Establishing strict priorities in IEEE 802.11p WAVE vehicular networks

Mohssin Barradi, Abdelhakim S. Hafid,
Network Research Laboratory,
University of Montreal, Montreal, Canada
{mbarradi, ahafid}@iro.umontreal.ca

Jose R. Gallardo
School of Information Technology and Engineering
University of Ottawa, Ottawa, Canada
gallardo@site.uottawa.ca

Abstract—The WAVE (Wireless Access in Vehicular Environments) concept includes seven channels within the DSRC band. One of them, known as the Control Channel (CCH), is the one used to exchange all safety-related messages. Messages sent over the CCH have to be processed with different priorities depending on how critical they are for vehicle safety. However, the MAC protocol currently adopted for WAVE, namely EDCA, stops short of that requirement; it does not establish strict priorities, but only relative advantages for some types of messages over the others. Another problem is that, since messages are broadcasted on the CCH, there are no acknowledgments. This means that it is not possible to know whether a transmission was successful or not, which eliminates the possibility to use the binary exponential backoff technique to reduce congestion. In this paper, we propose a simple but effective solution to both of these problems. We use simulations to analyze the performance of the modified MAC protocol and compare it to that of the original EDCA. The results show that the proposed scheme outperforms EDCA. Our comparison focuses on the reduced probability of collision for high-priority frames (gain) and on the increased delays for lower-priority frames (price to pay).

Keywords — WAVE; vehicular networks; IEEE 802.11p; EDCA MAC protocol; performance analysis.

I. INTRODUCTION

The transportation system has a significant impact on the economy and safety of the society. Safe, timely and low-cost transportation of people and goods is important for the well-being of commerce, industry, government, educational centers, families, etc. Moreover, because of its size, large number of components and complexity, the transportation system is extremely difficult to manage and supervise. The need for the development of intelligent systems able to enhance the transportation infrastructure in terms of safety, comfort and efficiency, was understood quite a long time ago. One way to do this is by creating a wireless communications network accessible by vehicles on the move, and able to provide them with safety-critical information as well as with a gateway to the global Internet.

In 1999, the FCC allocated 75 MHz of bandwidth in the 5.9 GHz band to create a nationwide vehicle-area network (VANET). This set of frequencies is known as DSRC (Dedicated Short-Range Communications) band [1]. Its purpose is to provide vehicle-to-roadside as well as vehicle-to-vehicle wireless communications; thus, stations on the roadside (roadside units - RSU) and mobile radio units located on board of vehicles (on board units - OBU) can share information related to road and traffic conditions and use it to improve the safety and efficiency of the transportation system. The resulting technology is known as WAVE (Wireless Access in Vehicular Environments); its overall architecture is defined in IEEE standards 1609.1, 1609.2, 1609.3, 1609.4, and P802.11p [2]. The latter is not yet in its final form, but is currently under development. The WAVE concept includes seven radio channels within the DSRC band. One of them is known as the Control Channel (CCH) and the remaining six are known as Service Channels (SCH). The CCH is the channel that road safety relies on, since all safety-related messages are exchanged using it. Service channels, on the other hand, are reserved for non-safety-related connections (e.g. to look for nearby hotels, to download a map or to check e-mail) and their goal is to allow people on vehicles to maintain access to the Internet.

In WAVE, a CCH classifier is defined, which is in charge of sorting packets into four access categories, each having a separate queue identified by their indexes (i.e., AC[3], AC[2], AC[1], and AC[0]). These four queues store traffic with different service needs, which are reflected in the corresponding channel-access parameters, such as time to sense the medium and decide if it is free (AIFS), the maximum length of the random backoff intervals (CW), and the maximum length of a transmission opportunity (TXOP). The authors in [3] present an example of each access category in CCH: (1) AC[3] concerns emergency information from the RSU (accidents, obstacles, slippery or missing traffic signs, etc.) and information generated by cars (vehicle with malfunctioning brakes, or speeding over a certain limit, etc.); (2) AC[2] concerns presence and speed information advertised by vehicles; and (3) AC[1] concerns information sent by vehicles asking for help when they pose no risk to other vehicles (e.g. when they overheat or run out of gas); and (4) AC[0] concerns information aimed at establishing new non-safety-related connections over the service channels. All these messages are broadcasted through the control channel (CCH).

To support the different types of safety messages appropriately, a vehicular network should establish strict/absolute priorities to access the wireless medium. However, the MAC protocol currently adopted for vehicular networks, namely Enhanced Distributed Channel Access (EDCA), described in the IEEE 802.11 standard [4], falls short of that requirement in the sense that it does not implement a strict-priority system, but only enforces service differentiation among the different types of messages. This differentiation...
raises two key issues: (1) A low-priority message being transmitted can collide with the transmission of a high-priority message causing for both transmissions to fail; and (2) A low-priority message may win the medium-access competition and start earlier, taking away precious time needed by high-priority messages.

EDCA’s effectiveness to reduce collisions relies on the so-called binary-exponential backoff for retransmissions, which consists of selecting larger random backoff periods as frames experience repeated collisions; the goal is to spread retransmissions over time, thus reducing the probability that the same frames experience new collisions. However, since all critical messages transmitted on the CCH are broadcasted, it is not possible to know whether a frame transmission is successful or not. This fact makes it impossible to implement the binary-exponential backoff technique; thus, there is no way to reduce the large number of collisions that arise when traffic load increases.

From the previous discussion, we can conclude that vehicular networks suffer from the lack of a MAC protocol with the capabilities to (1) implement strict priorities, needed for adequate support of safety messages; and (2) achieve relatively high throughput, when (broadcast) traffic load increases. In this paper, we propose a MAC protocol with these capabilities.

The remainder of the paper is organized as follows. Section II describes our proposed protocol that supports strict priorities and reductions collisions, when the load increases, using a form of broadcast acknowledgments. Section III evaluates our proposal. Section IV concludes the paper.

II. OUR PROPOSAL

In this section, we briefly outline the design and concept of the WAVE standard. This outline will help us describe our proposed protocol, which includes extensions to support the two missing capabilities mentioned in Section I: strict priorities and acknowledgments for broadcast messages.

A. WAVE

The physical layer in WAVE is defined in IEEE 802.11 and amended by IEEE 802.11p [2]; it relies on seven channels of 10 MHz bandwidth each. The WAVE spectrum is allocated in the upper 5.9 GHz range. Since the design allows both single- and multi-receiver units, the different channels cannot be used simultaneously; however, each station continuously alternates between the Control Channel (CCH) and one of the Service Channels (SCHs). Due to the strong delay requirements of the messages transmitted over the CCH (e.g. to keep vehicles from colliding), a period containing one CCH interval and one SCH interval should last no more than 100ms.

<table>
<thead>
<tr>
<th>AC[i]</th>
<th>CW\text{min}</th>
<th>CW\text{max}</th>
<th>AIFSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>15</td>
<td>1023</td>
<td>9</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>15</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>7</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>7</td>
<td>2</td>
</tr>
</tbody>
</table>

**TABLE I. DEFAULT VALUES OF EDCA PARAMETERS IN WAVE**

The MAC layer in WAVE is equivalent to the IEEE 802.11e Enhanced Distributed Channel Access (EDCA) Quality of Service (QoS) extension. The EDCA MAC protocol, an extension of the DCF (Distributed Coordination Function), provides traffic differentiation capabilities among four Access Categories AC[0] - AC[3]. The traffic differentiation in EDCA is achieved by assigning a specific set of parameters to each access category. These parameters include: (1) the Arbitration Inter Frame Space (AIFS), which defines the minimum time interval that the wireless medium has to be sensed idle in order for a node to decide that it is free; (2) contention windows (CW), which defines a random number generation window for the backoff mechanism; (3) transmit opportunity (TXOP) limit, which defines the longest time interval during which a WAVE device can transmit one or several frames. A frame that is too large to be transmitted in a single TXOP is fragmented into smaller frames.

The duration of AIFS for each access category AC is defined as follows:

\[
\text{AIFS}[\text{AC}] = \text{SIFS} + \text{AIFSN}[\text{AC}] \times \text{slot time}
\]

where SIFS (short IFS) is the time that separates atomic or indivisible frame exchanges, such as a data frame and its ACK. In turn, AIFSN[AC] (AIFS number) defines the minimum number of slots to wait before starting the backoff countdown, and slot time is the duration of a physical-layer time slot.

The timing relationship of EDCA is shown in Figure 1. A node corresponding to the AC with the smallest AIFS has an advantage over nodes from the other ACs since it will detect earlier that the medium has been freed and will start counting down its backoff interval. However, this advantage is not enough to establish strict priorities. The values of AIFSN[AC], CW/min[AC], and CW/max[AC], which are referred to as the EDCA parameters, are announced by the Access Point (AP) via beacon frames. Table I shows the default values for these parameters in WAVE for the different access categories [4].

B. STRICT PRIORITIES

Several contributions on traffic differentiation in the context of EDCA can be found in the open literature; however, most of these contributions (e.g. [5], [6]) consider only unicast (as opposed to broadcast) traffic. Others, like [7], [8], [9], propose interesting analytical studies that model the effect of the most important EDCA parameters. Unfortunately, they limit themselves to studying existing approaches and do not propose further enhancements that could improve performance in broadcast environments. Other alternatives on service differentiation can be found for instance in [10], and [11], where time is scheduled or reserved to improve, again, the...
To implement strict priorities we propose that, for a given access category (AC), the minimum time that a node has to sense the wireless medium to decide if it is idle be larger than the AIFS plus the largest possible backoff interval of all access categories corresponding to higher-priority frames.

The basic idea behind our proposal is the following. In the worst case, a higher-priority frame of class $k$ has to wait for $\text{AIFS}[k] + \text{CWmax}[k]$, $\text{AIFS}[k-1]$ has to wait, in the best case, only for $\text{AIFS}[k-1]$ assuming that the selected random backoff interval is zero. If we want to establish strict priorities, the former waiting period has to be smaller than the latter. Such condition is not satisfied if the standard parameter values are used.

We propose to establish that AC[3] messages have a strictly higher priority than AC[2] messages and that AC[2] messages, in turn, have a strictly higher priority than AC[1] messages. Since AC[1] messages are not as safety-critical as messages corresponding to AC[2] and AC[3], we propose that AC[1] messages do not have a strictly higher priority than AC[0] messages.

Table II shows the parameter values that we propose for EDCA, and Figure 2 shows the timing relationship among the different ACs. Figure 2(a) corresponds to the EDCA parameters included in the IEEE 802.11p standard and Figure 2(b) shows our proposed values. Notice that we suggest that $\text{AIFS}[2] = \text{AIFS}[3] + \text{CWmax}[3]$, $\text{AIFS}[1] = \text{AIFS}[2] + \text{CWmin}[2]$, and $\text{AIFS}[0] = \text{AIFS}[1] + \text{CWmin}[1]$. The fact that we use CWmax in the first expression is due to the introduction of acknowledgements and retransmission for the highest-priority class (AC[3]) (see next subsection for details). As mentioned before, the expected advantages obtained by establishing strict priorities is a reduction in delays and losses for higher-priority frames, and the price to pay for it is the introduction of longer end-to-end (E2E) delays for lower-priority frames.

C. Broadcast Acknowledgment

The MAC protocol currently adopted for WAVE does not allow the transmission of acknowledgments (ACK) for broadcast messages. Since all safety-related messages in the CCH are broadcasted, lost data frames cannot be identified or retransmitted. Thus, we cannot use the functionality of the binary-exponential backoff technique to reduce the large number of collisions expected to happen when the traffic load increases. In this section, we propose a cluster-based scheme that allows sending, with minimum overhead, acknowledgements for broadcast messages corresponding to the highest-priority class (AC[3]).

Active clustering requires the implementation of signalling protocols to exchange additional messages among neighbouring nodes to select cluster heads (CH) and gateways (GW); also, all participating network nodes must advertise cluster information periodically. All of this generates considerable overhead. Furthermore, the dynamic nature of VANETs will make the association and disassociation to and from clusters disrupt the stability of the network, especially for cluster-heads and gateways. Some papers propose a smooth Re-Clustering, which is not a bad idea for high mobility networks. In [12], for instance, the authors propose a second elected CH (SCH) that will be triggered to be the principal CH (PCH) when the PCH can no longer be a cluster-head. This approach reduces overhead and reconfiguration time, but unfortunately not enough for the protocol to be used in time-constrained environments.

In this paper, we selected passive clustering (PC) [13] since it is simple, easy to implement and, most importantly, virtually overhead-free. PC has the best performance [14] of all clustering protocols, especially in high-mobility and time-constrained networks, such as VANETs. Briefly explained, in PC each node starts with the Initial state and, by analyzing its neighbours’ status, it determines its role according to 4 states: Initial, Cluster head, Gateway, or Ordinary node. When sending data, a node piggybacks its cluster status information using 2 bits in the MAC sub-layer header of all outgoing frames. The receiver uses the cluster status information to determine its own state by counting the number of CHs and GWs within its communication range. We add the requirement that each gateway node will select one of the CHs to which it is associated as its home CH and will indicate it in its status information.

Our proposed scheme uses this partition of the vehicular network into clusters and establishes that cluster heads will

---

**Table II: Proposed Parameters for EDCA**

<table>
<thead>
<tr>
<th>AC[i]</th>
<th>CWmin</th>
<th>CWmax</th>
<th>AIFSmin</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>15</td>
<td>1023</td>
<td>19</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>15</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>7</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>7</td>
<td>2</td>
</tr>
</tbody>
</table>
send acknowledgments for the highest-priority (AC[3]) broadcast messages when received correctly from any of its cluster members, except for those gateway nodes that have selected a different home CH. The cluster head will also designate one of its cluster members as the one in charge to send acknowledgments for the highest-priority data frames broadcasted by the cluster head itself (cluster head is the source of the data). This designated node will preferentially not be a gateway to reduce the probability of interfering with transmissions generated in another cluster.

III. SIMULATIONS

In this section, we analyze the performance of the proposed modified EDCA MAC protocol, and compare it to the original EDCA MAC protocol. The metrics of interest are frame error rate, end-to-end delay and jitter.

In this study, as in reference [3], Frame Error Rate (FER) is the ratio of frames that are never successfully received because the maximum number of unsuccessful attempts is reached. In other words, if a frame is lost due to a collision, but eventually the same frame is retransmitted and successfully received, it will not be counted among the losses. End-to-end (E2E) delay is the time it takes a frame from the instant it reaches the head of its queue (HOQ) to the instant when it reaches its destination (see Eq. (2)). The jitter can be defined as the maximum difference in the delays experienced by any two frames (see Eq. (3)).

\[ E2ED = \text{Arrival Time} - \text{Arrival Time to HOQ} \]  
\[ \text{Jitter} = \text{Max} (E2ED) - \text{Min} (E2ED) \]  

Frame Error Rate (FER) is a key parameter since lost safety-related messages can cause severe problems by not warning vehicles of imminent danger. Timeliness of high priority safety messages also plays an important role in the safety of drivers, since a difference of a few milliseconds can make a difference between saving lives or not. We also include jitter since it can better measure the performance of delay-sensitive applications.

A. TOPOLOGY AND SETTING PARAMETERS

The simulations were carried out using ns-2.34. Each simulation lasts for 1000 seconds. We generated scenarios of a highway in a grid of 2400 meters × 400 meters, with up to 400 vehicles (OBU) and 4 access points (RSUs) as shown in Figure 3. All vehicles are moving with a random speed between 30 and 70 km/h, and turning around the grid. We varied the traffic load from 10% to 100%. All nodes (RSU and OBU) use the setting of 802.11a at 6Mbps for the physical layer. Table III shows the key parameters we used in our simulations. We built our implementation based on the last draft of 802.11p [15]. We adapted that version with the following key modifications: (1) the support of broadcast messages; (2) the implementation of sending ACK for every highest-priority frame correctly received; (3) Adjustment of the EDCA parameters according to Table II.

B. RESULTS

B.1. End-to-end delay and Jitter

We denote by ACold[0]-[3] the access categories with the default (old) parameters (see Table I), and by AC[0]-[3] the access categories with the new proposed parameters (see Table II).

We considered four scenarios to analyze the end-to-end delay and jitter:

- In the first scenario we fix the number of AC[3] nodes to two (RSU-2, and one car stopped beside); the fraction of messages that they generate is 10%; we then vary the traffic load from 20% to 100% with traffic generated by AC[2], AC[1] and AC[0] nodes stopped beside RSU-2.
- Similarly, in the second scenario we fix the number of ACold[3] nodes to two and the fraction of messages that they generate to 10%; we then vary the traffic load from 20% to 100% with traffic generated by ACold[2], ACold[1] and ACold[0] nodes.
- In the third scenario, we vary the traffic load from 10% to 100% using only AC[3] frames.
- Finally, in the fourth scenario, we vary the traffic load from 10% to 100% using only ACold[3] frames.

Figure 4 shows the one-hop E2E delay for frames corresponding to AC[1-3] (modified EDCA) in the first scenario, as a function of the traffic load. We observe that the E2E delay of AC[3] messages remains stable, close to 0.5 ms, even for the maximum load. On the other hand, we can see that the E2E delay of AC[2] and AC[1] frames increases with load, especially for AC[1] frames.

Figure 5 compares the one-hop E2E delay for highest-priority frames AC[3] (modified EDCA) and ACold[3] (original EDCA), obtained in the first and second scenarios, respectively, as a function of the traffic load. We can observe that the E2E delay of ACold[3] frames increases as the traffic

<table>
<thead>
<tr>
<th>SlotTime</th>
<th>0.000013</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFS</td>
<td>0.000032</td>
</tr>
<tr>
<td>PreambleLength</td>
<td>32</td>
</tr>
<tr>
<td>PLCPHeaderLength</td>
<td>40</td>
</tr>
<tr>
<td>SymbolDuration</td>
<td>0.000008</td>
</tr>
<tr>
<td>BasicModulationScheme</td>
<td>0</td>
</tr>
<tr>
<td>use 802.11a flag</td>
<td>true</td>
</tr>
<tr>
<td>RTSThreshold</td>
<td>2346</td>
</tr>
</tbody>
</table>

TABLE III: SIMULATION PARAMETERS

FIGURE 3: SIMULATION TOPOLOGY
load increases, reaching a maximum value above 0.6 ms, as opposed to the E2E delay for the extended EDCA, which remains below 0.54 ms. This confirms the superiority of our proposal, guaranteeing a stable low value for the E2E delay of highest-priority frames.

Figure 6 shows the one-hop E2E delay for highest-priority frames AC\[3\] (modified EDCA) and ACold\[3\] (original EDCA), obtained in the third and fourth scenarios, respectively. We can observe that, in both cases, the E2E delay increases as the traffic load increases, reaching maximum values of 7.45 ms for AC\[3\] and 4.15 ms for ACold\[3\]. We can see also that the E2E delay for highest priority frames AC\[3\] is always higher than the one of ACold\[3\]. The reason for this difference is the fact that AC\[3\] frames are being retransmitted when needed, which increases the total amount of time required for a successful transmission. It is important to mention that E2E delay values smaller than 7.5 ms is not bad; indeed, it has been reported that a delay smaller than 10 ms is still acceptable for this type of systems [16].

Figure 7 compares the jitter for highest-priority frames AC\[3\] (modified EDCA), obtained in the first and second scenario, respectively, as a function of the traffic load. We can observe that, in both cases, the frame error rate is null for traffic load values below 40%, and there is a slight increase after this threshold. The reason for this increase is the fact that lower-priority nodes tend to use most of the bandwidth left over by AC\[3\] nodes, so that AC\[3\] frames will very likely be generated when the channel is busy. That way, even if two AC\[3\] frames are not generated at the same time, they will likely be transmitted simultaneously when the channel becomes idle, provoking a collision. As expected, the presence of AC\[2\] traffic produces higher frame error rate values than those produced by AC\[1\] traffic.

Finally, Figure 10 compares the frame error rate experienced by highest-priority frames AC\[3\] (modified EDCA) and ACold\[3\] (original EDCA) in scenarios 3 and 4, respectively. We can observe that, while the frame error rate varies the traffic load from 20% to 100% with traffic generated by AC\[2\] nodes stopped beside RSU-2.

- In the second scenario, we fix the traffic generated by AC\[3\] nodes to 10% and vary the traffic load from 20% to 100% with traffic generated by AC\[1\] nodes.
- In the third scenario, we vary the traffic load from 10% to 100% using only AC\[3\] frames.
- Finally, in the fourth scenario, we vary the traffic load from 10% to 100% using only ACold\[3\] frames.

B.2. Frame Error Rate (FER)

We considered four scenarios to analyze the FER:

- In the first scenario, we fix the traffic generated by AC\[3\] nodes (RSU-2, and one car stopped beside) to 10% and vary the traffic load from 20% to 100% with traffic generated by AC\[2\] nodes stopped beside RSU-2.
- In the second scenario, we fix the traffic generated by AC\[3\] nodes to 10% and vary the traffic load from 20% to 100% with traffic generated by AC\[1\] nodes.
- In the third scenario, we vary the traffic load from 10% to 100% using only AC\[3\] frames.
- Finally, in the fourth scenario, we vary the traffic load from 10% to 100% using only ACold\[3\] frames.
corresponding to the modified EDCA remains very low, as described above and shown in Figure 8, the FER corresponds to the original EDCA increases to 25% for a traffic load of 100%. This confirms the improvement introduced by the new approach also in terms of the frame error rate experienced by the highest-priority frames.

IV. CONCLUSION

We propose very simple, but at the same time very effective modifications to the current EDCA protocol to solve two problems that would be present in WAVE vehicular networks if they were to be deployed as described in the current version of the draft standard IEEE 802.11p. The first adaptation would be to change the AIFSN parameter associated to the different access categories as described in Table II to allow the implementation of strict priorities for access categories 3 and 2. The second adaptation would be to allow the transmission of acknowledgments for successfully received AC[3] broadcast data frames, which in turn allows the transmitter of the relevant frames to know whether they were successfully received and, if not, to retransmit the frame using the binary exponential backoff technique, thus increasing the probability of success.

Our simulation results show the effectiveness of our proposal in reducing delay, jitter and losses for the most critical messages, even in high-traffic conditions. The price to pay here is an increase in the delay experienced by lower-priority frames which, it is worth mentioning, are able to tolerate these increased delays because they are not directly related to emergency situations.

V. REFERENCES


