Abstract—Intersection collision avoidance application is one of the highest-rated mid-term safety applications identified by the United States National Highway Traffic Safety Administration (NHTSA). Such safety applications are enabled by the Dedicated Short Range Communications (DSRC) at 5.9 GHz. The DSRC specification is based on the physical (PHY) and medium access control (MAC) layers of the IEEE 802.11a. Evaluating the performance of the IEEE 802.11a on vehicles’ mobility is a challenging task. This task can be achieved by either outdoor field studies using a large number of vehicles on different street topologies and traffic regulations, or by using computer simulation. For this type of task, computer simulation is the most feasible way of determining system performance. In this paper, we study, using computer simulations, the settings of two parameters of the IEEE 802.11a on the available bandwidth per vehicle, the average wait time per vehicle and the distance to the intersection for vehicles to react to warning message to avoid collisions. The two parameters under study are: (1) the number of retries, dot11shortretrylimit, to contend for a channel in a contention cycle, and (2) the lower bound limit for the number of time slots, aCWmin, to sense the channel. Our simulation results show that it is sufficient to set the aCWmin to 120 regardless of the value of the dot11shortretrylimit, and the distance to the intersection to 150m for the intersection collision avoidance system enabled by the DSRC.

I. INTRODUCTION

The Intelligent Transportation Society of America (ITSA) [3] and the United States Department of Transportation [4] have been working together for more than fifteen years on promoting the development and deployment of Intelligent Transportation Systems (ITS) technologies. ITS technologies will provide a safer traffic and will reduce deaths, injuries and economic losses from motor vehicle crashes. Such technologies include crash avoidance and notification systems, roadbed sensors, and on-board navigation systems. The United States Federal Communications Commission (FCC) authorized the 5.9 GHz dedicated short range communications (DSRC) for ITS technologies to be used in a wide range of advanced vehicle safety applications that require vehicle-to-infrastructure (V2I) and vehicle-to-vehicle (V2V) communications. Future vehicles will be equipped with DSRC devices that will communicate and collaborate to broadcast their safety-critical information to each other, such as speed, position, and heading. For the DSRC, there are seven non-overlapping 10MHz channels in the 5.850-5.925 GHz band. DSRC also supports data rates of 6, 9, 12, 18, 24, 36, 48, and 54 Mbps. The advantages of the DSRC are its capability of providing very low latency communications, and of transmitting broadcast messages to a maximum range of 1000 meters.

The National Highway Traffic Safety Administration (NHTSA) distributed a publication [5] that identifies intelligent vehicle safety applications enabled by the DSRC. The authors in [5] compiled a list of 34 communications-based vehicle safety application scenarios. Several of these safety-applications were selected as the highest-rated mid-term applications, such as pre-crash warning, cooperative forward collision warning, lane change warning, and intersection collision avoidance. In this paper, our main contribution focuses on the intersection collision avoidance application.

According to the National Center for Statistics and Analysis (NCSA) report [2], there are 2,481,619 crashes at intersections that caused 8,619 deaths, 848,000 injuries, and 1,625,000 in property damages. As one of the highest-rated vehicle safety applications, intersection collision avoidance system warns drivers of a possible collision with other vehicles at an intersection. The NHTSA publication [5] suggested the use of infrastructure sensors and/or DSRC communications to detect and collect information about the position, heading, velocity and turning status of all vehicles while approaching an intersection. The information obtained from the infrastructure sensors and/or DSRC communications is broadcasted to all vehicles approaching the intersection. An in-vehicle DSRC device that receives and processes such information determines whether or not a collision is imminent at the intersection and provides a warning to the driver.

In order to implement high-speed data transfer applications in the 5.9-GHz Intelligent Transportation Systems Radio Service (ITS-RS) Band, the physical layer (PHY) and the medium access control (MAC) layer specifications were developed as a standard for the DSRC [6]. The standard specification is based on and refers to the IEEE 802.11a MAC
and PHY layers.

The evaluation of the DSRC for ITS technologies can be made through real-time outdoor traffic experiments which will take a long time to setup, run, collect and analyze the collected data. These experiments will make the evaluation very difficult and infeasible at research stages of ITS technologies. In addition, these experiments will require large number of vehicles moving at a variety of street topologies and traffic conditions. An alternative approach is through the use of computer simulations. In our simulator, we control the mobility of vehicles, allow each vehicle to operate the IEEE 802.11a MAC layer and to transmit its safety-critical information to nearby vehicles, and allow each vehicle to either collide or avoid a collision with other vehicles based on their driving conditions and on the operation of the IEEE 802.11a. The main objective of developing an intersection traffic simulator is to measure the performance of the 802.11a and to find the optimal settings of its parameters for the vehicle intersection collision avoidance application.

The rest of the paper is organized as follows. In Section II, we discuss the background in IEEE 802.11 MAC layer. In Section III, we discuss the related work. In Section IV, we describe our motivation and contributions in this paper. In Section V, we describe our intersection traffic simulator. In Section VI, we show our simulation results. Finally, we conclude the paper in Section VII.

II. THE DISTRIBUTED COORDINATION FUNCTION

The MAC protocol for the DSRC is the IEEE 802.11a. The access mechanism for the 802.11 is the distributed coordination function (DCF), which is based on carrier sense multiple access with collision avoidance (CSMA/CA). This section briefly describes the basic access mechanism for the DCF. For a complete operation of the DCF, refer to the 802.11 standard [1].

A station that is equipped with an 802.11a device and has a new packet to transmit starts a contention cycle by sensing the channel for a period of time equals to a distributed interframe space (DIFS). If the channel is sensed busy during DIFS, the station continues to sense the channel until the channel has been sensed idle for DIFS. The station then starts the DCF contention operation. In DCF, stations contend for the channel using a DCF random backoff algorithm. A random backoff algorithm chooses a pseudorandom integer \( P \) between 0 and a Contention Window \( CW \) value. The \( CW \) is an integer between the PHY characteristics values \( aCWmin \) and \( aCWmax \) \( (aCWmin \leq CW \leq aCWmax) \). The \( CW \) takes an initial value of \( aCWmin \). The pseudorandom integer \( P \) is the number of time slots the station has to sense an idle channel before the station may transmit. A time slot has duration of \( aSlotTime \) seconds.

After sensing an idle channel for duration of DIFS, this station continues to sense the channel for \( P \) time slots. If the station senses an idle channel at a time slot, then it decrements its \( P \) value. If the station senses a busy channel, then it loses the channel, stops decrementing its \( P \) value and waits for the next contention cycle. Once all \( P \) time slots have been sensed idle, the station wins the channel and transmits its packet to a destination station. The destination station sends an acknowledgment \( (ACK) \) to the transmitter after a short interframe space \( (SIFS) \). The \( SIFS \) has duration of \( aSIFSTime \). The relationship between \( DIFS \) and \( aSIFSTime \) is given by \( DIFS = aSIFSTime + 2 \times aSlotTime \).

The transmission is successful if the transmitter receives the \( ACK \) signal. Otherwise, the transmitter attempts for another contention for the channel by doubling its \( CW \) value and generating a new \( P \) between 0 and the new \( CW \) value. This process is repeated until \( CW \) reaches \( aCWmax \). The transmitter also has \( dot11shortretrylimit \) attempts to contend for the channel if the transmission is unsuccessful. The value of \( CW \) stays at \( aCWmax \) until a transmission is successful or the number of retries \( dot11shortretrylimit \) is reached. If a transmission is successful or \( dot11shortretrylimit \) is reached, then \( CW \) takes again the initial value of \( aCWmin \). Table I shows the DSRC specifications for the parameters described in this section.

<table>
<thead>
<tr>
<th>TABLE I DSRC/IEEE 802.11a Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSRC parameter</td>
</tr>
<tr>
<td>----------------</td>
</tr>
<tr>
<td>aSlotTime</td>
</tr>
<tr>
<td>aSIFSTime</td>
</tr>
<tr>
<td>DIFS</td>
</tr>
<tr>
<td>aCWmin</td>
</tr>
<tr>
<td>aCWmax</td>
</tr>
<tr>
<td>dot11shortretrylimit</td>
</tr>
</tbody>
</table>

There is another optional technique for transmitting packets by using a four-way handshake protocol. This four-way handshake protocol uses request-to-send/clear-to-send (RTS/CTS) technique. A station that sends a request-to-send (RTS) may transmit packet only if a clear-to-send (CTS) is heard back from the destination. Otherwise, the sender attempts another transmission. This technique was introduced to solve the hidden-station problem: two stations that are out of the communication range of each other may transmit at the same time to a third station that is within the communication range of the other two.

III. RELATED WORK

In [7][8], the authors presented an intersection traffic simulator using the IEEE 802.11 and DOLPHIN protocols. In DOLPHIN, time is divided into slots and a vehicle transmits in a time slot based on non-persistent CSMA. The authors studied the PHY layer of the IEEE 802.11 against path loss and shadowing effects. They assumed that vehicles transmit their data when they are 50 m away from an intersection. The authors indicated that data collision and error rate are related to PHY layer errors only.

In [9], the authors developed a cooperative intersection collision warning system. The authors proposed top-level
specifications for intersection collision warning systems to reduce excessive and undesired warning messages. Based on their specifications, they developed a parameterized collision warning algorithm. The algorithm depends on the driver’s reaction time to warning messages and on the difference in time to a collision between two vehicles. Although the authors indicated that they used a DSRC device with IEEE 802.11a/b, the authors did not show the performance of the IEEE 802.11a/b on their proposed algorithm.

Researchers have also applied the IEEE 802.11 MAC protocol for highway V2V networks. In [10], the authors proposed to use the IEEE 802.11 MAC protocol to disseminate crash-warning messages on a highway via V2V networks. The message disseminates through multiple vehicles by measuring the distance between the sender and the receiver of the message. The farthest receiver has shorter backoff time, and hence, the receiver wins the channel and the message disseminates faster. In a similar approach, the authors in [11] proposed to use the IEEE 802.11 MAC protocol and a reference point on the road. The message disseminates through multiple vehicles by measuring the distance between a reference point and the vehicles. In [12], the authors proposed to use dual frequency wireless communication technologies. The first is a Short Distance Communication (SDC) system. It is a high-speed, short-distance V2V communications system. The second is a Long Distance Communication (LDC) system. It is a low-speed, long distance communication system employing centralized networking through a base station. The idea behind using an LDC is to exchange “info-tainment” data, such as text messages and video or image data, that has a low priority than safety critical information. The authors proposed to use the IEEE 802.11 MAC layer for the SDC system without showing its performance on their proposed system.

The authors in [14] conducted a field test to study drivers’ brake perception-reaction times (PRT) when amber light is activated at a high-speed signalized intersection. According to [14], the perception-reaction time (or brake reaction time) is defined as the time interval from the onset of the amber light to the instant when the brake pedal is pressed when a driver arrives at a signalized intersection. The field test was conducted on a four-way signalized intersection where drivers approached the intersection at a speed of 72 km/h. The observed PRTs ranges between 0.3 and 1.7 s, with a mean equals to 0.742 s, a median of 0.7 s, and a standard deviation of 0.189 s. The authors in [14] concluded that their study validated and is consistent with the 1.0 s, 85th percentile PRT that is recommended in traffic signal design procedures. The authors also concluded that the typical approach to modeling driver perception-reaction times is to consider a lognormal distribution with a mean equal to 0.67 s, a median of 0.65 s, and a standard deviation of 0.12 s.

The authors in [15] studied drivers’ preferred time-headway within a range of real-world driving speeds and conditions. The authors obtained a traffic flow data from Caltrans (The California Department of Transportation) for one of the eight-lane freeways in California. Their analysis showed that for a 1.5 hr period of time in a congested flow traffic, drivers successfully maintained a time headway between 1 and 2 s with vehicle speeds from 20 mph (~32 Km/h) to 60 mph (~96.5 Km/h) without rear-end collisions. The results obtained by [14] and [15] will be used as input parameters in our simulator in this paper.

IV. MOTIVATION AND CONTRIBUTIONS

Intersection collision avoidance application is one of the highest-rated mid-term safety applications identified by the NHTSA. The DSRC at 5.9 GHz has already been authorized by the FCC to be used for such safety applications. In [7][8], the authors studied the PHY layer of the DSRC for the vehicle traffic intersection. The authors assumed that vehicles start transmitting their safety-critical information when these vehicles are 50 m away from the intersection. When designing an intersection collision avoidance system, there are several factors that need to be considered: vehicles’ speed, drivers’ reaction time to warning messages, drivers’ deceleration, the number of vehicles contending for the channel, and the comfort of drivers. The authors studied only the PHY layer of the DSRC in their simulator. In addition, the authors used the DOLPHIN MAC protocol although the DSRC specifications require only the use of the IEEE 802.11a MAC layer. The authors did not show the performance of the IEEE 802.11a MAC layer on their intersection simulator. Similarly, the works in [9]-[12] used the preset values of the IEEE 802.11a/b specifications without studying the effect of these values on their proposed systems. To the best of our knowledge, the evaluation of the DSRC IEEE 802.11a MAC layer has not been studied before for the intersection collision avoidance system application.

We have previously developed an intersection traffic simulator using Visual C++ and OpenGL [13]. We studied the effect of DCF’s $aCW_{max}$ on the available bandwidth per vehicle. Our preliminary study showed that it is sufficient to set $aCW_{max}$ to 200 to avoid vehicles’ collisions at intersections. In this paper, we extend our previous analysis to find the optimal settings of the IEEE 802.11a DCF’s $aCW_{min}$ and dot11shortretrylimit on the available bandwidth per vehicle, the average wait time for a vehicle to transmit its safety-critical information, and the distance to the intersection for a vehicle to operate the DCF and react on warning messages.

V. INTERSECTION TRAFFIC SIMULATOR

In this section, we describe the structure of our intersection and the characteristics of vehicles and drivers’ behavior on this intersection.

A. Intersection Structure

As shown in Fig. 1, there are four unsignalized intersection legs. Each intersection leg has four lanes. The width of a lane is 4 m. There are also divisional islands between two opposite directions. The width of the island is 8 m. In this structure, vehicles travel on the right side of the road.

The NHTSA publication [5] suggested the use of
infrastructure sensors and/or DSRC communications to detect and collect information about all vehicles approaching an intersection. In this paper, we assume that there are sensors on the road for vehicles approaching an intersection, and for vehicles leaving the intersection. In Fig. 1, these sensors are indicated by the entry trigger line and exit trigger line. When vehicles approaching an intersection detect the entry trigger line, they start the execution of the IEEE 802.11a MAC layer. Vehicles continue the execution of the MAC layer while they are approaching the intersection zone. Vehicles stop contending for the channel and the execution of the MAC layer as soon as they detect the exit trigger line.

The entry trigger line is installed at a pre-determined distance, \( d_i \), from the intersection. This distance specifies the communication range of the DSRC, which is \( 2d_i \). If all intersection legs have an entry trigger line, then all vehicles that detect these lines are within the communication range of each other. Hence, we avoided the hidden-station problem. Once a vehicle leaves the intersection zone and detects the exit trigger line, it is unnecessary for this vehicle to contend for the channel. This allows additional available bandwidth for use by other vehicles approaching the intersection. We assume here that one of the seven DSRC channels is assigned to the intersection collision avoidance application.

We also consider the drivers’ reaction to warning messages. When the in-vehicle DSRC device receives broadcasted messages, it sends these messages to an in-vehicle electronic control unit (ECU). The ECU processes these messages and issues an alert to the driver if there is a possible collision with another vehicle at the intersection. The alert could be a combination of audible and indicator warnings. Humans perceive alerts and warning indicators differently from each other and with different response time. In this case, we consider the brake reaction time of drivers. The brake reaction time is the time between recognizing the alert and applying the brake. It is generated for each vehicle using the results of [14] by using a lognormal distribution with a mean of 0.67 s and a standard deviation of 0.12 s.

For each generated vehicle, a maximum speed and a maximum deceleration are also generated. The speed of vehicles is generated using a uniform distribution with a mean of 55 km/h and a standard deviation of 10 km/h. Similarly, the maximum decelerations of vehicles are generated using a uniform distribution with a mean of \( -4 \text{ m/s}^2 \) and a standard deviation of \( -0.5 \text{ m/s}^2 \). For all generated vehicles, we assume the length of a vehicle is 5 m.

### C. Vehicles Interactions:

Vehicles enter the network at the entry trigger line using their arrival time. Once a vehicle enters the network, it starts the IEEE 802.11a MAC protocol. If a vehicle at one of the intersection legs wins the channel, it broadcasts its velocity, heading, and position using its omni directional antenna and one of the assigned DSRC channels for the intersection collision avoidance application. We assume there is a device installed at the intersection that sends the ACK signal, as shown in Fig. 1. Hence, the basic access mechanism of the DCF in 802.11 is considered in this simulation. Table III shows the data packets and data rate used in this protocol.

![Image](image_url)

**Fig. 1.** The designed intersection structure. The entry trigger line triggers the vehicles to start contending for the channel. The exit trigger line triggers the vehicles to stop contending for the channel.

**TABLE II Vehicles’ and drivers’ characteristics**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Distribution</th>
<th>Mean: ( \mu )</th>
<th>Standard deviation: ( \sigma )</th>
</tr>
</thead>
<tbody>
<tr>
<td>vehicle’s arrival time</td>
<td>shifted negative exponential distribution</td>
<td>( \mu = 3.0 \text{s} )</td>
<td>min = 1.5 s</td>
</tr>
<tr>
<td>vehicles’ maximum speed</td>
<td>normal distribution</td>
<td>( \mu = 55 \text{ km/h} )</td>
<td>( \sigma = 10 \text{ km/h} )</td>
</tr>
<tr>
<td>vehicles’ maximum deceleration</td>
<td>normal distribution</td>
<td>( \mu = -4 \text{ m/s}^2 )</td>
<td>( \sigma = -0.5 \text{ m/s}^2 )</td>
</tr>
<tr>
<td>drivers’ break reaction time</td>
<td>lognormal distribution</td>
<td>( \mu = 0.67 \text{s} )</td>
<td>( \sigma = 0.12 \text{s} )</td>
</tr>
</tbody>
</table>

At the beginning of the simulation, the intersection is empty of vehicles. Vehicles are generated into the network at the entry trigger line for each intersection leg. Table II summarizes the characteristics of vehicles and drivers. Vehicles are generated into the network based on headway distributions among vehicles. To generate vehicles and simulate the arrival of vehicles at the entry trigger line, we used the shifted negative exponential distribution with a mean headway of 3.0 s and a minimum headway of 1.5 s. The arrival time for a vehicle in a lane is the summation of headways of its leading vehicles at that same lane. Here, we used the results obtained by [15] that the preferred time-headway is between 1 and 2s, and we took the average of 1.5 s as the minimum headway.
TABLE III Date rate and packets transmitted in 802.11a.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data bit rate</td>
<td>54 mbps</td>
</tr>
<tr>
<td>802.11 Header</td>
<td>34 bytes</td>
</tr>
<tr>
<td>802.11 ACK</td>
<td>14 bytes</td>
</tr>
<tr>
<td>Data packet</td>
<td>54 bytes = 2 bytes (speed) + 48 bytes (x-y-z position) + 1 byte (lane no.) + 1 bytes (direction) + 2 bytes (deceleration)</td>
</tr>
</tbody>
</table>

Vehicles from the other three intersection legs receive the broadcasted data using their in-vehicle DSRC device. Using the received data and its own speed, heading and position, the ECU in the receiving vehicle determines if its vehicle will be in the collision zone with the transmitting vehicle, as shown in Fig. 2. We assume the dimension of the collision zone is 10m × 10m. The ECU issues an alert to the driver, and the driver decelerates after a break reaction time. The break reaction time is applied only once when a driver receives the first alert. After the first alert, we assume the driver has been alerted, and the break reaction time will not be applied again with subsequent alerts on this intersection.

![Collision Zone Diagram](image)

Fig. 2. The collision zone is used to determine if two vehicles will be in the zone to issue an alert to the receiving vehicle.

D. Car following:

When a preceding vehicle is slower than the following vehicle, the following vehicle decelerates to match the speed of the preceding vehicle. This is similar in operation to the adaptive cruise control (ACC) system. In the meantime, if the following vehicle receives an alert of a possible collision with another vehicle traveling from another intersection leg, then this alert may take precedence over the ACC system. The in-vehicle ECU determines if the deceleration of the driver due to ACC is enough to issue the alert. The alert will be issued if the driver needs to decelerate further in order to avoid a collision at the intersection. In our simulator, the car-following system is independent of the intersection collision avoidance system. The received broadcasted messages are used to activate the intersection collision avoidance system and not the car-following system.

VI. SIMULATION RESULTS

Before running the simulator, we set `dot11shortretrylimit` to one of the values \{3, 5, 7\}, we set `aCWmax` to 960, and we set `aCWmin` to one of the values \{5, 30, 60, 120, 240, 480\}. Hence, we have 18 combinations of `dot11shortretrylimit` and `aCWmin`. For each one of these combinations, we ran the simulator 30 times for statistical analysis. Although each simulation-run takes several hours to complete, the real-time traffic operation of the system is set to 60 s in the simulator. In other words, if this system is implemented and a field test is conducted, then the system should be run for 60 s to collect the necessary data and be compared with the simulation data.

For each of the 18 combinations and 30 simulation runs, we set the entry trigger line to 50 m, 100 m and 150 m. We collected the bandwidth for each vehicle, the wait time for each vehicle, and the number of vehicle collisions. The following is a description of our simulation results.

A. Distance to the Intersection is 50 m

In [7][8], vehicles contend for the channel when they are 50 m away from the intersection. To validate whether or not vehicles can avoid a collision when they are 50 m away from the intersection, we set the entry trigger line (distance to the intersection, \(d_i\)) at 50 m in our simulator. After running the simulator, as described in the beginning of this section, we recorded the number of vehicle collisions at the intersection. Fig. 3 summarizes the average number of vehicle collisions for each setting of `aCWmin` and `dot11shortretrylimit`. The vehicles that were involved in collisions were traveling at an average speed of ~55 Km/h.

![Average Number of Collisions](image)

Fig. 3. Average number of vehicle collisions when the distance to the intersection is 50 m.

Therefore, setting the entry trigger line at distance \(d_i\) to 50 m is insufficient for vehicles to decelerate and avoid collisions, per our assumed drivers’ characteristics in Table III.

B. Distance to the Intersection is 100 m

We then increased the distance to the intersection, \(d_i\), to 100 m. Fig. 4 summarizes the average number of vehicle collisions at the intersection for each setting of `aCWmin` and `dot11shortretrylimit`. The recorded vehicle collisions in this setup were due to vehicles traveling at an average low speed of ~20 Km/h.
and retry again by choosing a new number of retries $P$ between 0 and 960 since $CW$ will eventually stays at 960 regardless of the value of $dot11shortretrylimit$. Therefore, from Fig. 8-10, setting $aCWmin$ to a value greater than 120 has a negligible affect on the setting of $dot11shortretrylimit$.

Notice from Fig. 5-10 that setting $dot11shortretrylimit$ to 7 always gives the minimum average wait time. We show in Fig. 11 the average wait time when $dot11shortretrylimit$ is set to 7 for $aCWmin$ 15, 30, 60, 120, 240 and 480. As expected, notice from Fig. 11 that the average wait time decreases for large number of vehicles (> 30) when $aCWmin$ increases from 15 to 60. The main reason is that the number of data collisions decreases with a wider range of random numbers. Further increase in $aCWmin$ to 240 causes a continuous decrease in average wait time.

When the number of vehicles is less than 30, then the average wait time increases when $aCWmin$ increases from 15 to 480. The reason is that choosing a large random number causes a vehicle to sense for an idle channel for a longer time.

From Fig. 11 we conclude that setting $aCWmin$ to 120 would provide a better choice than other settings for achieving an average minimum wait time.

Notice from Fig. 5 that as the number of $dot11shortretrylimit$ increases, the average wait time decreases. In this setup, $CW$ will initially be set to $aCWmin$ value of 15. Vehicles choose a random number $P$ between 0 and $CW$=15. When several vehicles have the same $P$ and then a data collision occurs, these vehicles will double the $CW$ value to 30 and retry again by choosing a new $P$ between 0 and $CW$=30. Thus, the range of random numbers between zero and the new $CW$ keeps increasing as the number of retries increases. Thus increasing $dot11shortretrylimit$ provides a wider range of random numbers and reduces the number of data collisions accordingly. When the number of data collisions is reduced, vehicles will wait less time to win the channel and transmit its safety-critical information to nearby vehicles.

Fig. 6-10 show the average wait time per vehicle when $aCWmin$ is set to 30, 60, 120, 240, and 480, respectively. If we look carefully to these figures (Fig. 6 through Fig. 10), notice that the gap between $dot11shortretrylimit$ decreases as $aCWmin$ is increased. Recall from Section II that the value of $CW$ stays at $aCWmax$ until a successful transmission or the number of retries $dot11shortretrylimit$ is reached. So, when $aCWmin$ increases and gets closer to $aCWmax$ of 960, a vehicle has to choose a random number $P$ between 0 and 960 and retry again by choosing a new number of retries $P$ between 0 and 960 since $CW$ will eventually stays at 960 regardless of the value of $dot11shortretrylimit$. Therefore, from Fig. 8-10, setting $aCWmin$ to a value greater than 120 has a negligible affect on the setting of $dot11shortretrylimit$.

Notice from Fig. 5-10 that setting $dot11shortretrylimit$ to 7 always gives the minimum average wait time. We show in Fig. 11 the average wait time when $dot11shortretrylimit$ is set to 7 for $aCWmin$ 15, 30, 60, 120, 240 and 480. As expected, notice from Fig. 11 that the average wait time decreases for large number of vehicles (> 30) when $aCWmin$ increases from 15 to 60. The main reason is that the number of data collisions decreases with a wider range of random numbers. Further increase in $aCWmin$ to 240 causes a continuous decrease in average wait time.

When the number of vehicles is less than 30, then the average wait time increases when $aCWmin$ increases from 15 to 480. The reason is that choosing a large random number causes a vehicle to sense for an idle channel for a longer time.

From Fig. 11 we conclude that setting $aCWmin$ to 120 would provide a better choice than other settings for achieving an average minimum wait time.
Fig. 8. Average wait time per vehicle when $aCWmin$ is set to 120.

Fig. 9. Average wait time per vehicle when $aCWmin$ is set to 240.

Fig. 10. Average wait time per vehicle when $aCWmin$ is set to 480.

2) Average Available Bandwidth:

We study next the average available bandwidth per vehicle in this setup. Fig. 12-17 show the average bandwidth per vehicle when $aCWmin$ is set to 15, 30, 60, 120, 240, and 480, respectively. Notice first that in all figures, the average bandwidth keeps increasing until about 10 vehicles are approaching the intersection. As the number of vehicles increases, the average bandwidth starts to decrease. The initial increase in bandwidth is due to the low occurrences of data collisions when a small number of vehicles (an average of less than 10 vehicles) contended for the channel. In our next analysis, we will study the available bandwidth when the number of vehicles approaching the intersection is greater than 10 vehicles.

We concluded in our wait-time analysis that setting $aCWmin$ to 120 would provide vehicles with an average wait time between 12 ms and 14 ms regardless of the setting of $dot11shortretrylimit$. If we study Fig. 15-17 that correspond to $aCWmin$ of 120, 240 and 480, respectively, we notice that the average available bandwidth (when the number of vehicles is 10) decreases as $aCWmin$ increases. For example, in Fig. 15, when $aCWmin$ is set to 120, the average bandwidth is $\sim$ 40 KBps when the number of vehicles is 10. Similarly, in Fig. 16, the average bandwidth is $\sim$ 27 KBps when $aCWmin$ is set to 240. Also in Fig. 17, the average bandwidth is $\sim$ 17 KBps when $aCWmin$ is set to 480. The main reason of this decrease in bandwidth is the amount of time a vehicle spends sensing the channel. As $aCWmin$ increases, a vehicle chooses a random number $P$ from a wider range of values. Hence, choosing a large random number causes a vehicle to sense for an idle channel for a longer time.

Furthermore, notice in Fig. 15-17 that as the number of vehicles increases to 70 vehicles, the average bandwidth decreases to $\sim$ 5 KBps, regardless of the setting of $aCWmin$ and the setting of $dot11shortretrylimit$. Recall again from our wait-time analysis and from Section II that the value of $CW_{st}$ stays at $aCW_{max}$ until a successful transmission or the number of retries $dot11shortretrylimit$ is reached. So, when $aCWmin$ increases and gets closer to $aCW_{max}$ of 960, a vehicle has to choose a random number $P$ between 0 and 960. Therefore, setting $aCWmin$ to a value greater than 120 has a negligible affect on the available bandwidth at large volume of vehicles ($\sim$ 70 vehicles).

Finally, we show in Fig. 18 the combined average available bandwidth when $dot11shortretrylimit$ is set to 7 for $aCWmin$ 15, 30, 60, 120, 240 and 480. Notice from this figure and Fig. 11 (the combined average wait time) that there is a bandwidth-wait time tradeoff for setting $aCWmin$. If a higher bandwidth is desired (when the number of contending vehicles is less than 20), we need to set $aCWmin$ to a value less than 120. Setting $aCWmin$ to a value less than 120 would provide a longer wait time as the number of vehicles increases.
Finally, we increased the distance to the intersection, $d_i$, to 150 m. We recorded that there are no vehicle collisions at the intersection. We also recorded the average wait time per vehicle and the average available bandwidth per vehicle. We noticed that increasing the distance $d_i$ by 50 m has a negligible effect on the average wait time and the average available bandwidth per vehicle compared with setting the distance $d_i$ to 100 m.

C. Distance to the Intersection is 150 m

Finally, we increased the distance to the intersection, $d_i$, to 150 m. We recorded that there are no vehicle collisions at the intersection. We also recorded the average wait time per vehicle and the average available bandwidth per vehicle. We noticed that increasing the distance $d_i$ by 50 m has a negligible effect on the average wait time and the average available bandwidth per vehicle compared with setting the distance $d_i$ to 100 m.

Fig. 12. Average available bandwidth per vehicle when $aCWmin$ is set to 15.

Fig. 13. Average available bandwidth per vehicle when $aCWmin$ is set to 30.

Fig. 14. Average available bandwidth per vehicle when $aCWmin$ is set to 60.

Fig. 15. Average available bandwidth per vehicle when $aCWmin$ is set to 120.

Fig. 16. Average available bandwidth per vehicle when $aCWmin$ is set to 240.

Fig. 17. Average available bandwidth per vehicle when $aCWmin$ is set to 480.

Fig. 18. Average available bandwidth when $dot11shortretrylimit$ is set to 7 for $aCWmin$ 15, 30, 60, 120, 240 and 480.
VII. CONCLUSION

In this paper, we developed an intersection traffic simulator to find the optimal parameters settings of the IEEE 802.11 DCF on the average wait time for a vehicle to win the channel and transmit its safety-critical information to nearby vehicles, and on the average available bandwidth per vehicle. The two parameters that we evaluated are the number of retries dot11shortretrylimit and the lower bound limit aCWmin for randomly choosing the number of idle time slots to sense the channel. From our simulation results, we can conclude that setting aCWmin to 120 and dot11shortretrylimit to any of the values \{3, 5, 7\} would suffice. Setting aCWmin to 120 would provide vehicles with a minimum wait time of \(~12 \text{ ms}\).

There is a bandwidth-wait time tradeoff for setting aCWmin when designing an intersection collision avoidance system. A higher bandwidth can be achieved by setting aCWmin to a value less than 120. However, this can only be fulfilled if the number of vehicles approaching an intersection is less than 20. As the number of vehicles increases, the bandwidth will be reduced to that of aCWmin=120, but the average wait time will increase.

We finally conclude that vehicles would need \(~150 \text{ m}\) from an intersection to operate the DSRC MAC protocol and avoid collisions at the intersection. This distance will provide drivers with sufficient time to respond to alerts and warning indicators without panic and sudden large decelerations.

REFERENCES