also has a chance to receive a packet from node R even when node X loses the packet from node S. In a fading channel, a packet can be lost randomly. Given a certain reception probability p, with the transmission at node S and node R, node X can receive a packet with a probability of $p(r)$ where

$$p(r) = 1 - (1 - p)^2 = 2p - p^2. \tag{5}$$

In Eq. (5), if p is 0.9, $p(r)$ will be 0.99. Therefore, we can easily know that if the relay nodes are properly selected, an enough packet reception ratio can be achieved with this simple retransmission mechanism. That means, we do not have to use explicit ACKs to check whether all intended receivers have received a packet or not.

FUZZBR uses a fuzzy logic based relay node selection algorithm to select relay nodes and retransmits a packet when these relay nodes do not receive the packet. By considering signal strength in the relay node selection, FUZZBR can choose a relay node which has a high reception probability and therefore can result a high reliability. In FUZZBR,

since all receiver nodes do not need to explicitly acknowledge the sender node, the protocol will not incur additional control message overhead.

We have to note that the value of *ack_delay_constraint* directly affects the outcome delay of FUZZBR. The ideal value of *ack_delay_constraint* depends on various factors including the network density, traffic rate and application requirement. Therefore, an adaptive *ack_delay_constraint* can be promising. However, this is not a focus of this research. In FUZZBR, we set *ack_delay_constraint* to be 0.04 second by default.

4. Simulation Results

We used network simulator ns-2 (version 2.34) [9] to conduct simulations. Simulation environments are shown in Table 2. We evaluated the protocols' performance in freeway scenarios and street scenarios. In the Freeway simulation, we used a freeway which had two lanes in each direction. All lanes of the freeway were 2000 m in length. The distance between any two adjacent lanes was 5 m. The maximum allowable vehicle velocity was 40 m/s. The freeway was generated by [12]. In the Freeway simulation, we evaluated the protocol's performance in various number of nodes. Two source nodes generated 50 packets with a rate of 10 packets per second. These two nodes were neighbors and being close to each other. This is to simulate a condition of two collided vehicles send data messages at the same time.

We also used SUMO [13], [14] and TraNS [15] to generate street scenarios. SUMO is a microscopic traffic simulator and TraNS is a realistic simulation generation tool that integrates SUMO and ns-2. In SUMO, a vehicle's speed is adapted to the speed of the leading vehicle. In our street scenarios, the maximal vehicle velocity was 18 m/s. For each of the street scenarios, we used a street area of 1700 m × 1700 m. The street consisted of 5 horizontal streets and 5 vertical streets and every street had one lane in each direction. The distance between any two neighboring intersections was 400 m. In the Street scenario, 619 nodes were moving toward their destinations (these destinations were selected randomly). Therefore, we can simulate a street which has various node densities on different road segments. We generated scenarios with various number of broadcast source nodes.

We used the Nakagami propagation model. The parameters of the Nakagami model are shown in Table 3. For each parameter, the first value indicates the parameter value used in Freeway scenarios and the value between the parentheses indicates the parameter value used in Street scenarios. We used these parameter values because they model a realistic wireless channel (including the effect of buildings) of vehicular ad hoc networks [16].

Other simulation parameters were the default settings of ns-2.34. The average transmission range was 250 m (due to fading, the transmission range may be reduced, but the transmission range will be 250 m when no fading existed). After the first 20 seconds (to allow the exchange of hello messages), senders sent messages with a packet size of 512 bytes. All nodes in the network were defined as intended receivers (the whole network area was defined as the destination region). The simulation time was 150 s. We launched simulations with 50 different scenarios, and analyzed the average value of the results.

FUZZBR was compared with Flooding, Weighted p-persistence [1], MPR Broadcast [5] and Enhanced MPR Broadcast [8]. As mentioned earlier, in Flooding, every receiver node rebroadcasts a packet upon the first reception. In the Weighted p-persistence scheme, a receiver node first calculates a broadcast probability according to the distance from the sender node divided by the transmission range (250 m) and rebroadcasts the packet with this probability. The greater the distance, the higher is the probability. In the MPR Broadcast, Enhanced MPR Broadcast and FUZZBR, a receiver rebroadcasts the packet only if it is itself specified as a relay node. In the Enhanced MPR Broadcast, a node retransmits a packet when the node does not receive an ACK from any intended receivers in a predefined time interval. Since a packet may be retransmitted multiple times depending on the reception status, to avoid an endless retransmission, for each packet, we set the maximum number of retransmission times to be 4. In the following simulation results, the error bars indicate the 95% confidence intervals.

4.1 Number of Messages

Figure 9 shows the number of messages per data packet for various numbers of nodes in Freeway scenario. We calculate this performance metric as the number of messages generated (including both ACK messages and data messages transmitted by all nodes in the network) divided by the number of data packets generated by the source nodes. Figure 10 shows the number of messages per data packet for various numbers of source nodes in Street scenario.

As shown in Fig. 9, in Flooding, since every node rebroadcasts a packet once, a large number of redundant

Table 2 Simulation environment.

	Freeway scenario	Street Scenario
Topology	2000 m, 4lanes	1700 m × 1700 m
Number of nodes	100 to 600	619
Mobility generation	Ref. [12]	SUMO + TraNS
Number of sources	2	5 to 55
Number of packets	50 packets at each source	
Packet size	512 bytes	
Data rate	10 packet per sec	1 packet per sec
MAC	IEEE 802.11 MAC (2 Mbps)	
Propagation model	Nakagami Model	
Simulation time	150 s	

Table 3 Parameters of Nakagami model: Freeway (Street).

gamma0_	gamma1_	gamma2_	d0_gamma_	d1_gamma_
1.9 (2.0)	3.8 (2.0)	3.8 (2.0)	200 (200)	500 (500)
m0_	m1_	m2_	d0_m_	d1_m_
1.5 (1)	0.75 (1)	0.75 (1)	80 (80)	200 (200)
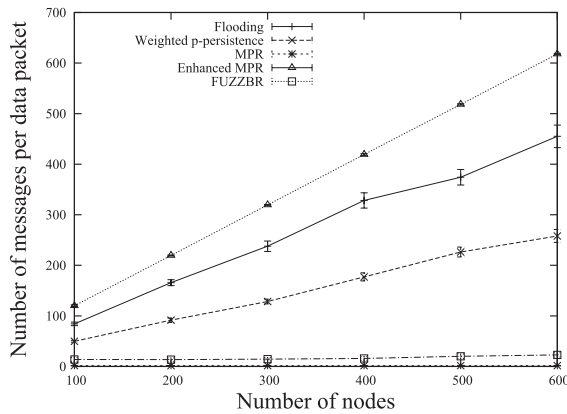
Fig. 9 Number of messages per data packet for various numbers of nodes in Freeway scenario.

Fig. 11 Packet dissemination ratio for various numbers of nodes in Freeway scenario.
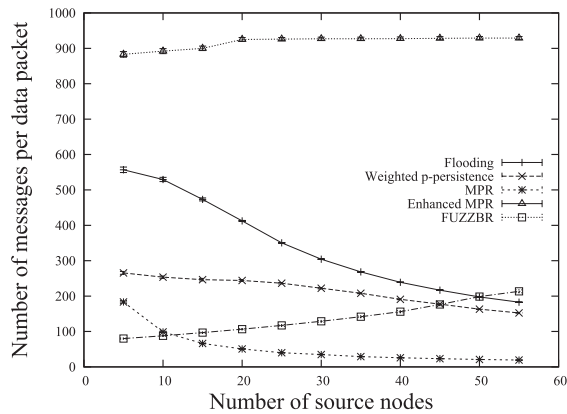

Fig. 10 Number of messages per data packet for various numbers of source nodes in Street scenario.

broadcasts are generated. As a result, many collisions occur and many packets are lost. The Weighted p-persistence scheme performs better than Flooding by using a probabilistic broadcast method. However, the Weighted p-persistence scheme cannot eliminate redundant rebroadcasts entirely. In the MPR Broadcast, Enhanced MPR Broadcast and FUZZBR, only the nodes which have been selected as relay nodes, rebroadcast the packets. Therefore, redundant broadcast can be efficiently reduced. However, since the Enhanced MPR Broadcast uses explicit ACKs to check reception conditions, a lot of ACK messages are generated. This is why the number of messages in Enhanced MPR Broadcast is larger than that in Flooding.

In FUZZBR, the number of nodes does not increase significantly with the increase of the node density because only the specified relay nodes rebroadcast a data message. The small number of messages is also because the lightweight retransmission mechanism does not incur additional control messages. The number of messages is 13.7 for 100 nodes, and 22.9 for 600 nodes. This slight increase is due to the retransmission which is occurred from collisions between a data packet and a hello packet. With the increase of the node density, the number of hello messages increases.
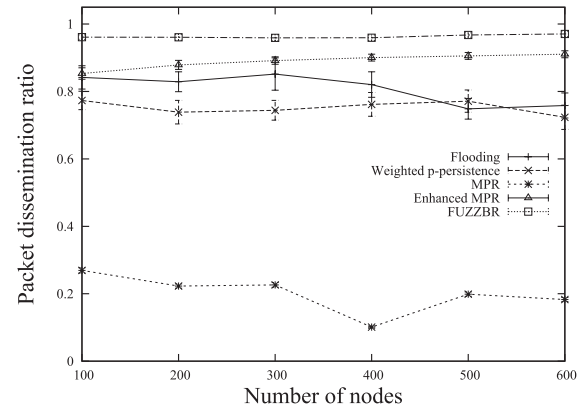
However, this does not incur too great an overhead because these messages are sent periodically. Moreover, we believe these messages are necessary for vehicular ad hoc networks.

In the Street scenario, the number of messages generated by FUZZBR is larger than the number in the Freeway scenario. This is because in the Street scenario, a node near the intersection has to disseminate a data message to multiple directions. We observe an increase of the number of messages with the increase of the number of sources. This is due to the increase of the number of packet collisions. However, this will not impair the advantage of FUZZBR because FUZZBR can achieve a high packet dissemination ratio (see Fig. 13).

The number of messages of Flooding decreases with the increase of the number of source nodes. This is because as the number of source nodes increases, the collisions between different traffic increase. As a result, Flooding loses more packets (see Fig. 13). Enhanced MPR Broadcast uses retransmissions to supplement these packet losses, resulting in a very large number of messages. The number of messages of MPR in Street scenario is larger than in the Freeway scenario. This is because, in the Street scenario, a node near the intersection selects multiple relay nodes. Although some relay nodes fail to receive a packet, the packet can be disseminated by other relay nodes.

4.2 Packet Dissemination Ratio

Figure 11 shows the packet dissemination ratio for various numbers of nodes in Freeway scenario. We calculate this performance metric as the number of data messages received by all nodes in the network divided by the multiplication of the number of data messages generated by the source nodes and the number of nodes in the network. As the number of nodes increases, the dissemination ratio of Flooding decreases. This is because in a high density network, many nodes try to broadcast at the same time and this introduces collisions and a drop in packet reception ratio. The Weighted p-persistence scheme works better than Flooding by reducing the number of broadcasts. However, since a
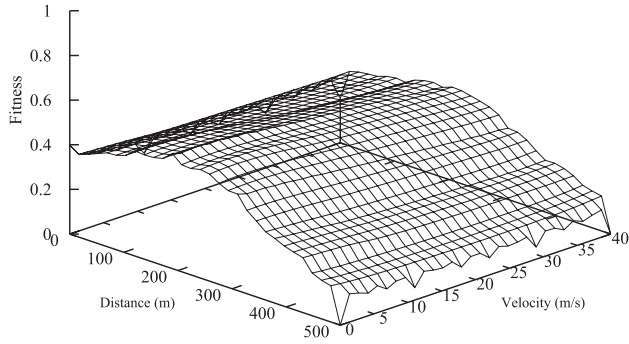
Fig. 12 Relay fitness for various distances and relative velocities.

Fig. 13 Packet dissemination ratio for various numbers of source nodes in Street scenario.

probabilistic method is used in Weighted p-persistence, the number of broadcasts increases as the number of nodes increases, leading to a drop in performance.

In MPR Broadcast scheme, although the number of broadcasts can be reduced efficiently, we observed a poor reception ratio. This is because in the MPR Broadcast scheme, a sender node usually selects the farthest node and consequently the selected node can often not receive the broadcast packet. Another reason is that the MPR Broadcast does not consider the node mobility in the relay node selection. Enhanced MPR Broadcast scheme performs better than MPR Broadcast because it does consider node mobility in the relay node selection. However, Enhanced MPR Broadcast does not consider the signal strength in the relay node selection, the selected relay node loses the packet with a high probability. Fortunately, Enhanced MPR Broadcast uses a retransmission mechanism to retransmits a packet when the packet is lost. However, a packet may not be delivered to a relay node successfully even with multiple retransmissions if the relay node is far away from the sender node.

Figure 12 illustrates the distribution of relay fitness values for various distances and relative velocities for FUZZBR. For this simulation, the received signal power at a certain distance is calculated by averaging the received signal powers of 10,000 packets at the same distance. By jointly considering inter-vehicle distance, node mobility and signal strength, FUZZBR can deal with node mobility and fading while providing large progress on the dissemination direction.

FUZZBR also retransmits a packet when a relay is failed. As a result, FUZZBR achieves a better packet reception ratio than the other protocols. In FUZZBR, when the number of nodes is larger than 300, we see a slight increase of packet dissemination ratio with the increase of the node density. As mentioned above, with the increase of the node density, the number of retransmissions increases. Consequently, the number of fading incurred packet losses decreases.

Figure 13 shows the packet dissemination ratio for various numbers of source nodes in Street scenario. With the increase of the number of source nodes, the collisions among different broadcast traffic increase. Therefore, the packet dissemination ratios of Flooding and Weighted p-
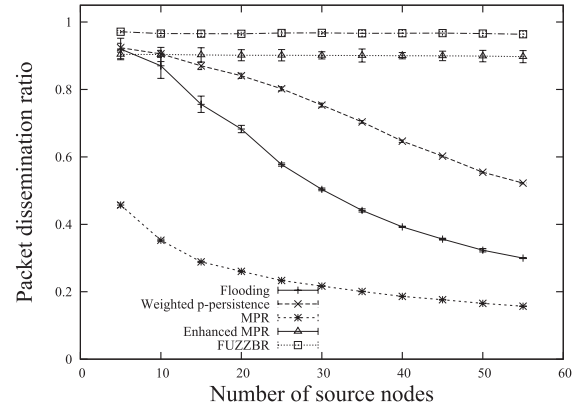
persistence decrease drastically. The packet dissemination ratio of MPR broadcast also drops when there is a large number of source nodes. However, these packet losses are because MPR only considers the increased radio range in the relay node selection. Therefore, the effect of the number of source nodes is not very significant.

In the Street scenario, FUZZBR also shows the highest packet dissemination ratio. This is due to the relay mechanism and the retransmission of FUZZBR. FUZZBR reduces the number of rebroadcasts by using only relay nodes to forward data messages. This can reduce the number of collisions. Since FUZZBR can select a proper relay node to forward data messages, the number of fading incurred packet losses decreases. Even if the collisions or packet losses occur, FUZZBR can retransmit a data message by using the retransmission mechanism. As shown in Fig. 11 and Fig. 13, FUZZBR provides at least 95% packet dissemination ratio in various node densities. Therefore, we can know that the fuzzy logic based relay node selection algorithm and the lightweight retransmission can provide enough reliability with a low control overhead.

4.3 End-to-End Delay

Figure 14 shows the average end-to-end delay between the source node and all intended receiver nodes for various numbers of nodes in Freeway scenario. In this paper, only the successfully delivered data messages are used for the end-to-end delay calculation. In Flooding, as the node density increases, the delay increases drastically. This is because, when the node density is high, the redundant broadcasts introduce many collisions and consequently the nodes that provide larger progress on distance may lose the data packets. As a result, the packets are delayed because they are delivered through sub-optimal paths (longer paths).

Also, as the number of rebroadcasts increases, the contention time at each node increases and therefore the end-to-end delay is increased even more. The end-to-end delay of Weighted p-persistence also increases as the number of source nodes increases because Weighted p-persistence can-
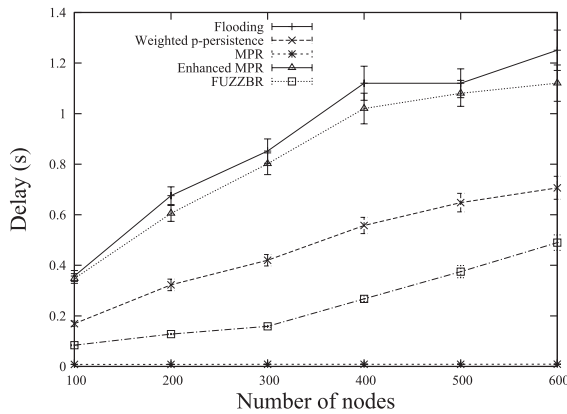
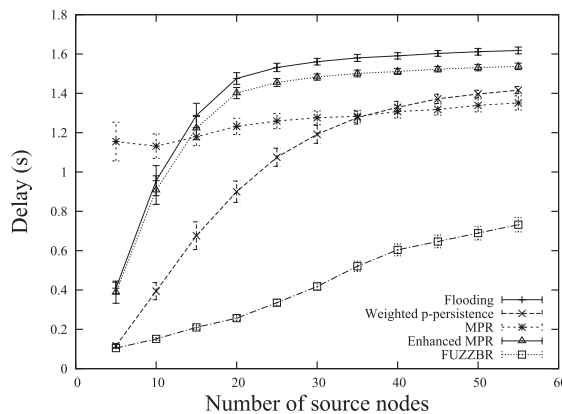**Fig. 14** End-to-end delay for various numbers of nodes in Freeway scenario.

Fig. 15 End-to-end delay for various numbers of source nodes in Street scenario.

not eliminate redundant broadcasts completely. MPR shows the lowest delay. This is because MPR chooses the farthest node as a relay node. The low delay of MPR is also because many data messages are lost at the relay node. Enhanced MPR Broadcast does not show good delay here because too many retransmissions increase the delay of a data message.

In FUZZBR, since a sender node selects a relay node considering inter-vehicle distance, node mobility and signal strength, there is a high probability that the selected relay nodes receive a packet without retransmissions. Although the selected relay nodes are usually not the farthest possible nodes, FUZZBR shows lower end-to-end delays. This is because FUZZBR reduces the contention time at each node by reducing the number of rebroadcasts. FUZZBR shows an increase of the end-to-end delay with the increase of the number of nodes. This is because with the increase of node density, the number of hello messages increases, resulting in a slight increase of MAC layer contention time at each node. However, this is acceptable because FUZZBR does show a low delay even in a high density network.

Figure 15 shows end-to-end delay for various numbers of source nodes in Street scenario. In the simulation, without loss of generality, for each data message, all nodes in

the network are defined as intended receivers. However, it makes no sense to average end-to-end delays for all nodes in the network. We consider 600 m is the distance within which a short propagation delay is required. Therefore, in the delay calculation, for each source node, we only use the receiver nodes of which the distance from the source node is smaller than 600 m. Due to the redundant rebroadcasts, Flooding and Weighted p-persistence show an exponential end-to-end delay increase with the increase of the number of source nodes. The end-to-end delay of MPR is not sensitive to the number of source nodes. That is because redundant rebroadcasts and packet collisions are not the main reasons for the packet losses in MPR. For Enhanced MPR Broadcast, a large number of control messages increase the contention time at each node. In Enhanced MPR Broadcast, too many retransmissions also increase the end-to-end delay.

FUZZBR shows a significant advantage over other protocols. Since only the specified relay nodes rebroadcast a packet, the collisions do not increase drastically with the increase of the number of source nodes. The low delay is also because FUZZBR uses a possibly short path to disseminate a data message by considering inter-vehicle distance in the relay node selection. Choosing a shorter path is very important in the Street scenario because there can be multiple paths to a receiver. If a longer path is used, the end-to-end delay will be increased drastically.

We have to note that, the delay of FUZZBR is sensitive to the *ack_delay_constraint*. The larger *ack_delay_constraint*, the larger the delay will be. A small *ack_delay_constraint* can guarantee a packet will be retransmitted on time. However, the value of *ack_delay_constraint* cannot be too small because that will result many unnecessary retransmissions. In a high density network, unnecessary retransmissions will increase the collisions and the MAC layer contention time at each node. As mentioned before, an adaptive *ack_delay_constraint* can improve the performance of FUZZBR. However, we leave this issue as a future work.

5. Conclusions and Future Works

We have proposed FUZZBR, a fuzzy logic based multi-hop broadcast protocol for vehicular ad hoc networks. FUZZBR reduces the number of broadcast messages efficiently by using only a subset of neighbor nodes to relay data messages. In FUZZBR, sender nodes specify relay nodes based on the inter-vehicle distance, node mobility and signal strength. The protocol employs fuzzy logic to use a combination of these metrics. To ensure reliability, FUZZBR also uses a lightweight retransmission mechanism to retransmit a packet when a relay fails. We used simulations to further evaluate the protocol's performance. The simulation results confirmed that FUZZBR offers a significant performance advantage over existing alternatives.

In this paper, we have chosen FUZZBR parameters based on overall performance in various network environments. The performance of FUZZBR can be improved if

we use different parameters for different scenarios. In our future work, we will work on an adaptive method which can optimize FUZZBR parameters based on network environments and application requirements.

Acknowledgement

### References

[1] N. Wisitpongphan and K.O. Tonguz, "Broadcast storm mitigation techniques in vehicular ad hoc networks," IEEE Wireless Commun., vol.14, no.6, pp.84–94, 2007.

[2] K. Suriyapaiboonwattana, C. Pornavalai, and G. Chakraborty, "An adaptive alert message dissemination protocol for VANET to improve road safety," IEEE Intl. Conf. on Fuzzy Systems, pp.1639–1644, 2009.

[3] M. Slavik and I. Mahgoub, "Stochastic broadcast for VANET," IEEE Consumer Communications and Networking Conference, pp.1–5, 2010.

[4] Y. Mylonas, M. Lestas, and A. Pitsillides, "Speed adaptive probabilistic flooding in cooperative emergency warning," Proc. 4th Annual Intl. Conf. on Wireless Internet, pp.1–7, Maui, Hawaii, 2008.

[5] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying for flooding broadcast messages in mobile wireless networks," 35th Annual Hawaii Intl. Conf. on System Sciences, pp.3866–3875, Big Island, Hawaii, 2002.

[6] L.O. Djedid, N. Lagraa, M. Yagoubi, and K. Tahari, "Adaptation of the MCDS broadcasting protocol to VANET safety applications," Intl. Conf. on Innovations in Information Technology, pp.534–538, 2008.

[7] J. Sahoo, E.H.K. Wu, P.K. Sahu, and M. Gerla, "BPAB: Binary partition assisted emergency broadcast protocol for vehicular ad hoc networks," Proc. 18th Intl. Conf. on Computer Communications and Networks, pp.1–6, San Francisco, USA, 2009.

[8] C. Wu, K. Kumekawa, and T. Kato, "A novel multi-hop broadcast protocol for vehicular safety applications," J. Inf. Process., vol.51, no.3, pp.930–944, 2010.

[9] The Network Simulator - ns-2, http://www.isi.edu/nsnam/ns/, Accessed on June 23, 2010.

[10] C. Wu, K. Kumekawa, and T. Kato, "A dynamic route change mechanism for mobile ad hoc networks," International Journal of Communication Networks and Distributed Systems, vol.7, no.1/2, pp.4–17, 2011.

[11] G.J. Klir, Ute St. Clair, and Y. Bo, Fuzzy set theory: Foundations and applications, Prentice-Hall, 1997.

[12] F. Bai, N. Sadagopan, and A. Helmy, "Important: A framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks," 22nd Annual Joint Conf. of the IEEE Computer and Communications Societies, pp.825–835, San Francisco, USA, 2003.

[13] Simulation of Urban Mobility (SUMO), http://sourceforge.net/apps/mediawiki/sumo/index.php, Accessed on Feb. 2011.

[14] D. Krajzewicz, G. Hertkorn, C. Rossel, and P. Wagner, "SUMO (Simulation of Urban MObility): An open-source traffic simulation," Proc. 4th Middle East Symposium on Simulation and Modelling (MESM2002), SCS European Publishing House, pp.183–187, 2002.

[15] TraNS (Traffic and Network Simulation Environment), http://trans.epfl.ch/, Accessed on Feb. 23, 2010.

[16] A. Khan, S. Sadhu, and M. Yeleswarapu, "A comparative analysis of DSRC and 802.11 over vehicular ad hoc networks," Project Report, Department of Computer Science, University of Californai, pp.1–8, Santa Barbara, 2009.

Celimuge Wu received the M.E. degree from Beijing Institute of Technology, Beijing, China, in 2006, and the Ph.D. degree from the University of Electro-Communications, Tokyo, Japan, in 2010. He is currently an assistant professor of the Graduate School of Information Systems, the University of Electro-Communications. His current research interests include mobile ad hoc networks, networking architectures and protocols.
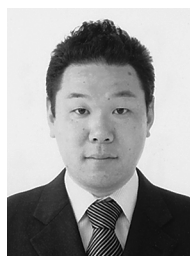
Satoshi Ohzahata received B.S., M.E., and D.E. degrees from the University of Tsukuba in 1998, 2000 and 2003, respectively. He was a Research Associate, Department of Computer, Information & Communication Sciences at Tokyo University Agriculture and Technology from 2003–2007, and was an assistant professor of the same university from 2007–2009. Since 2009, he has been an associate professor at Graduate School of Information Systems, the University of Electro Communication. His interests are mobile ad hoc networks, the Internet architecture in mobile environments and Internet traffic measurement. He is a member of IEEE, ACM and IPSJ.

Toshihiko Kato received the B.E., M.E. and Dr. Eng. degrees electrical engineering from the University of Tokyo, in 1978, 1980 and 1983, respectively. He joined KDD in 1983 and worked in the field of communication protocols of OSI and Internet until 2002. From 1987 to 1988, he was a visiting scientist at Carnegie Mellon University. He is now a professor of the Graduate School of Information Systems in the University of Electro-Communications in Tokyo, Japan. His current research interests include protocol for mobile Internet, high speed Internet and ad hoc network.