IEEE Access
practical innovations : open solutions

Received February 10, 2014, accepted February 24, 2014, date of publication March 11, 2014, date of current version March 18, 2014.

Digital Object Identifier 10.1109/ACCESS.2014.2309856

A Localized Adaptive Strategy to Calculate the Backoff Interval in Contention-Based Vehicular Networks

TAMER ABDELKADER[1] AND KSHIRASAGAR NAIK[2]
[1]Department of Information Systems, Ain Shams University, Cairo 11566, Egypt
[2]Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L3G1, Canada

Corresponding author: K. Naik (snaik@uwaterloo.ca)

ABSTRACT The dynamic nature of vehicular networks with their fast changing topology poses several challenges to setup communication between vehicles. Packet collisions are considered to be the main source of data loss in contention-based vehicular networks. Retransmission of collided packets is done several times until an acknowledgment of successful reception is received or the maximum number of retries is reached. The retransmission delay is drawn randomly from an interval, called the backoff interval. A good choice of the backoff interval reduces the number of collisions and the waiting periods of data packets, which increases the throughput and decreases the energy consumption. An optimal backoff interval could be obtained if global network information spread in the network in a short time. However, this is practically not achievable which motivates the efficient utilization of local information to approach the optimal performance. In this paper, we propose a localized adaptive strategy that calculates the backoff interval for unicast applications in vehicular networks. The new strategy uses fuzzy logic to adapt the backoff interval to the fast changing vehicular environment using only local information. We present four schemes of that strategy that differ in their behavior and the number of inputs. We compare the proposed schemes with other known schemes, binary exponential backoff, backoff algorithm, and an optimal scheme, in terms of throughput, fairness, and energy consumption. Results show that by proper tuning of the fuzzy parameters and rules, one of the proposed schemes outperform the other schemes, and approach the optimal results.

INDEX TERMS Vehicular network, backoff interval, fuzzy logic.

I. INTRODUCTION

The continual growth of transportation networks, combined with the recent advances in communication technologies, created the need for a wide variety of vehicular network applications [1]–[4]. These applications vary in the infrastructure and equipments needed and the quality of service (QoS) requirements. From the infrastructure perspective, as an example, location-based and map-based applications require a GPS device mounted on each vehicle. From the QoS perspective, safety applications [5], [6] are delay sensitive while broadcasting advertisements tolerate long delays. Vehicular network applications have been classified and categorized in [7] and [8]. While broadcast (one-to-many) and geocast (one-to-a-zone) applications have their strong presence in vehicular networking, such as in safety applications, unicast applications (one-to-one) are growing to be one of the key communication paradigms required for a number of VANET applications [9], especially in convenience and commercial applications. Examples of one-to-one vehicle applications include real time audio/video communication, instant message exchange, free flow tolling, parking availability notification, parking spot locator, remote vehicle diagnostics, and many others.

The growing demand for vehicular networking applications draws attention to the importance of resource allocation and the efficient use of the existing infrastructure [10], [11]. The application of an efficient medium access control (MAC) technique is crucial to achieve these goals. Fig. 1 shows a group of vehicles communicating with each other and with a road-side unit that acts as a gateway to the internet.

A. VEHICULAR MAC PROTOCOLS

In [12], the vehicular MAC protocols were classified into two categories: *contention-based* and *schedule-based*. In schedule-based MAC protocols [13]–[16], network users coordinate with each other, either using a central controller

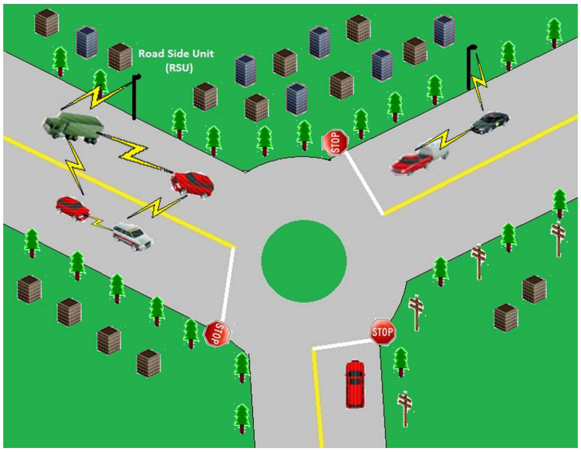

FIGURE 1. **Vehicles exchanging information with each other and with the road side unit.**

or in a distributed way, to allocate a transmission slot for each node. Schedule-based communication avoids collision between data packets. The main challenges in this approach are determining the collision free slots to be assigned, disseminating the assignment schedule to all the nodes in the network, synchronizing the nodes to the beginning of each slot, and adapting to changes in network loads and traffic topologies.

In contention-based methods, such as in [17] and [18], a random access scheme is implemented to regulate medium access by randomly delaying the transmissions, which reduces the probability of collisions, and hence, increases the network throughput. The random delay is bounded within an interval, known as the *backoff interval*. In general, a backoff scheme increases the backoff interval upon a failed transmission, and decreases it upon a successful transmission. Several backoff schemes were proposed to calculate the backoff interval [19]– [23]. These schemes implement a constant increment and decrement of the backoff interval which does not work well in a dynamic environment such as the vehicular networks. Long backoff intervals may lead to long idle periods of the medium, thereby reducing throughput.

In a dynamic environment, where network users frequently change their locations, their channel quality, and their connectivity to others, it is important to adapt the backoff interval to the changing network conditions. The backoff interval should map the availability of the channel bandwidth and should support its fair distribution among the network users. Network users should utilize the available network information to make a decision about the timing of the backoff interval. A good backoff scheme should have the following characteristics:

1) Localized: Uses only local information to decide about the length of the backoff interval. No global information should be needed.
2) Distributive: Does not use a central controller to manage the network.
3) Adaptive: Adapts to changes in network conditions, such as traffic load, speed, and network topology.

Examples of locally measured data at a node are number of successful and failed transmissions for the considered node, and the last backoff interval used. There is no exact model that maps this local information to a backoff interval. This motivated us to consider building a fuzzy model that uses logical rules to map the information to a backoff interval, as explained in Section III.

B. OUR CONTRIBUTION

In this paper, we propose an adaptive localized strategy based on fuzzy logic that uses only local information to generate a backoff interval. The strategy is adaptive in the sense that the backoff interval changes with the changing network conditions such as the traffic load, the interference levels, the network topology. Our adaptive strategy is based on human rules generated by a field expert which provides it a simplicity in interpretation and easiness to update. We created four schemes to show the sensitivity of fuzzy systems to its parameters. Three of the proposed schemes take one input ($Fuzzy - 1D$), and the fourth one takes two inputs ($Fuzzy - 2D$). The $Fuzzy - 1D$ schemes differ in their behavior, which is how the input is mapped to the output. The proposed schemes are:

- $Fuzzy - 1DS$: A selfish scheme, where the objective of each user is to capture the network channel regardless of its limited share.
- $Fuzzy - 1DG$: A generous scheme, where each user tends to give the channel to others if it finds the channel busy.
- $Fuzzy - 1DC$: A cautious scheme, where the objective of each user is to acquire, but not to exceed, its fair share of the channel.
- $Fuzzy - 2D$: A two-dimensional (2D) scheme that takes two inputs to improve the calculation of the backoff interval. Its objective is to distribute the channel fairly among the users while maximizing the network throughput. In this scheme, we combined the behavior of the three $Fuzzy - 1D$ schemes, as will be explained later, to improve on their performance.

The inputs to the proposed schemes are measured locally by every node, and they are:

- *Success Ratio* (S): The ratio of delivered packets to the total generated packets.
- *Recently-used Backoff Interval* (B_{RU}): The backoff interval used in the last channel access. This input is used in the $Fuzzy - 2D$ scheme only.

The objective of this work is to maximize the network throughput, minimize energy consumption, and improve fairness among the users. The proposed schemes are compared with known backoff schemes, mainly the binary exponential backoff (BEB) [24]– [26], the sensing backoff algorithm (SBA) [19], and an optimal scheme which requires the knowledge of the total number of nodes in the network [19].

C. SUMMARY OF ADVANTAGES

We can summarize the advantages of the proposed schemes as follows:

- Achieving throughput that is near to the optimal throughput expected, without overloading the network or the nodes' buffers with additional data exchange (as in the optimal scheme).
- Supporting the dynamic nature of vehicular networks by using only locally measured data.
- Enhancing the fairness between the network users by fairly distributing the bandwidth among them.
- Supporting green communication by reducing the number of attempts to send one packet without significant loss in throughput.

The rest of the paper is organized as follows. Section II presents an overview and a comparison of the backoff schemes presented for random access protocols. Section III explains the basics of Fuzzy logic with the aid of a numerical example. The fuzzy-based backoff schemes are presented in Section IV. Simulation results and performance comparisons are discussed in Section V. Finally, we state our conclusions and future work in section VI.

II. LITERATURE REVIEW

Backoff schemes are addressed in many research works. The role of a backoff algorithm is to avoid collisions of data packets between multiple users accessing the network, by delaying each user a random interval of time. This random delay is drawn uniformly from an interval known as the *backoff interval*. A backoff interval is increased or decreased according to the changing network conditions. Backoff algorithms differ in the amount of change they propose to the backoff interval. The more adaptive is the change, the better becomes the performance. In the following subsections we present three distributed backoff schemes and an optimal centralized scheme. After that, we finalize the section with a comparison and motivation for an adaptive scheme.

A. BINARY EXPONENTIAL BACKOFF (BEB)

Binary exponential backoff (*BEB*) is a simple and a widely-used algorithm in many MAC protocols [24]–[26]. In *BEB*, each node doubles the backoff interval up to a maximum backoff interval B_{max} after a collision occurs, and decreases the backoff interval to the minimum value B_{min} after a successful transmission. *BEB* can be summarized by the following set of equations:

$$\begin{cases} x \leftarrow \min(2x, B_{\max}), & \text{upon collision} \\ x \leftarrow B_{min}, & \text{upon successful transmission} \end{cases}$$

where x is the backoff interval from which a random delay d is drawn ($d \in [B_{min}, x]$). B_{min} and B_{max} are the minimum and the maximum backoff intervals, respectively. Their values are set in advance and do not change during the network operation. A major advantage of the *BEB* algorithm is its simplicity and good overall network throughput. However, a drawback of this scheme is the poor fairness in distributing the channel bandwidth among the different nodes of the network [27], [28]. A node with a successful transmission decreases its backoff interval to the minimum, which increases its chance of regaining access to the shared medium. On the other hand, a node with a failed transmission doubles its backoff interval, decreasing its chance to regain access to the medium. With high probability, the first node will continue to gain access to the medium, while the second node is deprived of accessing the medium. This explains the high-throughput achievement of this scheme.

B. MULTIPLICATIVE INCREASE LINEAR DECREASE (MILD)

The multiplicative increase linear decrease (*MILD*) algorithm is proposed in [27] to improve the fairness of the *BEB* algorithm. The contributions can be summarized in the following points:

1) Increasing the backoff interval by a factor of 1.5 (instead of 2 as in *BEB*) in the case of collision.
2) Gradual decrease of the backoff interval in case of successful transmission
3) Transferring the backoff interval with the data packet to other nodes

The following equations summarize the *MILD* scheme

$$\begin{cases} x \leftarrow \min(1.5x, B_{\max}), & \text{upon collision} \\ x \leftarrow x_{packet}, & \text{upon overhearing success} \\ x \leftarrow \max(x-1, B_{min}), & \text{upon successful transmission} \end{cases}$$

where x_{packet} is the backoff interval value included in the successfully transmitted packet. The step increase is decreased to 1.5 (instead of 2 in *BEB*). Upon successful transmission, the transmitting node decreases its backoff interval by one step, where a step is the transmission time of the request packet, *request-to-send (RTS)*. In addition, the receiving node of a successful transmission copies the backoff interval of the received packet to be used for its packets. This copy mechanism is based on the assumption that backoff intervals in an area indicate the contention levels in that area, and therefore, copying the backoff is a good means for sharing knowledge about the environment. The fairness performance is greatly improved in the *MILD* scheme using the copy mechanism. However, there are two main drawbacks for this scheme:

1) The overhead of storing the backoff interval in the packet header, which increases the packet length and, therefore, the probability of collisions.
2) The migration effect [19]: The backoff interval, which is assumed to indicate the contention level in a certain area, may travel to a different area where the contention level is different. This may lead some nodes to delay for intervals longer than required, leading to degraded throughput.

C. SENSING BACKOFF ALGORITHM (SBA)

The sensing backoff algorithm is proposed in [19] to address the drawbacks of the *MILD* algorithm. The main objective is to improve the fairness of the *BEB*, as in *MILD*, while avoiding the overhead and the migration problems. The contribution of *SBA* is in finding the optimal step increase and decrease. The following set of equations represent the *SBA* scheme:

$$\begin{cases} x \leftarrow \min(c_1.x, B_{max}), & \text{upon collision at transmitter} \\ x \leftarrow \max(x - c_2.c_3, B_{min}), & \text{upon overhearing success} \\ & \text{at neighbors} \\ x \leftarrow \max(x - 1, B_{min}), & \text{upon successful transmission} \\ & \text{at transmitter and receiver} \end{cases}$$

The step increase and decrease are predetermined by analytical analysis. According to [19], the authors find the optimal values for c_1, c_2 and c_3 to be (1.2, 0.8, 0.93) respectively. The *SBA* algorithm is found to outperform both the *BEB* and the *MILD* schemes in large networks in terms of throughput and fairness.

D. THE OPTIMAL BACKOFF (OPT)

In [19], an analytical study was conducted to find the optimal backoff interval which maximizes the network throughput, as a function of the number of nodes in the network, N. It was proved that the optimal backoff interval, B_{opt}, as a function of N is given by

$$B_{opt}(N) = 4N\gamma$$

where γ is the transmission time of a data packet. This result is found under the following assumptions:

- All the nodes are identical, in line-of-sight of each other and in the range of each other.
- Packet collisions are the only source of packet error.
- A successful transmission can be heard by all nodes. However, collisions can only be noticed by the packet transmitter, by means of lack of acknowledgment from its intended receiver.
- The backoff scheme is implemented in the pure ALOHA MAC protocol.
- A busy node will not process new packets until it successfully transmits the current packet. No packet preemption is allowed.
- All data packets are of the same size. The propagation delays are negligible.

The Optimal scheme requires the knowledge of the total number of nodes in the network to achieve the maximum throughput, which is not practically applicable in a dynamic environment such as a vehicular network. It is proved in [19] that the *OPT* algorithm achieves the maximum throughput when implemented in the pure ALOHA protocol. We use the *OPT* scheme performance results as a benchmark to compare with the results of the other schemes.

E. SUMMARY OF BACKOFF SCHEMES

From the previous discussion, it can be noticed that all the non-optimal backoff schemes decrease the backoff interval at the transmitter upon successful transmission and increase it upon collision. They differ in the amount of increase and decrease. Table 1 shows a summary of this comparison.

An important design issue is to adapt the changes in the backoff interval, whether increase or decrease, to the network conditions [29], such as the traffic load in the network, and the opportunity of a successful transmission. These conditions change with time. Therefore, in order to be considered in the backoff calculation, they should be measured from time to time. In the discussed schemes, the parameters used for the backoff increase or decrease are constant, and therefore, cannot capture the dynamic network conditions. In addition, there is no known function that can precisely map measurements of the network conditions to a change in the backoff interval of the nodes. As shown in the literature review, all practically implemented functions are heuristics. Therefore, we implemented a fuzzy inference system that will utilize locally measured data and map it to a change in the backoff interval.

TABLE 1. Comparison of the backoff schemes.

Transmission Status		BEB	$MILD$	SBA [1]
Successful	Transmitter	B_{min}	- 1	x 0.98
	Receiver	NA	$B_{transmitter}$	x 0.98
	Neighbors	NA	NA	- 0.8
Collision	Transmitter	x 2	x 1.5	x 1.2
	Receiver	NA	NA	NA
	Neighbors	NA	NA	NA
Drawbacks	Fairness	Poor	Good	Good
	Migration	No	Yes	No
	Constant change	Yes	Yes	Yes

III. FUZZY LOGIC

Fuzzy Logic (FL) [30]–[33] is a means of handling inexact information to get an exact output using natural human rules. These rules describe the behavior of the system that maps the inputs to the outputs, and therefore, should be set by an expert in the field to model the system under examination. The rules take the form of *if* − *then* statements. The fuzzy system is represented in the block diagram in Fig. 2.

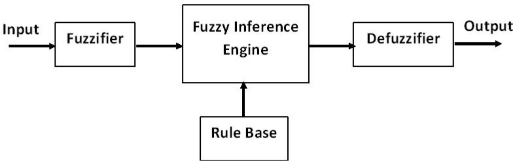

FIGURE 2. Fuzzy inference system block diagram.

Fuzzy logic has been used in wireless communication protocols and applications such as in admission control [34], [35], vehicular networks security [36], routing [37], [38], and MAC protocols [29], [39]. Adaptive techniques based on fuzzy logic are increasingly being proposed to control network behavior [40], [41].

To solve a problem using fuzzy logic, the following steps should be followed:

1) Define the input and output variables of the system with their lower and upper boundaries.
2) Normalize the variables by dividing each variable by its valid range (upper-lower). Some variables may need to have a wider or negative range.

---

[1] We used the optimal values found in [19]. In general, they are preassigned values for the three parameters, which do not dynamically change with network conditions.

3) Assign fuzzy (non-numeric) values to each variable within its normalized range.
4) Define the mapping relation between the numeric and fuzzy values for each variable. This process is called *fuzzification*.
5) Define the rules that map the fuzzy input values to the fuzzy output values.
6) Once we have numeric values for the inputs, they are converted into fuzzy (non-numeric) values (fuzzified). Then one or more rules will be applied to give fuzzy output values.
7) The fuzzy output values are mapped to numeric values using a process called *defuzzification*, where a geometric operation is applied to the output fuzzy area to give single numeric values.

As an example, assume we have a fuzzy backoff scheme with two variables: one input and a single output. The input is the success ratio, S, which is the number of delivered packets to the total number of transmission attempts (successful and failed) within a time window W.

$$S = \frac{Number\ of\ delivered\ packets}{Number\ of\ transmission\ attempts} \tag{1}$$

The output is the normalized backoff interval, $\hat{B}$, which is the ratio of the backoff interval to its valid range.

$$\hat{B} = \frac{B - B_{\min}}{B_{\max} - B_{\min}}$$
$$S, \hat{B} \in [0, 1] \tag{2}$$

where B is the actual backoff interval used for the considered network user, B_{max} and B_{min} are the upper and lower boundaries, respectively.

In the fuzzy domain, we assign each variable three fuzzy (non-numeric) values.

$$S_F \in \{Low, Medium, High\}$$
$$\hat{B}_F \in \{Small, Medium, Large\}$$

where S_F and $\hat{B}_F$ are the corresponding fuzzy values of S and $\hat{B}$ in the fuzzy domain. Each value is represented by a membership function, as shown in Fig. 3, that maps a range of numeric values to a fuzzy value. Commonly used shapes for membership functions are: triangular, trapezoidal, and Gaussian. The numerical inputs should undergo a *Fuzzification* process in which the numeric values are mapped to one or more of the fuzzy values. The number of fuzzy values and their start and end parameters are arbitrary and application dependent. They are usually set by an expert and are subject to tuning by simulation and practical experiments. Assume that we set the rules ($R1$ to $R3$) that map the input, S_F, to the output, $\hat{B}_F$, as follows:

$R1$: IF S_F is *Low*, THEN $\hat{B}_F$ is *Large*
$R2$: IF S_F is *Medium*, THEN $\hat{B}_F$ is *Medium*
$R3$: IF S_F is *High*, THEN $\hat{B}_F$ is *Small*

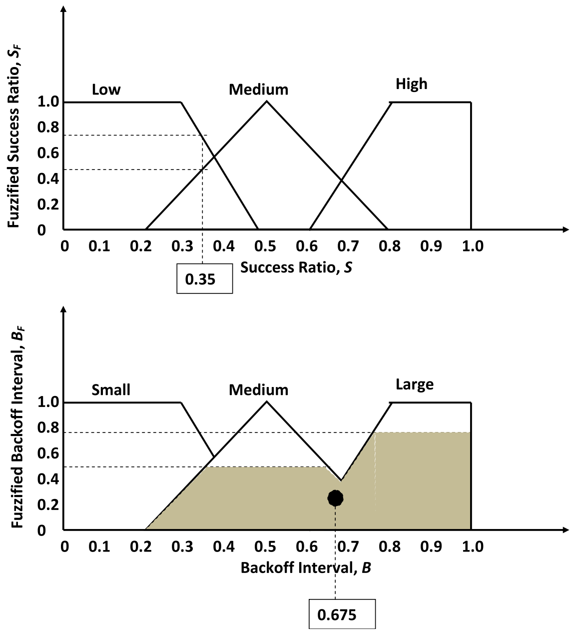

FIGURE 3. Membership functions for a fuzzy variable.

As a numeric example, assume we have a success ratio of 0.35. According to Fig. 3, it is mapped (fuzzified) to two fuzzy membership functions: *Low* with around 0.78 degree of belonging, and *Medium* with around 0.45 degree of belonging. The two fuzzified values of the success ratio are then used to fire one or more rules in the rule base. In our example, two rules are fired: $R1$ and $R2$. The two rules will result in the shaded area in two fuzzy values of the output (*Large* and *Medium*). To obtain a numeric value for the output, a *defuzzification* process is done in which a numeric value is chosen to represent the area shaded in the output function. Commonly used defuzzification functions include: mean, center of gravity (centroid), center of area (bisector), middle of maximum (MOM), and last of maximum (LOM). In our example, we choose the centroid, which is the most commonly used because it extracts most of the information in the output area. Using the centroid method, the output, which is the normalized backoff interval ($\hat{B}$), will be 0.675. To map $\hat{B}$ to the actual backoff value, we use Equation 2, to get $B = B_{min} + 0.675(B_{max} - B_{min})$.

IV. FUZZY LOGIC BASED BACKOFF-SCHEMES

We propose four schemes to calculate the backoff interval using fuzzy logic as an inference engine. Three of the four schemes use a single input, and the fourth one uses two inputs. A detailed explanation is provided in the following subsections.

A. ONE DIMENSIONAL FUZZY SCHEMES (Fuzzy − 1D)

These schemes take a single input, success ratio S, which is calculated, as in Equation 1, using locally counted acknowledged packets and total packets sent by the node under consideration. S is fuzzified into five fuzzy values:

$$S_F \in \{VeryLow(S_{vl}), Low(S_l), Medium(S_m), High(S_h), VeryHigh(S_{vh})\}$$

A low success ratio of a node means that there is a high probability of collision, which is most probably because of many users accessing the channel in a small time. In other words, the bandwidth is not sufficient for the number of users sharing the channel. Therefore, a recommended decision is to increase the backoff interval. A high success ratio of a node means that this node has many of its sent packets delivered successfully, which may be because the available bandwidth is more than that requested by the network users. A recommended decision may be to decrease the backoff interval or keep it the same according to the behavior of the scheme.

The output is the change (amount of increase or decrease) in backoff interval, dB. The fuzzy values for this variable are

$$dB_F \in \{HighNegative(dB_{hn}), SmallNegative(dB_{sn}), Zero(dB_0), SmallPositive(dB_{sp}), HighPositive(dB_{hp})\}$$

The change in backoff interval is defuzzified to have a numeric value in the range $[-1, 1]$, to represent the increase (positive range) and decrease (negative range). It is then multiplied by the recently-used interval to reflect the actual change in the backoff interval. In mathematical form:

$$dB_{FIS} = dB * B_{RU}$$

where dB is the numeric (defuzzified) value of dB_F and dB_{FIS} is the final output of the fuzzy inference system.

The new backoff interval B is obtained by adding dB_{FIS} to the recently-used backoff interval, B_{RU}, as follows:

$$B = dB_{FIS} + B_{RU}$$

Fig. 4 shows a block diagram representing the one-dimensional schemes.

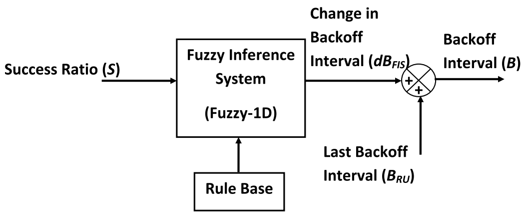

FIGURE 4. A block diagram representing the *Fuzzy* $-$ 1*D* backoff schemes.

We developed three schemes that differ in their behavior (rules mapping the input to the output). We set five rules in the rule set for each scheme. More rules requires finer tuning of the input/output mapping, and may lead to degraded performance if not tuned well. According to several experiments we conducted, we found five rules to perform well. However, extensive work should be done to verify the optimal number of rules required.

1) SELFISH FUZZY SCHEME (*FUZZY* $-$ 1*DS*)

In the selfish scheme, the goal of each network user is to capture the channel, regardless of the needs of others to use the channel. To represent this behavior, we set the rules as shown in Table 2, where dB_{S-F} instantiate dB_F.

TABLE 2. The fuzzy rule base for *fuzzy* $-$ 1*DS*.

$Rule$	S_F	dB_{S-F}
$R1$	S_{vl}	dB_{hp}
$R2$	S_l	dB_{sp}
$R3$	S_m	dB_0
$R4$	S_h	dB_{sn}
$R5$	S_{vh}	dB_{hn}

We defined the fuzzy rules according to the behavior of each scheme. The selfish scheme increases the backoff interval in case of low success ratio, which is an indicator of a busy channel. However, in case of high success ratio, it highly decreases the backoff interval regardless of the other's share in the bandwidth. This may lead to monopolizing the channel by the successful users.

2) GENEROUS FUZZY SCHEME (*FUZZY* $-$ 1*DG*)

In the generous scheme, each network user channel releases the channel to other users once it finds it busy. Therefore, it has no decreasing change in the backoff interval. Table 3 shows the rules we set for this scheme, where dB_{G-F} instantiate dB_F. In the generous scheme, each user increases the backoff interval when the success ratio is found to be low, assuming it is because of a busy channel. In the case of medium or higher success ratio, the scheme becomes satisfied with the achieved success ratio and does not decrease the backoff interval.

TABLE 3. The fuzzy rule base for *fuzzy* $-$ 1*DG*.

$Rule$	S_F	dB_{G-F}
$R1$	S_{vl}	dB_{hp}
$R2$	S_l	dB_{sp}
$R3$	S_m	dB_0
$R4$	S_h	dB_0
$R5$	S_{vh}	dB_0

3) CAUTIOUS FUZZY SCHEME (*FUZZY* $-$ 1*DC*)

In the cautious scheme, the goal of each network user is to avoid high jumps in the backoff interval. It avoids high increase when the channel is busy and high decrease when it assumes the channel is free. To represent this behavior, we set the rules as shown in Table 4, where dB_{C-F} instantiate dB_F. In the cautious scheme, users tend to avoid big changes to their backoff interval. If the channel is busy, they slightly increase their backoff interval. If the success ratio is high, they are satisfied. If it is low, they slightly decrease the backoff interval. This scheme provides a slow reaction to the network changes.

TABLE 4. The fuzzy rule base for *fuzzy* − 1*DC*.

$Rule$	S_F	dB_{C-F}
$R1$	S_{vl}	dB_{sp}
$R2$	S_l	dB_{sp}
$R3$	S_m	dB_{sp}
$R4$	S_h	dB_0
$R5$	S_{vh}	dB_{sn}

B. TWO DIMENSIONAL FUZZY SCHEMES (Fuzzy − 2D)

To improve the calculation of the backoff interval, we added the normalized recently-used backoff interval, $\hat{B}_{RU}$, as a second variable to the inference engine. The recently-used backoff interval (B_{RU}) is the one used by a node for the last transmitted packet. The backoff is normalized as shown in Equation 3.

$$\hat{B}_{RU} = \frac{B_{RU} - B_{\min}}{B_{\max} - B_{\min}} \tag{3}$$

The fuzzy-domain variable, $\hat{B}_{RU-F}$ corresponding to the real-domain $\hat{B}_{RU}$ is assigned three values as follows:

$$\hat{B}_{RU-F} \in \{Low(B_l), Medium(B_m), High(B_h)\}$$

This variable is an indicator of the last state of the traffic load in the network. A large backoff interval indicates a crowded channel and a medium or small backoff interval indicates a medium or low traffic load, respectively. For example, a decision may be taken to increase the backoff interval if the recently-used backoff is small or medium and the success ratio is low, while, for the same success ratio, the backoff interval may be kept with minimal or no change if the recently-used backoff interval is high.

Fig. 5 shows a block diagram of the two-dimensional scheme. Table 5 shows the rule set for this scheme.

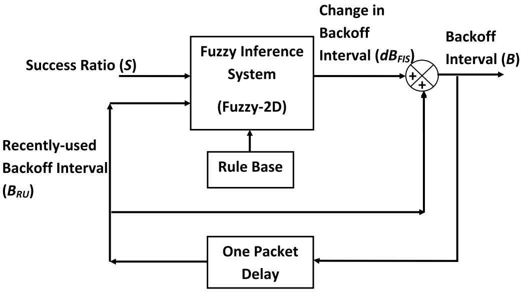

FIGURE 5. Block diagram showing the inputs and the output of the *Fuzzy* − 2*D* backoff scheme.

The following general formula applies to all the rules:

$$IF\ Input1\ is\ S_F\ AND\ Input2\ is\ \hat{B}_{RU-F}$$
$$THEN\ Output\ is\ dB_F$$

For example, $R1$ states that if the success ratio, S_F, is very low, S_{vl}, and the recently-used backoff, B_{RU-F}, is low, B_l, then the required change in backoff, dB_F, is high positive (a large increase). The interpretation of rule $R1$ is that the node has an unsuccessful history in transmissions which may be because of its small backoff interval. Therefore, it is recommended that the node increases its backoff interval by a large amount and try again.

TABLE 5. The fuzzy rule base for *fuzzy* − 2*D*.

$Rule$	S_F	$\hat{B}_{RU-F}$	dB_F
$R1$	S_{vl}	B_l	dB_{hp}
$R2$	S_{vl}	B_m	dB_{hp}
$R3$	S_{vl}	B_h	dB_{sp}
$R4$	S_l	B_l	dB_{hp}
$R5$	S_l	B_m	dB_{sp}
$R6$	S_l	B_h	dB_{sp}
$R7$	S_m	B_l	dB_{sp}
$R8$	S_m	B_m	dB_{sn}
$R9$	S_m	B_h	dB_{sn}
$R10$	S_h	B_l	dB_0
$R11$	S_h	B_m	dB_{sn}
$R12$	S_h	B_h	dB_{hn}
$R13$	S_{vh}	B_l	dB_0
$R14$	S_{vh}	B_m	dB_0
$R15$	S_vh	B_h	dB_{sn}

V. SIMULATION AND RESULTS

We conduct three different sets of experiments to compare the performance of the different backoff schemes: *BEB*, *SBA*, *OPT*, *Fuzzy* − 1*DS*, *Fuzzy* − 1*DG*, *Fuzzy* − 1*DC*, and *Fuzzy* − 2*D*. The first set has been published before in [42], which represents the short version of this work, while the other two are new. Each point in the figures is the average of ten simulation runs. We use the throughput per packet-time, instead of per second, to evaluate the throughput of each scheme, because it is known that the optimal throughput of pure ALOHA is 0.186 data packets per one packet-time. The packet time is the time expected to send one data packet and receive acknowledgment assuming no collisions. Throughput is represented, mathematically, as follows:

$$Thr = \frac{\sum_{i=1}^{N} p_i}{T_s} \times \frac{(D + ACK)}{R} \tag{4}$$

where p_i is the total number of delivered packets for node i during the simulation time, N is the number of nodes in the network, T_s is the simulation time, D and ACK are the sizes of one data packet and one ACK packet, respectively, in bytes, and R is the transmission rate in bytes per second. To evaluate fairness, we use *Jain's Fairness Index (JFX)* [43], [44].

$$JFX = \frac{(\sum_{i=1}^{N} p_i)^2}{N \sum_{i=1}^{N} p_i^2} \tag{5}$$

which evaluates the distribution of throughput in terms of delivered packets among the network users. In addition, we compare the average backoff interval and the number of attempts for each delivered packet as an indicator of energy consumption during transmission. The first set of experiments were conducted with packet collisions as the only source of failure. Another set of experiments is performed with varying channel quality for each node beside the packet collision.

A. NETWORK SETUP

We consider a network of N nodes connected to each other via wireless links, where N varies from 25 to 250. Nodes are distributed randomly in an area of $1500 \times 1500\ m^2$, and have an infinite backlog of constant-size packets. In order to compare with the *OPT* scheme, we used the same assumptions listed in [19]. The parameters for the network and the access protocol are listed in Table 6.

TABLE 6. Network and protocol parameters.

Simulation Parameter	Value
Network Area, A	$1500 \times 1500\ m^2$
Simulation time, T_s	50 $seconds$
Data packet size, D	1072 $bytes$
ACK packet size, ACK	304 $bytes$
Transmission rate, R	1 $Mbps$
Minimum Backoff Interval, B_{min}	3
Maximum Backoff Interval, B_{max}	1023

In order to have an updated and accurate estimate of the channel, we calculate the success ratio for the last ten transmission attempts. The success ratio is the ratio of successful attempts to the total attempts.

Impact of Mobility: Although not mentioned explicitly, impact of vehicles' mobility is considered in simulations. Because of the open-area nature of the network, it is expected for each vehicle to have a line-of-sight circle of at least 300m radius. If we assume that vehicles move randomly in the network area with a relative speed of 60mph (25m/s), then the connectivity between two vehicles could last for at most 600/25 = 24 sec. After the 24 seconds, some vehicles will leave the network area, and other new ones will join the network. Given these estimated values, we can consider the network to look as static for around 24 seconds, because the set of nodes does not change. After 24 seconds, a new set of nodes (some from the old set plus some new) is in the network. Therefore, any protocol that keeps a history of more than 24 seconds will not function properly. In our scheme, we keep a history of ten transmission attempts. Each attempt could take a maximum of around one second, assuming the maximum backoff interval used and the parameter values in Table 6. In conclusion, our scheme assumes the network to be static for at most ten seconds, which is far below the maximum estimated time for the network to look as static.

B. FUZZY SYSTEM SETUP

The membership functions have a triangular shape for all the inputs and the output. A triangular shape is represented by three points (a_0, x_0, a_1), where a_0 and a_1 are the leftmost and rightmost points, respectively, of the triangle base, and x_0 is the triangle head point on the horizontal axis. The head point has the value of 1 on the vertical axis. The parameters used for the fuzzy inference systems are listed in Table 7.

After applying the inputs to the fuzzy system, one or more rules fire. Each rule gives an area of possibility for the output. The areas of all the firing rules are aggregated. Then the output is extracted by calculating the centroid of the aggregated area.

TABLE 7. The fuzzy system parameters.

Variable	Fuzzy value	a_0	x_0	a_1
S_F	Very Low	0	0	1
	Low	0	0.3	0.6
	Medium	0	0.5	1
	High	0.4	0.7	1
	Very High	0	1	1
B_{RU-F}	Low	0	0	1
	Medium	0	0.5	1
	High	0	1	1
dB_F	High Negative	-1.0	-1.0	-0.33
	Low Negative	-0.67	-0.33	0
	Zero	-0.33	0	0.33
	Positive	0	0.33	0.67
	High Positive	0.33	1.0	1.0

C. SIMULATION RESULTS

The first set of experiments were conducted by assuming that packet collisions are the only source of packet failure. We assume that the channel quality is good for all nodes. To represent the channel quality in the simulations, we use a packet failure probability, $\alpha \in [0, 1]$. The first set of experiments are conducted with $\alpha = 0$. In the next set of experiments, we vary α to account for other sources of packet failure.

1) CONSTANT CHANNEL QUALITY, $\alpha = 0$

Fig. 6 shows the average backoff interval achieved by all schemes. The *OPT* scheme has a linear backoff with the number of nodes. The *SBA* scheme starts at a near optimal value for small number of nodes, and then gets away with a smaller value at large number of nodes. The $Fuzzy - 1D$ schemes appear to have been trapped in high backoff interval values regardless of the number of nodes. The $Fuzzy - 2D$ approach provides an increasing backoff interval with the number of nodes and then settles at a large value when the number of nodes is large.

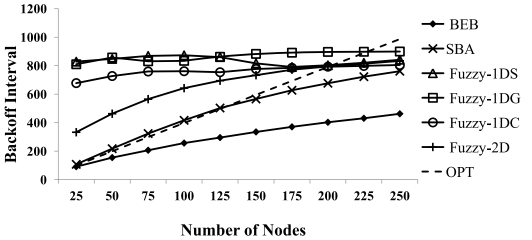

FIGURE 6. Comparison of the backoff interval for all the schemes.

As known for the pure ALOHA protocol, the maximum throughput that can be achieved is 0.186 data packets per one packet time. This throughput is achieved for networks with large number of nodes (theoretically infinity) when the packet transmission rate is double the packet generation (or arrival) rate. Fig. 7 shows the throughput for all schemes. It can be noticed that all the fuzzy schemes approach the optimal throughput for large number of nodes. The $Fuzzy - 1DS$ scheme achieves higher throughput than all other schemes for

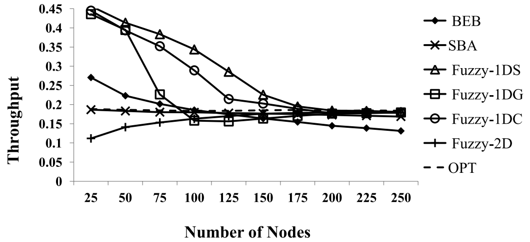

FIGURE 7. Throughput of the backoff schemes.

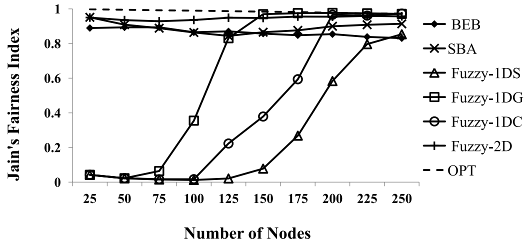

FIGURE 8. Fairness index of the backoff schemes.

small number of nodes in the network. This can be explained using the fairness figure, Fig. 8, where the $Fuzzy - 1DS$ scheme is shown to have the worst fairness. This means that it achieves high throughput by allowing some nodes to monopolize the network bandwidth, while the other nodes are starving.

It can also be noticed in the fairness figure, Fig. 8, that $Fuzzy-2D$ provides the fairest distribution of the bandwidth among all the other protocols, even at small number of nodes. This can be explained using the backoff and throughput figures, Figs. 6 and 7. From the backoff figure, Fig. 6, it is shown that $Fuzzy - 2D$ provides a backoff interval which is a little higher but almost parallel to the *OPT* scheme, which decreases the overall throughput a little lower than the optimal, as shown in Fig. 7. The $Fuzzy-2D$ scheme is considered to be the best one among the non-optimal schemes.

Fig. 9 can further explain the performance of the different schemes. The figure displays the ratio of failed packet transmissions to the total transmission attempts (successful+failed). The good performance of the Fuzzy schemes can be explained by their control of the transmission attempts, while *BEB* and *SBA* schemes attempt to access the medium blindly, thereby increasing the rate of collisions.

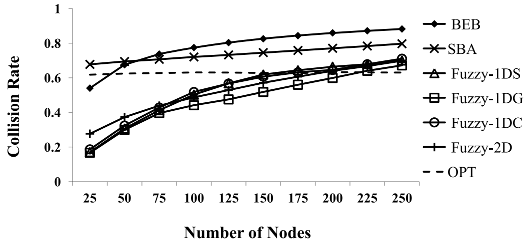

FIGURE 9. Number of collisions to the total transmission attempts.

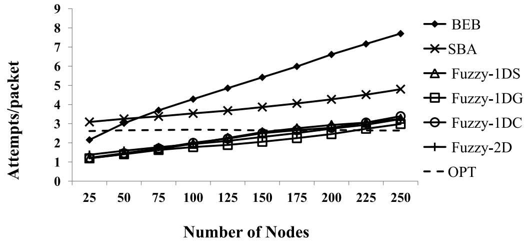

FIGURE 10. Average number of transmission attempts per received packed.

In communication networks, a major source of energy consumption is packet transmissions and reception [45], [46]. By decreasing the number of transmissions and avoiding redundant transmissions, we can reduce energy consumption. Therefore, one of the objectives of this work is to control the number of transmissions and reduce it as much as possible without sacrificing the high throughput. Fig. 10 shows the number of attempts for each delivered packet. *BEB* and *SBA* protocols were found to have the highest energy consumption because of their short positive change in the backoff interval when the channel is busy which causes the highest collision rate, as shown in Fig. 9. Protocols with high positive jumps have a lower energy consumption because they quickly escape the collision period.

2) DEGRADING CHANNEL QUALITY, α

In the next set of experiments, we fixed the number of nodes $N = 100$, and introduced the channel quality as a new variable. Channel quality is represented using the packet failure probability variable, α. When $\alpha = 0$, then packet collisions are the only source of packet failure. When $\alpha > 0$, then each node may have a bad channel quality that contributes to the packet failure together with packet collisions. The *OPT* scheme is no longer optimal because it assumes collisions to be the only source of failure. However, we continue showing it to be consistent with the previous experiments.

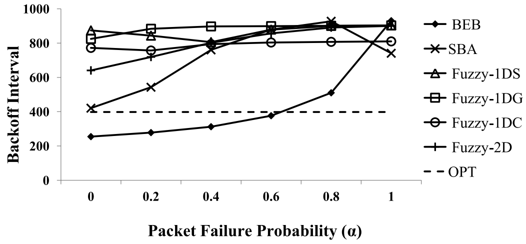

FIGURE 11. Comparison of the backoff interval for all the schemes.

Fig. 11 shows the average backoff interval for all the schemes under different failure rates. All schemes, except for the *OPT* increase the backoff interval with the increased failure probability assuming that it is because of packet collisions. The increase stops when the maximum backoff is reached. The fuzzy schemes tend to assign high backoff intervals even for small failure probabilities.

As expected, the throughput decreases for all the schemes with the increasing failure probability, as shown in Fig. 12. The decrease tends to be linear with the probability failure.

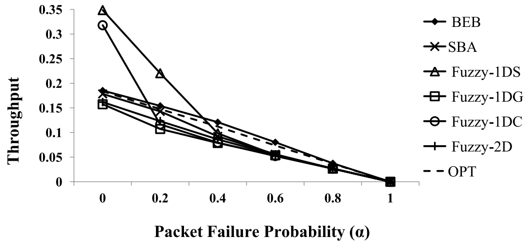

FIGURE 12. Throughput of the backoff schemes.

The worse the channel becomes, the fairest is the throughput distribution between the competing users. As shown in Fig. 13, all the schemes approach the highest fairness index, except for the selfish scheme, $Fuzzy-1DS$, which favors the nodes capturing the channel to keep in its state while the other nodes are starving.

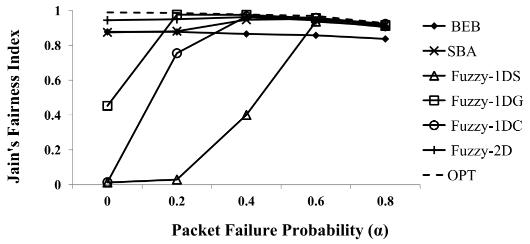

FIGURE 13. Fairness index of the backoff schemes.

Because of the increased packet failure, more attempts are needed to deliver a packet, and more energy is consumed. Fig. 14 shows an exponential increase in the number of attempts for each delivered packet. Schemes which assign lower backoff intervals suffer from higher number of attempts.

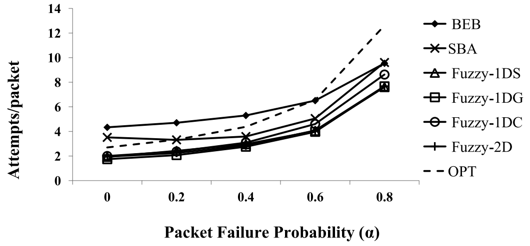

FIGURE 14. Average number of transmission attempts per received packed.

3) CHANGING FUZZY SYSTEM PARAMETERS

In the next set of experiments, we compare two versions of $Fuzzy-2D$ with different fuzzy system parameters. The new system parameters are to affect the membership functions (MFs) of the input variables. The previous version of $Fuzzy-2D$ was assigned a wide range for its MFs. We will call it $Fuzzy-2Dw$. The following set of experiments compare it with a narrow MFs $Fuzzy-2D$, which will be called $Fuzzy-2Dn$. The parameters for the new version is in Table 8.

TABLE 8. The $fuzzy-2Dn$ system parameters.

Variable	Fuzzy value	a_0	x_0	a_1
S	Very Low	0	0	0.34
	Low	0.1667	0.334	0.5
	Medium	0.334	0.5	0.667
	High	0.5	0.667	0.833
	Very High	0.667	1	1
B_{RU}	Low	0	0	0.35
	Medium	0.25	0.5	0.75
	High	0.65	1	1
dB	High Negative	-1.0	-1.0	-0.33
	Low Negative	-0.67	-0.33	0
	Zero	-0.33	0	0.33
	Positive	0	0.33	0.67
	High Positive	0.33	1.0	1.0

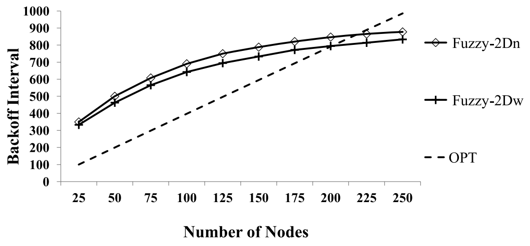

FIGURE 15. Comparison of the backoff Interval for all the schemes.

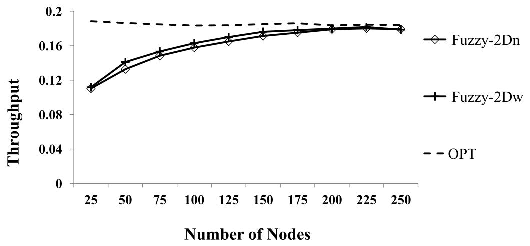

FIGURE 16. Throughput of the backoff schemes.

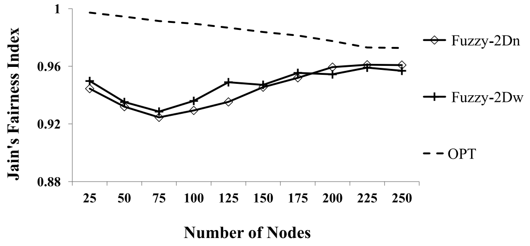

FIGURE 17. Fairness index of the backoff schemes.

Fig. 15 shows a slight increase in the backoff interval for the narrow MFs version, $Fuzzy-2Dn$, but with the same behavior. The increased backoff results in a little lower throughput, as shown in Fig. 16, and around the same fairness (Fig. 17). A higher backoff usually leads to lower collision

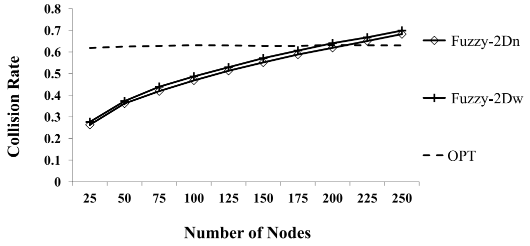

FIGURE 18. **Number of collisions to the total transmission attempts.**

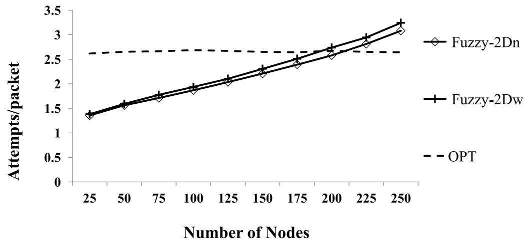

FIGURE 19. **Average number of transmission attempts per received packed.**

rate (Fig. 18), and therefore, lower attempts per delivered packets (Fig. 19).

VI. CONCLUSION

In contention-based vehicular networks, a wireless channel is shared among many users. Packet collisions occur due to multiple uncoordinated access to the shared medium. Backoff schemes provide a simple solution to decrease collisions, and therefore increase throughput and fairness. A number of backoff schemes were previously proposed to improve the calculation of backoff intervals. In this paper, we studied and compared the *BEB*, *SBA*, and *OPT* backoff schemes. The *OPT* scheme requires the knowledge of the number of nodes in the network which is impractical in dynamic networks such as vehicular networks. All the other schemes increase the backoff interval in the case of collisions and decrease it in case of successful transmissions. They differ in the amount of increase and decrease. However, they all assign a constant change which does not perform well with the dynamic network conditions.

We proposed backoff schemes that adapt the backoff interval according to locally measured network parameters: packet success ratio (measured at each node), and the recently-used backoff interval. The schemes use a fuzzy inference system to find the backoff interval based on a set of fuzzy rules. We simulated the proposed schemes and the previously proposed schemes using ALOHA as a random access protocol. Simulation results show that the fuzzy schemes behave differently according to their rules. The scheme, $Fuzzy - 2D$, was found to achieve better performance in terms of fairness and energy consumption than the other schemes, and approaches the optimal throughput. Moreover, the fuzzy schemes do not incur any communication overhead because of their localized nature which avoids exchange of information, and therefore saves energy consumed in transmissions and receptions.

REFERENCES

[1] H. T. Cheng, H. Shan, and W. Zhuang, "Infotainment and road safety service support in vehicular networking: From a communication perspective," *Mech. Syst. Signal Process.*, vol. 25, no. 6, pp. 2020–2038, Aug. 2011.

[2] J. Ott and D. Kutscher, "Drive-thru internet: IEEE 802.11b for 'automobile' users," in *Proc. 23rd Annu. Joint Conf. IEEE Comput. Commun. Soc.*, vol. 1. Mar. 2004, p. 373.

[3] A. N. S. Das, B. Zhou, G. Pau, and M. Gerla, "AdTorrent: Digital billboards for vehicular networks," in *Proc. IEEE/ACM Int. Workshop Vehicleto-Veh. Commun.*, Jul. 2005, pp. 1–20.

[4] H. Hartenstein, B. Bochow, A. Ebner, M. Lott, M. Radimirsch, and D. Vollmer, "Position-aware ad hoc wireless networks for intervehicle communications: The fleetnet project," in *Proc. 2nd ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, New York, NY, USA, 2001, pp. 259–262.

[5] T. ElBatt, S. K. Goel, G. Holland, H. Krishnan, and J. Parikh, "Cooperative collision warning using dedicated short range wireless communications," in *Proc. 3rd Int. Workshop Veh. Ad Hoc Netw.*, New York, NY, USA, 2006, pp. 1–9.

[6] C. Maihöfer and R. Eberhardt, "Time-stable geocast for ad hoc networks and its application with virtual warning signs," *Comput. Commun.*, vol. 27, no. 11, pp. 1065–1075, 2004.

[7] F. Bai, H. Krishnan, V. Sadekar, G. Holl, and T. Elbatt, "Towards characterizing and classifying communication-based automotive applications from a wireless networking perspective," in *Proc. IEEE Workshop Autom. Netw. Appl.*, Dec. 2006, pp. 1–25.

[8] F. D. Da Cunha, L. A. Vilas, A. C. Viana, and A. A. F. Loureiro, "Data communication in VANETs: A Survey, challenges and applications," Dept. Comput. Sci., Federal Univ. Minas Gerais, Minas Gerais, Brazil, Tech. Rep., Jan. 2014.

[9] M. Boban, O. K. Tonguz, and J. Barros, "Unicast communication in vehicular ad hoc networks: A reality check," *IEEE Commun. Lett.*, vol. 13, no. 12, pp. 995–997, Dec. 2009.

[10] The intelligent transportation systems society of Canada, Toronto, ON, Canada. *Intelligent Transportation Systems* [Online]. Available: http://www.itscanada.ca

[11] S. Zeadally, R. Hunt, Y.-S. Chen, A. Irwin, and A. Hassan, "Vehicular ad hoc networks (VANETS): Status, results, and challenges," *Telecommun. Syst.*, vol. 50, no. 4, pp. 217–241, 2012.

[12] F. Yu and S. Biswas, "Self-configuring TDMA protocols for enhancing vehicle safety with DSRC based vehicle-to-vehicle communications," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 8, pp. 1526–1537, Oct. 2007.

[13] L. H. van, T. Nieberg, H. Kip, and P. Havinga, "Advantages of a TDMA based, energy-efficient, self-organizing MAC protocol for WSNs," in *Proc. IEEE 59th Veh. Technol. Conf.*, vol. 3. May 2004, pp. 1598–1602.

[14] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves, "Energy-efficient collision-free medium access control for wireless sensor networks," in *Proc. 1st Int. Conf. Embedded Netw. Sensor Syst.*, New York, NY, USA, 2003, pp. 181–192.

[15] T.-K. Liu, J. Silvester, and A. Polydoros, "Performance evaluation of R-ALOHA in distributed packet radio networks with hard real-time communications," in *Proc. IEEE 45th Veh. Technol. Conf.*, vol. 2. Jul. 1995, pp. 554–558.

[16] R. Verdone, "Multihop R-ALOHA for intervehicle communications at millimeter waves," *IEEE Trans. Veh. Technol.*, vol. 46, no. 4, pp. 992–1005, Nov. 1997.

[17] N. H. Vaidya. *Mobile Ad-Hoc Networks: Routing, MAC, and Transport Issues* [Online]. Available: http//www.crhc.uiuc.edu/ nhv

[18] R. M. Yadumurthy, C. H. Adithya, M. Sadashivaiah, and R. Makanaboyina, "Reliable MAC broadcast protocol in directional and omni-directional transmissions for vehicular ad hoc networks," in *Proc. 2nd ACM Int. Workshop Veh. Ad Hoc Netw.*, New York, NY, USA, 2005, pp. 10–19.

[19] Z. J. Haas, S. Member, and J. Deng, "On optimizing the backoff interval for random access schemes," *IEEE Trans. Commun.*, vol. 51, no. 12, pp. 2081–2090, Dec. 2003.

[20] B. Jae Kwak, N. Oak Song, L. E. Miller, and S. Member, "Performance analysis of exponential backoff," *IEEE/ACM Trans. Netw.*, vol. 13, no. 2, pp. 343–355, Apr. 2005.

[21] N. O. Song and B. J. Kwak, "On the stability of exponential backoff," *J. Res. Nat. Inst. Stand. Technol.*, vol. 108, no. 4, pp. 289–297, 2003.

[22] P. Soni and A. Chockalingam, "Analysis of link-layer backoff schemes on point-to-point Markov fading links," *IEEE Trans. Commun.*, vol. 51, no. 1, pp. 29–32, Jan. 2003.

[23] Y. Xiao, F. Li, K. Wu, K. Leung, and Q. Ni, ''On optimizing backoff counter reservation and classifying stations for the IEEE 802.11 distributed wireless LANs,'' *IEEE Trans. Parallel Distrib. Syst.*, vol. 17, no. 7, pp. 713–722, Jul. 2006.

[24] J. Goodman, A. G. Greenberg, N. Madras, and P. March, ''Stability of binary exponential backoff,'' *J. ACM*, vol. 35, no. 3, pp. 579–602, Jun. 1988.

[25] D. Raychaudhuri and K. Joseph, ''Performance evaluation of slotted aloha with generalized retransmission backoff,'' *IEEE Trans. Commun.*, vol. 38, no. 1, pp. 117–122, Jan. 1990.

[26] J. Hastad, T. Leighton, and B. Rogoff, ''Analysis of backoff protocols for multiple access channels,'' in *Proc. 9th Annu. ACM Symp. Theory Comput.*, New York, NY, USA, 1987, pp. 241–253.

[27] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, ''MACAW: A media access protocol for wireless LAN's,'' *SIGCOMM Comput. Commun. Rev.*, vol. 24, no. 4, pp. 212–225, Oct. 1994.

[28] S. Xu and T. Saadawi, ''Does the IEEE 802.11 MAC protocol work well in multihop wireless ad hoc networks?'' *IEEE Commun. Mag.*, vol. 39, no. 6, pp. 130–137, Jun. 2001.

[29] T. Abdelkader, K. Naik, A. Nayak, and F. Karray, ''Adaptive backoff scheme for contention-based vehicular networks using fuzzy logic,'' in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Aug. 2009, pp. 1621–1626.

[30] D. Dubois and H. Prade, *Fuzzy Sets and Systems: Theory and Applications*. Orlando, FL, USA: Academic, 1980.

[31] L. A. Zadeh, *A Theory of Approximate Reasoning*, vol. 9, J. E. Hayes, D. Michie, and L. I. Mikulich, Eds. New York, NY, USA: Elsevier, 1979, pp. 149–194.

[32] T. F. G. J. Klir, *Fuzzy Sets, Uncertainty, and Information*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1988.

[33] F. O. Karray and C. de Silva, *Soft Computing and Intelligent Systems Design: Theory, Tools and Applications*. New York, NY, USA: Addison-Wesley, 2004.

[34] L. Barolli, T. Yamada, S. Yokoyama, A. Koyama, T. Suganuma, and N. Shiratori, ''A fuzzy admission control scheme for high-speed networks,'' in *Proc. 12th Int. Workshop Database Expert Syst. Appl.*, Washington, DC, USA, 2001, pp. 157–161.

[35] J. Ye, X. S. Shen, and J. W. Mark, ''Call admission control in wideband CDMA cellular networks by using fuzzy logic,'' *IEEE Trans. Mobile Comput.*, vol. 4, no. 2, pp. 129–141, Mar. 2005.

[36] T. Alpcan and S. Buchegger, ''Security games for vehicular networks,'' *IEEE Trans. Mobile Comput.*, vol. 10, no. 2, pp. 280–290, Feb. 2011.

[37] H. Liu, J. Li, Y.-Q. Zhang, and Y. Pan, ''An adaptive genetic fuzzy multi-path routing protocol for wireless ad hoc networks,'' in *Proc. Int. Workshop Softw. Eng., Artif. Intell., Netw. Parallel/Distrib. Comput., 1st Int. Conf. Self-Assembling Wireless Netw.*, May 2005, pp. 468–475.

[38] W. Arnold, H. Hellendoorn, R. Seising, C. Thomas, and A. Weitzel, ''Fuzzy routing,'' *Fuzzy Sets Syst.*, vol. 85, no. 2, pp. 131–153, 1997.

[39] Q. Ren and Q. Liang, ''Fuzzy logic-optimized secure media access control (FSMAC) protocol wireless sensor networks,'' in *Proc. IEEE Int. Conf. Comput. Intell. Homeland Security Personal Safety*, Apr. 2005, pp. 37–43.

[40] S. Lekcharoen and C. C. Fung, ''An adaptive fuzzy control traffic shaping scheme over wireless networks,'' in *Proc. APCC*, Oct. 2007, pp. 177–182.

[41] S. Lekcharoen and C. Jittawiriyanukoon, ''QoS sensitive fuzzy backoff schemes in policing mechanisms,'' in *Proc. 2nd Int. Conf. Mobile Technol., Appl. Syst.*, Nov. 2005, pp. 1–6.

[42] T. Abdelkader, K. Naik, and A. Nayak, ''Using fuzzy logic to calculate the backoff interval for contention-based vehicular networks,'' in *Proc. 7th IWCMC Conf.*, Jul. 2011, pp. 783–788.

[43] D.-M. Chiu and R. Jain, ''Analysis of the increase and decrease algorithms for congestion avoidance in computer networks,'' *Comput. Netw. ISDN Syst.*, vol. 17, no. 1, pp. 1–14, Jun. 1989.

[44] R. Jain, *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. New York, NY, USA: Wiley, 1991.

[45] R. Palit, A. Singh, and K. Naik, ''Modeling the energy cost of applications on portable wireless devices,'' in *Proc. 11th Int. Symp. Model., Anal. Simul. Wireless Mobil Syst.*, New York, NY, USA, 2008, pp. 346–353.

[46] R. Palit, K. Naik, and A. Singh, ''Estimating the energy cost of communication on portable wireless devices,'' in *Proc. 1st IFIP WD*, Nov. 2008, pp. 1–5.

TAMER ABDELKADER received the Ph.D. degree in electrical and computer engineering from the University of Waterloo, Waterloo, ON, Canada, in 2012. He is currently an Assistant Professor with the Information Systems Department, Faculty of Computers and Information Sciences, Ain Shams University, Egypt. He is the author of several publications in the IEEE journals and conferences. His research interests include most of mobile networking problems, such as routing, access control, energy cost modeling, quality-of-service provisioning, resource allocation, mobility modeling, and many others.

KSHIRASAGAR NAIK received the B.Sc. degree in engineering from Sambalpur University, India, and the M.Tech. degree from IIT Kharagpur. He was a Software Developer with Wipro Technologies, Bangalore, for three years. Next, he received the M.Math. degree in computer science from the University of Waterloo and the Ph.D. degree in electrical and computer engineering from Concordia University, Montreal. He was a Faculty Member with the University of Aizu, Japan, and the Carleton University, Ottawa. He is currently a Full Professor with the Department of Electrical and Computer Engineering, University of Waterloo. He was a Co-Guest Editor of three special issues of the IEEE Journal on Selected Areas in Communications. He was an Associate Editor of the *Journal of Peer-to-Peer Networking and Applications* from 2008 to 2013. He is serving as an Associate Editor of the *International Journal of Parallel, Emergent and Distributed Systems* and the *IEEE Transactions on Parallel and Distributed Systems*. In addition, he is serving as a Regional Editor (America) of the *Journal of Circuits, Systems, and Computers*. His research interests include dependable wireless communication, resource allocation in wireless networks, mobile computing, peer-to-peer communication, vehicular networks, energy efficiency of smartphones and tablet computers, energy performance testing of mobile apps, and communication protocols for smart power grids. His book entitled *Software Testing and Quality Assurance: Theory and Practice* (Wiley, 2008) has been adopted as a text in many universities around the world. His second book entitled *Software Evolution and Maintenance* (Wiley, 2014) is in press.

• • •