Toward Communication Strategies for Platooning: Simulative and Experimental Evaluation

Michele Segata, Bastian Bloessl, Stefan Joerer, Christoph Sommer, Mario Gerla, *Fellow, IEEE*,
Renato Lo Cigno, *Senior Member, IEEE*, and Falko Dressler, *Senior Member, IEEE*

***Abstract*—Platooning, which is the idea of cars autonomously following their leaders to form a road train, has huge potential to improve traffic flow efficiency and, most importantly, road traffic safety. Wireless communication is a fundamental building block: It is needed to manage and maintain the platoons. To keep the system stable, strict constraints in terms of update frequency and communication reliability must be met. We investigate different communication strategies by explicitly taking into account the requirements of the controller, exploiting synchronized communication slots, and transmit power adaptation. As a baseline, we compared the proposed approaches to two state-of-the-art adaptive beaconing protocols that have been designed for cooperative awareness applications, namely, the European Telecommunications Standards Institute (ETSI) Decentralized Congestion Control (DCC) and Dynamic Beaconing (DynB). Our simulation models have been parameterized and validated by means of real-world experiments. Our results demonstrate that the combination of synchronized communication slots with transmit power adaptation is perfectly suited for cooperative driving applications, even on very crowded freeway scenarios.**

***Index Terms*—Automated highways, cooperative systems, networks, vehicles.**

I. INTRODUCTION

SINCE research on Vehicular Ad Hoc Networks (VANETs) started more than 20 years ago, many applications based on intervehicular communication (IVC) have been proposed; only few have been implemented and tested in field operational tests (FOTs) [1]. Cooperative Adaptive Cruise Control (CACC), which has become widely known as platooning, is among these applications. Being investigated since the 1980s, e.g., within the PATH project [2], it is still an active topic due to the challenging problems it raises. One of the core reasons behind such a huge interest is the benefits that platooning could provide once deployed.

Platooning can enhance the travel experience covering consumption issues, safety, and comfort: First, it has the potential to improve the traffic flow and to reduce the fuel consumption, reducing jams on freeways and decreasing pollution [3], [4]. Second, platooning can improve drivers' safety if a system fault is less likely than a human error, which is the main cause of accidents [4]. Last but not least, a vehicle autonomously following its leaders permits the driver to relax, as shown in the recent SARTRE project [5].

From a research point of view, platooning is extremely challenging, as it involves several research fields, including control theory, communications, vehicle dynamics, and traffic engineering. From a communication perspective, the main reasons are the requirements in terms of latency and reliability. For what concerns networking, any controller designed for supporting platooning, such as the CACCs in [6] and [7], needs frequent and timely information about vehicles in the platoon to avoid instabilities that might lead to crashes. For this reason, CACC is often cited as one of the most visionary applications of IVC [8], [9].

Looking at the state of the art in the IVC protocol design, we see that many protocols are focusing on cooperative awareness. This is particularly the case for all protocols investigated in large-scale FOTs. In the USA and Europe, the IEEE Dedicated Short-Range Communications (DSRC)/Wireless Access in Vehicular Environments (WAVE) stack [10] and its European Telecommunications Standards Institute (ETSI) counterpart, i.e., the ITS-G5 stack [11], [12], dominate all practical tests.

The biggest advantage of the aforementioned protocol stacks is their ability to provide cooperative awareness while keeping the network load reasonable. Alternative approaches presented in the literature argue that more aggressive approaches such as Dynamic Beaconing (DynB) [13], [14] are needed to support traffic safety applications. A platooning system has a recommended information update frequency of 10 Hz [7]; thus, delays on the order of several hundred milliseconds due to congestion control mechanisms cannot be tolerated. Whether these communication requirements can be satisfied by the plain DSRC/WAVE or ETSI ITS-G5 stacks is still unclear, and further work is needed before platooning can become a reality.

In this paper, which extends our contribution presented in [15], we study the suitability of state-of-the-art beaconing-based cooperative awareness protocols for platooning and highlight the challenges that are still open, proposing and investigating

---

Manuscript received March 30, 2015; revised July 15, 2015, August 21, 2015, and October 2, 2015; accepted October 7, 2015. Date of publication October 9, 2015; date of current version December 14, 2015. The review of this paper was coordinated by the Guest Editors.

M. Segata is with the Institute of Computer Science, University of Innsbruck, 6020 Innsbruck, Austria, and also with the Department of Information Engineering and Computer Science, University of Trento, 38123 Trento, Italy (e-mail: segata@ccs-labs.org).

B. Bloessl, C. Sommer, and F. Dressler are with the Department of Computer Science, University of Paderborn, 33098 Paderborn, Germany (e-mails: bloessl@ccs-labs.org; sommer@ccs-labs.org; dressler@ccs-labs.org).

S. Joerer is with the Institute of Computer Science, University of Innsbruck, 6020 Innsbruck, Austria (e-mail: joerer@ccs-labs.org).

M. Gerla is with the Department of Computer Science, University of California at Los Angeles, Los Angeles, CA 90032 USA (e-mail: gerla@cs.ucla.edu).

R. Lo Cigno is with the Department of Information Engineering and Computer Science, University of Trento, 38123 Trento, Italy (e-mail: locigno@disi.unitn.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVT.2015.2489459

communication protocols that can help tackle them. We only consider platoons of homogeneous cars, and we do not take into account mixed platoons made of cars, trucks, buses, or vans. We investigate the reliability of wireless communications for platooning under high channel load imposed by a large number of platoons, showing how the proposed approaches support applications' needs and how they compare to two adaptive beaconing solutions, i.e., the current ETSI standard Decentralized Congestion Control (DCC) [11], [12], [16] and DynB [14].

Based on these insights, we explore design options for communication strategies suitable for the platooning application and finally derive a novel set of communication protocols that are able to build upon the standard IEEE 802.11p protocol. As a baseline for an extensive set of simulation studies, we report on an experimental validation of the simulation model using a platoon of four cars equipped with IEEE 802.11p-compliant devices. Our results clearly indicate that a combination of synchronized communication slots with transmit power adaptation performs best in our platooning scenario. This even holds for very dense network scenarios such as on a crowded freeway.

Our main contributions can be summarized as follows.

- We first describe in detail the CACC controller we consider in this paper and present its main features and benefits (see Section III).
- We define a set of different communication strategies, specifically taking into account CACC controller requirements (see Section IV).
- To calibrate our simulation models, we performed a measurement campaign on the road using four cars driving in a platoon (see Section V).
- We compare the proposed approaches against state-of-the-art adaptive beaconing strategies from a network and an application layer perspective, showing the substantial benefits of our approaches for platooning (see Section VI). In this analysis, we assume the communication channel to be dedicated to the platooning application. We then relax this assumption and test the impact of human-driven vehicles using DCC on our approach and *vice versa*. Finally, we test the controller in an emergency braking scenario under different message generation rates to understand the real requirements of the CACC we consider.

In the remainder of this paper, to avoid ambiguity, we use the terms collision and congestion when referring to the network, and crash and traffic jam when referring to vehicles.

II. RELATED WORK

A. CACC

The platooning research community initially focused on the problems connected to the automated control of vehicles because the design of such a system is a nontrivial task. Indeed, the characteristic that makes a CACC different from a standard Adaptive Cruise Control (ACC) is the capability to *closely* follow the car in front by making use of wireless links to communicate with nearby vehicles. A standard ACC exploits only data provided by the radar, thus the distance and the relative speed to the vehicle in front. Such a system must keep a safety gap on the order of 1 to 2 s [7], [17], making it unsuitable for close following as required by platooning. Smaller intervehicle gaps would make the system unstable and might lead to crashes [17].

In the literature, we can find several CACC controllers, each of those employing different communication patterns and having different characteristics. The CACC designed in [7] makes use of communication only between direct followers. In this case, the distance that can be maintained by the controller has to be speed-dependent as for ACC. The headway time, however, can range from 0.5 to 0.7 s, much smaller than for a standard ACC. Another type of controller uses data communicated from both the vehicle in front and the platoon leader [6]. The benefit is that the system can be proven to be stable under a constant-spacing policy, i.e., the intervehicle distance does not need to be speed-dependent. This means that the intervehicle gap can be fixed and chosen in meters and not in seconds as for a standard ACC or the CACC in [7]. For example, the FOTs in the PATH and SARTRE projects [5], [6] used this kind of controller, and the distances chosen in the experiments were between 5 and 7 m. Other solutions can configure the logical topology to adapt to network conditions [18], [19].

CACC controllers have been investigated since the beginning by the pioneering projects PATH [2] and Auto21 CDS [20], but they are still under continuous improvement either by academic research [7], [18], [19] or by car manufacturers, as in the SARTRE project [5]. What differentiates pioneering projects from recent studies is the "philosophy." In the case of PATH or Auto21 CDS, platoons were designed to run on dedicated freeways, managed by a centralized system [21]. The idea in SARTRE, instead, is that platoons autonomously form and can travel on public motorways mixed with human-driven vehicles. In both cases, network conditions are a major concern: 802.11-based networks can suffer high packet loss ratios even in moderate channel load conditions, and given the frequent updates needed by the CACC to ensure platoon stability [17], the impact of the network performance on the safety of the overall system is nonmarginal.

B. IVC Protocols for Cooperative Awareness

Cooperative awareness is among the most prominent VANET applications. Broadcast-based IVC, or beaconing, has been identified as the underlying protocol primitive and has been investigated in detail by the vehicular networking community. The consensus is to periodically send beacons to all vehicles in communication range to improve cooperative awareness in general. ETSI defined cooperative awareness messages (CAMs) for this purpose, supported by decentralized environmental notification messages (DENMs) for event-triggered safety warnings. This line of research is still featuring very diverse proposals.

Most recently, ETSI ITS-G5 has announced a new standard taking into consideration the network dynamics and the need for congestion control, i.e., DCC [11], [12], [16], which features a variety of protocol variants. In the scope of this paper, we consider the complete ETSI CAM generation algorithm, which includes Transmit Rate Control, Transmit Power Control (TPC), Transmit Datarate Control, and DCC Sensitivity Control

TABLE I
DCC PARAMETERS FOR THE CCH, AC_VI

	RELAXED	ACTIVE	RESTRICTIVE
b_i	0.15	0.20	0.40
Tx power	33 dBm	ref	−10 dBm
Packet interval	0.04 s	ref	1 s
Datarate	3 Mb/s	ref	12 Mb/s
CCA threshold	−95 dBm	ref	−65 dBm

mechanisms. Each algorithm component controls beacon generation rate, transmit power, physical-layer data rate, and clear channel assessment (CCA) threshold, respectively. Furthermore, the latest release includes a set of vehicle dynamics-based rules for CAM triggering.

A state machine drives each component of the algorithm. The active state depends on the currently observed channel busy ratio, which is the amount of time the channel was sensed as busy by the physical layer. In this paper, we consider the three-state state machine designed for the control channel (CCH), i.e., the channel designated to CAMs [12]. The state change decision is taken by monitoring the busy ratio over two time windows, i.e., T_{down} and T_{up}. The protocol, at time t, computes $b_{\text{down}} = \max\{b_{t-T_{\text{down}}}, \ldots, b_t\}$ and $b_{\text{up}} = \min\{b_{t-T_{\text{up}}}, \ldots, b_t\}$, where b_t, $b_{t-T_{\text{down}}}$, and $b_{t-T_{\text{up}}}$ are the channel loads measured at times t, $t - T_{\text{down}}$, and $t - T_{\text{up}}$, respectively. The protocol then performs a state change by comparing these values with thresholds $b_{\min}$ and $b_{\max}$. State change is performed according to the following rules.

- If $b_{\text{down}} < b_{\min}$, set the state to RELAXED.
- If $b_{\text{up}} \geq b_{\max}$, set the state to RESTRICTIVE.
- Otherwise, set the state to ACTIVE.

The ACTIVE state can be further divided in substates. Each ACTIVE substate i defines a maximum channel load b_i and its DCC parameters. States are ordered according to channel load so that $b_{i-1} < b_i, i = 1, \ldots, N+1$, where N is the number of ACTIVE states, and $b_{N+1} = b_{\max}$. In the ACTIVE substates, state transitions are performed by finding the state id $i = \max(i_{\text{up}}, i_{\text{down}})$ such that

$$b_{i_{\text{up}}-1} \leq b_{\text{up}} < b_{i_{\text{up}}} \tag{1}$$

$$b_{i_{\text{down}}} < b_{\text{down}} \leq b_{i_{\text{down}}+1}. \tag{2}$$

For the CCH, however, only a single ACTIVE state is considered. Table I lists the parameters for configuring the three CCH states for the AC_VI access category, the one we consider for CAM messages. A "ref" value indicates that the corresponding parameter is unchanged when switching from the old to the new state. In this paper, we use the default parameters listed in [11], [12], and [16]. Finding and using a different set of parameters that maximizes DCC performance is out of the scope of this paper.

By following state change rules and configuring DCC with the parameters in Table I, the state machine is in the RELAXED state for channel loads lower than 15%, in the ACTIVE state for channel loads between 15% and 20%, and in the RESTRICTIVE state for channel loads higher than 20% (this is confirmed by the example shown in [11, Fig. 6]).

The part of the standard described so far considers the DCC mechanism only. CAM triggering rules are described in a dedicated standard (ETSI EN 302 637-2 [12]) and are built on top of DCC rules. EN 302 637-2 redefines minimum and maximum CAM generation intervals, i.e., 0.1 and 1.0 s, respectively. Moreover, the minimum CAM generation interval is further restricted based on the current DCC state. In particular, in [16], a set of DCC profiles (DPs) was defined to characterize traffic streams in the access, the network, and the transport layers. DPs are numbered from 0 to 32 (0 being the traffic with the highest priority), and each is associated with a T_{off} parameter that regulates packet interval rules for each DCC state.

According to [12], CAMs belong to the DP2 profile; thus, the minimum interval is restricted to 95, 190, and 250 ms for the RELAXED, the ACTIVE, and the RESTRICTIVE states, respectively [16, Table 1].

Finally, the standard defines vehicle dynamics-based triggering conditions to alert nearby vehicles if there are sudden changes in the state of the vehicle; of course, these can be extended according to new applications' requirements. In particular, the following are the three conditions.

- The absolute difference between last sent heading and current heading exceeds 4°.
- The distance between last sent position and current position exceeds 4 m.
- The absolute difference between last sent speed and current speed exceeds 0.5 ms.

If any of the aforementioned conditions is met and the minimum packet interval T_{off} has elapsed, a CAM should be immediately sent. Moreover, if the CAM is triggered due to the dynamics-dependent conditions, the protocol must schedule three consecutive CAMs with an interval equal to the time elapsed since the last CAM generation. The packet interval must be reset to the maximum (i.e., 1 s) when all the repetitions are sent.

Another approach, DynB [13], [14], tries to maintain network load at a fixed predefined value. Similar to DCC, DynB monitors the channel busy ratio and computes the interval to be used for sending the next beacon accordingly but adapts more aggressively to the current channel conditions. More formally, let N be the number of neighbors computed using frames received from nearby vehicles, I_{des} be the desired (i.e., the minimum) beacon interval, and b_t and b_{des} be the measured and the desired busy ratio, respectively. The beacon interval I is computed as

$$I = I_{\text{des}}(1 + rN) \tag{3}$$

where $r = b_t/b_{\text{des}} - 1$, clipped in [0, 1]. The idea of the protocol is that, if the channel load does not exceed a certain threshold, then the number of collisions should be small.

C. Platooning-Specific IVC Support

As aforementioned, reducing channel congestion and dealing with packet losses in IVC have been tackled with several proposals. These approaches are very beneficial for the network, keeping the load under control and packet losses at an acceptable level. Most of them, however, are not application aware; hence, they cannot meet specific application requirements. As a consequence, this might harm single applications such as platooning, which requires a constant and reliable flow of information. Platooning is not the only application that might suffer from this, as witnessed by the amount of papers that are trying to take into account specific application requirements [9], [22]–[25]. Due to this reason, the IVC community has recently started to investigate the impact of communication characteristics on platooning performance.

As an example, Lei *et al.* [26] showed the impact of different packet loss rates on the string stability of CACC, considering a controller with constant time headway policy. Fernandes and Nunes [9] analyzed strategies to improve communication reliability considering five different protocols, all based on time-division multiple access (TDMA). Furthermore, they proposed a dynamic adaptation of CACC parameters to cope with different situations. Böhm *et al.* instead analyzed the coexistence of CAMs (used for platooning) with (DENM, showing how the choice of different medium access control (MAC) layer priority classes for the two categories heavily affects the effectiveness of data dissemination [27]. From a communication point of view, some infrastructure might be used to make transmissions more efficient [28], or additional channels can be incorporated for better scalability [29]. Other communication technologies have been also analyzed. For example, Abualhoul *et al.* [30] proposed the use of visible light communication to communicate between immediate followers.

All these papers point out that the integration with the standardized cooperative awareness applications is very challenging and needs further investigations. In this paper, we propose a set of communication strategies explicitly taking the requirements of the CACC controller into account. This opens up new opportunities for the integration of cooperative awareness oriented beaconing with application-tailored protocols.

III. CONTROLLER MODEL

In this paper, we consider the discretized version of the CACC controller detailed in [17]. The model assumes that each vehicle i (with i being its position in the platoon) knows the position x_{i-1}, the speed $\dot{x}_{i-1}$, and the acceleration $\ddot{x}_{i-1}$ of the preceding vehicle, as well as the speed $\dot{x}_0$ and the acceleration $\ddot{x}_0$ of its platoon leader. The distance to the preceding vehicle can be obtained using a radar (and to some extent, also variations of the distance), whereas all other parameters must be conveyed through wireless communications.

Let Δ_t be the time interval of the sampled system and n be the current sampling step (or discrete time). Processing time is negligible compared with Δ_t. For each vehicle i

$$\dot{x}_i[n] = \frac{(x_i[n] - x_i[n-1])}{\Delta_t} \tag{4}$$

$$\ddot{x}_i[n] = \frac{(\dot{x}_i[n] - \dot{x}_i[n-1])}{\Delta_t} = \frac{(x_i[n] - 2x_i[n-1] + x_i[n-2])}{\Delta_t^2} \tag{5}$$

and the CACC control law computes the control input (i.e., the desired acceleration) to maintain the platoon as

$$u_i[n] = \alpha_1 u_{i-1}[n] + \alpha_2 u_0[n] + \alpha_3 \dot{\varepsilon}_i[n] + \alpha_4 \left(\dot{x}_i[n] - \dot{x}_0[n]\right) + \alpha_5 \varepsilon_i[n] \tag{6}$$

where

$$\varepsilon_i[n] = x_i[n] - x_{i-1}[n] + l_{i-1} + \text{gap}_{\text{des}} \tag{7}$$

$$\dot{\varepsilon}_i[n] = \dot{x}_i[n] - \dot{x}_{i-1}[n]. \tag{8}$$

In (7), l_{i-1} is the length of the preceding vehicle, whereas gap_{des} is the desired intervehicle gap, in meters. The distance to the front vehicle is given by the radar, which always provides up-to-date information with negligible error; hence, the term $x_i[n] - x_{i-1}[n] + l_{i-1}$ is considered to be exact. Terms $u_{i-1}[n]$, $u_0[n]$, $\dot{x}_{i-1}[n]$, and $\dot{x}_0[n]$ are instead obtained via wireless communication; thus, their value is affected by errors and can be outdated, as the beaconing process is slower than Δ_t and it is not fully reliable. Their value will be the one received with the last beacon message from the leader or from the front vehicle.

The α_i parameters in (6) are defined as

$$\alpha_1 = 1 - C_1; \quad \alpha_2 = C_1; \quad \alpha_5 = -\omega_n^2 \tag{9}$$

$$\alpha_3 = -\left(2\xi - C_1(\xi + \sqrt{\xi^2 - 1})\right)\omega_n \tag{10}$$

$$\alpha_4 = -C_1(\xi + \sqrt{\xi^2 - 1})\omega_n. \tag{11}$$

C_1 is a weighting factor between the accelerations of the leader and the preceding vehicle, which we set to 0.5, ξ is the damping ratio, set to 1; and ω_n is the bandwidth of the controller, set to 0.2 Hz as in [31].

As stated in [17], the desired acceleration u_i computed by the controller is not instantaneously applied because of the actuation lag introduced by the mechanical components. Such lag can be modeled by a first-order low-pass filter [7], [17]. In [17], the lag is assumed to be $\tau = 0.5$ s. We compute the actual acceleration as

$$\ddot{x}_i[n] = \beta \cdot u_i[n] + (1 - \beta) \cdot \ddot{x}_i[n-1]; \beta = \frac{\Delta_t}{\tau + \Delta_t}. \tag{12}$$

The acceleration of the platoon leader is governed by a standard ACC. We implemented a discretized version of the ACC detailed in [17], computing the desired acceleration as

$$u_i[n] = -\frac{1}{T}\left(\dot{\varepsilon}_i[n] + \lambda \delta_i[n]\right) \tag{13}$$

$$\delta_i[n] = x_i[n] - x_{i-1}[n] + l_{i-1} + T\dot{x}_i[n]. \tag{14}$$

In (13) and (14), T is the headway time in seconds, and λ is a design parameter. If no other car is in front of the platoon leader, the speed converges to a desired value.

The real acceleration of the car is computed as in (12). To ensure the stability of the system under the presence of a first-order lag, the following must hold [17]:

$$T \geq 2\tau; \qquad \lambda > 0. \tag{15}$$

We set $T = 1.5$ s and $\lambda = 0.1$.

IV. COMMUNICATION PROTOCOLS

The set of communication protocols for platooning we propose is based on the IEEE 802.11p/IEEE 1609.4 physical (PHY)/MAC; hence, scheduled messages contend for the channel in a carrier sense multiple access with collision avoidance (CSMA/CA) fashion.

We adopt the platooning controller employed in [6] and [17] [see Eq. (6)] where the inputs to the system are the leader's and the front vehicle's speed and acceleration. For the design of the algorithms, we exploit the specific requirements of the controller. In particular, we assume that each vehicle is aware of its position in the platoon and uses this information to decide *how* and *when* to send a beacon.

To decide *how*, we can exploit the fact that, in addition to the leader, each vehicle needs to communicate its speed and acceleration only to the vehicle immediately behind. The transmit power can be thus reduced to increase spatial reuse and avoid interfering with cars that are "not interested" in receiving such data. Leaders can instead use high transmit power to reach all vehicles within the platoon.

In general, TPC is complex because it must cope with highly dynamic networks [32], but for the application we consider, the setup is simplified due to the linear topology of the platoon. One issue to consider is the effect of different types of vehicles. For example, a truck with an antenna placed on its front might not be able to communicate with a car behind [33], requiring ad hoc power calibration to overcome the problem. This is, however, out of the scope of this paper; hence, we set a fixed transmit power value of 0 dBm for the followers.

To decide *when* to send, we exploit the vehicle's position within the platoon. The leader can send its beacon first, and then the others can follow in a cascading fashion, i.e., the second vehicle, the third, and so on. Notice that this is different from a standard TDMA approach, as with TDMA, every node participating in the communication obeys the same rules. In our approach, only nodes within a platoon cooperate in a TDMA fashion to reduce intraplatoon channel contention.

Algorithm 1: SLB protocol.

```
onStartup():
    if $myRole = leader$ then
        schedule(sendBeacon, beaconInterval);
    end
sendBeacon():
    sendBroadcast(getVehicleData());
    schedule(sendBeacon, beaconInterval);
onBeacon($beacon$);
    updateCACC(beacon);
    if $beacon.sender = leader$ then
        onLeaderBeacon(beacon);
    end
onLeaderBeacon($beacon$);
    unschedule(sendBeacon);
    schedule(sendBeacon, myPosition · offset);
```

The pseudocode of this slotted approach is listed in Algorithm 1. The idea is to divide the time after a beacon from the leader into slots and have each vehicle send its beacon in the time slot corresponding to its position in the platoon. As shown in Algorithm 1, only the leader starts to send beacons at protocol startup. The followers use the beacon received from the leader for synchronization, computing the time at which they should send the beacon depending on their position and a time offset. To avoid that a lost beacon from the leader blocks the protocol, each vehicle, upon sending a beacon, always schedules another send event after one beacon interval. Upon reception of leader's beacon, this event is updated to synchronize with the leader.

Fig. 1. Cars used for the experimental validation.

The rationale behind this protocol is to reduce random channel contention by adding synchronization among nodes. Moreover, even if there is no interplatoon collaboration, the leaders can end up roughly synchronizing with other platoons when performing CSMA/CA at the MAC layer. In the remainder of this paper, we refer to this slotted beaconing protocol as SLB and SLBP (without and with TPC, respectively).

In this paper, we consider a platoon size of 20 cars. Thus, we define a slot time offset between consecutive vehicles equal to 5 ms, i.e., the beacon interval (0.1 s) divided by the platoon size. The leader always broadcasts messages with the maximum transmit power (20 dBm). The followers, when TPC is disabled, use 20 dBm as well and 0 dBm otherwise.

To obtain a deep understanding of the benefits of each of the two proposals, we compare them with a baseline approach that uses standard static beaconing, i.e., periodic broadcasting. This protocol uses only CSMA/CA; hence, nodes randomly contend for channel access. We refer to this approach, with and without TPC, as STB and STBP, respectively. Transmit power values for leaders and followers are the same as in SLB and SLBP.

V. EXPERIMENTAL VALIDATION

In the first step, we performed a set of experiments with real cars (see Fig. 1). The goal is to validate and calibrate the network model we employ in simulations against real-world measurements. In the experiment, we used four cars and drove on a private road to safely maintain a distance of 5 m when driving at 20 km/h. The vehicles were driven by humans in respect of the Austrian legislation while the system was automatically recording network statistics without requiring any action from the driver.

TABLE II
PARAMETERS EMPLOYED IN THE EXPERIMENTAL VALIDATION

Parameter	Value
Beacon frequency	10, 20, and 25 Hz
Tx power (leader)	20 dBm
Tx power (followers)	20, 10, and 0 dBm
Modulation	QPSK R=½

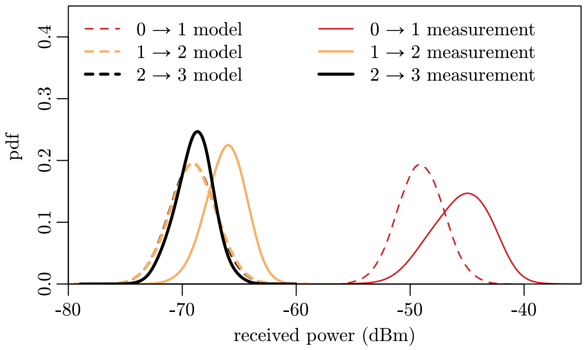


Fig. 2. Comparison of received power distributions between the experimental test bed (measurement) and the simulation environment (model) for the communication between immediate followers (0 being the leader; 3 being the last car). Transmit power of 20 dBm is for the leader, and 0 dBm is for the followers. The 1 → 2 and 2 → 3 model curves overlap because the distance and the transmit power between vehicles 1 and 2 and vehicles 2 and 3 are the same.

For the communication, we used two Cohda Wireless MK2[1] and two Unex DCMA-86P[2] devices, both IEEE 802.11p compliant. We connected each device to a Mobile Mark ECOM9-5500 dipole antenna with a gain of 9 dBi, magnetically mounted on the rooftop of the vehicles.

We implemented STB and SLB and tested them while repeatedly driving on a 2-km stretch of road using the parameters shown in Table II. Each experiment lasted roughly 30 s and was repeated three times to collect results in different environmental conditions. The same conditions (number of cars, protocols, parameters, etc.) were reproduced in a simulation scenario to calibrate the simulation model.

In our experimental setup, we always received at least 99% of frames sent, making the frame error rate not valid for comparison. For this reason, we only compare the received power distribution.

We model fading at the receiver with a Rice distribution with a strong line-of-sight (LOS) component. We assume this as, for this paper, we only take into account cars: As stated in [34], if the first Fresnel zone is less than 40% obstructed, then shadowing has no major impact on signal strength, and we experimentally verified the statement in another measurement campaign [35]. With a strong LOS component in a Rician channel, we can approximate the amplitude with a lognormal distribution [36]; thus, we assume lognormal fading.

Fig. 2 shows the comparison between the simulation and the experimental results for 20- and 0-dBm transmit power values for the leader and the followers, respectively. Before running the experiments, we tested the equipment by pairing the network interface controllers using a cable with a 90-dB attenuator and found that one device is transmitting with lower transmit power than selected and reporting incorrect received signal strength. These tests were used to calibrate and equalize the received power values prior to analyzing the data.

The first aspect we focus on is the shape of the distribution. Different real-world experiments show slightly different standard deviations; the one we choose for the simulation ($\sigma = 2$ dBm) is a good compromise between all of them, matching also the LOS measurements we reported in [35].

The second aspect is the average received power. In the simulation, we employed a free-space path loss model with $\alpha = 2.0$. As can be shown, the average received power is slightly higher in the experiment. This is due to the antennas we used as they provide high transmission gain. In the simulation, we instead considered theoretical isotropic antennas with no gain, as frequently used in vehicular simulations. For the time being, we are mainly interested in the shape of the resulting curves rather than the exact quantities, thus ensuring better comparability with other simulation studies.

VI. SIMULATIONS AND RESULTS

To compare the different approaches, we use PLEXE, a dedicated open-source[3] platooning simulator [37]. The simulator is based on the well-known Veins [38] framework and provides a high level of detail and realism, featuring mixed scenarios with ACC and CACC controlled vehicles [6] and human behavioral car-following models. A fully fledged IEEE 802.11p/IEEE 1609.4 network stack [39], [40] permits us to develop and to evaluate arbitrary freeway scenarios, high-level applications, and communication protocols. In these aspects, our model differs from the model in [41], which assumes completely automated and dedicated freeways. Our vision is the deployment in a more flexible scenario containing both fully automated vehicles and vehicles controlled by traditional car-following models, like a generalization of the SARTRE philosophy. Simulating such a mixed scenario is easily possible in our modeling approach.

In this paper, we want to understand the characteristics and the behavior of the protocols and network conditions in a "stressful" configuration. In the main part of our evaluation, we focus on the performance on a global scale, both from a networking and, most importantly, from an application perspective. Finally, to give an idea of the real delay requirements, we analyze the performance of CACC in light of the control model presented in Section III using an emergency braking scenario.

A. *Simulation Model and Setup*

Table III summarizes all simulation parameters. To model channel phenomena, we employ a free-space propagation loss model with $\alpha = 2.0$ plus lognormally distributed fading with $\sigma = 2.0$, as obtained from the model calibration in Section V. For PHY and MAC layer models, we use the IEEE 802.11p and IEEE 1609.4 models presented in [40]. However, we disable

---

[1] http://www.cohdawireless.com/product/mk2.html

[2] http://unex.com.tw/product/dcma-86p2

[3] http://plexe.car2x.org

TABLE III
NETWORK AND ROAD TRAFFIC SIMULATION PARAMETERS

	Parameter	Value
communication	Path loss model	Free space ($\alpha = 2.0$)
	Fading model	Log-normal ($\sigma = 2.0$)
	PHY/MAC model	802.11p/1609.4 single channel
	Frequency	5.89 GHz (CCH)
	Bitrate	6 Mb/s (QPSK R = ½)
	Access category	AC_VI
	MSDU size	200 B
	Transmit power	20 and 0 dBm
	Sensitivity	-95 dBm
	Noise floor	-95 dBm
	CCA-threshold	-95, -85, and -65 dBm
mobility	Number of cars	160, 320, and 640
	Number of lanes	4
	Platoon size	20 cars
	Car length	4 m
	Intra-platoon distance	5 m
	Inter-platoon distance	$\simeq 41$ m
	Speed	100 km/h
controllers	C_1	0.5
	ω_n	0.2 Hz
	ξ	1
	T	1.5 s
	λ	0.1
	τ	0.5 s
DynB	I_{des}	0.1 s
	b_{des}	0.25

the switching between CCH and service channel using only the CCH. The bit rate for STB, SLB, and DynB is 6 Mb/s, which has been reported to be optimal for highly demanding vehicular applications [42].

Regarding the application layer, packets have a MAC service data unit (MSDU) size of 200 B. They use the AC_VI access category and are sent with a beacon frequency of 10 Hz, the minimum required by CACC [7]. The fixed beacon interval value of 10 Hz only holds for STB and SLB. DynB and DCC compute their own beacon intervals; furthermore, DynB uses a static transmit power value of 20 dBm. The implementation and the parameters for DynB are taken from [14] and listed in Table III; the DCC parameters are set to their default values, as given in Table I. All protocols send standard CAMs that, as defined by ETSI, are broadcast frames [12]. Thus, they are not acknowledged.

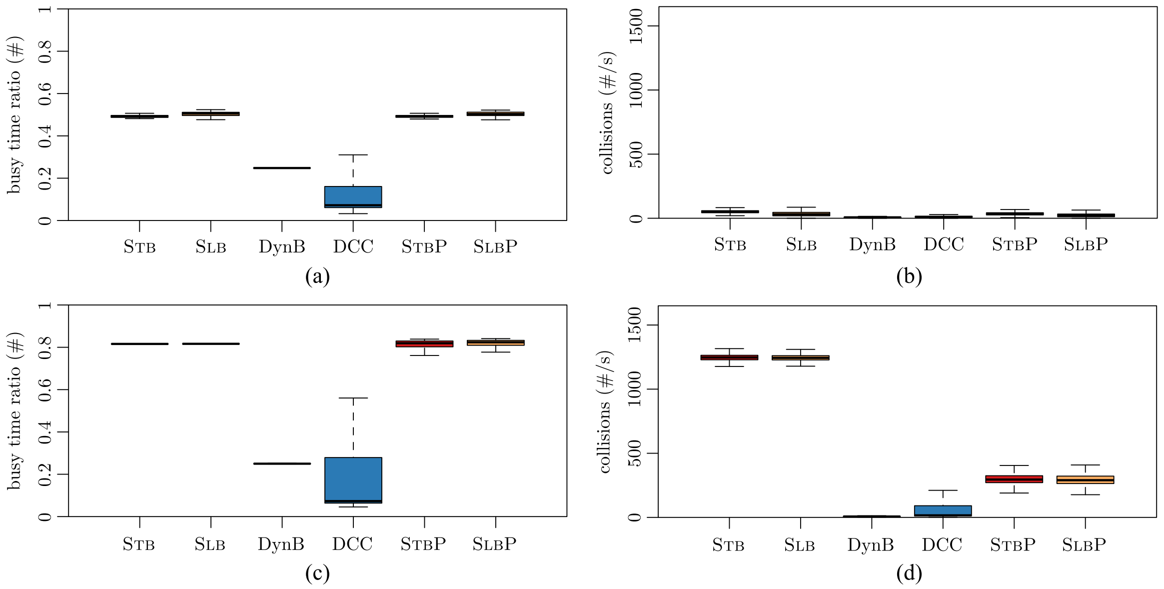
The last physical layer parameter we change for all protocols but DCC is the CCA threshold. The CCA threshold is used to assess channel busy status when a station misses the preamble portion of a frame, for example, when multiple frames are simultaneously received (cumulative interference). For the CCA threshold, we use values of −65 dBm (as defined in the standard [43, 18.3.10.6]), −85 dBm (the minimum required sensitivity for the lowest modulation and coding scheme), and −95 dBm (equal to model's minimum sensitivity). In the IEEE 802.11 standard, sensitivity is different from the CCA threshold. For an IEEE 802.11-compliant device, sensitivity is defined as the power threshold, above which 90% of the preambles are correctly detected [43, 18.3.10.6]. In our model, instead, the sensitivity is the frame detection threshold, and it is set to −95 dBm. Any frame received with a power lower than the sensitivity is simply ignored.

We simulate a stretch of a four-lane freeway filled by platoons of 20 cars each, for a total number of cars of 160, 320, and 640, respectively. Such a high number of vehicles might seem unreasonable, but we choose it for two reasons. First, we want to understand if there is any upper limit, meaning that we want to know if the protocols are always behaving as expected, i.e., if they stop working properly above a certain vehicle density. Second, such high densities are well possible on big freeways during rush hours, and platooning might exactly be the application we want to run in such situations. Therefore, understanding whether it can be supported or not is crucial. We choose 640 as the upper bound value because, in this case, the interference domain of each vehicle does not cover the entire scenario; thus, adding further cars does not affect the ones in the middle. Other relevant parameters are the intraplatoon vehicle distance gap_{des}, set to 5 m, and the speed of the platoons, set to 100 km/h. In this paper, we assume a constant speed because we focus on the analysis of the network. Each simulation experiment (each combination of density and protocol) has been repeated ten times to improve the confidence in the results. As further processing step, we partially removed data collected at simulation borders to get rid of border effects that biased the evaluation. More details about this procedure are given during the discussion of the single metrics.

B. General Networking Performance

The simulations we run for the analyses in Section VI-B–D assume the wireless channel to be dedicated to the platooning application. This eases the interpretation of the results and permits to obtain some fundamental understanding of the network's behavior. In Section VI-E, we relax this assumption and simulate a highway with automated vehicles (running our beaconing protocols) and human-driven vehicles (running DCC).

We begin our analysis by looking at two generic network metrics, i.e., channel busy ratio and collisions. Channel busy ratio is the amount of time the physical layer declares the channel as busy over a certain time window. Each vehicle samples and records the busy ratio once a second throughout the entire simulation. Collisions instead count the number of frames that each vehicle was not able to decode due to interferences. This metric is sampled and recorded once a second as well. Sampled values for both metrics are grouped in box plots, which display the first and third quartiles as a box and the median as the center line, as well as the minimum and maximum values with whiskers. The box plots do not include data collected during simulation warm-up. Moreover, for the 640-vehicle scenario, we discard data of cars at the border of the simulation to avoid border effects. In particular, we removed the data of 15% of the vehicles (7.5% at the front and the tail) after verifying that this amount was enough to get rid of such effects. Vehicles at the head and at the tail of the stream experience lower channel congestion because the interference domain of each car is smaller than the size of the scenario. For this reason, border vehicles falsify the results by showing higher performance than the actual one.



(a) (b)

(c) (d)

Fig. 3. Busy time ratio and collisions for the 160- and 640-car scenarios. CCA threshold set to −95 dBm. (a) Busy time ratio for the 160-car experiment. (b) Collisions per second for the 160-car experiment. (c) Busy time ratio for the 640-car experiment. (d) Collisions per second for the 640-car experiment.

Fig. 3 shows busy ratio and collisions for the 160- and 640-car experiments for a CCA threshold of −95 dBm. From a network perspective, 160 cars do not cause network overload. This is shown by a maximum channel load on the order of 50% and a very limited amount of collisions. DynB keeps the channel load at the desired level (25%), whereas for DCC, the load spans between 5% and 20% due to its dynamic behavior. For that which concerns STB, STBP, SLB, and SLBP, instead, we see that TPC in low-density scenarios is not helpful because the network is not saturated. Moreover, the slotted approach shows a slight improvement in network utilization and collision reduction, but the difference is statistically irrelevant.

For the high-density scenario (640 cars), results are totally different. The dynamic approaches are still capable of keeping the load and the collisions under control by adapting their behavior to the high number of nodes simultaneously contending for the channel. STB and SLB, in contrast, completely saturate the channel reaching about 80% channel load and a large amount of collisions per second. Using TPC in STBP and SLBP improves the performance in terms of collisions. Even if channel load is close to complete saturation, the number of collisions is drastically reduced compared with STB and SLB. Results for the 320-car scenarios show similar behavior. For the sake of brevity, however, we omit the graphs for such scenarios.

C. Application Layer Perspective

According to busy ratio and collisions, the dynamic approaches definitely show better performance. DynB and DCC are indeed designed to improve the overall network conditions without considering specific application requirements. In platooning systems, missed (or omitted) packets can harm the application and thus passengers' safety. If a CACC controller misses data packets, it is forced to perform a "blind" control action, i.e., it computes the desired acceleration based on old outdated information, which might result in instabilities or crashes. To measure protocols' effectiveness from the application layer perspective, however, we would need to have precise information about the theoretical requirements of the controller, which are out of scope of this paper. We thus define an application layer metric, which is parameterized on a maximum tolerable delay.

In particular, let δ_{req} be the maximum allowable intermessage delay, and let $\mathcal{D}$ be the set of all intermessage delays collected by a vehicle. We define the set of all delays satisfying the requirement δ_{req} as

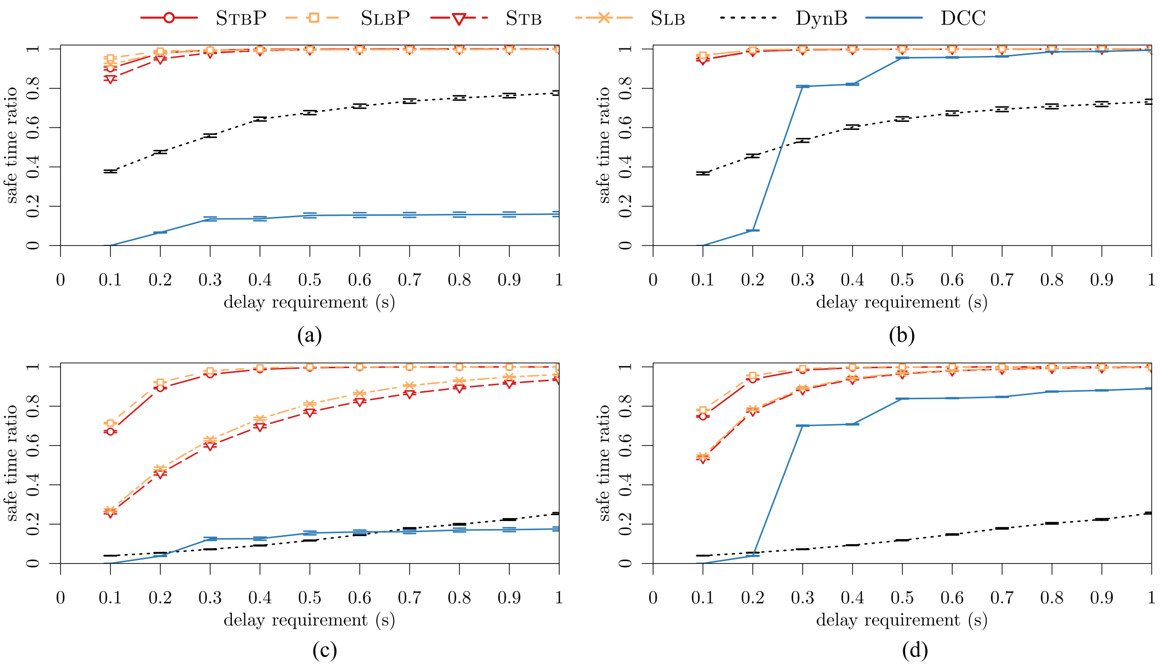
$$\mathcal{D}_{\text{safe}} = \{d : d \in \mathcal{D} \wedge d \leq \delta_{\text{req}} + \Delta\} \tag{16}$$

where Δ is a small grace period in which the information is still useful, which accounts for uncertainties such as MAC layer backoffs. In our computation, we set $\Delta = 10$ ms, which is assumed to be the CACC controller sampling time [7], [37]. The safe time ratio metric r_{safe} is defined as

$$r_{\text{safe}} = \frac{\sum_{d_{\text{s}} \in \mathcal{D}_{\text{safe}}} d_{\text{s}}}{\sum_{d \in \mathcal{D}} d}. \tag{17}$$

For example, if all delays in $\mathcal{D}$ are equal 200 ms, by setting $\delta_{\text{req}} = 100$ ms, we obtain $r_{\text{safe}} = 0$, meaning that the vehicle was never in safe conditions. Conversely, by setting $\delta_{\text{req}} = 300$ ms, we obtain $r_{\text{safe}} = 1$; thus, the requirement was always satisfied.

Fig. 4 shows the results for the r_{safe} metric. Each point in the plot represents the average with 95% confidence intervals among all cars and all simulations for that particular configuration. We removed border vehicles for the 640-car scenario. Moreover, we discarded all the interarrival times of leader messages for the first platoon in each lane for the 640-car scenario, as for DynB, the first leaders are beaconing way faster than other cars because of their border position. Because each vehicle expects to receive data from the leader and the vehicle



(a) (b)

(c) (d)

Fig. 4. Safe time ratios of both leader and front messages for the 160- and 640-car scenarios. CCA threshold set to -95 dBm. (a) Leader messages: 160 cars. (b) Front messages: 160 cars. (c) Leader messages: 640 cars. (d) Front messages: 640 cars.

immediately in front, we plot the metric for both kinds of packets and we refer to them as leader and front messages.

In Fig. 4, DynB shows the consequences of forcing the busy ratio to a fixed value. To keep channel utilization at the desired level, DynB needs to increase the beacon interval indefinitely. For the 160-car scenario, the performance is still reasonable, but in the 640-car case, even for a delay requirement of 1 s, the vehicles are in a safe state for less than 30% of the time. DynB, however, shows a fair behavior, as the metric is similar for both leader and front messages.

DCC, instead, behaves "orthogonally" with respect to DynB. DCC shows similar performance independently of the number of vehicles but behaves unfairly, as r_{safe} for front messages is higher than for leader messages. This is due to the low transmit power employed in the Restrictive state (-10 dBm), making communication with the leader almost impossible.

Concerning StbP and SlbP, instead, it is clear that taking into account specific application requirements can bring enormous benefits. Not using TPC indeed results in poor performance in a high-density scenario. Stb and Slb perform better than dynamic approaches but worse than their TPC counterparts. In the 640-car scenario, for a delay requirement of 100 ms, using TPC results in a performance gain of roughly 40% and 20% for leader and front messages, respectively. In the worst case (r_{safe} for 640 cars, leader messages), the protocols ensure that vehicles are in safe conditions roughly 70% of the time for the most demanding delay requirement (0.1 s). Overall, the slotted approach provides slightly but not significantly better results. Front vehicle messages are easier to receive due to the small distance between consecutive vehicles. The benefits of the slotted approach are, however, influenced by the size of the platoon: The bigger the platoon, the smaller the channel contention among vehicles in the same platoon.

The results show that, even if StbP and SlbP cause in general a higher number of collisions (see Fig. 3), such collisions interest data frames that are not needed by the CACC application. StbP and SlbP manage to deliver leader and front vehicle messages in 90% of the time within 200 ms in the most demanding scenario. To keep the collisions under control, instead, the dynamic approaches need to lower the beacon rate, causing extremely large delays. In summary, an increase in the collisions count does not necessarily worsen application layer performance.

D. Impact of CCA Threshold

Here, we briefly analyze the impact of the CCA threshold on the performance of Slb and SlbP. The CCA threshold defines the amount of energy required to declare the channel as busy when the preamble portion has been missed. For example, during a transmission, a station will not hear preambles of frames sent by other stations. Thus, when the transmission is completed, the station is required to measure the amount of energy in the channel to understand if there are ongoing communications. The IEEE 802.11 standard mandates a CCA threshold of -65 dBm, but in this paper, we consider -85 and -95 dBm as well.

Fig. 5 shows the safe time ratio of leader messages for Slb and SlbP for the 160-car scenario. When the number of cars is limited and all nodes use full transmit power, the CCA threshold has no impact at all (for the values we consider). Conversely, when using TPC, a lower CCA threshold helps increasing the awareness about other vehicles' transmissions, thus decreasing the number of collisions. In particular, for a delay requirement of 0.1 s, r_{safe} increases roughly 5% when using a CCA threshold of -95 dBm.

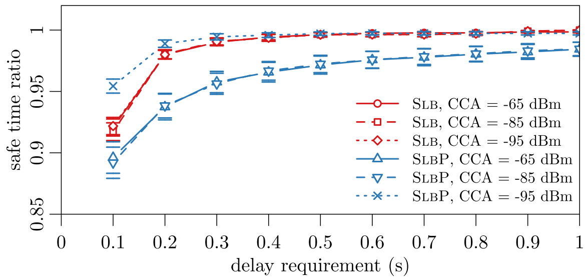


Fig. 5. Safe time ratio of leader messages for the slotted approaches in a 160-car scenario for different values of the CCA threshold.

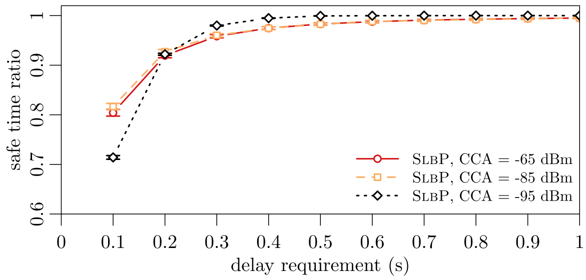


Fig. 6. Safe time ratio of leader messages for the slotted approach with TPC in a 640-car scenario for different values of CCA threshold.

For a larger number of vehicles (see Fig. 6, 640-car scenario), a low CCA threshold is too conservative and reduces spatial reutilization. Indeed, for a delay requirement of 0.1 s, the safe time ratio for −95 dBm performs roughly 10% worse than higher CCA thresholds.

By considering all the cases, the best approach would be to adapt the threshold based on the network load, as DCC mandates or as considered in other works [44]. In the absence of such a mechanism, however, a threshold of −65 dBm as mandated by the IEEE 802.11 standard provides, on average, the best performance.

E. Coexistence With DCC

In the analysis considered so far, we assumed the channel to be dedicated to the platooning application, together with a 100% market penetration. In particular, during the introduction of this technology, the road will be shared among human and autonomously driven vehicles, and human-driven vehicles might use different applications concurrently accessing the channel. We thus modified the 640-vehicle scenario by filling two lanes with 320 platooning vehicles using STBP and SLBP whereas the remaining two lanes with human-driven vehicles using DCC.

Fig. 7 shows safe time ratios of leader and front messages for both STBP and SLBP. We apply the same border removal procedure used in previous plots. The performance is comparable with the one in Fig. 4: All approaches have a safe time ratio greater than 85% for a delay requirement of 100 ms. Due to the load caused by STBP and SLBP, DCC goes in RESTRICTIVE state, thus using the minimum transmit power of −10 dBm. As a result, STBP and SLBP are unaffected by human-driven vehicles, but DCC suffers a large amount of packet losses. DCC is indeed not designed to coexist with other protocols; thus, its parameters should be tuned differently. This is, however, out of scope for this paper.

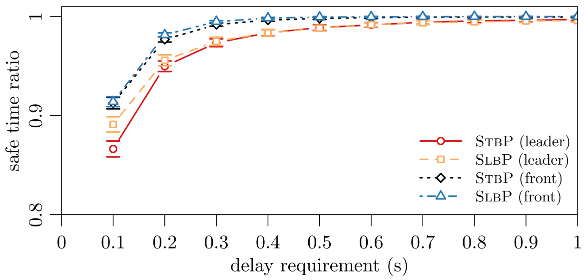


Fig. 7. Safe time ratio of leader and front messages for STBP and SLBP for the scenario with human-driven vehicles. CCA threshold set to −65 dBm.

F. Impact of Communication on CACC Performance

Here, we make a base study of the performance of the CACC described in Section III. The idea here is to provide a generic understanding of what the requirements of the controller might be. To this purpose, we simulated a single platoon of 20 cars running at 130 km/h using SLB to avoid any kind of channel contention among vehicles. We perform full-stop braking with different deceleration rates and different beacon frequencies and measure the minimum distance between any pair of cars as a measure of the reliability of the system. This will give a basic idea of the value for δ_{req} previously defined. Providing a definite value is out of the scope of this paper, as this requires a dedicated study.

We used a beacon rate from 1 to 10 Hz in steps of 1 Hz and from 10 to 20 Hz in steps of 5 Hz. The leader decelerations we used are 2, 4, 6, and 8 m/s^2. The maximum deceleration for the follower vehicles was set to 9 m/s^2; hence, vehicles have a higher braking capability than what is performed by the leader. This is a different setup from the real-world experiments in [45], where the driving conditions of the vehicles were different. In a first experiment, maximum deceleration was inhomogeneous among the different trucks and they were driving on different lanes to avoid a real crash, whereas in a second experiment, following trucks had a stronger braking capability than front trucks. Even if the maximum braking capabilities are the same, in our experiments, the system can become unsafe because of delayed message reception.

We repeat each simulation ten times, and we take the minimum distance over all pairs of cars and simulation runs for each specific configuration of beacon frequency and deceleration rate. We thus show a worst case analysis.

Fig. 8 shows the resulting minimum distances. The abscissa is plotted in logarithmic scale to highlight CACC behavior for beacon frequencies up to 10 Hz. For the sake of comparison with the δ_{req} values used in Fig. 4, we added a second scale showing the intermessage interval corresponding to a particular beacon frequency. When the minimum distance is 0 m, this means that two cars crashed into each other.

The first noticeable fact is that the allowable intermessage interval for such emergency braking scenario highly depends on the deceleration rate. The higher the deceleration, the lower the tolerable delay. For example, for a 2-m/s^2 deceleration rate, 0.5 s of intermessage interval seems enough to avoid a crash, as the worst-case minimum distance is 2.5 m, whereas when decelerating at 8 m/s^2, a delay of 0.33 s can result in a crash.

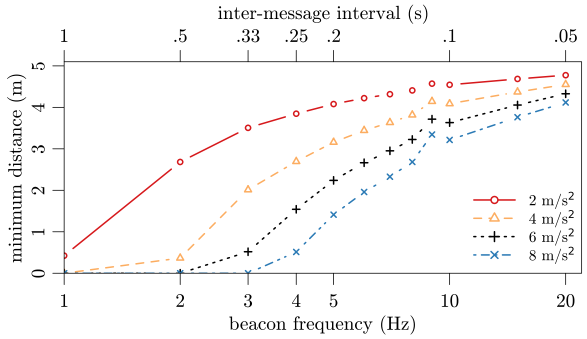


Fig. 8. Minimum distances after the complete stop of the platoon as function of the beacon interval for different leader decelerations.

A noncrash, however, does not necessarily translate into a "safe" situation, as drivers might feel uncomfortable when coming too close to the vehicle in front. In any event, the graph suggests that an intermessage interval greater than 0.3 s might harm system's safety and result in crashes. To obtain a definite value, however, it is required to study the controller more in depth, for example, by considering different controller parameters' values, different actuation lags, and different intermessage intervals for leader and front messages.

To conclude, by considering the results in Fig. 4(c) and by comparing them with the performance in Fig. 8, STBP and SLBP by being able of providing data to the application with a minimum rate of 5 Hz in 90% of the time, even in the occurrence of an emergency situation the vehicles would avoid a crash. Indeed, Fig. 8 shows that the CACC is robust to a "blind control" action when few packets are lost. In contrast, the dynamic approaches show unacceptable performance for a CACC application, having an update time larger than 1 s 80% of the time.

These results raise two different interesting issues: *i)* The network load may be able to be further reduced by adapting the beacon frequency to the current acceleration or, in general, to the "stability" of the platoon, and *ii)* further joint research between the networking and the vehicular control community is needed to identify controllers that are efficient, reliable, and stable under DynB, as well as under emergency situations.

VII. CONCLUSION

This paper has discussed the feasibility of different beaconing solutions for an automated platooning system. The goal of the paper was to compare the network and application layer performance of state-of-the-art DynB solutions, i.e., DynB and ETSI DCC, against the four possible alternatives we propose. The design of these new protocols stems from the specific requirements of the application, which suggest a TDMA-like approach coupled with TPC. The results prove that considering high-level requirements can greatly improve the performance from the application layer perspective while avoiding severe network congestion. Furthermore, we have briefly discussed the impact of the choice of different CCA thresholds, showing that, in the absence of a dynamically changing threshold, the value mandated by the IEEE 802.11 standard gives the best performance on average. We also considered a mixed scenario where some human-driven vehicles concurrently access the channel using DCC, showing that the performance of our approaches is unaffected but that DCC would need to be reparameterized. Finally, we have obtained the requirement that a CACC application has in an emergency braking scenario, showing that the maximum tolerable delay depends on the dynamics of the maneuver. Maximum delay, however, should not be bigger than 0.2 to 0.3 s. We believe that these results can help the community in the development of efficient, dynamic, and application-aware beaconing protocols for platooning.

REFERENCES

[1] C. Sommer and F. Dressler, *Vehicular Networking*. Cambridge, U.K.: Cambridge Univ. Press, Nov. 2014.

[2] S. Shladover, "PATH at 20—History and major milestones," in *Proc. IEEE ITSC*, Toronto, ON, Canada, Sep. 2006, pp. 22–29.

[3] B. van Arem, C. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 429–436, Dec. 2006.

[4] A. Davila and M. Nombela, "Platooning—Safe and eco-friendly mobility," presented at the Soc. Automotive Eng. World Congr. Exhib., Detroit, MI, USA, Apr. 24–26, 2012.

[5] C. Bergenhem, Q. Huang, A. Benmimoun, and T. Robinson, "Challenges of platooning on public motorways," in *Proc. 17th World Congr. Intell. Transp. Syst.*, Busan, Korea, Oct. 2010, pp. 1–12.

[6] R. Rajamani, H.-S. Tan, B. K. Law, and W.-B. Zhang, "Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 4, pp. 695–708, Jul. 2000.

[7] J. Ploeg, B. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer, "Design and experimental evaluation of cooperative adaptive cruise control," in *Proc. IEEE ITSC*, Washington, DC, USA, Oct. 2011, pp. 260–265.

[8] F. Dressler, H. Hartenstein, O. Altintas, and O. K. Tonguz, "Inter-Vehicle Communication—Quo Vadis," *IEEE Commun. Mag.*, vol. 52, no. 6, pp. 170–177, Jun. 2014.

[9] P. Fernandes and U. Nunes, "Platooning with IVC-enabled autonomous vehicles: Strategies to mitigate communication delays, improve safety and traffic flow," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 91–106, Mar. 2012.

[10] *Wireless Access in Vehicular Environments*, IEEE Std. 802.11p-2010, Jul. 2010.

[11] "Intelligent Transport Systems (ITS); Decentralized congestion control mechanisms for intelligent transport systems operating in the 5 GHz range; Access layer part," ETSI, Sophia Antipolis, France, TS 102 687 V1.1.1, Jul. 2011.

[12] "Intelligent Transport Systems (ITS); Vehicular communications; Basic set of applications; Part 2: Specification of cooperative awareness basic service," ETSI, Sophia Antipolis, France, EN 302 637-2 V1.3.1, Sep. 2014.

[13] C. Sommer *et al.*, "How shadowing hurts vehicular communications and how dynamic beaconing can help," in *Proc. 32nd IEEE INFOCOM, Mini-Conference*, Turin, Italy, Apr. 2013, pp. 110–114.

[14] C. Sommer *et al.*, "How shadowing hurts vehicular communications and how dynamic beaconing can help," *IEEE Trans. Mobile Comput.*, vol. 14, no. 7, pp. 1411–1421, Jul. 2015.

[15] M. Segata *et al.*, "Towards inter-vehicle communication strategies for platooning support," in *Proc. 7th IFIP/IEEE Int. Workshop Nets4Cars-Fall*, Saint Petersburg, Russia, Oct. 2014, pp. 1–6.

[16] "Intelligent Transport Systems (ITS); Harmonized Channel Specifications for Intelligent Transport Systems operating in the 5 GHz frequency band," ETSI, Sophia Antipolis, France, TS 102 724 V1.1.1, Oct. 2012.

[17] R. Rajamani, *Vehicle Dynamics and Control*, 2nd ed. New York, NY, USA: Springer-Verlag, 2012.

[18] M. di Bernardo, A. Salvi, and S. Santini, "Distributed consensus strategy for platooning of vehicles in the presence of time varying heterogeneous communication delays," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 102–112, Feb. 2015.

[19] S. Santini *et al.*, "A consensus-based approach for platooning with intervehicular communications," in *Proc. 34th IEEE INFOCOM*, Hong Kong, Apr. 2015, pp. 1158–1166.

[20] S. Hallé, B. Chaib-draa, and J. Laumonier, "Car platoons simulated as a multiagent system," in *Proc. 4th Workshop Agent-Based Simul.*, Montpellier, France, Apr. 2003, pp. 57–63.

[21] L. Baskar, B. De Schutter, J. Hellendoorn, and Z. Papp, "Traffic control and intelligent vehicle highway systems: A survey," *IET Intell. Transp. Syst.*, vol. 5, no. 1, pp. 38–52, Mar. 2011.

[22] N. An, T. Gaugel, and H. Hartenstein, "VANET: Is 95% probability of packet reception safe?" in *Proc. 11th ITST*, Saint Petersburg, Russia, Aug. 2011, pp. 113–119.

[23] N. An, M. Maile, D. Jiang, J. Mittag, and H. Hartenstein, "Balancing the requirements for a zero false positive/negative forward collision warnings," in *Proc. 10th IEEE/IFIP WONS*, Banff, AB, Canada, Mar. 2013, pp. 191–195.

[24] S. Joerer *et al.*, "Fairness kills safety: A comparative study for intersection assistance applications," in *Proc. 25th IEEE PIMRC*, Washington, DC, USA, Sep. 2014, pp. 1442–1447.

[25] M. Segata and R. Lo Cigno, "Automatic emergency braking—Realistic analysis of car dynamics and network performance," *IEEE Trans. Veh. Technol.*, vol. 62, no. 9, pp. 4150–4161, Oct. 2013.

[26] C. Lei *et al.*, "Impact of packet loss on CACC string stability performance," in *Proc. 11th ITST*, Saint Petersburg, Russia, Aug. 2011, pp. 381–386.

[27] A. Böhm, M. Jonsson, and E. Uhlemann, "Co-existing periodic beaconing and hazard warnings in IEEE 802.11p-based platooning applications," in *Proc. 10th ACM VANET*, Taipei, Taiwan, Jun. 2013, pp. 99–102.

[28] J. Ahn, Y. Wang, B. Yu, F. Bai, and B. Krishnamachari, "RISA: Distributed road information sharing architecture," in *Proc. 31st IEEE INFOCOM*, Orlando, FL, USA, Mar. 2012, pp. 1494–1502.

[29] C. Han, M. Dianati, R. Tafazolli, X. Liu, and X. S. Shen, "A novel distributed asynchronous multichannel MAC scheme for large-scale vehicular ad hoc networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 7, pp. 3125–3138, Sep. 2012.

[30] M. Abualhoul, M. Marouf, O. Shagdar, and N. Fawzi, "Platooning control using visible light communications: A feasibility study," in *Proc. IEEE ITSC*, The Netherlands, Oct. 2013, pp. 1535–1540.

[31] P. Fernandes, "Platooning of IVC-enabled autonomous vehicles: Information and positioning management algorithms, for high traffic capacity and urban mobility improvement," Ph.D. dissertation, Dept. Elect. Comput. Eng., Univ. Coimbra, Coimbra, Portugal, Apr. 2013.

[32] T. Tielert, D. Jiang, H. Hartenstein, and L. Delgrossi, "Joint power/rate congestion control optimizing packet reception in vehicle safety communications," in *Proc. 10th ACM VANET*, Taipei, Taiwan, Jun. 2013, pp. 51–60.

[33] C. Bergenhem, E. Hedin, and D. Skarin, "Vehicle-to-vehicle communication for a platooning system," in *Proc. Transp. Res. Arena*, Athens, Greece, Apr. 2012, pp. 1222–1233.

[34] M. Boban, T. Vinhosa, J. Barros, M. Ferreira, and O. K. Tonguz, "Impact of vehicles as obstacles in vehicular networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 1, pp. 15–28, Jan. 2011.

[35] M. Segata *et al.*, "Vehicle shadowing distribution depends on vehicle type: Results of an experimental study," in *Proc. 5th IEEE VNC*, Boston, MA, USA, Dec. 2013, pp. 242–245.

[36] S. Abbas and A. Sheikh, "On understanding the nature of slow fading in LOS microcellular channels," in *Proc. 47th IEEE VTC*, Phoenix, AZ, USA, May 1997, pp. 662–666.

[37] M. Segata *et al.*, "PLEXE: A platooning extension for veins," in *Proc. 6th IEEE VNC*, Paderborn, Germany, Dec. 2014, pp. 53–60.

[38] C. Sommer, R. German, and F. Dressler, "Bidirectionally coupled network and road traffic simulation for improved IVC analysis," *IEEE Trans. Mobile Comput.*, vol. 10, no. 1, pp. 3–15, Jan. 2011.

[39] *IEEE Standard for Wireless Access in Vehicular Environments (WAVE)—Multi-channel Operation*, IEEE Std. 1609.4, Feb. 2011.

[40] D. Eckhoff, C. Sommer, and F. Dressler, "On the necessity of accurate IEEE 802.11p models for IVC protocol simulation," in *Proc. 75th IEEE VTC-Spring)*, Yokohama, Japan, May 2012, pp. 1–5.

[41] P. Fernandes and U. Nunes, "Platooning of autonomous vehicles with intervehicle communications in SUMO traffic simulator," in *Proc. IEEE ITSC*, Madeira Island, Portugal, Sep. 2010, pp. 1313–1318.

[42] D. Jiang, Q. Chen, and L. Delgrossi, "Optimal data rate selection for vehicle safety communications," in *Proc. 5th ACM VANET*, San Francisco, CA, USA, Sep. 2008, pp. 30–38.

[43] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, IEEE Std. 802.11-2012, Mar. 2012.

[44] R. K. Schmidt, A. Brakemeier, T. Leinmüller, F. Kargl, and G. Schäfer, "Advanced carrier sensing to resolve local channel congestion," in *Proc. 8th ACM VANET*, Las Vegas, NV, USA, Sep. 2011, pp. 11–20.

[45] R. Zheng *et al.*, "Study on emergency-avoidance braking for the automatic platooning of trucks," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1748–1757, Aug. 2014.

Michele Segata received the B.Sc. and M.Sc. degrees in computer science from the University of Trento, Trento, Italy, in 2009 and 2011, respectively. He is currently working toward the joint Ph.D. degree with the Computer and Communication Systems Group, University of Innsbruck, Innsbruck, Austria, and with the Advanced Networking Systems Group, University of Trento. His Ph.D. grant is funded by the BIT Joint School for Information Technology, which is a joint effort of the local governments of Bolzano, Italy; Innsbruck; and Trento.

He is currently working on simulation models, network protocols, and management strategies for semiautonomous driving, with particular focus on platooning. Moreover, he has also worked on safety applications for vehicular networks, software-defined radio prototypes development, and control algorithms for platooning.

Bastian Bloessl received the diploma degree in computer science from the University of Würzburg, Würzburg, Germany, in 2011.

He is currently a Research Assistant with the University of Paderborn, Paderborn, Germany, joining the Distributed Embedded Systems Group in 2014. Before going to Paderborn, he pursued the Ph.D. degree with the Computer and Communication Systems Group, University of Innsbruck, Innsbruck, Austria. Currently, he is using software-defined radio-based prototypes to assess the performance and robustness of vehicular and sensor networks.

Stefan Joerer received the B.Sc. and M.Sc. degrees in computer science in 2009 and 2011, respectively, from the University of Innsbruck, Innsbruck, Austria, where he has been working toward the Ph.D. degree with the Computer and Communication Systems Group since 2011.

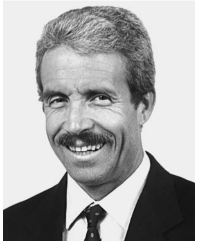
He was a Visiting Scholar with the Wireless Networking Group (WiNG) of Prof. A. Jamalipour in the School of Electrical and Information Engineering, The University of Sydney, Darlington, Australia, in 2014. His current research is focused on questions regarding intervehicle communication for road traffic safety applications.

Mr. Joerer received the prestigious Australia Award Endeavour Research Fellowship to visit The University of Sydney in 2014.

Christoph Sommer received the M.Sc. (Dipl.-Inf. Univ.) degree in computer science and the Ph.D. (Dr.-Ing. with distinction) degree in engineering from the University of Erlangen, Erlangen, Germany, in 2006 and 2011, respectively.

In 2010, he was a Visiting Scholar with the research group of O. K. Tonguz with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, USA. In 2012, he was a Visiting Scholar with the research group of M. Gerla with the Computer Science Department, University of California, Los Angeles, CA, USA. Until 2014, he was a Postdoctoral Research Fellow with the Computer and Communication Systems Group, University of Innsbruck, Innsbruck, Austria. He is currently an Assistant Professor (AkadR a.Z.) with the University of Paderborn, Paderborn, Germany, joining the Distributed Embedded Systems Group in 2014. He is the author of the textbook *Vehicular Networking* (Cambridge University Press, 2014). His research is focused on questions regarding traffic efficiency, safety, and security aspects of Car-to-X communication in heterogeneous environments.

Dr. Sommer has been a member of the Association for Computing Machinery/Springer Wireless Networks (WINET) editorial board since 2011.



Mario Gerla (F'03) received the Engineering degree from the Politecnico di Milano, Milan, Italy, and the Ph.D. degree from the University of California, Los Angeles (UCLA), CA, USA.

At UCLA, he was part of the team that developed the early ARPANET protocols under the guidance of Prof. L. Kleinrock. From 1973 to 1976, at Network Analysis Corporation, New York, he helped transfer ARPANET technology to government and commercial networks. Since 1976, he has been with the faculty of UCLA, where he has designed and implemented network protocols, including ad hoc wireless clustering, multicast (on-demand multicast routing protocol and CodeCast), and Internet transport (TPC Westwood). He has lead the $12M, six-year Office of Naval Research MINUTEMAN project, designing the next-generation scalable airborne Internet for tactical and homeland defense scenarios. He is now leading two advanced wireless network projects under U.S. army and IBM funding. His team is developing a vehicular test bed for safe navigation, urban sensing, and intelligent transport. A parallel research activity explores personal communications for cooperative networked medical monitoring (see www.cs.ucla.edu/NRL for recent publications).

Renato Lo Cigno (M'94–SM'11) received the master's degree in electronic engineering with a specialization in telecommunications from the Politecnico di Torino, Turin, Italy, in 1988.

During 1998–1999, he was with the Computer Science Department, University of California, Los Angeles, CA, USA, as a Visiting Scholar. He was with the Politecnico di Torino until 2002. He is currently an Associate Professor with the Department of Information Engineering and Computer Science (DISI), University of Trento, Trento, Italy, where he leads the Advanced Networking Systems Group in computer and communication networks. He has coauthored more than 150 papers in international peer-reviewed journals and conferences. His current research interests are in the performance evaluation of wired and wireless networks, modeling and simulation techniques, congestion control, peer-to-peer networks, and networked systems in general, with specific attention toward applications and sustainable solutions.

Mr. Lo Cigno is a Senior Member of the Association for Computing Machinery (ACM). He has been the General Chair of the IEEE International Conference on Peer-to-Peer Computing and the General Chair and Technical Program Committee (TPC) Chair of the ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots (WMASH) and the IEEE Annual Conference on Wireless On-demand Network Systems and Services (WONS). He was an Area Editor of *Computer Networks*. He has served on many Technical Program Committees of IEEE and ACM conferences, including the IEEE International Conference on Computer Communications (INFOCOM); the IEEE Global Communications Conference (GLOBECOM); the ACM International Conference on Communications; the ACM International Conference on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM); the IEEE Vehicular Networking Conference; and the International Conference on Emerging Networking Experiments and Technologies (CoNext).

Falko Dressler (SM'08) received the M.Sc. and Ph.D. degrees from the University of Erlangen-Nuremberg, Nuremberg, Germany, in 1998 and 2003, respectively.

He is currently a Full Professor of computer science and the Head of the Distributed Embedded Systems Group, Department of Computer Science, University of Paderborn, Paderborn, Germany. He is the author of the textbooks *Self-Organization in Sensor and Actor Networks* (Wiley, 2007) and *Vehicular Networking* (Cambridge University Press, 2014). He has been an IEEE Distinguished Lecturer and an ACM Distinguished Speaker in the fields of intervehicular communication, self-organization, and bioinspired and nano-networking. His research objectives include adaptive wireless networking, self-organization techniques and embedded system design with applications in ad hoc and sensor networks, vehicular networks, industrial wireless networks, and nano-networking.

Dr. Dressler is a Senior Member of the Association for Computing Machinery (ACM) (SIGMOBILE) and a member of GI (GIBU, KuVS). He regularly serves on the Program Committees of leading IEEE and ACM conferences. He is an Editor of journals such as the IEEE TRANSACTIONS ON MOBILE COMPUTING and Elsevier's *Ad Hoc Networks*, *Computer Communications*, and *Nano Communication Networks*. He was a Guest Editor of special issues on self-organization, autonomic networking, vehicular networks, and bioinspired communication for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, Elsevier's *Ad Hoc Networks*, and others. He was the General Chair of the IEEE/ACM International Conference on Bio-inspired Information and Communications Technologies (BIONETICS) in 2007; IEEE/International Federation for Information Processing Annual Conference on Wireless On-demand Network Systems and Services (WONS) in 2011; the IEEE Vehicular Networking Conference (VNC) in 2014; and the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc) 2016; a Technical Program Committee Co-Chair of the International Conference on Computer Communications; IEEE VNC; the IEEE Vehicular Technology Conference; IEEE Global Communications Conference (GLOBECOM); and the ACM International Conference on Modeling, Analysis, and Simulation of Wireless and Mobile Systems (MSWiM); and Poster/Demo Chair of ACM MobiCom.