Dynamic Channel Selection for Real-time Safety Message Communication in Vehicular Networks

Yunhao Bai†, Kuangyu Zheng†, Zejiang Wang*, Xiaorui Wang†, and Junmin Wang*
†Department of Electrical and Computer Engineering, The Ohio State University, USA
*Department of Mechanical Engineering, University of Texas at Austin, USA

***Abstract*—Ensuring the real-time delivery of safety messages is an important research problem for Vehicle to Vehicle (V2V) communication. Unfortunately, existing work relies only on one or two pre-selected control channels for safety message communication, which can result in poor packet delivery and potential accident when the vehicle density is high. If all the available channels can be dynamically utilized when the control channel is having severe contention, safety messages can have a much better chance to meet their real-time deadlines. In this paper, we propose *MC-Safe*, a multi-channel V2V communication framework that monitors all the available channels and dynamically selects the best one for safety message transmission. MC-Safe features a novel channel negotiation scheme that allows all the vehicles involved in a potential accident to work collaboratively, in a distributed manner, for identifying a communication channel that meets the delay requirement. Our evaluation results both in simulation and on a hardware testbed with scaled cars show that MC-Safe outperforms existing single-channel solutions and other well-designed multi-channel baselines by having a 12.31% lower deadline miss ratio and an 8.21% higher packet delivery ratio on average.**

I. Introduction

Enhancing driving safety is a major objective of the current research on Vehicle to Vehicle (V2V) communication [1][2], due to the fact that car accidents cost nearly 1.3 million lives every year [3]. By exchanging safety messages with critical vehicle information (e.g., car speed, location, direction), vehicles can be notified in real time if there is a potential accident. Such safety considerations have already been included in the current V2V standards, such as Wireless Access in Vehicular Environments (WAVE), a widely adopted vehicle communication protocol. The proposed WAVE protocol provides both safety and data services on the Dedicated Short-Range Communication (DSRC) band, using the IEEE 802.11p standard [4]. According to the requirement of the US government, every vehicle should be equipped with DSRC device to enhance road safety.

In order to deliver safety messages in a timely manner, delay is one of the most important requirements for V2V communication. Generally, there are two types of safety messages: Periodic safety messages (e.g., GPS location, speed) and event-driven safety messages (e.g., driving actions like braking and lane changing) [5]. Both types have some deadline requirements, but the event-driven messages commonly have more stringent deadlines. For example, on a crowded highway, missing the deadline of a safety message that alerts a sudden braking action may cause a rear-end collision. Similarly, if a lane-changing message is not delivered in real time, unsafe lane merging could cause severe accidents. The V2V communication deadline for transmitting event-driven safety message can be as short as 20ms [5], based on the car speeds and their distance to each other. Even for periodic messages, some soft deadline requirements are also necessary, because these messages contain time-sensitive critical information about the vehicle's dynamics, such as velocity, yawing rate, and position, which may become outdated after a short period of time. The recommended transmission frequency for periodic safety messages is at least 10Hz in WAVE standard.

However, meeting the V2V communication deadline is challenging, particularly when the vehicle density is high. For example, the vehicle density of a downtown area is commonly more than 5,000 vehicles per square mile and can be 1.5 times during the rush hours [6]. Such a high density can cause safety messages to miss their deadlines, due to significant wireless channel contentions on the control channel used by the current WAVE (802.11p) protocol, for two major reasons: First, when more vehicles compete for the limited bandwidth resources, the packet delay can become unbounded with the CSMA (Carrier-Sense Multiple Access) mechanism. Second, a high vehicle density can lead to a higher chance of having the well-known hidden terminal problem, which in turn can result in more packet dropping at the receiver vehicle. Such a channel contention problem is mainly due to the fact that WAVE is designed to transmit safety messages only on the control channel, despite that seven non-overlapping channels are actually available in the DSRC band for V2V communication. If other channels can also be dynamically utilized when the control channel is having severe contention, safety messages can have a much better chance to meet their real-time deadlines.

Most existing work on improving the communication performance of safety messages focuses on adapting transmission rate and power [7][8][9], as well as message priority or period [10][11][12]. Although those methods can be effective when the vehicle density is low, they still transmit all the safety messages on the single control channel where the total network capacity is limited. There are indeed some recent studies that investigate multiple channels for safety message communication [13][14][15]. However, instead of dynamically selecting the best channel for real-time communication, they

This work was supported, in part, by NSF under grants CPS-1645657 and CNS-1421452.

2576-3172/18/$31.00 ©2018 IEEE
DOI 10.1109/RTSS.2018.00016

mainly have two pre-selected channels for safety and non-safety messages, respectively. For example, some studies have proposed to adjust the time interval length of staying on the control channel, in order to improve the vehicle's safety message transmission rate [13][14]. Due to the limited channel choices, they can still have inferior performance when the vehicle density is high. To our best knowledge, existing work does not *dynamically* select the best channel from all the channel resources provided by DSRC to improve the *real-time performance* of safety messages.

In this paper, instead of utilizing only one or two pre-selected channels for safety message communication, we propose *MC-Safe*, a multi-channel V2V communication framework that monitors all the available channels and dynamically selects the best one for safety message transmission in an emergency scenario. MC-Safe features a novel channel negotiation scheme that is activated whenever two or more cars are estimated to have a potential accident. All the involved cars work collaboratively in a distributed manner to identify a channel that can meet the specified delay and packet error ratio requirements for every car. Afterward, all the involved cars switch to the selected channel for real-time safety message communication, without suffering the interference from other cars in the vehicular network. MC-Safe is robust to varying channel conditions because it can automatically adapt its model to better estimate channel delays. Our evaluation results both in simulation and on a hardware testbed with scaled cars show that MC-Safe outperforms existing single-channel solutions and other well-designed multi-channel baselines, by having a 12.31% lower deadline miss ratio and an 8.21% higher packet delivery ratio on average. Compared to WAVE, the state-of-the-practice solution, MC-Safe successfully reduces the average delay of safety message transmission from 300ms to 20ms when the vehicle density is high.

Specifically, this paper makes two major contributions:

- We observe that the existing work on V2V safety message communication focuses only on one or two pre-selected channels, and thus may have a low packet delivery ratio and a high deadline miss ratio when the car density is high. Accordingly, we propose to explore all the seven available non-overlapping channels for better real-time performance and a lower chance of having collisions.
- We design MC-Safe, a multi-channel V2V communication framework that monitors all the available channels and dynamically selects the best one for safety message transmission. MC-Safe features a novel channel negotiation scheme that lets all the involved cars work collaboratively to identify a channel that meets the desired delay and packet error rate requirements.

The rest of the paper is organized as follows: Section II discusses the related work. Section III motivates our work by comparing message transmission on one single channel or multiple channels under different road situations. Section IV introduces the design of MC-Safe. Section V presents the evaluation results. Section VI concludes the paper.

II. Related Work

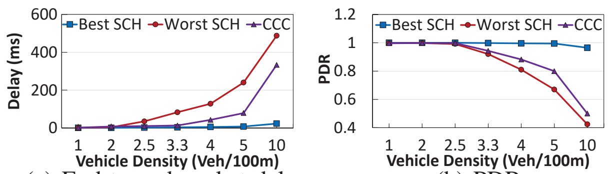
Due to the complex wireless environment of V2V network, many studies conduct analysis of the general transmission performance in current V2V WAVE protocol [16][17][7][18]. Campolo et al. have shown that the transmission delay can increase dramatically when the vehicle density becomes high [17]. On a typical road intersection with only 50m distance between vehicles, the control channel can be saturated [18]. These studies demonstrate that the current 802.11p standard is not sufficient for the transmission of delay-sensitive data, especially for the V2V real-time message communication.

Meanwhile, many recent studies are proposed to improve safety message transmission in the current V2V network based on WAVE standard. One major direction is alleviating the control channel workload by adjusting the message transmission rate or power [7][8][9]. Some studies also try to utilize other knobs, such as message priority, beaconing frequency or duplicated packets [10][12][19][13][20]. For example, Xiang et al. [12] propose to add priorities to different messages to avoid collision on the control channel. Although the aforementioned methods improve the performance of safety message broadcasting, they still transmit on the single control channel where the total network capacity is limited.

Recently, some studies begin to consider safety message broadcasting under the multi-channel scenarios [14][21][15]. For example, CRN-VANETs [21] aims to reduce data contention in the control channel but it does not consider the stringent time requirements in a potential accident. Ghandour et al. propose to form a sub-network for each channel, in order to deliver the event-driven message in time [15]. Yao et al. proposes to calculate the optimal bandwidth resource allocation for multi-channel V2V network. However, those studies mainly use one or two pre-selected channels for safety and non-safety message transmission without explicit delay consideration, and do not consider real-time transmission requirements. In sharp contrast, MC-Safe dynamically selects a channel that meets the delay and packet error rate requirement, through distributed channel negotiation among vehicles.

III. Motivation

We now motivate the design of MC-Safe by investigating the real-time communication performance of WAVE, the state-of-the-practice, in a typical road scenario. We use the ns-2 network simulator to test the WAVE multi-channel access algorithm and associated lower layer functions. For the test, we consider a six-lane highway: Each lane's width is four meter, and the total road length is 1000m. There are six Road-Side Units (RSU) serving as receivers on the service channels (SCHs, i.e., channels other than the control channel), and the RSU is equipped with multiple radios so it can work on all the available service channels. We use Poisson distribution to model the distance of vehicles on the road, which is widely used for the road traffic analysis [22]. The vehicle density is set with parameter λ (the average inter-vehicle distance) of Poisson distribution; The transmission interval is set to be 20ms and the packet size is 300 bytes. Vehicles on the common control channel (CCC) are transmitting beacons with



(a) End-to-end packet delay (b) PDR

Fig. 1: Average packet transmission delay and Packet Delivery Ratio (PDR) under a fixed service vehicle ratio=0.8.

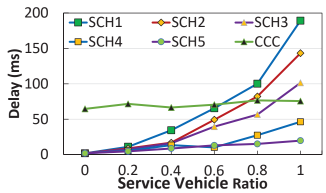


Fig. 2: Average packet transmission delay of each SCH and the CCC given fixed vehicle density 2000 veh/mile^2.

a frequency of 10Hz, and the transmission interval of non-safety services is 20ms. We apply a realistic V2V network propagation model measured at 5.9GHz band [23]. In the motivation test, we choose the vehicles of interest as two neighboring cars in the middle of the road segment. We define the ”service vehicle ratio” as the number of vehicles that are transmitting on channels other than the CCC divided by the total number of vehicles on the road. The vehicle using non-safety services is randomly assigned to one channel among SCH1 to SCH5. We adjust the vehicle density and service vehicle ratio to test different traffic scenarios.

For the real-time vehicle control system for accident prevention, a typical control period is usually 20ms [5], and every packet needs to be received within its period to have the correct control action. We first evaluate the case that the two cars are transmitting on the CCC, which is also used by all the other cars at the same time. As shown in Figure 1, the average package delay of using the CCC already becomes longer than 20ms when the vehicle density is just 4 veh/100m (about 1600 per square mile), which is lower than the 2000 veh/mile^2 vehicle density in most suburban areas [6]. In fact, the delay increases almost exponentially due to the Distributed Coordination Function (DCF) mechanism used in the 802.11 protocol. Note that a message delivered after the 20ms delay requirement is outdated for real-time vehicle control and could even be misleading. Meanwhile, the Packet Delivery Ratio (PDR) of using the CCC drops significantly as the vehicle density increases. Thus, transmitting safety messages only on the single control channel can result in a long delay with a poor PDR. On the other side, the delay and PDR of the best service channel (i.e., the service channel with the best delay or PDR result) remain small when the vehicle density increases. The reason is that the best service channel is far less crowded compared to the CCC. However, in the worst case, a service channel could perform even worse than the CCC, indicating that the channel selection upon an emergency scenario is not trivial: A bad choice of the service channel could harm the performance of the real-time vehicle control system.

Figure 2 shows the average packet delay for each channel under a fixed vehicle density 2000 veh/mile^2. As service

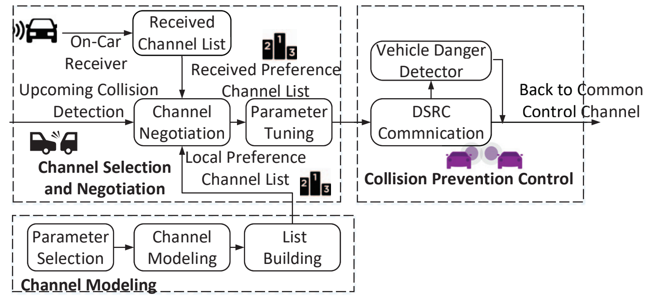


Fig. 3: Overview of the MC-Safe framework.

vehicle ratio increases, the difference between different service channels become larger (from 0.23ms to 170ms). When the ratio is below 0.6, almost all service channels can outperform the CCC. However, when service vehicle ratio increases, the service channel must be chosen carefully, otherwise it will have even worse performance, such as the cases using SCH1, SCH2 and SCH3. This evaluation provides us a strong motivation to consider using other channels instead of the CCC to do real-time safety message transmission, especially when the vehicle density is high (i.e., higher chance to have accident), and to choose the channel carefully to avoid long delay and poor PDR.

IV. Design of MC-Safe

In this section, we first introduce the overview of MC-Safe, with its general work flow and major components. We then introduce the detailed design of each part.

A. Design Overview

MC-Safe aims to dynamically select the best channel for real-time safety message transmission in a pre-crashing scenario. Figure 3 shows the overview of MC-Safe. Generally, when a possible collision for cars is detected (with the vehicle trajectory predictor), distributed MC-Safe on the involved cars will start negotiation to find the best channel between cars, establish the communication in a short time, and enable real-time and reliable safety message exchange for collision prevention control. When the dangerous condition is resolved (e.g., car distance becomes longer than a threshold), every car will change its communication back onto the CCC. MC-Safe realizes these functions with two major components: 1) Channel Modeling and 2) Channel Selection and Negotiation.

Channel Modeling. Channel modeling component on each vehicle is conducted periodically to estimate the packet delay and delivery ratio on all channels, with the delay and Packet Delivery Ratio (PDR) requirements of the safety message. Based on such information, it evaluates the conditions of all the available channels and constructs a local Channel Preference List (CPL), which is ordered by channel quality from the best to the worst, as the input to the channel selection and negotiation component.

Channel Selection and Negotiation. This component is invoked before any potential accident to find the best common channel for the vehicles involved in the potential accident to perform real-time safety message communication. After a potential accident is detected, based on their own CPLs and the CPLs received from other vehicles, the vehicles involved

TABLE I: Typical Signal Deadline Requirement in CAN.

Message Type	Deadline	Jitter	Actuator
Steering Control	5ms	0.2ms	Motor Controller
Speed Control	20ms	0.7ms	Vehicle Controller
Emergency Brake	40ms	0.5ms	Brake
Shift in Progress	20ms	1.4ms	Motor Controller

in the accident start the negotiation process quickly to select the channel with the best quality for all the involved vehicles. If the selected channel cannot meet the requirements of safety message transmission, MC-Safe will tune and suppress the non-safety transmission of other vehicles on the selected channel to meet the requirements.

The key parts in the design of MC-Safe are to 1) estimate the communication quality of each channel by channel modeling (Section IV.B), and 2) find the best channel among all involved vehicles by channel negotiation and selection (Section IV.C). The important notations used in the rest of the design of MC-Safe are listed as follows:

- p: Probability of transmission failure of one packet.
- T_i: Transmission interval of the safety packet.
- d_i: The delay of a single packet.
- Ω: The Maximum allowed interval between two consecutive packets.
- δ, Δ: Threshold probabilities for packet delivery ratio requirement and delay requirement, respectively.
- p_b: The probability of backing off in the DCF mechanism.
- CW_i: The maximum backoff counter of i^{th} retransmission.

B. Channel Modeling

Channel modeling aims to determine whether a given channel can meet the real-time requirements of the collision prevention control system. We first introduce how to quantify the requirements of the control system, then we introduce how to estimate the performance for each channel in the real world.

1) Delay Requirement: Different emergency conditions can have different requirements of delay and Packet Delivery Ratio (PDR). For example, the adaptive cruise control system or hard brake reaction system require a 25Hz sampling rate [24]; the lateral motion control proposed in [25] requires a sampling rate of 50Hz. In order to determine the requirement for each element in vehicle dynamics, we derive the deadlines from the Control Area Network (CAN) scheduling standard [26]. The requirement for each signal in the CAN system is listed in Table I. With the consideration of variation, requirements are set as the deadline listed in Table I minus the jitter.

2) Channel Modeling: First, we use two well-known constraints adopted by many networked control systems (e.g., autonomous vehicles, industrial automation, and robots [27][28]) to evaluate the performance and stability of each channel: a) Maximum Allowable Transfer Interval (MATI) and b) Maximum Allowable Delay (MAD) requirement. However, due to specific V2V conditions in our case (i.e., sophisticated backoff mechanism in 802.11p), we further revised the formulation, with details presented later in Equations (5)(7). The original constraints are defined as follows:

$$\begin{aligned} \text{MATI}: &\quad 1 - p^{\lceil \frac{\Omega}{T_i} \rceil} \geq \delta \\ \text{MAD}: &\quad d_i \leq \Delta \end{aligned} \tag{1}$$

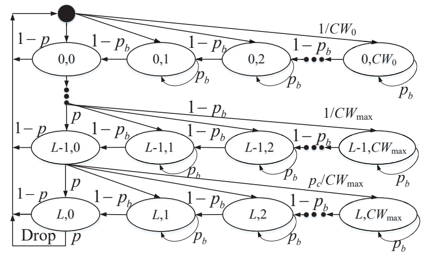


Fig. 4: Markov chain model of 802.11 backoff mechanism.

where the thresholds (Ω, δ and Δ) are determined by the control system. Specifically, Δ is determined from the CAN standard for automobiles, and Ω is determined by the recommended value in vehicular safety applications [28]. Generally speaking, MAD requires that a packet should not have a delay larger than the threshold Δ, and MATI requires that the probability is larger than δ to receive at least one packet successfully in the T_i period of time.

To derive the transmission failure probability p and packet delay d_i in Equation (1), we model the backoff procedure in 802.11p as a Markov process (shown in Figure 4) [29], which is proved to be sufficiently accurate compared to the performance measured in real world [29][30][31]. Base on the analysis proposed by Yao et al. [16], the backoff probability p_b and failure probability p shown in Figure 4 can be calculated as follows:

$$p_b = 1 - e^{-N_c \tau} \tag{2}$$

$$p = 1 - e^{-N_c \tau} e^{-N_h \tau T_v / \sigma} \tag{3}$$

where N_c is the number of vehicles in the carrier sensing area of current considered channel; N_h is the number of hidden terminals. T_v is the hidden-terminal vulnerable period; τ is the transmission probability in one slot, whose value can be found in [29] with p_b. σ is the length of a time slot. Since we are modeling the delay on the SCH, we assume that there are always packets in the MAC queue waiting for transmission [13]: The saturated situation can give us the upper bound of these parameters. We also assume the packets to be sent has the highest priority in Enhanced Distributed Channel Access (EDCA) because they contain safety-related data.

Based on the above models, we propose to calculate the total packet delay d_i in the following way, which is the summation of two parts: The service delay $\frac{1}{\mu}$ and the queuing delay $\mathrm{E}[W]$. Using the value of p_b and p, we can calculate $\frac{1}{\mu}$ as the summation for the backoff delay for each retransmission multiplied by its corresponding probability. Mathematically, $\frac{1}{\mu}$ can be calculated as:

$$\frac{1}{\mu} = \sum_{i=0}^{L} (1-p) p^i \frac{\min(2^{\mathrm{i}} CW_0, CW_{max})}{2} \mathrm{E}[X] \tag{4}$$

where $\mathrm{E}[X]$ is the expected time for the backoff counter decreasing by one, and it can be calculated based on probability of a time slot being idle p_i, being busy p_b, having a collision p_c and respective time values. $(1-p)p^i$ represents the probability of one packet going through i times of retransmission, and the other terms in the summation represent the expected delay for the i^{th} retransmission, i.e., s_i. The queuing delay $\mathrm{E}[W]$ can be directly calculated with M/G/1 queue model [16].

However, in Equation (1), T_i is not a constant value in 802.11p and cannot be used directly. Even though in the 802.11 backoff mechanism, it has a default upper bound retransmission times L shown in Figure 4, there are specific related delay requirements for the V2V real-time communication. Therefore, we propose to derive a new form of MATI requirement compatible with 802.11p. For MC-Safe we define a new parameter, the maximum allowable times of retransmission L_m, which can be calculated as:

$$\arg\max_{L_m} \sum_{i=0}^{L_m} s_{\mathrm{mod}(i,L)} <= \Omega \tag{5}$$

The expected delay s_i for i^{th} retransmission is:

$$s_i = \frac{\min(2^i CW_0, CW_{max})}{2} \mathrm{E}[X] \tag{6}$$

then MATI in Equation (1) can be re-written as follows:

$$1 - p^{L_m} \geq \delta \tag{7}$$

L_m is easy to get according to the requirements and has solid physical meaning in 802.11p compared to T_i. By using above proposed constraint in Equation (7), MC-Safe on every vehicle can efficiently conduct the above channel analysis distributively by checking the number of vehicles within one-hop distance N_c, and the number of hidden terminal from the beacons of one-hop neighbors N_h on each channel.

3) Model Adaptation: The above channel estimation utilizes the N_c and N_h information from the beacon messages. However, sometimes the dynamic road environment can be complex and the beacon messages can be interfered with errors. Therefore, we further design a dynamic model adaptation scheme in MC-Safe to improve the channel estimation accuracy. To estimate the packet drop rate and delay, besides using the beacon messages, MC-Safe also measures the network status by leveraging the PHY layer sensing information to adjust idle probability p_i, collision probability p_c and busy probability p_b. In each period, it monitors each channel and collects the counts of the idle slot, the busy slot and the collision slot, then calculates the empirical probabilities for p_i', p_c' and p_b'. Then, to get the final results for adaptation (i.e., p_i^f, p_c^f and p_b^f) in the coming period, MC-Safe uses a weighted summation of the theoretical values and empirical values in the previous period with a dynamic factor α, e.g., p_i^f is updated as follows:

$$p_i^f = \alpha p_i + (1 - \alpha) p_i' \tag{8}$$

To determine the value of α, MC-Safe periodically uses the monitored empirical probabilities (p_i^f, p_c^f, p_b^f) in the current period as the ground truth, and calculates the value of α that provides the best estimation accuracy. Then, the updated α is used in the estimation for the next coming period. This adaptation process is applied continuously, so that the estimation can be updated in a timely manner to reflect the current road situation. The adaptation period can be set based on different factors like channel conditions and road situations.

With the above scheme, MC-Safe can then select the channel meeting MATI and MAD requirements by combining the theoretical model and empirical statistics. To further reduce the computation overhead, we build an offline lookup table to store the theoretical values related to N_c and N_h. Each element in the table stores p_i, p_c and p_b for one specific N_c and N_h pair. With the look-up table, the modeling process can be finished within 13μs, which is smaller than the required time for generating Acknowledgment (ACK) frame .

4) Channel Preference List: After we get both the estimated delay and the error rate for each channel, we can categorize it into one of three types:

- Type 1: This channel meets both the MATI and MAD requirements.
- Type 2: This channel does not meet the MATI and MAD requirements at the same time. However, by suppressing the transmission of other nodes within the one-hop distance, this channel may become a Type-1 channel.
- Type 3: This channel can meet either the MATI or MAD requirements unless the two-hop neighbors are suppressed because of hidden terminals.

The building process of CPL is not event-triggered. Note that building up CPL may incur considerable computation overheads. Thus, instead of building CPL when channel negotiation is needed, we maintain CPL periodically based on the beacons from other vehicles and the adaptation algorithm to incrementally adjust the channel types. From the beacons, we can get the information like the vehicle dynamics, MAC address and SCH the vehicle is using of each near-by vehicle.

C. Channel Selection and Negotiation

Here, we introduce our proposed approach for real-time channel selection and negotiation. First, we utilize the Constant Yawing Rate and Acceleration (CYRA) model to calculate Time-To-Collision (TTC) in order to detect potential accidents [32]. The CYRA model is widely used to estimate the vehicle trajectory, especially on accurate prediction for short time movements. We also use the CYRA model to calculate T_{ava}, which represents remaining time for a channel to be Type 1 due to mobility issues. For example, if one channel is occupied by a cluster of vehicles coming behind, T_{ava} for that channel will be smaller than those of other channels; we only calculate T_{ava} for Type-1 channels and set T_{ava} for other channels to 0. T_{ava} is transmitted along with the CPL as a reference for channel selection.

After detecting a potential accident, channel selection and negotiation starts, which consists of the following steps:

1) Channel Negotiation: When a vehicle identifies the potential danger with another vehicle, it needs to negotiate and choose a service channel to use. This is similar to the typical rendezvous problem in cognitive radio system [33][34]. To reduce the queuing delay of using the CCC, we give the negotiation packet the highest priority by placing it in the head of the MAC queue. Also, to reduce the service delay, we limit the CW_{max} to be equal to CW_0: The backoff window remains a constant in the backoff procedure. We also try to decrease CW_0, but it will only give us more retransmissions, making the packet delay even longer than using CW_0 directly. Note that the MAC queue manipulation is conducted only for negotiation packets, because they are sent out only one time. This cannot be periodically used for safety messages on the CCC to meet MATI/MAD requirements. Otherwise, it would cause severe congestion to other cars that are not involved in the accident.

2) Channel Negotiation: MC-Safe selects a coordinator to determine the final channel to use for the accident prevention on the channel negotiation. Based on the periodically broadcast safety messages from all vehicles (which contain the vehicle position, velocity, MAC address, etc.), the vehicle with the smallest MAC address becomes the coordinator, because this method incurs little computation and is widely adopted. Once the coordinator has been determined, the negotiation process works as follows: First, the vehicles with a greater MAC address will start the negotiation process by sending the negotiation request and its own CPL to the vehicle with smallest MAC address (the coordinator). After all the CPLs are received by the coordinator, the coordinator will start the channel selection process. The channel selection process contains several list matching procedures. The final channel to use is chosen according to the rule listed below:

- If there exist available Type-1 channels, the channel with the largest T_{ava} is chosen; If these channels have the same T_{ava}, then the one with the least interference nodes is selected;
- If there is no common Type-1 channel, a Type-2 channel which requires the least suppression effort is selected.
- If there is no common Type-2 channel either, a Type-3 channel with the least hidden terminals is chosen.

In order to quickly settle the channel to use and avoid unnecessary negotiation delay, for MC-Safe we propose to adopt the *max-min* policy to compare T_{ava}, which means that T_{ava} is determined by the smallest value in the CPLs received by the vehicle with the smallest MAC address. For example, T_{ava} is 3.2 for Channel-1 and 3.0 for Channel-2 in the CPL of vehicle C. Then, vehicle C (the coordinator) receives CPL from vehicle A indicating T_{ava} is 4.0 for Channel-1 and 2.4 for Channel-2. Vehicle C also receives vehicle B's CPL indicating T_{ava} is 1.2 for Channel-1 and 3.1 for Channel-2, vehicle C will set T_{ava} for Channel-1 to be 1.2 and Channel-2 to be 2.4. If Channel-1 and Channel-2 are both Type-1 channels, Channel-2 is selected for vehicle A, B and C to communicate on. The vehicle with the smallest MAC address will make the decision and transmit it in the ACK frame to all other vehicles involved in this potential accident, and change its own channel. After the ACK is received by the other nodes, they will also change to the chosen channel in the ACK frame accordingly. As a result, all n vehicles involved in a potential accident will select the same channel. Though multiple rounds of negotiation can make the channel selection result more robust, the possible extra negotiation delay can be intolerable given a critical condition on the road. Thus, we use the above proposed one-time handshake format to settle the negotiation.

Since MC-Safe aims to establish communication under an emergency situation, the delay for the negotiation should be as small as possible. The negotiation delay can be calculated as follows: As we already put the negotiation packet in the head of the packet queue and the backoff counter does not increase due to retransmission, the delay for transmitting the negotiation packet consists of only the service delay s_0. After the negotiation packet is received by the other side, the ACK will be ready after the period of time defined as Short InterFrame Space (SIFS) [4] and not go through the backoff procedure again. After the ACK frame is received by the transmitter, the negotiation process is finished. The involved vehicles starts to switch channel and communicate with each other. According to our proposed model, the total negotiation delay d_n incurred by the negotiation can be calculated as:

$$d_n = \frac{s_0}{1-p} + \text{SIFS} + T_{ACK} + T_s \tag{9}$$

where SIFS is the waiting time to generate ACK frame; T_{ACK} is the transmission time for the ACK frame; T_s is the channel switch overhead. The average T_s is only 2.87ms, which is negligibly short compared with the negotiation delay, according to our evaluation using the Intel 3945 Network Interface Card [35]; For a medium vehicle density (8 vehicle per 100 meters) and a typical speed for a vehicle (40mph) on the road, the average negotiation delay is 22ms and the traveling distance of a vehicle during the negotiation process is less than 1m. For heavy congestion cases, the negotiation can be longer than 100ms, but car speeds normally are reduced with a higher vehicle density, so the traveling distance during negotiation is still limited. Thus, the negotiation delay for MC-Safe is sufficiently small.

We also test the total delay for channel negotiation involving more than two vehicles. The total delay for three, four and five vehicles can be 38ms, 46ms and 54ms with a vehicle density 8 vehicles per 100 meters, respectively. The traveling distance for each vehicle is still smaller than 1m. The delay for more than five vehicles can be longer but it is rare to have an accident involving more than four vehicles according to the report issued by the US government [36].

3) Interference Suppression: If there is no common Type-1 channel available, the suppression mechanism is applied to enforce the MATI and MAD requirements. Specifically, when the channel is selected, several packets during the DSRC communication will be broadcast to other vehicles to suppress those vehicles that are not involved in the potential accident and sending non-safety messages, in order to enforce the MATI and MAD requirement of the urgent safety messages (from experiments, we find five packets are normally enough for suppression). The suppression scheme is implemented by adopting a *p*-persist mechanism: For each transmitting packet, it has a probability of p_t to be transmitted. The probability p_t is sent in the broadcast packet and is calculated by dividing the number of one-hop neighbors N_c by the maximum number of allowed direct neighbor defined in the MATI table. For the case of channel in Type 3, the channel cannot satisfy MATI and MAD by only suppressing the nodes within one-hop distance. Thus, hidden terminals need to be suppressed as well. Note that, in our extensive simulation, the worst situation is rare, and usually there is at least one common Type-1 or Type-2 channel.

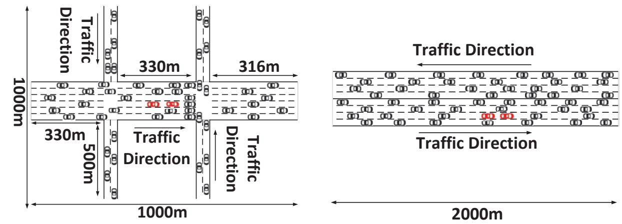
D. Discussion

1) Overhead Analysis: Here we analyze the time and space overheads of MC-Safe. The time overheads mainly come from two components: 1) Channel Modeling, 2) Negotiation. For channel modeling, as the theoretical parameters (p_b, p_c, p_i) are estimated offline with the lookup table, the time overhead of channel modeling comes from online monitoring, which includes estimating p and d_i for each channel and checking MATI/MAD requirements. The overhead for those two parts is 8μs and 5μs based on the measurements on our hardware testbed, respectively. For negotiation, the complexity of the algorithm is $\mathcal{O}(n\log m)$ (m is the number of involved vehicles and n is the number of available service channels). Given a limited number of service channels (six in WAVE) and involved vehicles (e.g., three vehicles), the measured computation overhead is small (6.43μs) compared with other terms in Equation (9) and so negligible. The space overhead mainly comes from the offline lookup table. In our implementation, we set the maximum of N_c and N_h to be 100 and 20, respectively, and each element in the lookup table contains three single precision float-point numbers (p_b, p_c, p_i), so the space overhead is $20 \times 100 \times 3 \times 4$ bytes (20kb) in total.

2) Reliability of Negotiation Messages: The message reliability of MC-Safe can be ensured by its MAC protocol, 802.11p: The negotiation request will be retransmitted if the sender does not receive an ACK from the coordinator within a specified time interval until it reaches the retry limit. In the rare cases when there are indeed packet losses (after retry limit is violated), MC-Safe could select a channel that is not the best. However, MC-Safe just becomes Random in this case, whose performance is still much better than using CCC directly (see Section V for results). To ensure that all the vehicles involved select the same channel for negotiation, the coordinator must receive ACK from every other vehicle (regardless of the retry limit) after it sends out the channel decision.

3) MAC Queue Manipulation: Queue length control and packet scheduling are well-studied topics. Similar with previous work [16][13], MC-Safe here uses the typical First-In-First-Out (FIFO) scheduling scheme for the general cases. But other more advanced schemes can also be integrated for specific demands. For example, using Earliest Deadline First (EDF) or Random Early Detection (RED) can lead to smaller end-to-end delay by packet canceling and admission control. According to our test with the RED model, the delay is reduced by 2.4% at the cost of dropping more packets due to the admission control.

4) Congestion Control: MC-Safe is also orthogonal to the dynamic congestion control schemes using the CCC in current protocols (e.g., ETSI ITS-G5). For ITS-G5, it adjusts the broadcast beacon frequency between 1Hz to 10Hz depending on the busy condition of the CCC. For MC-Safe, the negotiation overhead is small based on the analysis above and the suppression process only works on SCH, thus placing no extra overhead on the CCC. Note that, for other protocols such as the IEEE 1609 protocol families (WAVE), they do not have such congestion control mechanisms and the beacon frequency is recommended to be 10Hz in order to make the V2V network stable. Therefore, a slower beacon frequency will cause a longer refreshing delay and degrade the performance for the real-time vehicle control.



(a) Intersection (Int) (b) Highway

Fig. 5: The road topology of our two test scenarios. The vehicles of interest are marked with red color.

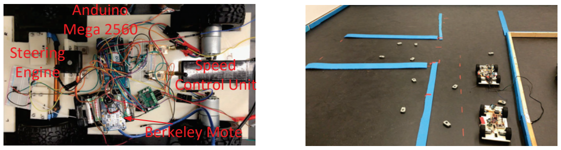
V. Evaluation

In this section, we conduct the evaluation of MC-Safe. We first introduce the experiment setup. We then test MC-Safe and other baselines under a typical case, and examine MC-Safe's performance in different scenarios in simulation (Sections V-B to V-F). At last, we test the performance of MC-Safe on a hardware testbed (Section V-G).

A. Experiment Setup

1) Road and Vehicle Setup: We test MC-Safe in two major road scenarios. The first one is a typical traffic scenario in the urban area with one main road and two branches: The main road is one-way with 6 lanes; The branches are one-way 2-lane roads; The width of each lane is 4 meters and the area of the scenario is one square kilometer. The second one is a bi-directional highway scenario of 8 lanes with a total length of 2000m. Upon arriving at the end of one direction, vehicles re-enter the scenario at the start point of the other direction, thus the total number in the simulation remains constant. Figure 5 shows the two traffic scenarios and the vehicles in red color form a potential rear-end collision scenario: The heading vehicle suddenly stops and the following vehicle starts the channel selection process when the following car receives the hard-brake message. We focus mainly on the scenarios that involve two vehicles, because 80% (1258 out of 1577) of multi-car accidents involve only two vehicles according to the report issued by the US government [36]. As mentioned in the motivation example, we formulate the inter-vehicle distance in Poisson distribution as a typical traffic modeling method [22]. Moreover, in order to minimize the variance brought by random factors from the traffic distribution, we run 20 independent and identically tests to get the average results.

2) DSRC Setup: The DSRC radio on each car broadcasts safety information on the CCC periodically and the broadcast frequency is 10Hz as recommended. The size of safety message is set to be 300 bytes [9]. The transmission power is set to a recommended value (20dbm) [4]. Among all available data rates defined in 802.11p protocol (*3Mbps*, *6Mbps*, *9Mbps*, *12Mbps*) [4], we choose the lowest *3Mbps* to test the worst-case scenario in terms of delay performance. The propagation model is chosen to be the same model proposed by Cheng [23]. The traffic on SCHs follows the Poisson distribution with an



(a) The scaled car (b) Road Setup

Fig. 6: Hardware testbed of MC-Safe.

arrival rate ranging from 2ms to 30ms to emulate different kinds of applications.

3) Hardware Testbed Setup: We implement a prototype of MC-Safe using small-scaled model cars. As shown in Figure 6(a), the control algorithm is implemented with the Arduino Mega 2560 broad, which connects to the steering engine and speed control actuator to realize the driving control using a typical PID controller. Although MC-Safe should be implemented on real DSRC devices for V2V communication, as a proof of concept system, we use Tmote Sky motes as the wireless communication device and implement MC-Safe using TinyOS. The initial speed of the two scaled cars are set to 150cm/s (60mph for a real car). Figure 6(b) shows the road setup for our experiment. We create one two-lane road with each lane's width as 50cm. In the experiment, we deploy other Tmote Sky motes (each representing a car) randomly as interference nodes.

4) Baselines for comparison with MC-Safe:

- **Common Control Channel (CCC)**: Similar to the state-of-the-practice solution WAVE, CCC relies only on the control channel to transmit safety messages.
- **Random**: When a pre-crashing condition is detected, each randomly selects one of the service channels (i.e., channels other than the control channel) for safety message transmission. After that, the channel to be used is finalized in the negotiation process.
- **The Least Congested Channel First (LCCF)**: It chooses the channel that has the least vehicles within the one-hop distance. Thus, it may have degraded performance when there are more hidden terminals on the same channel.
- **Ideal**: One channel is reserved and always available *only* for the vehicles involved in this particular potential accident. However, in a real V2V network, other vehicles may also use this channel when they are likely to have collision. Hence, this method is unrealistic, but just serving as a baseline with the performance upper bound.

B. Analysis of a Typical Scenario

Here we investigate the whole channel selection process from the time point when the danger is identified by MC-Safe to the time point when the danger is resolved. We set the density to be 7 vehicles per 100 meters. Vehicles of interest are selected in the middle of the road segment to avoid the inaccuracy of the two ends in the road segment (marked red in Figure 5).

Figure 7 shows the full traces of message transmission delay of the three methods (MC-Safe, LCCF and Random) during the whole process. Each trace contains 3 phases: The channel negotiation phase, the DSRC communication phase, and the back to CCC phase. The packet delay at first is large

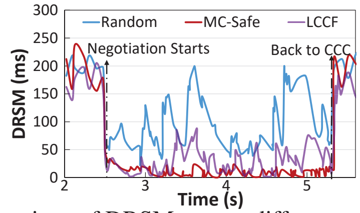
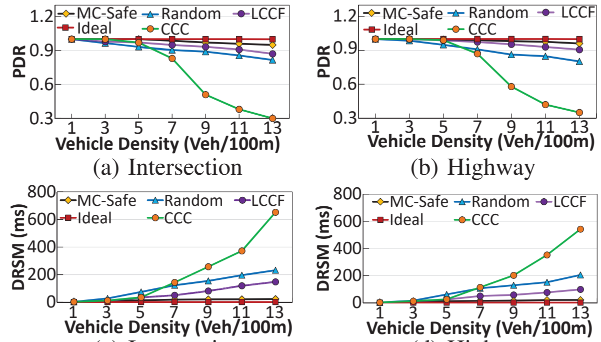


Fig. 7: Comparison of DRSM among different methods during the whole channel selection process.



(a) Intersection (b) Highway

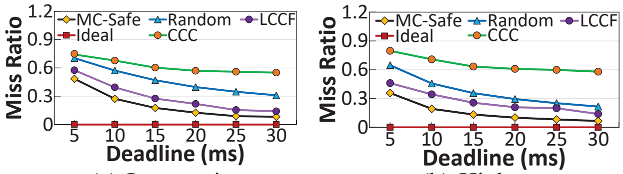
(c) Intersection (d) Highway

Fig. 8: PDR and DRSM under different vehicle densities.

because the vehicle is on CCC whose condition has large variations. When the following vehicle receives the hard-brake message from the heading car, it starts to negotiate for a common channel to use (at around 2.5s). When the negotiation starts, the delay drops from 200ms to 50ms-75ms because the negotiation packet is placed at the head of the MAC queue and its backoff counter does not double for retransmissions. After the ACK frame is received, the two vehicles can communicate with each other with a negligible delay (10ms-15ms). After the danger is resolved, the two vehicles switch back to CCC and start to beacon again. Figure 7 shows that MC-Safe can achieve the best performance compared to the baselines. Random has an average delay of 120ms during DSRC communication, which is much larger than the 20ms requirement. The delay performance is also not stable: The largest delay can be around 200ms, which will result in a large blackout period and degrade the performance of control algorithm. LCCF has an average delay of 40ms, which is much smaller than that of Random but it still cannot meet the typical control requirement (20ms). LCCF shows some spikes in the DSRC communication phase, indicating that LCCF has large variations and suffers from unexpected high delay at some time points because LCCF does not consider the hidden terminal issue and suffers a lot of retransmissions in the MAC layer.

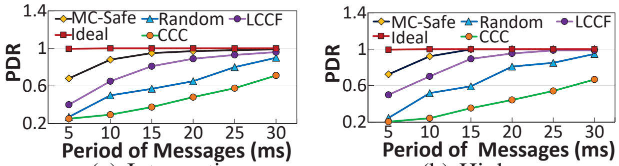
C. Different Vehicle Densities

In this set of experiments, we compare MC-Safe with three baselines (Random, LCCF, Ideal) under different vehicle densities. We test two typical scenarios (shown in Figure 5) and assign different speeds to the vehicles. To emulate a real-life scenario, a random number of cars are assigned on different service channels for non-safety services with a service vehicle ratio of 0.5. Figure 8 shows the Delay of Received Safety Message (DRSM) and Packet Delivery Ratio (PDR) for the intersection and highway, respectively.



(a) Intersection (b) Highway

Fig. 9: Deadline miss ratio under different deadline values.



(a) Intersection (b) Highway

Fig. 10: Comparison with different safety message periods.

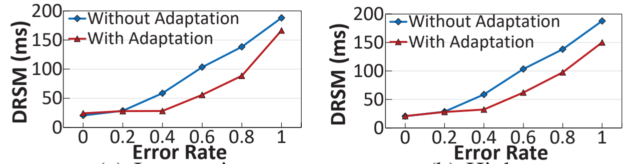
MC-Safe outperforms LCCF by 8.21% on average in terms of PDR and 23.4% in terms of delay. CCC is unusable when the traffic density reaches to 7 vehicles per 100 meters, which is common in the urban area, especially in the rush hour. The delay for a single packet can be as long as 160ms, which is 8 times the CAN deadline with a PDR of only 60%. The delay of Random increases rapidly (from 15.91ms to 61.32ms) as the density increases from 3 to 5 vehicles per 100 meters. LCCF has a similar delay to MC-Safe but it only considers the one-hop neighbors. Therefore, as the vehicle density increases, its PDR begins to drop due to the increase of its two-hop hidden terminal neighbors, which is not considered in its design. MC-Safe maintains a high PDR for all the vehicle densities, with the delay performance closest to that of Ideal. For a higher traffic density where there is no suitable channel for real-time communication (7 veh/100m and above) because the packet delay cannot meet the MAD requirement, MC-Safe can still achieve an average delay below 20ms while LCCF has an average delay of more than 50ms.

D. Different Deadlines and Message Periods

Here, we examine the impacts of two key design factors: deadline and safety message period.

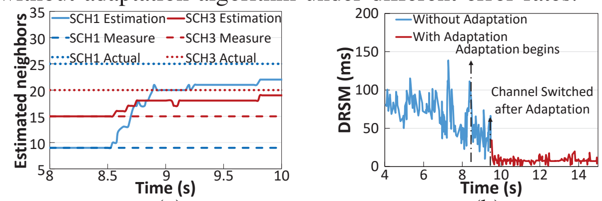
First, we consider the impact of different deadline values. Figure 9 shows the deadline miss ratio when the deadline changes from 5ms to 30ms. For CCC, the deadline miss ratio is 57% for the 30ms deadline, which is 48% higher than that of MC-Safe. MC-Safe is 20% lower than that of Random on average and is 28.7% better under the tightest deadline (5ms). We can also see that MC-Safe outperforms LCCF by 12.31% on average, 10% for the relatively loose ones and 21.7% under the tight ones. The main reasons are two folds: First, LCCF does not consider the hidden terminal thus for one packet it has to re-transmit many times, causing a high variance on the packet delay. Second, it does not consider the control requirements, thus lacking of a well-designed suppression mechanisms as well.

We then test different safety message periods. Results in Figure 10 indicate that MC-Safe can achieve the best PDR compared with other baselines. When the safety message period is small, the queue is actually saturated and reaches its throughput limitation: Some packets are dropped due to the overflow of the MAC queue. The PDR of MC-Safe reaches



(a) Intersection (b) Highway

Fig. 11: Comparison of DRSM with adaptation algorithm and without adaptation algorithm under different error rates.



(a) (b)

Fig. 12: Comparison of DSRM before and after applying adaptation algorithm in MC-Safe in the face of beacon errors. The adaptation monitoring starts at 8.5s, and the channel switch (from SCH1 to SCH3) with adaptation is at 9.6s, with much shorter delay afterwards.

about 70% in the experiment, compared with the 20% PDR of CCC, 36% PDR of Random and 51% PDR of LCCF for a 5ms message period. Though LCCF has the shortest service time and the largest queue utilization, it suffers from hidden terminals heavily and has a lower PDR than MC-Safe. As the safety message period becomes longer, the difference between MC-Safe and the baselines becomes smaller. Nevertheless, for the period of 30ms, MC-Safe can still outperform other baseline by 6%.

E. Effectiveness of Model Adaptation

In this part, we evaluate our model parameter adaptation algorithm based on online monitoring. To test the adaptation performance, we deliberately alter the beacon content of the currently used channel of some vehicles, thus inserting error to the model parameters, i.e., N_c and N_t. Therefore, the reported number of one-hop neighbors (N_c) for each service channel will be inaccurate. As shown in Figure 11, the error rate ranges from 0 (totally accurate) to 1 (totally wrong) when vehicles send their channel usage information in the beacon. These errors will have two consequences: First, due to the inserted error, MC-Safe will not select the right channel to use, leading to lower PDR and longer delay; Second, the error will also affect the suppression algorithm, because the probability of virtual collision depends on the reported N_c. For MC-Safe, it uses the adaption algorithm to solve these issues by adaptively monitoring the channels, and re-estimates the number of nodes on the channel in the 1s time window. Figure 11 shows the improvements due to the adaptation algorithm: The average delay decreases from 58ms to 36ms when the error is 40%, and it is also improved from 103ms to 62ms under 60% error. However, when the error is above 80%, the improvement gets smaller as the error is too large to be corrected. On average, the delay with the adaptation algorithm is 18% lower, and it can have up to 35% improvement in the best case.

Figure 12 shows an example of applying the adaptation algorithm. Here, we deliberately start the adaptation algorithm

TABLE II: Comparison of collision probability. MC-Safe achieves the lowest collision probability.

Inter-vehicle Distance	MC-Safe	LCCF	Random	CCC
close (25m)	18%	42%	60%	78%
Relatively close (30m)	11%	25%	38%	50%
Relatively safe (40m)	0%	0%	6%	15%

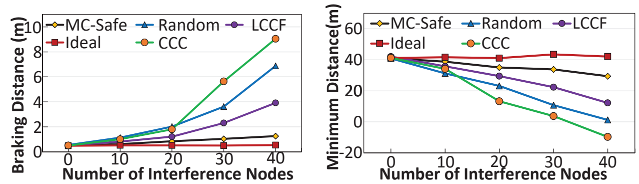
from 8.5s. Before that, the channel decision is made only based on the received beacons. Therefore, SCH1 is selected with the lowest estimated number of one-hop neighbors. However, due to beacon errors, SCH1 is not the best channel and cannot meet the delay requirement. Therefore, MC-Safe suffers a high delay more than 50ms. From 8.5s, the channel monitoring for adaptation begins, and the estimation results of both channels become closer to their actual values (Figure 12(a)). Based on the adaptation, SCH3 is selected with the suppression mechanism applied at 9.5s. Compared to the result before adaptation, the delay of MC-Safe is much improved. We can see that this adaptation algorithm provides MC-Safe better dynamic performance in the face of beacon errors. Note that, here we use 1s as an example time window value to illustrate the effectiveness of the adaptation algorithm. In a real system, this value can be tuned based on different factors such as channel conditions, car velocities and road situations.

F. Comparison of Accident Probability

In this section, we compare the accident probability of MC-Safe with other baselines in a real-life rear-end collision scenario, to see how MC-Safe can help in a pre-crashing scenario. We set the vehicle density as 7 vehicles per 100 meters and their speeds as 40mph. When the heading vehicle takes the hard braking action, the following vehicle will change the channel to communicate with the heading one. After the safety message is received, the following vehicle will brake at the largest deceleration allowed. Here we use the typical three-phase braking model [37] to calculate the braking distance based on how soon the safety message is received (i.e., message delay). With different vehicle distances, we list the calculated accident probabilities in Table II. Results show that in the real case, MC-Safe can achieve the lowest collision probability due to more timely (i.e., shorter delay) and reliable (i.e., higher PDR) safety message communication.

G. Hardware Testbed Experiment

To evaluate the effectiveness of MC-Safe in the real-world scenario, two emergency scenarios are investigated with our hardware testbed: 1) Rear-end collision, which is similar to previous simulation scenario; 2) Lane changing, one abruptly changes its lane, and the other car tries to avoid the collision when it detects the lane change action through DSRC. For the first scenario, we investigate the *stopping distance* of MC-Safe and the baselines when different number of interference nodes are deployed. The stopping distance is defined as how long the following car travels between the moment when the heading car sends out the braking message and the moment when the following car is fully stopped. For the second scenario, we investigate the *minimum distance* during the whole lane change process. the minimum distance is defined as the shortest distance between the two scaled cars from the moment when one car starts to change its lane to the moment it merges into



(a) (b)

Fig. 13: Comparison of control performance with MC-Safe and different baselines. (a) Stopping distance for different baselines in the hard-braking case. (b) The minimum distance between the two scaled cars in the lane changing case.

the new lane completely. The larger the distance is, the better performance one scheme achieves. We use this metric because if the lane-changing message can be received with a shorter delay, the other car can act earlier to stay farther away from the lane-changing car. Hence, the greater the distance, the better the real-time performance.

Figure 13(a) shows the result for the first emergency scenario: MC-Safe has the closest performance to Ideal compared with other baselines. With 30 interference nodes, the stopping distance of MC-Safe is 1.32m shorter than that of LCCF, and 2.53m shorter than that of Random. Note that 2.53m for scaled cars is approximately 20.24m for real cars. Such a short stopping distance (due to better real-time communication) would significantly lower the probability of having a rear-end collision. For the lane changing scenario, Figure 13(b) shows that MC-Safe provides the largest minimum distance in this scenario. The minimum distance between the two vehicles can be decreased by 18.24cm on average when using MC-Safe for 40 interference node case compared with LCCF. This distance is about the width of one vehicle in the real case (1.47m). Note that a distance lower than zero means the accident avoidance system fails and a collision has occurred (e.g., CCC with 40 interference nodes). With MC-Safe, a collision can be avoided even for a dense traffic situation.

VI. Conclusion

Existing work on safety message communication relies only on one or two pre-selected control channels, which can result in poor packet delivery and potential accident when the vehicle density is high. In this paper, we have proposed *MC-Safe*, a multi-channel V2V communication framework that monitors all the available channels and dynamically selects the best one for safety message transmission. MC-Safe features a novel channel negotiation scheme that is activated whenever two or more cars are determined to have a potential collision. All the involved cars work collaboratively in a distributed manner to identify a channel that can meet the delay and packet error rate requirements for every car. Afterward, all the involved cars switch to the selected channel for real-time safety message communication. Our evaluation results show that MC-Safe outperforms existing single-channel solutions and other well-designed multi-channel baselines, by having a 12.31% lower deadline miss ratio and an 8.21% higher packet delivery ratio on average. Compared to WAVE, the state-of-the-practice solution, MC-Safe successfully reduces the average delay of safety message transmission from 300ms to 20ms when the vehicle density is high.

References

[1] J. Harding, G. Powell, R. Yoon, J. Fikentscher, C. Doyle, D. Sade, M. Lukuc, J. Simons, and J. Wang, "Vehicle-to-Vehicle Communications: Readiness of V2V technology for Application," Tech. Rep., 2014.

[2] E. C. Eze, S. Zhang, and E. Liu, "Vehicular Ad Hoc Networks (VANETs): Current State, Challenges, Potentials and Way Forward," in *the 20th International Conference on Automation and Computing (ICAC)*, 2014.

[3] R. Martin, "Annual global road crash statistics." [Online]. Available: http://asirt.org/initiatives/informing-road-users/road-safety-facts/road-crash-statistics

[4] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an International Standard for Wireless Access in Vehicular Environments," in *the 67th IEEE Vehicular Technology Conference (VTC)*, 2008.

[5] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein, "Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information," *IEEE Transactions on Vehicular Technology*, 2009.

[6] D. Newton, "Density, car ownership, and what it means for the future of los angeles," 2010. [Online]. Available: http://la.streetsblog.org/2010/12/13/density-car-ownership-and-what-it-means-for-the-future-of-los-angeles/

[7] Y. Yao, X. Chen, L. Rao, X. Liu, and X. Zhou, "LORA: Loss Differentiation Rate Adaptation Scheme for Vehicle-to-Vehicle Safety Communications," *IEEE Transactions on Vehicular Technology*, 2017.

[8] X. Chen, L. Kong, X. Liu, L. Rao, F. Bai, and Q. Xiang, "How Cars Talk Louder, Clearer and Fairer: Optimizing the Communication Performance of Connected Vehicles via Online Synchronous Control," in *the 35th IEEE International Conference on Computer Communications (INFOCOM)*, 2016.

[9] M. Torrent-Moreno, J. Mittag, P. Santi, and H. Hartenstein, "Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information," *IEEE Transactions on Vehicular Technology*, 2009.

[10] L. Zhang and S. Valaee, "Congestion Control for Vehicular Networks With Safety-Awareness," *IEEE/ACM Transactions on Networking*, 2016.

[11] Y. Park and H. Kim, "Application-Level Frequency Control of Periodic Safety Messages in the IEEE WAVE," *IEEE Transactions on Vehicular Technology*, 2012.

[12] Q. Xiang, X. Chen, L. Kong, L. Rao, and X. Liu, "Data Preference Matters: A New Perspective of Safety Data Dissemination in Vehicular Ad Hoc Networks," in *the 34th IEEE International Conference on Computer Communications (INFOCOM)*, 2015.

[13] Q. Wang, S. Leng, H. Fu, Y. Zhang *et al.*, "An IEEE 802.11 p-based Multichannel MAC Scheme with Channel Coordination for Vehicular Ad Hoc Networks," *IEEE Transactions on Intelligent Transportation Systems*, 2012.

[14] Y. Yao, K. Zhang, and X. Zhou, "A Flexible Multi-Channel Coordination MAC Protocol for Vehicular Ad Hoc Networks," *IEEE Communications Letters*, 2017.

[15] A. J. Ghandour, M. Di Felice, H. Artail, and L. Bononi, "Dissemination of Safety Messages in IEEE 802.11 p/WAVE Vehicular Network: Analytical Study and Protocol Enhancements," *Pervasive and Mobile Computing*, 2014.

[16] Y. Yao, L. Rao, and X. Liu, "Performance and Reliability Analysis of IEEE 802.11 p Safety Communication in a Highway Environment," *IEEE Transactions on Vehicular Technology*, 2013.

[17] F. Martelli, M. E. Renda, G. Resta, and P. Santi, "A Measurement-based Study of Beaconing Performance in IEEE 802.11p Vehicular Networks," in *the 31st IEEE International Conference on Computer Communications (INFOCOM)*, 2012.

[18] A. F. Long Le and R. Baldessari, "Vehicular Wireless Short-Range Communication for Improving Intersection Safety," *IEEE Communications Magazine*, 2009.

[19] Y. Wu, Y. Zhu, H. Zhu, and B. Li, "CCR: Capacity-Constrained Replication for Data Delivery in Vehicular Networks," in *the 32nd IEEE International Conference on Computer Communications (INFOCOM),* 2013.

[20] G. Guo and L. Wang, "Control over Medium-Constrained Vehicular Networks with Fading Channels and Random Access Protocol: a Networked Systems Approach," *IEEE Transactions on Vehicular Technology*, 2015.

[21] A. J. Ghandour, K. Fawaz, H. Artail, M. Di Felice, and L. Bononi, "Improving Vehicular Safety Message Delivery through the Implementation of a Cognitive Vehicular Network," *Ad hoc networks*, 2013.

[22] F. Bai and B. Krishnamachari, "Spatio-Temporal Variations of Vehicle Traffic in VANETs: Facts and Implications," in *the sixth ACM international workshop on VehicularAr InterNETworking (VANET)*, 2009.

[23] L. Cheng, B. E. Henty, D. D. Stancil, F. Bai, and P. Mudalige, "Mobile Vehicle-to-Vehicle Narrow-Band Channel Measurement and Characterization of the 5.9 GHz Dedicated Short Range Communication (DSRC) frequency band," *IEEE Journal on Selected Areas in Communications*, 2007.

[24] S. Oncu, J. Ploeg, N. Van de Wouw, and H. Nijmeijer, "Cooperative Adaptive Cruise Control: Network-Aware Analysis of String Stability," *IEEE Transactions on Intelligent Transportation Systems*, 2014.

[25] Z. Shuai, H. Zhang, J. Wang, J. Li, and M. Ouyang, "Lateral Motion Control for Four-Wheel-Independent-Drive Electric Vehicles Using Optimal Torque Allocation and Dynamic Message Priority Scheduling," *Control Engineering Practice*, vol. 24, pp. 55–66, 2014.

[26] K. Tindell and A. Burns, "Guaranteeing message latencies on control area network (CAN)," in *the 1st International Controller Area Network Conference (CAN)*, 1994.

[27] Y. Sadi, S. C. Ergen, and P. Park, "Minimum Energy Data Transmission for Wireless Networked Control Systems," *IEEE Transactions on Wireless Communications*, 2014.

[28] G. Karagiannis *et al.*, "Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions," *IEEE communications surveys & tutorials*, 2011.

[29] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on selected areas in communications*, 2000.

[30] Y. Zheng, K. Lu, and D. Fang, "Performance Analysis of IEEE 802.11 DCF in Imperfect Channels," *IEEE Transactions on Vehicular Technology*, 2006.

[31] H. Zhai, Y. Kwon, and Y. Fang, "Performance Analysis of IEEE 802.11 MAC Protocols in Wireless LANs," *Wireless Communications and Mobile Computing*, 2004.

[32] R. Schubert, E. Richter, and G. Wanielik, "Comparison and Evaluation of Advanced Motion Models for Vehicle Tracking," in *the 11th International Conference on Information Fusion (ISIF)*, 2008.

[33] Z. Lin, H. Liu, X. Chu, and Y.-W. Leung, "Jump-Stay based Channel-Hopping Algorithm with Guaranteed Rendezvous for Cognitive Radio Networks," in *the 30th IEEE International Conference on Computer Communications (INFOCOM)*, 2011.

[34] N. C. Theis, R. W. Thomas, and L. A. DaSilva, "Rendezvous for Cognitive Radios," *IEEE Transactions on Mobile Computing*, 2011.

[35] M. Harris and S. Harvey, "Channel Switching Overhead for 802.11 b," *Technical Report*, 2009.

[36] E.-H. Choi, F. Zhang, E. Y. Noh, S. Singh, and C.-L. Chen, "Sampling Design Used in the National Motor Vehicle Crash Causation Survey," 2010.

[37] K. Hedrick, "Brake System Modeling, Control and Integrated Brake/Throttle Switching Phase I," *California PATH Research Report*, 1997.