This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

An Adaptive Broadcasting Strategy for Efficient Dynamic Mapping in Vehicular Networks

Federico Mason, Marco Giordani, *Member, IEEE*, Federico Chiariotti, *Member, IEEE*, Andrea Zanella, *Senior Member, IEEE*, Michele Zorzi, *Fellow, IEEE*

***Abstract*—In this work, we face the issue of achieving an efficient dynamic mapping in vehicular networking scenarios, i.e., obtaining an accurate estimate of the positions and trajectories of connected vehicles in a certain area. State-of-the-art solutions are based on the periodic broadcasting of the position information of the network nodes, with an inter-transmission period set by a congestion control scheme. However, the movements and maneuvers of vehicles can often be erratic, making transmitted data inaccurate or downright misleading. To address this problem, we propose to adopt a dynamic transmission scheme based on the actual positioning error, sending new data when the estimate overcomes a preset error threshold. Furthermore, the proposed method adapts the error threshold to the operational context according to an innovative congestion control algorithm that limits the collision probability among broadcast packet transmissions. This threshold-based strategy can reduce the network load by avoiding the transmission of redundant messages, and is shown to improve the overall positioning accuracy by more than 20% in realistic urban scenarios.**

***Index Terms*—Vehicular networks; broadcasting; vehicular tracking; congestion control.**

I. Introduction

In recent years, there has been a growing interest in vehicular communications, which have rapidly emerged as a means to support safe and efficient transportation systems through inter-vehicular networking [2]. From a safety perspective, vehicular networks can mitigate the severity of traffic accidents by notifying the vehicles about dangerous situations in their surroundings, including bad road conditions and approaching emergency vehicles [3]. Moreover, they can also support other services, ranging from real-time multimedia streaming to interactive gaming and web browsing [4].

Lately, the automotive industry has been considering new 5G service classes such as Ultra Reliable Low Latency Communication (URLLC) to support more safety-critical applications [5], e.g., security distance warning, cooperative perception, or driver assistance. In order to enable these systems to operate with low latency (less than 10 ms for safety-related services) and high reliability (i.e., up to 99.999% for high degree of automation [6]), the research has focused on the design of novel architectures that guarantee the timely and accurate positioning of vehicles (with errors below 1.5 m for where-in-lane localization, according to the recommendations of the National Highway Traffic Safety Administration (NHTSA) [7]).[1] The vehicular scenario often involves rapid dynamics and unpredictable changes in the network topology [10], and thus requires position updates to be disseminated as timely as possible, ideally at the very same instant they are generated.

In this regard, the traditional approach is to have each vehicle broadcast periodic updates with its positioning information. However, the intrinsically variable topology of vehicular networks might make periodic broadcasting strategies inefficient: long inter-transmission intervals may prevent the timely dissemination of positioning information in safety-critical situations, while very frequent broadcasting may overload the wireless medium with useless data and increase the number of packet collisions [11]. Congestion avoidance mechanisms have thus been proposed in the literature to regulate information distribution as a function of the network load [12]. These techniques dynamically adapt to the number of neighboring vehicles [13] or assign priorities to vehicles based on their operating conditions [14], but usually disregard the level of positioning accuracy that is finally achieved.

To solve these issues, more sophisticated information distribution solutions that explicitly consider the Quality of Information (QoI) [15] have been investigated. These strategies, however, have been typically proposed for ad hoc and sensor networks and may not be directly applied to the vehicular environment. Some other proposals, e.g., [16]–[18], consider the value of the possible position information updates, only broadcasting those that maximize the utility for the target applications. However, congestion control is not considered, and practical validation in real vehicular deployments is still missing.

Following this rationale, in this paper we face the challenge of ensuring accurate position estimation of vehicles while minimizing the network load in a cost-effective way. The main novelty of our work consists in the following points:

- We find a mathematical expression of the packet collision probability as a function of the vehicular traffic density.

Federico Mason, Marco Giordani, Andrea Zanella, and Michele Zorzi are with the Department of Information Engineering, University of Padova, Padova, Italy (email: {masonfed, giordani, zanella, zorzi}@dei.unipd.it). Federico Chiariotti was with the Department of Information Engineering, University of Padova, Padova, Italy, and is now with Aalborg University, Denmark (email: fchi@es.aau.dk).

A preliminary version of this paper that did not consider congestion control was presented at the *25th European Wireless Conference (EW)*, May 2019 [1].

[1]Positioning is typically provided by on-board Global Positioning System (GPS) receivers, which may not always have the required accuracy [8]. For this reason, data fusion techniques [9] have recently been considered by combining several positioning strategies (including, but not limited to, dead reckoning, map matching, and camera image processing) into a single solution that is more robust and precise than any individual approach.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

- Based on the above-mentioned relation, we design a new congestion control mechanism that exploits network topology information to reduce the packet collision probability. Compared with traditional channel-based congestion control, our solution can better adapt to fluctuating conditions of the environment, as is typically the case in the vehicular ecosystem.
- We design a threshold-based broadcasting algorithm that *(i)* estimates the positioning error of the vehicle and its neighbors within communication range, based on the Constant Turn Rate and Acceleration (CTRA) model, and *(ii)* makes vehicles distribute state information messages if the estimated error is above a predefined threshold. Our method is computationally efficient, and can be executed in real time even with the limited on-board computational resources of mid-range and budget car models.
- We investigate the performance of the proposed scheme compared to state-of-the-art solutions in realistic scenarios generated by Simulation of Urban MObility (SUMO) [19], an open simulator designed to model the traffic of large road networks. Hence, a simulator implemented in `python` is used to model the vehicular network and the related communication and tracking processes.
- We provide guidelines on the optimal broadcasting strategy for a given set of automotive-specific parameters, including the transmission periodicity and the number of subcarriers.

The performance of our approach is compared with a baseline *periodic broadcasting* solution that instructs vehicles to broadcast state information at regular intervals. Simulation results show that the proposed algorithm, in spite of its simplicity, can reduce the average position estimation error by more than 10%, and its 95th percentile by more than 20%, compared to the periodic broadcast approach. Analytical validation of our theoretical framework is also provided in the highway scenario, i.e., where traffic dynamics can be analytically represented as a rectilinear motion.

The remainder of this paper is organized as follows. In Sec. II we present a selection of the most relevant related work. In Sec. III we introduce our system model. In Secs. IV and V we describe our broadcasting strategies and congestion control mechanisms, respectively, and derive the expression for the packet collision probability as a function of the vehicular traffic density. In Sec. VI we validate our theoretical analysis through simulations and present our main findings and results. Finally, in Sec. VII we provide conclusions and suggestions for future work.

II. Related Work

In a Connected and Intelligent Transportation Systems (C-ITS) scenario, vehicles are equipped with on-board sensors, which are used to gather observations about the surrounding environment. These observations are then broadcast within the network through wireless technologies, and are used to implement a tracking system [20], [21] whose target state is the set of positions of all surrounding neighbors. The performance of the tracking framework depends on the degree of coordination among the vehicles and on how the data is processed. The most common choice is to adopt a Bayesian filtering approach, typically based on the Kalman Filter (KF) [22], the Unscented Kalman Filter (UKF) [23], or the Particle Filter (PF) [24]. A tracking framework based on the UKF and the CTRA motion model is presented in [25]. In [26], route information and digital map data are jointly processed by a particle filter algorithm. In [27], position forecasting is achieved by using a Hidden Markov Model [28] and the Viterbi algorithm [29]. We highlight that, in all the Bayesian filters, the performance greatly depends on the algorithm settings, e.g., the process and estimation noise covariances, which must be known *a priori* [30].

Conventional tracking approaches mainly focus on the real-time estimation of the target state. However, most advanced C-ITS applications also require a prediction of vehicles' future trajectories. Long-term forecasting can be achieved by simply applying the predictive step of a Bayesian filter to the last available state estimate. However, this solution is very sensitive to imperfections of the motion model: to overcome this issue, more sophisticated approaches have been proposed in the literature. In [31], the output of a KF is used to perform a parametric interpolation of the future path of the target vehicle. In [32], dead reckoning is used to improve the performance of packet forwarding in a highway scenario. Another possibility consists of describing vehicle position prediction as a time series forecasting problem [9]. Hence, Machine Learning (ML) techniques can improve target state estimation over a large time horizon: in [33], Support Vector Machines are used to forecast vehicle trajectories, allowing the estimation of target positions when the GPS signal is not available. In [34], a neural network is trained with historical traffic data and then used to predict vehicles' speeds. Although the ML approach generally guarantees high performance, it is highly scenario-dependent and requires a large amount of data for the initial training, which is often not publicly available. Also, ML solutions may suffer from significant computational complexity and may be hardly executed on board of the vehicles. ML techniques are also often combined with Bayesian algorithms: in [35], the authors present a system that makes use of a Hidden Markov Module to estimate vehicle maneuvers and a Support Vector Machine to predict future vehicle trajectories. In [36], a Radial Basis Function classifier is used to compute the inner parameters of a particle filter, which estimates the long-term motion of the target. In [37], the results of a maneuver recognition system are combined together with the output of a tracking system based on the CTRA motion model.

Regardless of the complexity of the tracking framework, the overall system performance degrades if on-board sensor measurements are not sufficiently accurate. Users can share local information to compensate the low quality of the input data: C-ITS nodes periodically broadcast their own system state by using the Dedicated Short Range Communication (DSRC) technology [38] and the Wireless Access in Vehicular Environment (WAVE) standard [39]. However, the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) channel access scheme may cause congestion in scenarios

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

with high vehicular density: consequently, the transmitted information may be lost due to packet collisions. Defining novel congestion control schemes, which suit the characteristics of modern vehicular networks, is a problem of interest. Over the years, many researchers have proposed different Medium Access Control (MAC) strategies that adapt inter-vehicle communications to channel conditions. In [40], the authors present a rate-adaption strategy that ensures channel stability through vehicular networking. The convergence of the proposed algorithm is theoretically proved and guidance for the choice of the algorithm parameters is provided. In [41], the hidden terminal problem is avoided by adopting a time-slotted structure where each vehicle is assigned a dedicated slot for each frame, during which it can alert its neighbors about its future transmissions. In [42], the authors focus on improving congestion control in road intersections by using a locally-distributed strategy based on ML, where dedicated road infrastructures have the task of deleting redundant communications and assigning specific CSMA/CA parameter settings to different clusters of transmission requests.

A solution for reducing the channel occupancy is to select the optimal transmission strategy as a function of the instantaneous positioning error of nearby vehicles. The authors in [43], for example, propose a broadcasting strategy in which vehicles trigger new transmissions whenever the estimates of their neighbors' errors are above a predetermined threshold. However, such analysis is provided only for specific case scenarios and the framework that predicts future vehicles' states is quite obsolete with respect to current vehicular tracking techniques. Similarly, in [44], the transmission rate by which new information is disseminated through the network is regulated according to both the positioning error and the estimated number of packet collisions. Nevertheless, the authors assume that vehicles are always aware of the number of packets lost during each timeslot, which may not be always true in vehicular scenarios where most packet collisions are caused by the hidden terminal problem and, thereby, cannot be directly sensed by other nodes. In [45], the authors analyze the inter-vehicle communication dynamics that cause the hidden terminal problem. In the same work, the limitations of the CSMA/CA protocol are addressed by varying the vehicle communication range according to the channel occupancy. However, the validity of the approach is proven only in a highway scenario and cannot be generalized to more complex and unpredictable environments.

The adaptation of the broadcasting period according to the channel conditions, as well as the implementation of accurate tracking frameworks, plays a key role in the future development of vehicular networks. In this perspective, our work aims at designing new communication strategies that can minimize broadcasting operations while ensuring accurate position estimation.

III. System Model

In this section, we present the system model that is considered in our study. In Sec. III-A, we theoretically model a C-ITS network as a time-varying Euclidean graph, whose nodes and edges represent vehicles and their communication links, respectively. In Sec. III-B, we define a performance metric that takes into account both the tracking errors and the vehicle positions. Finally, in Sec. III-C and III-D we describe the tracking system implemented by each vehicle and the communication channel through which vehicles' state information is broadcast, respectively.

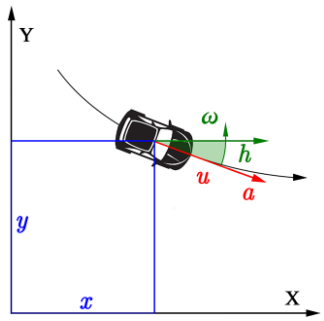


Fig. 1: Graphical representation of the vehicle state $s(t) = \Big(x(t),\, y(t),\, h(t),\, u(t),\, a(t),\, \omega(t)\Big)$ at time t.

A. General Model

We represent a C-ITS network as a Euclidean graph $G = (V, E)$, i.e., an undirected graph whose vertices are points on a Euclidean plane [9]. V represents the set of nodes, i.e., vehicles, while E represents the set of edges. We say that two vehicles $v_i, v_j \in V$, $i \neq j$, are connected by the edge $< v_i, v_j > \in E$ if the average Signal to Noise Ratio $(\text{SNR})_{i,j}$ between them is higher than a threshold Γ_{thr} that makes correct packet reception possible, i.e., $E = \{< v_i, v_j >: i \neq j, \Gamma_{i,j} > \Gamma_{\text{thr}}\}$. This implies that $< v_i, v_j > \in E$ if and only if the Euclidean distance between v_i and v_j is lower than a certain communication range r. Since the composition of the edge set depends on the positions of the vehicles, the topology of the network is time-varying, e.g., new edges can be activated or disabled according to how vehicles move.

In our model, we assume that vehicles move in a two-dimensional space; while not always realistic, this hypothesis does not compromise the accuracy of our analysis. To highlight the time dependency of the network, we denote by $G(t) = (V(t), E(t))$ the network graph at time t. For simplicity, we assume that time is divided into discrete timeslots of duration T_t, so that the system evolves in steps. Hence, we define the neighbor set $N_i(t)$ of $v_i \in V(t)$ at time t as the set of all the vehicles connected to v_i by an edge in $E(t)$, i.e., $N_i(t) = \{v_j \in V(t) :< v_i, v_j > \in E(t)\}$.

The behavior of each vehicle $v_i \in V(t)$ is represented by a 6-tuple $s_i(t) = \Big(x_i(t),\, y_i(t),\, h_i(t),\, u_i(t),\, a_i(t),\, \omega_i(t)\Big)$, which we call *vehicle state*. In particular, x_i and y_i are the Cartesian coordinates of v_i on the road topology, h_i is the vehicle's heading direction, u_i and a_i are the vehicle's tangent velocity and acceleration, respectively, and ω_i is the vehicle's angular velocity as exemplified in Fig. 1. The physical distance between the positions of vehicles v_i and v_j at time t is given by $d(s_i(t), s_j(t)) = \sqrt{(x_i(t) - x_j(t))^2 + (y_i(t) - y_j(t))^2}$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

B. Error Function

In our model each vehicle $v_i \in V(t)$ aims at tracking the position of each neighbor $v_j \in N_i(t)$ at any time t. For the rest of the work, we call *ego vehicle* any vehicle $v_i \in V(t)$ that is tracking a group of neighbors, which are named *target vehicles*. The details of the tracking framework of the ego vehicle will be described in Sec. III-C. Hence, we denote by $\hat{N}_i(t)$ the subset of $V(t)$ containing the target vehicles, by $\hat{s}_{i,j}(t)$ the state estimate of $v_j \in \hat{N}_i(t)$ maintained by v_i, and by $\hat{s}_{i,i}(t)$ the state estimate of v_i (which is the estimate of s_i performed by the ego vehicle itself). We highlight that the real neighbor set $N_i(t)$ might differ from $\hat{N}_i(t)$: the ego vehicle could have no knowledge of a vehicle in $N_i(t)$ (*undetection*), some vehicles in $\hat{N}_i(t)$ might actually be outside the coverage area of v_i (*false alarm*), and communication and tracking errors might lead the ego vehicle to consider an actual neighbor to be outside its communication range (*misdetection*).

Under these hypotheses, the performance of the ego vehicle in terms of position estimation accuracy can be assessed by an *error function* $\mathcal{F}(v_i, t)$, which is a weighted average of the position estimation errors made by the ego vehicle with respect to itself and all its target vehicles. Since computing the ego vehicle's estimation error requires the knowledge of its real state, no node in the network can know it exactly. Furthermore, each node in the network only has partial and often outdated knowledge about its neighbors. We use the error function to evaluate the performance of different tracking systems in our simulations, as we can compute it offline using our knowledge of the ground truth trajectories. We define the error function as

$$\mathcal{F}(v_i,t) = \frac{1}{|\hat{N}_i(t)|+1}\Bigg(\lambda_{i,i}(t)d(\hat{s}_{i,i}(t), s_i(t)) + \sum_{v_j \in \hat{N}_i(t)} \lambda_{i,j}(t) d(\hat{s}_{i,j}(t), s_j(t))\Bigg). \quad (1)$$

In (1), $d(\hat{s}_{i,i}(t), s_i(t))$ and $d(\hat{s}_{i,j}(t), s_j(t))$ represent the error made by v_i in estimating its own state $s_i(t)$ and the neighbor state $s_j(t)$, respectively, $|\hat{N}_i(t)|+1$ represents the total number of estimations carried out by v_i, and $\lambda_{i,j}(t)$ is the generalized logistic function defined as

$$\lambda_{i,j}(t) = A_\lambda + \frac{E_\lambda - A_\lambda}{\left(C_\lambda + D_\lambda e^{-B_\lambda(d(s_i(t),s_j(t))-d_0)}\right)^{1/\nu_\lambda}}. \quad (2)$$

The reader should note that the function in (2) fades away as the distance between the ego vehicle and its neighbors grows above d_0. Therefore, the error function in (1) is more sensitive to positioning errors of vehicles that are closer to the ego vehicle itself. In such a way, we are giving more importance to information updates which may increase the safety of the considered vehicles. The function that describes $\lambda_{i,j}(t)$ is known as Richards' curve [46] and was originally proposed for plant growth modeling: the values of its parameters will be listed in Tab. I.

To evaluate the tracking accuracy performance of the whole network, we define $\mathcal{F}(t)$ as the average of $\mathcal{F}(v_i, t)$ among all vehicles $v_i \in V(t)$:

$$\mathcal{F}(t) = \frac{1}{|V(t)|}\sum_{v_i \in V(t)} \mathcal{F}(v_i, t). \quad (3)$$

C. Tracking System

To minimize the positioning error defined in (1), each vehicle $v_i \in V(t)$ must estimate its state and the state of every other neighbor $v_j \in \hat{N}_i(t)$ at any time t. To reach this goal, the ego vehicle exploits both the information gathered by its on-board sensors and the information received from its neighbors through inter-vehicle communications. To allow the estimation of $s_i(t)$, we assume that, in every timeslot, the ego vehicle's on-board sensors provide a new observation $o_i(t)$ of $s_i(t)$. Hence, the ego vehicle can model the evolution of its own state through a Bayesian approach, obtaining the system

$$\begin{cases} s_i(t+1) = f(s_i(t)) + \zeta(t), \\ o_i(t) = m(s_i(t)) + \eta(t). \end{cases} \quad (4)$$

In (4), the first equation describes the evolution of the vehicle state $s_i(t)$, while the second equation describes the relation between $s_i(t)$ and the state observation $o_i(t)$. In particular, $f(\cdot)$ is a function describing the CTRA motion model given in [47], while $m(\cdot)$ is a function representing the vehicle's measurement system. Moreover, $\zeta(t)$ and $\eta(t)$ represent the process and measurement noises, respectively, and are modeled as independent Gaussian processes with zero mean and covariance matrices Q and R. Once all the parameters in (4) are defined, the ego vehicle can estimate its own state by using a Bayesian filter. In our system, each vehicle implements a UKF algorithm exploiting the *sigma points* parameterization given in [48]. The UKF is an extended version of the classical KF that can handle non-linear system equations. In particular, the UKF summarizes the state information into a set of particles, i.e., the *sigma points*, which can evolve through $f(\cdot)$ and $m(\cdot)$ even under non-linear conditions. After carrying out the predictive and update steps, the filter builds the final state as a weighted average of the *sigma points*, where the weights are based on the system uncertainty. From (4), the ego vehicle can obtain its own estimate $\hat{s}_{i,i}(t)$ and the related covariance matrix $P_{i,i}(t)$, which represents the uncertainty of the state estimation, in each timeslot t.

To allow the ego vehicle to track its neighbors, an additional UKF is deployed for each known neighbor $v_j \in \hat{N}_i(t)$. This UKF has the task of computing the state estimate $\hat{s}_{i,j}(t)$ and is fed by the information broadcast by v_j. Indeed, in any timeslot t, each vehicle $v_j \in V(t)$ can decide to transmit the estimate $\hat{s}_{j,j}(t)$ and the related covariance matrix $P_{j,j}(t)$. The time frame by which new transmissions are initiated depends on the selected broadcasting strategy, as described in Sec. IV. Each message transmitted by v_j is received by all the vehicles in $N_j(t)$ after a certain communication delay (provided that the transmission is not interfered, as we will explain later). Whenever the ego vehicle gets a message from another node that was not previously in its neighbor set, it initializes a new UKF with initial state and uncertainty set equal to those just received. Instead, if the transmitter was known to the ego vehicle, the tracking system is updated by setting

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

$\hat{s}_{i,j}(t) = \hat{s}_{j,j}(t)$ and $P_{i,j}(t) = P_{j,j}(t)$. In both cases, the predictive step of the UKF is used to evolve future estimates of v_j until a new update is received. In this way, the tracking of neighbor vehicles is carried out without transmitting the complete history of observations but only the last computed state estimation, thus saving precious channel resources. We highlight that the UKF directly uses the uncertainty matrix to compute the *sigma points*' average: the transmission of $P_{j,j}(t)$ is necessary to allow v_i to predict the future evolution of $\hat{s}_{i,j}(t)$. If a vehicle v_i does not receive state updates from a neighbor $v_j \in \hat{N}_i(t)$ for a period longer than Δ_{track}, v_j is removed from $\hat{N}_i(t)$.

D. Channel Access Model

Inter-vehicle communications are modeled following the IEEE 802.11p standard, which defines the Physical (PHY) and MAC layer features of the DSRC transmission protocol [38]. DSRC defines seven different channels at the PHY layer, each containing $n_{sc,tot} = 52$ subcarriers [39]. In this work, we assume that only a limited number of subcarriers $n_{\text{sc}} \leq n_{sc,tot}$ can be used for broadcasting state information messages, while the rest is reserved for other applications. DSRC implements the CSMA/CA scheme at the MAC layer, where nodes listen to the wireless channel before sending. We consider an ideal 1-persistent CSMA/CA scheme, capable of successfully arbitrating the channel access among in-range vehicles in such a way that a single transmission per subcarrier and timeslot is enabled, even in case of multiple potential transmitters. However, we assume that collisions can still occur among out-of-range vehicles that transmit towards the same receiver, an issue known in the literature as hidden node problem. Therefore, the transmission from a vehicle v_i to a vehicle v_j will suffer from a hidden terminal collision if any of v_j's neighbors that are out of v_i's range start a transmission that overlaps in time and frequency with v_i's signal. We also design and implement a congestion control algorithm to reduce the channel collision probability. More details will be given in Sec. V.

IV. Broadcasting Strategies

In this section, we describe the communication strategies that are used to regulate inter-vehicle communications in our model. In particular, two different solutions are considered, namely Periodic Broadcasting (PB) (Sec. IV-A), which is already implemented by most C-ITS applications, and Error Threshold Broadcasting (ETB) (Sec. IV-B), our original proposal.

A. Benchmark: Periodic Broadcasting

In the PB scenario each vehicle $v_i \in V(t)$ chooses an *inter-transmission period* $T_{\text{per},i}$. If $T_{\text{per},i}$ remains constant over time, the communication process of v_i follows an almost regular time-frame. This strategy represents the benchmark solution of our analysis as it emulates the broadcast behavior of current vehicular applications which transmit Cooperative Awareness Messages (CAMs) and Basic Safety Messages (BSMs) at regular frequency. Reducing $T_{\text{per},i}$ allows any neighbor $v_j \in N_i(t)$ to receive the state estimate $\hat{s}_{i,i}$ of v_i more frequently, at the expense of increasing the probability of channel access collision. Moreover, a new transmission is initiated each time a neighbor $v_k \notin \hat{N}_i(t)$ is sensed and no transmissions were initiated in the previous two timeslots. This feature is added to allow v_k to update its neighbor set $\hat{N}_k(t)$, thus reducing the undetection probability. The PB strategy is described in Algorithm 1, where T_t is the timeslot duration, $T_{\text{last-tx},i}$ represents v_i's last transmission and *"new neighbor"* is a Boolean variable indicating that a vehicle $v_k \notin \hat{N}_i(t)$ is detected at time t.

Algorithm 1 Periodic Broadcasting (PB) strategy (Sec. IV-A)

Input: $T_{\text{per},i} > 0$, $T_{\text{last-tx},i} > 0$, *new neighbor* $\in$ {*True, False*}
Output: *transmit* $\in$ {*True, False*}

transmit $\leftarrow$ *False*
$T_{\text{last-tx},i} \leftarrow T_{\text{last-tx},i} + T_t$
if $T_{\text{last-tx},i} > T_{\text{per},i}$ **or** (*new neighbor* **and** $T_{\text{last-tx},i} > 2T_t$) **then**
    transmit $\leftarrow$ *True*
end if
if *transmit* **then**
    $T_{\text{last-tx},i} \leftarrow \max\{T_{\text{last-tx},i} - T_{\text{per},i}, 0\}$
end if
return *transmit*

B. New Proposal: Error Threshold Broadcasting

In the ETB scenario each vehicle $v_i \in V(t)$ chooses an *error threshold* $E_{\text{thr},i}$ and regulates its communication behavior so that the overall position estimation error never exceeds $E_{\text{thr},i}$. To reach this goal, the ego vehicle defines an additional UKF, which replicates the UKF operations of all the neighbor nodes that are tracking the ego vehicle itself. This filter propagates the ego vehicle's state by using only its predictive step with no sensor input, as done by the other vehicles. Each time the ego vehicle triggers a new communication, the filter state is updated mimicking the operation performed by neighbor vehicles upon receiving the ego vehicle's state update. Hence, in each timeslot t, the ego vehicle knows both the *a posteriori* state estimate $\hat{s}_{i,i}(t)$, which is the output of its UKF filter, and the *a priori* state estimate $\hat{s}^p_{i,i}(t)$, which is the output of its purely predictive filter and represents the state estimate of v_i made by its neighbor vehicles. In each timeslot, the two different estimates are compared: if the difference $d(\hat{s}_{i,i}(t), \hat{s}^p_{i,i}(t))$ exceeds $E_{\text{thr},i}$, a new transmission is initiated. We observe that, as before, the communication process can vary according to some specific events. A maximum inter-transmission period $T_{\max}$ is defined to mitigate the undetection of new neighbors, and additional transmissions are initiated in case new neighbors are detected. This strategy is described in Algorithm 2, where T_t, $T_{\text{last-tx},i}$ and *"new neighbor"* are defined as in Sec. IV-A.

An intuitive understanding of the PB and ETB dynamics is provided by Figs. 2a and 2b, which represent the evolution of the position error $d(\hat{s}_{i,i}(t), \hat{s}^p_{i,i}(t))$ according to the transmission process in the two cases. In the PB scenario, we can observe that new transmissions are initiated in a regular fashion, regardless of the value of $d(\hat{s}_{i,i}(t), \hat{s}^p_{i,i}(t))$. Instead, in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

Algorithm 2 Error Threshold Broadcasting (ETB) strategy (Sec. IV-B)

Input: $E_{\text{thr},i} > 0$, $T_{\text{last-tx},i} > 0$, *new neighbor* $\in$ {*True, False*}, $\hat{s}_{i,i}(t) \in \mathbb{R}^6$, $\hat{s}^p_{i,i}(t) \in \mathbb{R}^6$

Output: *transmit* $\in$ {*True, False*}

transmit $\leftarrow$ *False*
$T_{\text{last-tx},i} \leftarrow T_{\text{last-tx},i} + T_t$
if $d(\hat{s}_{i,i}(t), \hat{s}^p_{i,i}(t)) > E_{\text{thr},i}$ **or** $T_{\text{last-tx},i} > T_{\max}$ **or** (*new neighbor* **and** $T_{\text{last-tx},i} > 2T_t$) **then**
    transmit $\leftarrow$ *True*
end if
if *transmit* **then**
    $T_{\text{last-tx},i} \leftarrow \max\{T_{\text{last-tx},i} - T_{\max}, 0\}$
    $\hat{s}^p_{i,i}(t) \leftarrow \hat{s}_{i,i}(t)$
end if
return *transmit*

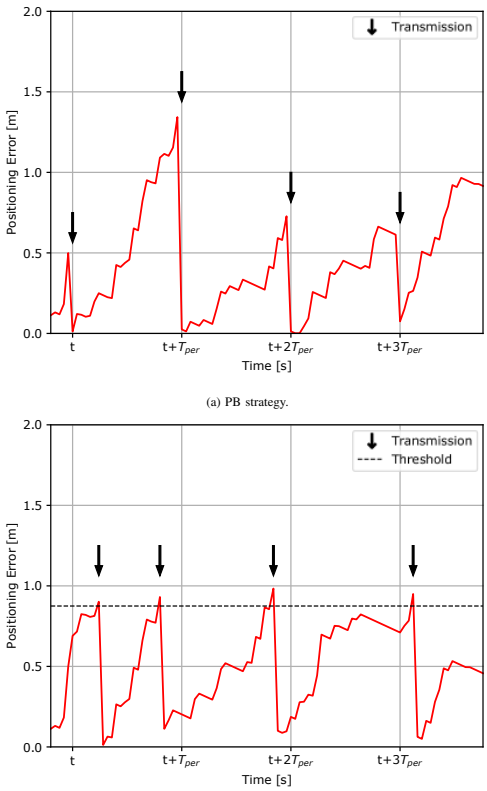
the ETB scenario, new transmissions are initiated only when $d(\hat{s}_{i,i}(t), \hat{s}^p_{i,i}(t))$ is above a certain threshold.[2]

In both the PB and ETB scenarios, the best approach would require any vehicle $v_i \in V(t)$ to determine the optimal value of $T_{\text{per},i}$ and $E_{\text{thr},i}$ at any time t. Such values can be computed by the network infrastructure through an exhaustive and computationally heavy approach that iterates on all the possible values of the number of available subcarriers, the vehicular density, the characteristics of the road map, and other automotive-specific parameters. However, this approach is only possible in simulations, as both the computational and communication loads would be too large for real scenarios. It is far more practical to use less resource-heavy congestion control techniques, as we will show in the next section.

V. Congestion Control

Considering the dynamic nature of the vehicular networks, the potential of the broadcasting strategies described in Sec. IV can be fully expressed when coupled with congestion control mechanisms that regulate information distribution as a function of the network load and minimize the packet collision probability. Practically, congestion control allows each vehicle $v_i \in V(t)$ to independently adjust the values of $T_{\text{per},i}$ and $E_{\text{thr},i}$, so as to adapt the communication process in real time. In Sec. V-A, we first consider a benchmark congestion control mechanism, i.e., Channel Sensing Congestion Control (CSCC), that is based on the LIMERIC protocol [40], which is generally acknowledged as one of the most widely adopted schemes for congestion control [49] and, like most state-of-the-art approaches, relies on channel sensing. In Sec.V-B we design an alternative congestion control approach, i.e., Neighbor Aware Congestion Control (NACC), that exploits network topology information to reduce the packet collision probability. We also find an expression for the packet collision probability as a function of the network load. Finally, in Sec. V-C we specify how congestion control can be practically integrated with the proposed ETB strategy.

[2]We highlight that the complexity of the ETB strategy is only slightly higher than that of the PB strategy: each vehicle needs to maintain an additional UKF which performs only the predictive step in each timeslot.



(a) PB strategy.

(b) ETB strategy.

Fig. 2: Temporal evolution of the positioning error for the broadcasting strategies described in Sec. IV.

A. Benchmark: Channel Sensing Congestion Control

In the CSCC scenario, each vehicle $v_i \in V(t)$ constantly listens to the wireless channel and estimates the amount of resources that it is allowed to use to avoid congestion. Now let us assume that v_i is assigned to subcarrier $c_i \in \{0, 1, ..., n_{\text{sc}} - 1\}$. In each timeslot, v_i senses the channel and determines if a new transmission has been initiated. We denote by $\mathcal{C}_{\ell,i}(t)$ the local channel busy ratio, i.e., the fraction of time during which v_i senses the channel busy during the last $\Lambda^{\text{avg}}_{\mathcal{C}}$ timeslots. Hence, every $\Lambda^{\text{update}}_{\mathcal{C}}$ timeslots, the value of $\mathcal{C}_{\ell,i}(t)$ is smoothed as

$$\mathcal{C}_{v,i}(t) = 0.5 \cdot \mathcal{C}_{v,i}(t) + 0.5 \cdot \mathcal{C}_{\ell,i}(t). \quad (5)$$

In (5), $\mathcal{C}_{v,i}(t)$ is the vehicle average channel busy ratio and represents the channel occupancy sensed by v_i over the subcarrier c_i. In the CSCC approach, each vehicle $v_i \in V(t)$ aims at keeping the value of $\mathcal{C}_{v,i}(t)$ as close as possible to a target value $\mathcal{C}_{\text{target}}$. Practically, every $\Lambda^{\text{update}}_{\mathcal{C}}$ timeslots, v_i evaluates the difference between $\mathcal{C}_{v,i}(t)$ and $\mathcal{C}_{\text{target}}$ and updates accordingly the value of $\rho_i(t)$, which is the fraction of time that v_i can exploit to transmit over the wireless channel. This procedure is described in detail in Algorithm 3, where $\rho_i(t)$ takes values between $\rho_{\min}$ and $\rho_{\max}$, and $[x]^b_a = \min(\max(x, a), b)$. The values of the parameters in Algorithm 3 are chosen according to [40] and will be specified in Sec.VI.

When v_i adopts the PB strategy described in Sec. IV-A, the value of $T_{\text{per},i}$ at time t is updated as $T_{\text{per},i}(t) = 1/\rho_i(t)$. In

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

Algorithm 3 CSCC protocol (Sec. V)

Input: $\mathcal{C}_{v,i}(t) > 0$, $\rho_i(t-1) > 0$
Output: $\rho_i(t) > 0$
  if $\mathcal{C}_{\text{target}} - \mathcal{C}_{v,i}(t) > 0$ **then**
    $\delta = \min\left(\beta \cdot (\mathcal{C}_{\text{target}} - \mathcal{C}_{v,i}(t)), \delta_{\max}\right)$
  else
    $\delta = \max\left(\beta \cdot (\mathcal{C}_{\text{target}} - \mathcal{C}_{v,i}(t)), \delta_{\min}\right)$
  end if
  $\rho_i(t) = [(1-\alpha)\cdot\rho_i(t-1) + \delta]_{\rho_{\min}}^{\rho_{\max}}$
  return $\rho_i(t)$

case v_i is adopting the ETB strategy described in Sec. IV-B, $\rho_i(t)$ should be determined as a function of $E_{\text{thr},i}$, as explained in Sec. V-C.

B. New Proposal: Neighbor Aware Congestion Control

In the NACC approach, each vehicle $v_i \in V(t)$ computes the value of $\rho_i(t)$ as a function of its knowledge about its neighbors' positions. In particular, vehicles can increase or decrease the channel occupancy with the aim of minimizing the packet collision probability. To design NACC, we first theoretically model the communication that takes place in a group of vehicles when CSMA/CA is implemented at the MAC layer, as is the case for IEEE 802.11p operations. Then, we describe how a user can estimate the number of neighbors that may potentially result in packet collisions. Finally, we find a relation between the vehicular density sensed by a user and the packet collision probability itself.

In the following, for the tractability of the analysis, we make some simplifying assumptions. First, we assume an ideal CSMA/CA mechanism, capable of perfectly arbitrating the channel access among in-range nodes, so that only one node at a time can transmit. However, nodes that are mutually hidden can transmit simultaneously. We consider that the communication channel is error-free and that packets are always received if the distance between the vehicles is below the communication range r, except in case of collisions with hidden nodes. We assume that packet capture is impossible, i.e., that if two packets collide, the receiver cannot decode either. We also assume that vehicles attempt to transmit packets according to a Poisson process, and that all vehicles have the same traffic rate, i.e., transmission probability, ρ. Finally, to make the model tractable from a mathematical point of view, we assume that the vehicular density is constant in the considered map. Deriving the optimal congestion control strategy without these assumptions is possible, but would require a more cumbersome analysis and some extra signaling overhead for the vehicles. Although the policy obtained under these simplifying assumptions may turn out to be suboptimal in a more realistic setting, still it provides a reasonable strategy that, as proved by our simulation results, can achieve better performance than current state-of-the-art solutions.

1) CSMA/CA Analysis: We saw in Sec. III-D that vehicles access the channel following an ideal 1-persistent CSMA/CA protocol. We now consider a group of $\mathcal{V}$ vehicles that share the same subcarrier $c \in \{0, 1, ..., n_{\text{sc}} - 1\}$. We assume that all these vehicles are always mutually in-range, i.e., at a distance lower than r, and that no other vehicle can interfere with their communication. In case of multiple vehicles with pending packets, the ideal CSMA/CA algorithm will let just one of them transmit in a time slot. Denoting by q_t the number of vehicles that want to access the channel during the timeslot t, we can write $q_t = \max(q_{t-1} + a_t - 1, 0)$, where a_t is the number of new vehicles attempting a transmission while at most one vehicle managed to transmit in timeslot t_1 (provided that $q_{t-1} > 0$). The memoryless nature of the Poisson arrival process allows us to calculate the probability of a_t given q_{t-1}:

$$P(a_t = a | q_{t-1} = q, \rho, \mathcal{V}) = \binom{\mathcal{V} - q}{a} \rho^a (1-\rho)^{\mathcal{V}-q-a}. \quad (6)$$

Naturally, $P(a_t = a | q_{t-1} = q, \rho, \mathcal{V}) = 0$ if $a > \mathcal{V} - q$. Since the probability of accessing the channel is the same for all vehicles, the probability that a specific vehicle trying to access the channel will transmit is $(q_t + 1)^{-1}$.

We observe that, if ρ and $\mathcal{V}$ are fixed, the channel dynamics at the end of any timeslot t are completely characterized by the number of users that need to transmit, i.e., the queue size q_t. Hence, we can describe the overall system as a Markov Chain, whose state q_t is in the set $Q = \{0, \ldots, \mathcal{V} - 1\}$ and whose transition probability matrix $\mathbf{T}(\rho, \mathcal{V})$ is given in (7).

The steady-state distribution of q is the left eigenvector $\Pi(\rho, \mathcal{V}) = [\Pi_0(\rho, \mathcal{V}), \Pi_1(\rho, \mathcal{V}), ..., \Pi_{\mathcal{V}-1}(\rho, \mathcal{V})]$ of $\mathbf{T}(\rho, \mathcal{V})$ with eigenvalue equal to 1, normalized so that it is a valid probability distribution:

$$\begin{cases} \Pi(\rho, \mathcal{V})\mathbf{T}(\rho, \mathcal{V}) = \Pi(\rho, \mathcal{V}); \\ \sum_{q=0}^{\mathcal{V}-1} \Pi_q(\rho, \mathcal{V}) = 1. \end{cases} \quad (8)$$

We can compute the eigenvector with well-known algebraic methods and normalize it to get the full distribution of q. Hence, we can obtain the probability of different transmission events; in particular, the probability that no transmissions are initiated during a timeslot t is given by

$$P(q_{t-1}=0, a_t=0 | \rho, \mathcal{V}) = \Pi_0(\rho, \mathcal{V}) \cdot P(a_t=0 | q_{t-1}=0, \rho, \mathcal{V}). \quad (9)$$

2) Vehicle Position Distribution: We recall that our objective is to minimize the number of packet collisions, which in our model are caused only by the hidden terminal problem. Let us consider that the ego vehicle $v_i \in V(t)$ is sending a packet to any vehicle $v_j \in N_i(t)$, whose distance from v_i is

$$T_{q,z}(\rho, \mathcal{V}) = \begin{cases} 0, & z < q - 1; \\ P(a_t = z - q + 1 | q_{t-1} = q, \rho, \mathcal{V}), & 0 < q < \mathcal{V},\ q - 1 \le z < \mathcal{V}; \\ P(a_t = z + 1 | q_{t-1} = q, \rho, \mathcal{V}), & q = 0,\ 0 < z < \mathcal{V}; \\ P(a_t \le 1 | q_{t-1} = q, \rho, \mathcal{V}), & q = 0,\ z = 0. \end{cases} \quad (7)$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

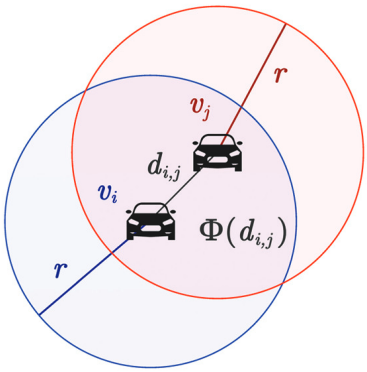


Fig. 3: Intersection $\Phi(d_{i,j})$ of the communication ranges of v_i and v_j.

$d_{i,j}$. To compute the collision probability of v_i, we should estimate how many neighbors of v_j can interfere with the communication: we denote this value by $N_{\text{ht},i}(t)$. Since the vehicular density is constant, we can assume $N_j(t) \simeq \hat{N}_i(t)$. Hence, $N_{\text{ht},i}(t)$ can be estimated as

$$\hat{N}_{\text{ht},i}(t) = \frac{\hat{N}_i(t)+1}{n_{\text{sc}}} \frac{\pi r^2 - \mathbb{E}\left[\Phi(d_{i,j})\right]}{\pi r^2}. \tag{10}$$

In (10), $\frac{\hat{N}_i(t)+1}{n_{\text{sc}}}$ is the estimate of the number of vehicles contained in the communication area of v_j that are using the same subcarriers as v_i, while $\Phi(d_{i,j})$ is the intersection of the communication areas of v_i and v_j. Hence, $\pi r^2 - \Phi(d_{i,j})$ is the size of the area within the coverage of v_j but not of v_i, i.e., the area from which a transmission would be hidden from v_i, possibly causing a hidden node collision. A graphical representation of this scenario is reported in Fig. 3 while the mathematical expression of $\Phi(d_{i,j})$ is given by

$$\Phi(d_{i,j}) = 2r \left(r \arccos\left(\frac{d_{i,j}}{2r}\right) - \frac{d_{i,j}}{2} \sqrt{1 - \left(\frac{d_{i,j}}{2r}\right)^2} \right). \tag{11}$$

The probability distribution of $d_{i,j}$ is equal to $f_d(d_{i,j}) = \frac{2d_{i,j}}{r^2}$. Given (11), the mean value of $\Phi(d_{i,j})$ can be computed as

$$\mathbb{E}\left[\Phi(d_{i,j})\right] = \int_0^{2r} \Phi(\sigma) f_d(\sigma) d\sigma = r^2 \left(\pi - \frac{3\sqrt{3}}{4}\right). \tag{12}$$

Replacing $\mathbb{E}\left[\Phi(d_{i,j})\right]$ in (10) we finally obtain the expression

$$\hat{N}_{\text{ht},i}(t) = \frac{\hat{N}_i(t)+1}{n_{\text{sc}}} \frac{3\sqrt{3}}{4\pi}. \tag{13}$$

3) Packet Collision Probability: On average there are $\hat{N}_{\text{ht},i}(t)$ vehicles which can interfere with the communication of the ego vehicle. Hence, according to our channel model, the probability that the transmission will not fail corresponds to the probability that none of those $\hat{N}_{\text{ht},i}(t)$ interfering nodes transmits during t. Assuming that the ego vehicle $v_i \in V(t)$ has transmission probability ρ_i, while all its interfering vehicles have the same transmission probability $\rho = \rho_i$ and do not interact with other network nodes during t, the packet collision probability P_{coll} of $v_i \in V(t)$ at time t can be derived from

Algorithm 4 NACC protocol (Sec. V-B)

Input: $\hat{N}_i(t) \geq 0$
Output: $\rho_i(t) > 0$
$\hat{N}_{\text{ht},i}(t) = \frac{\hat{N}_i(t)+1}{n_{\text{sc}}} \frac{3\sqrt{3}}{4\pi}$
$\rho_i(t) = \rho$ such that $\Pi_0(\rho, \hat{N}_{\text{ht},i}(t))(1-\rho)^{\hat{N}_{\text{ht},i}(t)} = P_{\text{thr}}$
return $\rho_i(t)$

(9), obtaining

$$P_{\text{coll}}(\rho, \hat{N}_{\text{ht},i}(t)) = 1 - \Pi_0(\rho, \hat{N}_{\text{ht},i}(t))(1-\rho)^{\hat{N}_{\text{ht},i}(t)}. \tag{14}$$

Assuming that the ego vehicle uses the same transmission probability as the interfering nodes, each vehicle $v_i \in V(t)$ can regulate $\rho_i(t)$ such that its collision probability equals a predetermined threshold P_{thr}:

$$P_{\text{coll}}(\rho_i(t), \hat{N}_{\text{ht},i}(t)) = P_{\text{thr}}. \tag{15}$$

This equation can be solved using Eq. (14), since the vehicle knows the value of $\hat{N}_{\text{ht},i}(t)$. We need to find the solution of:

$$1 - \Pi_0(\rho_i(t), \hat{N}_{\text{ht},i}(t))(1-\rho_i(t))^{\hat{N}_{\text{ht},i}(t)} = P_{\text{thr}} \tag{16}$$

In the following, we omit the time dependency from the notation for the sake of readability. Since we know that the steady-state distribution is a proper distribution, and that the eigenvalues are a continuous function of ρ_i, we can prove that a real-valued solution exists by using the intermediate value theorem. We know that if $\rho_i = 0$, no vehicle ever transmits and $\Pi_0(0, \hat{N}_{\text{ht},i}) = 1$ for any value of $\hat{N}_{\text{ht},i}$, and if $\rho_i = 1$, all vehicles always try to transmit, and $\Pi_{\hat{N}_{\text{ht},i}-1}(1, \hat{N}_{\text{ht},i}) = 1$ for any value of $\hat{N}_{\text{ht},i}$. The problem then has a solution for any $P_{\text{thr}} \in [0,1]$. If the number of potential interferers $\hat{N}_{\text{ht},i}$ is small, we can compute the eigenvector explicitly and find the solution, while the solution for larger numbers of potential interferers requires higher-order polynomial equations, which can only be solved numerically. The explicit solution of the equation for the case with two potential interferers is described in Appendix A.

The described procedure is implemented by our proposed NACC protocol to minimize channel congestion as a function of the network load, as described in Algorithm 4. In particular, each vehicle $v_i \in V(t)$ changes the value of $\rho_i(t)$ according to the vehicular density in its surroundings. In case the vehicle is using the PB strategy described in Sec. IV-A, the value of $T_{\text{per},i}$ is updated as $T_{\text{per},i}(t) = \frac{1}{\rho_i(t)}$. Conversely, if the ETB strategy described in Sec. IV-A is implemented, $\rho_i(t)$ will be updated as a function of $E_{\text{thr},i}$, as explained in Sec. V-C. We highlight that, by adjusting the value of $T_{\text{per},i}$ in this way, we violate the assumption regarding the distribution of the packet inter-transmission time considered in the definition of the system Markov model. Indeed, with the PB strategy, the time between two subsequent transmissions is constant rather than geometrically distributed, while in the ETB strategy scenario it depends on the position error evolution. This approximation may impair the performance of our congestion control: in particular, we expect to observe a significant performance reduction in the case of the PB strategy.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

C. Implementing Congestion Control for the ETB Strategy

Both the CSCC and the NACC approaches improve the efficiency of the broadcasting strategies described in Sec. IV by adapting the inter-transmission period to the deployment scenario. As stated previously, to combine a congestion control scheme with the ETB strategy, we have to relate the inter-transmission period to the error threshold. Practically, we need to build a map $\mathcal{M}$ such that the transmission period $T_{\text{per}} = \mathcal{M}(E_{\text{thr}})$ yields an average map estimation error close to E_{thr}. Unfortunately, the relation between E_{thr} and T_{per} is subject to multiple factors and cannot be easily modeled. $\mathcal{M}$ depends on how the position estimation error of vehicles evolves in time, i.e., on both the road map and the users' behaviors.

To reach our goal, we hence resorted to a pragmatic approach. By simulating a purely predictive UKF in the considered scenario, we derive an empirical estimate of the statistical distribution $P(e_h \leq E_{\text{thr}})$ of the position estimation error e_h after h timeslots since the last update, for any $h \geq 0$. Denoting by H_{thr} the number of timeslots in which the error e_h exceeds the threshold E_{thr}, we can set $T_{\text{per}} = \mathbb{E}[H_{\text{thr}}] \cdot T_t$, where T_t is the timeslot duration. Now, pretending that the errors e_h can be modeled as independent random variables, the complementary cumulative distribution function of H_{thr} can be expressed as

$$P(H_{\text{thr}} > H) = \prod_{h=1}^{H} P(e_h \leq E_{\text{thr}}), \tag{17}$$

from which we easily get

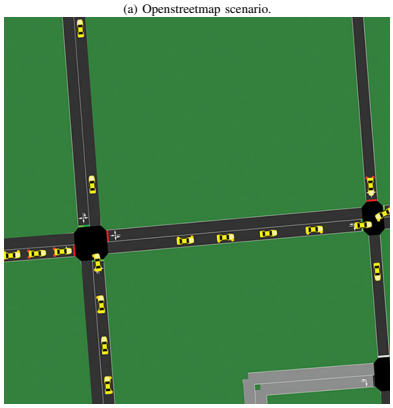
$$T_{\text{per}} = T_t \sum_{H=1}^{\infty} \prod_{h=1}^{H} P(e_h \leq E_{\text{thr}}). \tag{18}$$

Equation (18) hence provides the desired map $\mathcal{M}$ from the error threshold E_{thr} to the inter-transmission period T_{per}. This function can also be used to determine the value ρ of the broadcast policy ETB, which can be computed as $\rho = \frac{1}{\mathcal{M}(E_{\text{thr}})}$. We highlight that this approach requires that vehicles know the distribution of the position estimation error in the map. In a realistic scenario, such information can be provided to vehicles by the road infrastructure, or pre-programmed into the channel access algorithm (possibly with multiple choices, depending on the road conditions). The investigation of such aspects, however, is left to future work.

VI. Performance Analysis and Simulation Results

In this section we evaluate the performance of the proposed ETB strategy for broadcasting operations, compared to a traditional PB approach. Moreover, we exemplify how the proposed NACC mechanism can improve the performance of the broadcasting strategies by exploiting network topology information, with respect to the benchmark CSCC scheme that relies only on channel sensing. Specifically, Sec. VI-A presents our simulation parameters, which are listed in Tab. I, Sec. VI-B analytically validates our theoretical framework in a highway scenario, while Sec. VI-C provides some numerical performance results in a realistic urban C-ITS scenario.

(a) Openstreetmap scenario.



(b) SUMO scenario.

Fig. 4: Representation of a portion of the urban map considered for the performance evaluation.

A. General parameters

Vehicles communicate in the legacy band, i.e., at 5.9 GHz, using the IEEE 802.11p protocol described in Sec. III-D, which is implemented in `python` in our simulations. We use the common piecewise log-distance propagation loss model with Nakagami fading [50], using the parameters derived in [51], which yield a maximum communication range r of about 140 m. In order to simulate the packet error rate for a given SNR, we use the analytical model given in [52], with a physical layer rate R_{PHY} Mb/s and a packet size $P_{\text{size}} = 200$ bytes. In case of collision, we assume that both colliding packets are always lost with no packet capture. The communication delay is set to $T_d = 100$ ms, corresponding to one timeslot T_t. When congestion control is implemented, T_{per} and E_{thr} are automatically adjusted as a function of the network load. When not implementing a congestion control scheme, instead, the settings of both the PB and ETB strategies must be defined *a priori*, which means that all the vehicles adopt the same T_{per} and E_{thr}, respectively. In our simulations we adopt an exhaustive approach and consider $N_{\text{set}} = 30$ different settings. In particular, we make the inter-transmission period T_{per} vary from 0 to 10 seconds while the error threshold E_{thr} ranges between 0 and 42 meters: the trade-off involves both estimation accuracy and broadcasting overhead.

For our simulations, we use real road map data imported from OpenStreetMap (OSM), an open-source software that combines wiki-like user generated data with publicly available information. In particular, we consider the OSM map of New

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

TABLE I: General parameters.

Parameter	Value	Description	Parameter	Value	Description
T_{sim}	100 s	Simulation duration	T_{per}	$\{0, \ldots, 10\}$ s	Inter-transmission period
N_{sim}	20	Number of runs	E_{thr}	$\{0, \ldots, 42\}$ m	Error threshold
T_t	100 ms	Timeslot duration	n_{sc}	$\{2, 4, 6, 8, 10\}$	Number of subcarriers
T_d	100 ms	Communication delay	$\Lambda_{\mathcal{C}}^{\text{avg}}$	10	Timeslot interval in which $\mathcal{C}_\ell$ is computed
Δ_{track}	10 s	Maximum tracking duration	$\Lambda_{\mathcal{C}}^{\text{update}}$	2	Timeslot interval between $\mathcal{C}_v$ updates
$v_{\max}$	13.89 m/s	Maximum speed	K	$\lvert V\rvert / n_{\text{sc}}$	Maximum number of neighbors
d_0	42 m	Safety distance	α	0.1	Algorithm speed parameter
A_S	0.5168 km^2	Area size	β	$(2 - \alpha)/K$	Algorithm convergence parameter
d_v	120 vehicles/km^2	Vehicular density	$\{\delta_{\min}, \delta_{\max}\}$	$\{-1, 1\}$	Lower and upper bounds of δ
$\lvert V\rvert$	62	Number of nodes	$\{\rho_{\min}, \rho_{\max}\}$	$\{-1, 1\}$	Lower and upper bounds of ρ
$\{A_\lambda, B_\lambda, C_\lambda, D_\lambda, E_\lambda, \nu\}$	$\{1, 0.05, 1, 1, 0, 0.2\}$	Logistic function parameters	$\mathcal{C}_{\text{target}}$	0.68	Target Channel Busy Ratio
q	1	Process noise parameter	P_{thr}	0.3	Collision probability threshold
$R_{1,1}$	1.18535 m^2	Position accuracy along x	P_{size}	200 B	Packet size
$R_{2,2}$	1.18535 m^2	Position accuracy along y	P_{tx}	16 dBm	Transmission power
$R_{3,3}$	0.5 (m/s)^2	Speed accuracy	W	-96 dBm	Noise floor
$R_{4,4}$	0.39 (m/s^2)^2	Acceleration accuracy	R_{PHY}	3 Mb/s	Physical layer rate
$R_{5,5}$	0.09211 rad^2	Heading accuracy	r	140 m	Communication range
$R_{6,6}$	0.01587 (rad/s) ^{2}	Turn rate accuracy	$n_{\text{sc,tot}}$	52	Maximum number of subcarriers

York City, as represented in Fig. 4a, so as to characterize a dynamic urban environment. In order to consider realistic mobility routes that are representative of the behavior of vehicles in the network, we simulate the mobility of cars using SUMO, as represented in Fig. 4b. The vehicles move through the street network according to a `randomTrip` mobility model, which generates trips with random origins and destinations, and speeds that depend on the realistic interaction of the vehicle with the road and network elements. The maximum speed is set to $v_{\max} = 13.89$ m/s, which is consistent with current speed limits. Given $v_{\max}$, we set $d_0 = 42$ m, which corresponds to the distance traveled in 3 seconds by a vehicle running at the maximum speed. In this way, d_0 represents the maximum *safety distance* that should be held in an urban scenario. Following the work in [53], we consider a vehicular density of $d_v = 120$ vehicles/km^2 for medium traffic conditions. Given the total road map area of $A_S = 0.5168$ km^2, the number of vehicles deployed in the considered scenario is $|V| = 62$.

The mobility data produced by SUMO are then processed by an ad hoc `python` simulator, which is designed to model both the tracking and the communication processes performed by the system vehicles. As we assessed in Sec. III, the behavior of each node $v_i \in V(t)$ can be fully represented by its state $s_i(t)$. Measurements of the components of $s_i(t)$ are affected by a non-negligible noise which is modeled as a Gaussian process with zero mean and covariance matrix R. The diagonal elements of R are given in Tab. I and are derived from the models in [54]–[56]. We define $Q = qI$, where q is the process noise covariance parameter and I denotes the identity matrix.

Tab. I also reports the parameters of the congestion control schemes from Sec. V. For what concerns the CSCC approach, we use the same parameters suggested in [40]. Therefore, we set $\mathcal{C}_{target}$ to 0.68, so that vehicles aim at occupying the channel about 68% of the time, $\alpha = 0.1$, which ensures a sufficiently high convergence speed, and $\beta = (2 - \alpha)/K$, so that the algorithm's convergence is guaranteed for any K. We observe that K represents the maximum number of users sharing the same communication channel that in our scenario is on average $|V|/n_{\text{sc}}$. For what concerns our proposed NACC approach, we set the collision probability threshold to $P_{\text{thr}} = 0.3$. We choose this value to allow a fair comparison between the CSCC and the NACC approaches. Indeed, setting $\mathcal{C}_{\text{target}} = 0.68$ and $P_{\text{coll}} = 0.3$, we obtain similar values of mean inter-transmission time $\overline{T}_{\text{tx}}$ for each combination of broadcasting strategy and congestion control mechanism.

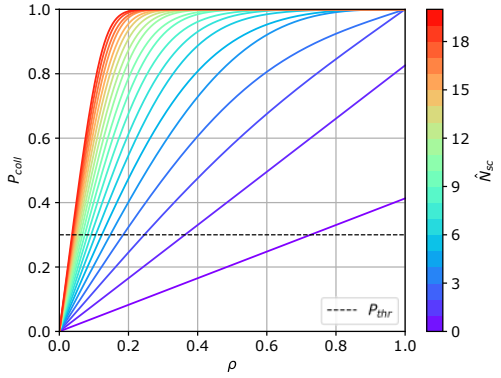
To evaluate the performance of the proposed broadcasting strategies in the simulations, we take into account four main factors, namely:

- *Average positioning error*, i.e., the average error of the ego vehicle when estimating its own position and that of its neighbors, which is given by (1);
- *95th percentile of the positioning error*, i.e., the positioning error threshold exceeded only by the worst 5% of the vehicles;
- *Detection error*, i.e., the sum of the misdetection (i.e., unknown vehicles in the ego vehicle communication area) and false detection (i.e., vehicles that are believed to be in the neighborhood but are actually beyond the communication range) event probabilities;
- *Packet collision rate*, i.e., the average number of packet collisions per vehicle and per second that occur because of the hidden terminal problem.

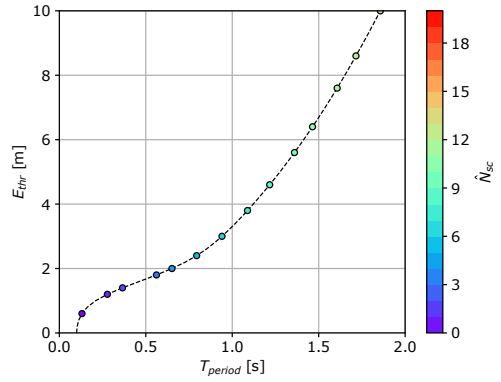
B. Theoretical Analysis and Validation

In Fig. 5a, we plot P_{coll} as a function of ρ_i for different values of $\hat{N}_{\text{sc},i} = \lceil \frac{\hat{N}_i + 1}{n_{\text{sc}}} \rceil$, which represents the estimate of the number of users in the communication range of the ego vehicle using its same subcarrier. By looking at Fig. 5a, we observe that P_{coll} increases with both the transmission probability and the number of interfering neighbors. When using the NACC approach, the ego vehicle sets the transmission probability ρ_i according to both $\hat{N}_{\text{sc},i}$ and P_{coll}. Hence, the value of $T_{\text{per},i}$ is updated as $\frac{1}{\rho_i}$ while the value of $E_{\text{thr},i}$ is updated as $\mathcal{M}^{-1}\left(\frac{1}{\rho_i}\right)$. In Fig. 5b, we represent the function $\mathcal{M}^{-1}$ used for this purpose. In particular, the colored dots represent the different values of $E_{\text{thr},i}$ that are chosen according to both $\hat{N}_{\text{sc},i}$ and P_{coll}.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications



(a) Collision probability.



(b) Function $\mathcal{M}^{-1}$.

Fig. 5: Congestion control analysis.

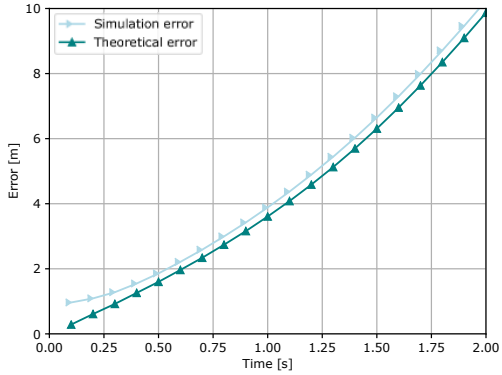
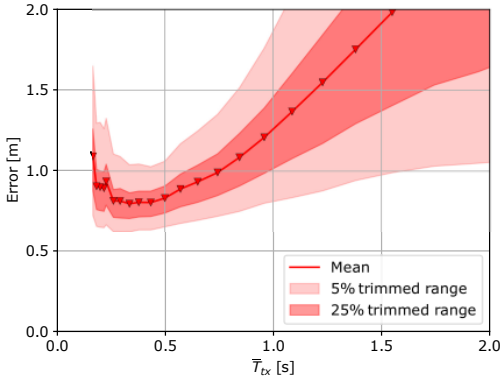
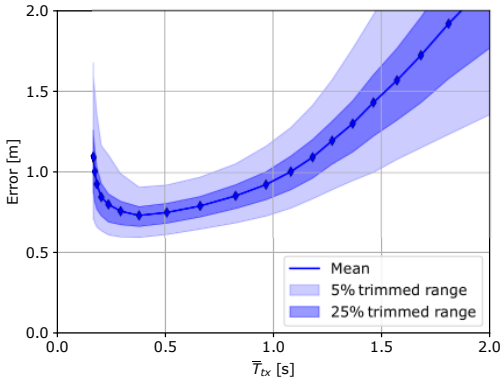


Fig. 6: Simulation vs. theoretical positioning error in a highway scenario.

To validate our analysis, we show how the average tracking error evolves considering the output of the Kalman filter and the empirical results obtained through simulation. In the first case, the motion of a vehicle is analytically represented as a rectilinear motion, disregarding the users' driving imperfections, and the vehicle's speed and direction are assumed constant. In the second case, mobility traces are generated using SUMO. In Fig. 6, we represent the average positioning error obtained in the two different cases as a function of time. We observe that the two data trends are very similar, thereby validating our theoretical framework. The gap between the two curves is due to driving imperfections which cannot be predicted by the rectilinear motion model.



(a) PB strategy.



(b) ETB strategy.

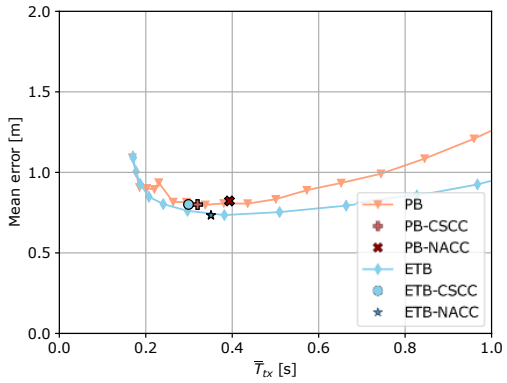
Fig. 7: Positioning error statistics as a function of $\overline{T}_{\text{tx}}$ with $n_{\text{sc}} = 8$.

C. Simulation Results

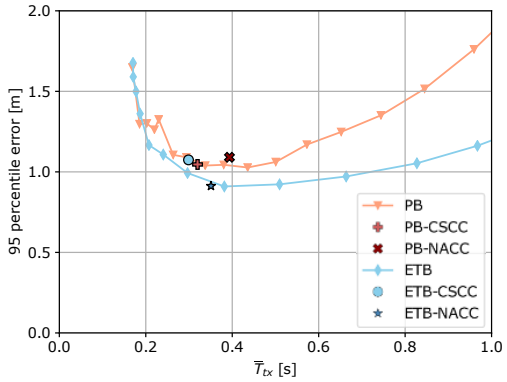
In the rest of the section, the performance of the broadcasting strategies and of the congestion control schemes that we described in Sec. IV and Sec. V, respectively, are analyzed via simulations according to the evaluation methodology described in Sec. VI-A. Hence, the results are derived through a Monte Carlo approach, where N_{sim} independent simulations of duration $T_{\text{sim}} = 100$ s are repeated to get different statistical quantities of interest. This approach has the advantage to increase the level of realism and generalizability of our analysis compared to the evaluation provided in the previous section, and makes it possible to estimate the system performance accounting for realistic channel behaviors and traffic dynamics. All the results are obtained with a realistic communication channel model, as described in Sec. VI-A.

At first, we fix the number of the available subcarriers to $n_{\text{sc}} = 8$. Later, we will verify how different values of n_{sc} may influence the simulation outcomes. In Fig. 7a and Fig. 7b, we analyze the statistics of the positioning error as a function of the mean inter-transmission time $\overline{T}_{\text{tx}}$, which is an indicator of the total channel occupancy. We highlight that $\overline{T}_{\text{tx}}$ does not coincide with the inter-transmission period used in the PB strategy. Indeed, while T_{per} is defined *a priori* and can take all the values within the set $\{0.1 \text{ s}, ..., 10 \text{ s}\}$, $\overline{T}_{\text{tx}}$ is an outcome of the simulation. In particular, in a realistic scenario, $\overline{T}_{\text{tx}}$ never goes below the value of 0.2 s, i.e., two timeslots, because of the channel access contention. From Fig. 7a and Fig. 7b, we can also observe how the limits of CSMA/CA affect the positioning error: when the number of channel access requests

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications



(a) Mean positioning error.



(b) 95 percentile of the positioning error.

Fig. 8: Positioning error statistics as a function of $\overline{T}_{\text{tx}}$ with $n_{\text{sc}} = 8$.

is too high, i.e., $\overline{T}_{\text{tx}} < 0.3$ s, the channel gets congested and, consequently, the performance of the overall system degrades. Indeed, the positioning error can be described by a convex curve, with a minimum for $\overline{T}_{\text{tx}} \approx 0.3$ s; this value represents the level of channel occupancy that guarantees the best position estimation accuracy. By comparing Fig. 7a with Fig. 7b, we can observe that the ETB strategy outperforms the benchmark PB strategy. In particular, ETB proves to have a slightly lower average error and a significantly lower error variance.

By optimizing the channel access requests, the ETB strategy has a lower positioning error than the benchmark strategy. Fig. 8a and Fig. 8b show a direct comparison between the considered broadcasting strategies; in particular, Fig. 8a reports the mean error while Fig. 8b shows the 95th percentile of the error. In both cases, the ETB strategy ensures better position estimation accuracy for the same level of channel occupancy. The marks in Fig. 8a and Fig. 8b represent the performance of the congestion control schemes designed in Sec.V. Since congestion control can adapt the values of T_{per} and E_{thr} and the communication strategy to the scenario in real time, we obtain a single outcome for each combination of broadcasting strategy and congestion control approach. First, we observe that all the deployed solutions succeed in maintaining the channel occupancy close to the optimal working point, i.e., $\overline{T}_{\text{tx}} \approx 0.3$ s. Among all the possible solutions, the combination of the ETB strategy with the NACC approach ensures the best performance. In particular, this scheme outperforms the classical approach used in the literature, which is represented by

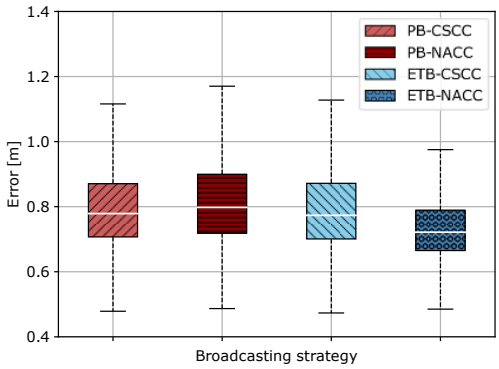


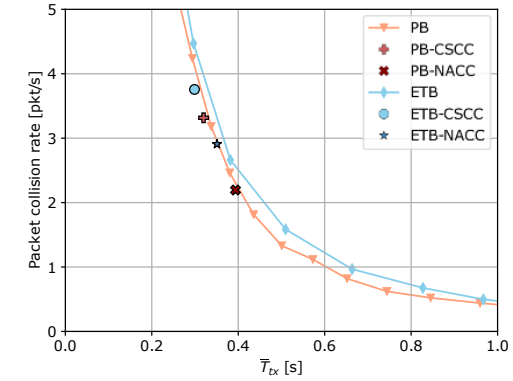
Fig. 9: Boxplot of the positioning error with $n_{\text{sc}} = 8$.

the combination of the PB strategy with the CSCC approach, obtaining a 10% gain when considering the mean error and a 20% gain when considering the 95th percentile of the error, thereby achieving better positioning accuracy for vehicles, a requirement of utmost importance for safety applications [7], [57].

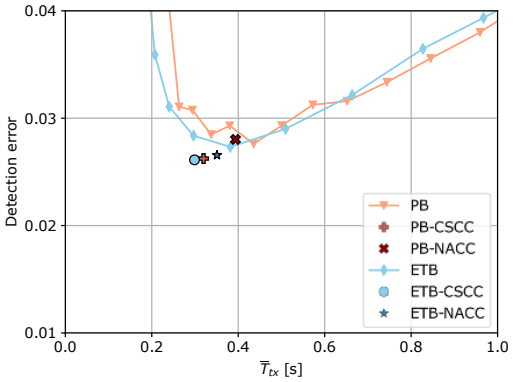
The full positioning error statistics of the four congestion control solutions are shown as a boxplot in Fig. 9. In the figure, each box is delimited by the first and the third quartiles of the error distribution. The box's center lines represent the median of the error, and the whiskers show the 95% confidence intervals. Outliers are represented as dots. We can see that our solution is the only technique that guarantees that the third quartile is below 0.7 m and that the confidence interval is below 1.0 m: this ensures that positioning estimates are not affected by local variations.

In Fig. 10 we show the packet collision rate and the detection error probability. Taking into account the broadcasting strategies without the congestion control schemes, we observe that both techniques present almost identical trends. As we can observe from Fig. 10a, the amount of information that gets lost in the channel significantly increases when $\overline{T}_{\text{tx}} < 0.5$ s, independently of the deployed strategy. This phenomenon explains the degradation of the positioning estimation accuracy that we observe in Fig. 7a and Fig. 7b. Looking at Fig. 10a, the combination of the PB strategy with the NACC approach provides the lowest packet collision rate. This is due to the fact that PB-NACC is the most conservative approach among those we analyzed, i.e., it minimizes the number of transmissions compared to the other communication schemes. On the other hand, PB-NACC exhibits larger positioning estimation errors, as shown in Fig. 9. We highlight that the channel congestion does not affect only the positioning error but also the probability of misdetection and false alarm of a neighbor vehicle. Indeed, by looking at Fig. 10b, we can observe that the detection error probability quickly increases as soon as $\overline{T}_{\text{tx}} < 0.25$ s. All the schemes that implement congestion control have very similar detection error probabilities: since the strategies' optimal working point is $\overline{T}_{\text{tx}} \approx 0.3$, we conclude that minimizing the positioning error does not necessarily imply an increase of the misdetection and undetection events. Note that, because of the long simulation time, we could perform only 20 runs with each parameter setting, so that the

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TWC.2020.2994782, IEEE Transactions on Wireless Communications

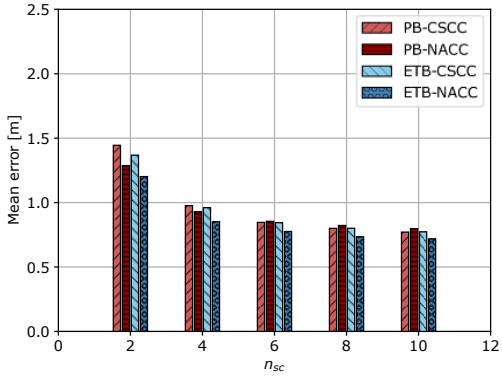


(a) Packet collision rate.

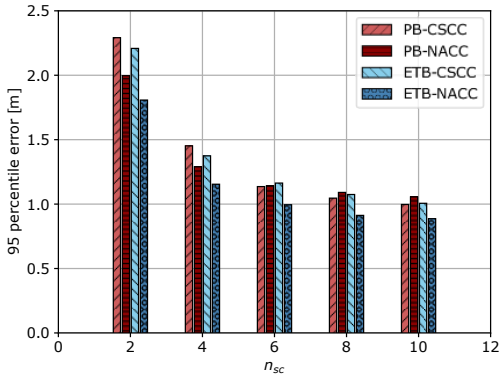


(b) Detection error probability.

Fig. 10: Collision and detection statistics as a function of $\overline{T}_{\text{tx}}$ with $n_{\text{sc}} = 8$.



(a) Mean positioning error.



(b) 95th percentile of the positioning error.

Fig. 11: Positioning error statistics as a function of n_{sc}.

estimate of the detection error probability (which is a rare event) exhibits some statistical oscillations. Nonetheless, we observe that all the schemes that implement congestion control do not deviate from the curves defined by the exhaustive simulations. As already mentioned, the combination of the PB strategy with the NACC approach presents a slightly higher $\overline{T}_{\text{tx}}$ and, therefore, is characterized by a different packet collision rate and detection error probability.

In order to validate these results in a more general scenario, we analyze the performance of the four possible congestion control schemes with different numbers of subcarriers n_{sc}. The results of this analysis are reported in Fig. 11a and Fig. 11b. We observe that the solution combining the ETB strategy and the NACC approach outperforms the other schemes for any value of n_{sc}, considering both the mean positioning error (in Fig. 11a) and the 95th percentile of the positioning error (in Fig. 11b). In particular, our solution outperforms state-of-the-art solutions by up to 20% mean error reduction and up to 30% 95th percentile error reduction. For what concerns the other techniques, we observe that the combination of ETB and CSCC performs poorly for $n_{\text{sc}} \leq 4$, while it leads to better results when the number of subcarriers is greater. Conversely, the combination of PB and NACC performs well for $n_{\text{sc}} \leq 4$ but does not fully exploit the available resources when $n_{\text{sc}} \geq 6$.

Overall, it is possible to provide some guidelines for the configuration of the optimal broadcasting strategy and related parameters in a vehicular deployment. First, Figs. 7a and 7b demonstrate that the proposed ETB solution delivers better localization accuracy compared to a traditional PB approach since state updates are disseminated based on the estimated positioning error of each vehicle. Such approach was also shown to reduce the temporal variance of the positioning error, thereby achieving more robust predictions. Second, Fig. 8a and Fig. 8b illustrate that the proposed NACC strategy outperforms the baseline CSCC strategy in terms of both average localization error and variance, thereby motivating efforts towards the design of a congestion control scheme that exploits network topology information to regulate the periodicity of broadcasting opportunities. Finally, in Figs. 11a and 11b we proved that increasing the number of subcarriers n_{sc} has the benefit of reducing the average number of channel access collisions due to the hidden node problem: as a consequence, the mean localization error is reduced by more than 50% when n_{sc} grows from 2 to 10. In general, among all the possible solutions, the combination of the proposed ETB strategy with the NACC approach ensures the best performance, for any subcarrier configuration.

VII. Conclusions and Future Work

In this work, we studied the trade-off between ensuring accurate position information and preventing congestion of the communication channel in vehicular networks and designed an innovative threshold-based broadcasting algorithm that forces vehicles to distribute state information if the estimated positioning error is above a certain error threshold. We also adopted a new congestion control mechanism that adapts the inter-transmission period according to network topology

information. We showed through simulations that the proposed approach outperforms a conventional broadcasting strategy, which relies on the periodic transmission of state information and channel sensing, since it reduces the positioning error with no additional resources.

As part of our future work, we will test our broadcasting and congestion control frameworks in more complex scenarios, e.g., by considering different road maps and traffic conditions. Besides, we are interested in improving the tracking accuracy of our communication strategy by exploiting the ML paradigm, particularly federated learning, as a tool to decide whether and when to transmit state information updates.

APPENDIX

In this Appendix, we derive the solution of the transmission rate equation in (16) with $\hat{N}_{\text{ht},i} = 2$, which is equivalent to having just two possible interferers. If we apply Eq. (7), we obtain:

$$T(\rho_i) = \begin{bmatrix} 1-\rho_i^2 & \rho_i^2 \\ (1-\rho_i) & \rho_i \end{bmatrix} \tag{19}$$

Since the state of the Markov model represents the number of vehicles waiting to transmit right after a transmission opportunity, there is no case in which both vehicles will be in the queue at the same time, and state 2 is unreachable. We can now solve the system in Eq. (7):

$$\begin{cases} (1-\rho_i^2)\Pi_0 - \Pi_0 + (1-\rho_i)\Pi_1 = 0 \\ \Pi_0 + \Pi_1 = 1 \end{cases}, \tag{20}$$

which results in the following expression for Π_0 and, consequently, for P_{coll}:

$$\Pi_0 = \frac{1-\rho_i}{\rho_i^2 - \rho_i + 1} \tag{21}$$

$$P_{\text{coll}} = 1 - \frac{(1-\rho_i)^3}{\rho_i^2 - \rho_i + 1}. \tag{22}$$

Eq. (16) then becomes:

$$\rho_i^3 - (2 + P_{\text{thr}})\,\rho_i^2 + (2 + P_{\text{thr}})\,\rho_i - P_{\text{thr}} = 0. \tag{23}$$

The solution is now easy to compute. For $P_{\text{thr}} = 0.3$, the optimal transmission rate is $\rho_i \simeq 0.152$.

ACKNOWLEDGMENTS

Part of this work was supported by the US Army Research Office under Grant no. W911NF1910232:"Towards Intelligent Tactical Ad hoc Networks (TITAN)."

REFERENCES

[1] F. Mason, M. Giordani, F. Chiariotti, A. Zanella, and M. Zorzi, "Quality-Aware Broadcasting Strategies for Position Estimation in VANETs," in *25th European Wireless Conference*, May 2019.

[2] H. Hartenstein and L. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, vol. 46, no. 6, pp. 164–171, Jun. 2008.

[3] E. Hossain, G. Chow, V. C. Leung, R. D. McLeod, J. Mišić, V. W. Wong, and O. Yang, "Vehicular telematics over heterogeneous wireless networks: A survey," *Computer Communications*, vol. 33, no. 7, pp. 775–793, May 2010.

[4] M. Amadeo, C. Campolo, and A. Molinaro, "Enhancing IEEE 802.11 p/WAVE to provide infotainment applications in VANETs," *Ad Hoc Networks*, vol. 10, no. 2, pp. 253–269, Sep. 2012.

[5] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected Vehicles: Solutions and Challenges," *IEEE Internet of Things Journal*, vol. 1, no. 4, pp. 289–299, Aug. 2014.

[6] 3GPP, "Service requirements for enhanced V2X scenarios (Release 15)," *TS 22.186*, Sept 2018.

[7] National Highway Traffic Safety Administration (NHTSA), "Federal motor vehicle safety standards; V2V communications," *Federal Register*, vol. 82, no. 8, pp. 3854–4019, 2017.

[8] N. Alam and A. G. Dempster, "Cooperative Positioning for Vehicular Networks: Facts and Future," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1708–1717, Dec. 2013.

[9] L. N. Balico, A. A. F. Loureiro, E. F. Nakamura, R. S. Barreto, R. W. Pazzi, and H. A. B. F. Oliveira, "Localization Prediction in Vehicular Ad Hoc Networks," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 4, pp. 2784–2803, May 2018.

[10] S. Yousefi, M. S. Mousavi, and M. Fathy, "Vehicular Ad Hoc Networks (VANETs): Challenges and Perspectives," in *6th International Conference on ITS Telecommunications*, Jun. 2006, pp. 761–766.

[11] C. Joo and A. Eryilmaz, "Wireless scheduling for information freshness and synchrony: Drift-based design and heavy-traffic analysis," *IEEE/ACM Transactions on Networking*, vol. 26, no. 6, pp. 2556–2568, Dec. 2018.

[12] L. Wischhof and H. Rohling, "Congestion control in vehicular ad hoc networks," in *International Conference on Vehicular Electronics and Safety*, Oct. 2005, pp. 58–63.

[13] G. Caizzone, P. Giacomazzi, L. Musumeci, and G. Verticale, "A power control algorithm with high channel availability for vehicular ad hoc networks," in *International Conference on Communications (ICC)*, vol. 5, May 2005, pp. 3171–3176.

[14] C.-L. Huang, Y. P. Fallah, and R. Sengupta, "Analysis of aggregated power level and rate-power control designs for status update messages in VANETs," in *IEEE 6th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, Oct. 2009, pp. 615–620.

[15] C. Bisdikian, L. M. Kaplan, and M. B. Srivastava, "On the quality and value of information in sensor networks," *ACM Transactions on Sensor Networks (TOSN)*, vol. 9, no. 4, p. 48, Jul. 2013.

[16] M. Giordani, A. Zanella, T. Higuchi, O. Altintas, and M. Zorzi, "Investigating value of information in future vehicular communications," in *IEEE 2nd Connected and Automated Vehicles Symposium (CAVS)*, Sep. 2019.

[17] M. Giordani, T. Higuchi, A. Zanella, O. Altintas, and M. Zorzi, "A framework to assess value of information in future vehicular networks," in *1st ACM MobiHoc Workshop on Technologies, mOdels, and Protocols for Cooperative Connected Cars*, Jul. 2019, pp. 31–36.

[18] T. Higuchi, M. Giordani, A. Zanella, M. Zorzi, and O. Altintas, "Value-anticipating V2V communications for cooperative perception," in *IEEE Intelligent Vehicles Symposium (IV)*. IEEE, Jun. 2019, pp. 1947–1952.

[19] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO-Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, Dec. 2012.

[20] A. Boukerche, H. A. Oliveira, E. F. Nakamura, and A. A. Loureiro, "Vehicular ad hoc networks: A new challenge for localization-based systems," *Computer Communications*, vol. 31, no. 12, pp. 2838–2849, Jul. 2008.

[21] H. S. Ramos, A. Boukerche, R. W. Pazzi, A. C. Frery, and A. A. F. Loureiro, "Cooperative target tracking in vehicular sensor networks," *IEEE Wireless Communications*, vol. 19, no. 5, pp. 66–73, Oct. 2012.

[22] R. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME - Journal of basic Engineering*, vol. 82, pp. 35–45, Jan. 1960.

[23] E. A. Wan and R. V. D. Merwe, "The unscented Kalman filter for nonlinear estimation," in *IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium*, Oct. 2000, pp. 153–158.

[24] P. Del Moral, "Nonlinear filtering: Interacting particle resolution," *Comptes Rendus de l'Académie des Sciences-Series I-Mathematics*, vol. 325, no. 6, pp. 653–658, 1997.

[25] P. Lytrivis, G. Thomaidis, M. Tsogas, and A. Amditis, "An advanced cooperative path prediction algorithm for safety applications in vehicular networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 3, pp. 669–679, Sep. 2011.

[26] A. U. Peker, O. Tosun, and T. Acarman, "Particle filter vehicle localization and map-matching using map topology," in *IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2011, pp. 248–253.

[27] A. T. Akabane, R. W. Pazzi, E. R. M. Madeira, and L. A. Villas, "Modeling and Prediction of Vehicle Routes Based on Hidden Markov Model," in *IEEE 86th Vehicular Technology Conference (VTC-Fall)*, Sep. 2017.

[28] A. B. Poritz, "Hidden markov models: a guided tour," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Apr. 1988, pp. 7–13.

[29] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, Apr. 1967.

[30] M. Roth, G. Hendeby, and F. Gustafsson, "EKF/UKF maneuvering target tracking using coordinated turn models with polar/Cartesian velocity," in *17th International Conference on Information Fusion (FUSION)*, Jul. 2014.

[31] C. M. Kang, S. J. Jeon, S. Lee, and C. C. Chung, "Parametric trajectory prediction of surrounding vehicles," in *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, Jun. 2017, pp. 26–31.

[32] T. King, H. Füßler, M. Transier, and W. Effelsberg, "Dead-reckoning for position-based forwarding on highways," in *International Workshop on Intelligent Transportation (WIT)*, Mar. 2006, pp. 199–204.

[33] D. Wang, J. Liao, Z. Xiao, X. Li, and V. Havyarimana, "Online-SVR for vehicular position prediction during GPS outages using low-cost INS," in *IEEE 26th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, Aug. 2015, pp. 1945–1950.

[34] J. Park, D. Li, Y. L. Murphey, J. Kristinsson, R. McGee, M. Kuang, and T. Phillips, "Real time vehicle speed prediction using a neural network traffic model," in *The International Joint Conference on Neural Networks*, Jul. 2011, pp. 2991–2996.

[35] N. Deo, A. Rangesh, and M. M. Trivedi, "How Would Surround Vehicles Move? A Unified Framework for Maneuver Classification and Motion Prediction," *IEEE Transactions on Intelligent Vehicles*, vol. 3, no. 2, pp. 129–140, Jun. 2018.

[36] C. Hermes, C. Wohler, K. Schenk, and F. Kummert, "Long-term vehicle motion prediction," in *IEEE Intelligent Vehicles Symposium (IV)*, Jun. 2009, pp. 652–657.

[37] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Nov. 2013, pp. 4363–4369.

[38] Y. J. Li, "An overview of the DSRC/WAVE technology," in *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*. Springer, Nov. 2010, pp. 544–558.

[39] D. Jiang and L. Delgrossi, "IEEE 802.11p: Towards an international standard for wireless access in vehicular environments," in *Vehicular Technology Conference (VTC-Spring)*, May 2008, pp. 2036–2040.

[40] G. Bansal, J. B. Kenney, and C. E. Rohrs, "LIMERIC: A Linear Adaptive Message Rate Algorithm for DSRC Congestion Control," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 9, pp. 4182–4197, Nov. 2013.

[41] H. A. Omar, W. Zhuang, and L. Li, "VeMAC: A TDMA-Based MAC Protocol for Reliable Broadcast in VANETs," *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1724–1736, Sep. 2013.

[42] N. Taherkhani and S. Pierre, "Centralized and localized data congestion control strategy for vehicular ad hoc networks using a machine learning clustering algorithm," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 11, pp. 3275–3285, Nov. 2016.

[43] S. Rezaei, R. Sengupta, H. Krishnan, X. Guan, and R. Bhatia, "Tracking the position of neighboring vehicles using wireless communications," *Transportation Research Part C: Emerging Technologies*, vol. 18, no. 3, pp. 335–350, Jun 2010.

[44] C. Huang, Y. P. Fallah, R. Sengupta, and H. Krishnan, "Adaptive intervehicle communication control for cooperative safety systems," *IEEE Network*, vol. 24, no. 1, pp. 6–13, Jan. 2010.

[45] Y. P. Fallah, C. Huang, R. Sengupta, and H. Krishnan, "Analysis of information dissemination in vehicular ad-hoc networks with application to cooperative vehicle safety systems," *IEEE Transactions on Vehicular Technology*, vol. 60, no. 1, pp. 233–247, Jan. 2011.

[46] F. Richards, "A flexible growth function for empirical use," *Journal of experimental Botany*, vol. 10, no. 2, pp. 290–301, Jun. 1959.

[47] M. Tsogas, A. Polychronopoulos, and A. Amditis, "Unscented Kalman filter design for curvilinear motion models suitable for automotive safety applications," in *7th International Conference on Information Fusion*, vol. 2, Jul. 2005, pp. 1295–1302.

[48] S. J. Julier, "The scaled unscented transformation," in *2002 American Control Conference (IEEE Cat. No.CH37301)*, vol. 6, May 2002, pp. 4555–4559 vol.6.

[49] B. Cheng, A. Rostami, M. Gruteser, J. B. Kenney, G. Bansal, and K. Sjoberg, "Performance evaluation of a mixed vehicular network with CAM-DCC and LIMERIC vehicles," in *IEEE 16th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, Jun. 2015, pp. 1–6.

[50] Z. Kang, K. Yao, and F. Lorenzelli, "Nakagami-m fading modeling in the frequency domain for OFDM system analysis," *IEEE Communications Letters*, vol. 7, no. 10, pp. 484–486, Oct. 2003.

[51] J. Benin, M. Nowatkowski, and H. Owen, "Vehicular network simulation propagation loss model parameter standardization in ns-3 and beyond," in *IEEE Southeastcon*, Mar. 2012.

[52] F. Abrate, A. Vesco, and R. Scopigno, "An analytical packet error rate model for WAVE receivers," in *IEEE Vehicular Technology Conference (VTC Fall)*. IEEE, Sep. 2011.

[53] M. Boban, X. Gong, and W. Xu, "Modeling the evolution of line-of-sight blockage for v2v channels," in *IEEE 84th Vehicular Technology Conference (VTC-Fall)*. IEEE, Sep. 2016.

[54] B. Kim and K. Yi, "Probabilistic and holistic prediction of vehicle states using sensor fusion for application to integrated vehicle safety systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2178–2190, Oct. 2014.

[55] T. Driver, "Long-Term Prediction of GPS Accuracy: Understanding the Fundamentals," *20th International Technical Meeting of the Satellite Division of The Institute of Navigation*, vol. 2007, pp. 152–163, Sep. 2007.

[56] G. Falco, M. Pini, and G. Marucco, "Loose and Tight GNSS/INS Integrations: Comparison of Performance Assessed in Real Urban Scenarios," *Sensors*, vol. 2017, p. 27, Jan. 2017.

[57] 3GPP, "Technical Specification Group Radio Access Network; Study on NR positioning support," *TR 38.855*, Mar 2019.

Federico Mason received the bachelor's and master's degree in telecommunication engineering from the University of Padua, Italy, in 2016 and 2018, respectively, where he is currently pursuing the Ph.D. degree. His current research interests include the analysis and development of artificial intelligence algorithms to optimize complex telecommunication systems.

Marco Giordani [S'17-M'20] received his B.Sc. (2013) in Information Engineering, M.Sc. (2015) in Telecommunication Engineering, and Ph.D. (2020) in Information Engineering from the University of Padova, Italy, where he is now a postdoctoral researcher and adjunct professor. He visited New York University (NYU), USA, and TOYOTA Infotechnology Center Inc., USA, in 2016 and 2019, respectively. He has been collaborating with several academic and industrial research partners, including InterDigital, NYU, TOYOTA, and NIST. He is the recipient of several awards, including the "IEEE Daniel E. Noble Fellowship Award" from the IEEE Vehicular Technology Society, the "Francesco Carassa Prize" from the Italian Telecommunications and Information Theory Group (GTTI), and the "Young Researcher Award" from the Human Inspired Technology Research Center (HIT). He has authored more than 40 published papers in the area of millimeter wave wireless networks, two of which have received Best Paper Awards. His research interests focus on protocol and architecture design for 5G/6G cellular and vehicular networks.

Federico Chiariotti [S'15 M'19] is currently a post-doctoral researcher at Aalborg University, Denmark. He received his Ph.D. in information engineering in 2019 from the University of Padova, Italy. He received the bachelor's and master's degrees in telecommunication engineering (both cum laude) from the University of Padova, in 2013 and 2015, respectively. In 2017 and 2018, he was a Research Intern with Nokia Bell Labs, Dublin. He has authored over 30 published papers on wireless networks and the use of artificial intelligence techniques to improve their performance. He was a recipient of the Best Paper Award at the Workshop on ns-3 in 2019 and the Best Student Paper Award at the International Astronautical Congress in 2015. His current research interests include network applications of machine learning, transport layer protocols, Smart Cities, bike sharing system optimization, and adaptive video streaming.

Andrea Zanella [S'98-M'01-SM'13] is a Full Professor at the Department of Information Engineering (DEI), University of Padova, Italy. He received the Laurea degree in Computer Engineering in 1998 from the same University and the PhD in 2001. During 2000, he spent 9 months with Prof. Mario Gerla's research team at the University of California, Los Angeles (UCLA). Andrea Zanella is one of the coordinators of the SIGnals and NETworking (SIGNET) research lab. His long-established research activities are in the fields of protocol design, optimization, and performance evaluation of wired and wireless networks. He has been serving as Technical Area Editor for the IEEE Internet of Things Journal, and Associate Editor for the IEEE Transactions on Cognitive Communications and Networking, IEEE Communications Surveys and Tutorials, and Digital Communications and Networks.

Michele Zorzi [F'07] received his Laurea and PhD degrees in electrical engineering from the University of Padova, Italy, in 1990 and 1994, respectively. During the academic year 1992-1993 he was on leave at the University of California at San Diego (UCSD). In 1993 he joined the faculty of the Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy. After spending three years with the Center for Wireless Communications at UCSD, in 1998 he joined the School of Engineering of the University of Ferrara, Italy, where he became a professor in 2000. Since November 2003 he has been on the faculty of the Information Engineering Department at the University of Padova. His present research interests include performance evaluation in mobile communications systems, WSN and Internet of Things, cognitive communications and networking, 5G mmWave cellular systems, vehicular networks, and underwater communications and networks. He is the recipient of several awards from the IEEE Communications Society, including the Best Tutorial Paper Award (2008, 2019), the Education Award (2016), and the Stephen O. Rice Best Paper Award (2018). He was the Editor in Chief of IEEE Wireless Communications from 2003 to 2005, of the IEEE Transactions on Communications from 2008 to 2011, and of the IEEE Transactions on Cognitive Communications and Networking from 2014 to 2018. He served the IEEE Communications Society as a Member-at-Large of the Board of Governors from 2009 to 2011, as Director of Education in 2014-15, and as Director of Journals in 2020-21.