Machine Learning for Disseminating Cooperative Awareness Messages in Cellular V2V Communications

Luca Lusvarghi, *Graduate Student Member, IEEE*, and Maria Luisa Merani, *Senior Member, IEEE*

***Abstract*—This paper develops a novel Machine Learning (ML)-based strategy to distribute aperiodic Cooperative Awareness Messages (CAMs) through cellular Vehicle-to-Vehicle (V2V) communications. According to it, an ML algorithm is employed by each vehicle to forecast its future CAM generation times; then, the vehicle autonomously selects the radio resources for message broadcasting on the basis of the forecast provided by the algorithm. This action is combined with a wise analysis of the radio resources available for transmission, that identifies subchannels where collisions might occur, to avoid selecting them. Extensive simulations show that the accuracy in the prediction of the CAMs' temporal pattern is excellent. Exploiting this knowledge in the strategy for radio resource assignment, and carefully identifying idle resources, allows to outperform the legacy LTE-V2X Mode 4 in all respects.**

***Index Terms*—CAM, LTE-V2X, safety messages, vehicular machine learning, wireless intelligence.**

I. Introduction and State of the Art

PRESENT days witness an increased and widespread sensitivity to road safety and sustainable transports. Day 1 safety applications are already present on vehicles, to increase space awareness and grant the car and its driver more time to react to unexpected situations. Safety will be further improved by upcoming applications, whose distinctive feature is to rely on vehicular communications. The onset of vehicular networking represents a major turning point, as it lies the basis for Day N services, where fully autonomous and cooperative driving will turn into reality and the goal of a secure and more environment-friendly transports will be accomplished.

Long Term Evolution Vehicle-to-Everything (LTE-V2X) is the first cellular standard for vehicular communications. Its Mode 4 envisions a distributed radio resource selection and reservation procedure that represents the baseline approach for safety services; as a matter of fact, communications occur with no network assistance. The performance of Mode 4 has been investigated by numerous works [1]–[6]. Recently, some studies outlined that it falls short when dealing with aperiodic packet flows [7], [8], and also struggles when transmitting messages of variable size [9]. New Radio (NR)-V2X, the LTE-V2X evolution in the Fifth Generation (5 G) of cellular networks, inherits the majority of LTE-V2X core choices and, as a consequence, the question of how to effectively schedule aperiodic traffic remains unanswered [10], [11].

The current work intends to offer a contribution to this domain, taking a fresh look at the problem of aperiodic safety message dissemination. It concentrates on the main traffic type that LTE-V2X was designed to deliver, namely, Cooperative Awareness Messages (CAMs), application-layer packets standardized by the European Telecommunications Standards Institute (ETSI), and it proposes to harness Machine Learning (ML) to broadcast such messages.

As matter of fact, ML has stirred an unprecedented interest and consensus in numerous wireless settings. This major branch of artificial intelligence is often seen as the appropriate tool to pick the lock of complex problems encountered in, e.g., radio resource allocation and optimization; with no ambition for completeness, [12]–[14] represent captivating examples in the field. The survey in [15] offers an excellent portrayal of the recent ML applications to the specific domain of vehicular networks. However, to the authors' knowledge, none of the studies in the field have scouted the adoption of ML in V2V safety communications.

When the literature on CAM distribution is explored, it is worth recalling [16], which determined an accurate message broadcasting threshold and broadcasting interval to avoid high rate of unnecessary messages. In [17], a novel CAM triggering condition on winding roads was also proposed. Moreover, the authors of [18] explored a simple mechanism to confine the queueing delays suffered by CAMs which coexist with different traffic types. In all of the above papers, the authors modified the generation pattern of CAM traffic. As opposed to such contributions, the current work adheres to the ETSI EN 302 637-2 standard for CAM generation [19], and forecasts when the next CAMs will be produced; it then reserves resources accordingly, modifying Mode 4 scheduling in a very effective manner.

In analogy to the present study, [20] and [21] investigated the issue of radio resource assignment in vehicular networks. In [20], CAMs were compressed to reduce the channel load;

Manuscript received September 9, 2021; revised April 13, 2022; accepted April 22, 2022. Date of publication April 28, 2022; date of current version July 18, 2022. This work was supported by the University of Modena and Reggio Emilia, Italy, through the FAR 2021 Research Grant. The review of this article was coordinated by Dr. S. Misra. *(Corresponding author: Luca Lusvarghi.)*

The authors are with the Department of Engineering "Enzo Ferrari", University of Modena and Reggio Emilia, 41125 Modena, Italy, and also with the Consorzio Nazionale Interuniversitario per le Telecomunicazioni, CNIT, 43100 Parma, Italy (e-mail: luca.lusvarghi5@unimore.it; marialuisa.merani@unimore.it).

Digital Object Identifier 10.1109/TVT.2022.3170982

yet, the authors themselves evidenced that compressing and decompressing is time-consuming. In [21], the resource allocation method of Mode 4 was tuned to different packet sizes. However, it has to be observed that the size of CAMs is not known *a priori* and can vastly vary, which in practice prohibits the adoption of the solution in [21].

On a different rim, [22] and [23] considered message delivery for cooperative awareness, but focused on Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA), the access strategy adopted in the Medium Access Control (MAC) sublayer of 802.11p. Namely, [22] considered a simplified, periodic model for CAM traffic and leveraged on full-duplex transceivers; [23] highlighted the impact of realistic mobility patterns on 802.11p operation. On the contrary, our contribution is centered on LTE-V2X Mode 4, and on improving the CAM delivery process via ML.

Additional references are [24]–[27]: the authors of [24] examined LTE-V2X Mode 3, hence the scenario where the eNodeB controls the allocation of resources to V2V communications; the authors of [25] investigated a centralized multicast/broadcast approach too. Conversely, our solution is totally decentralized, as LTE-V2X Mode 4 mandates. The study in [26] faced the design of V2V communications and employed the sub-6 GHz band exclusively for the control plane, whereas the data plane was positioned at mmWave frequencies. Similarly to [26], [27] considered mmWave communications and allowed for multi-hop transmissions among vehicles. On the contrary, the current investigation is sub-6 GHz centered and examines single-hop transmissions, adhering to the standard for cellular V2V communications. Overall, this paper paves a new research path on LTE-V2X Mode 4, as:

- it addresses the question of whether ML can help in serving aperiodic traffic in Mode 4, given the latter is a recognized benchmark for safety communications in a vehicular environment;
- it provides a largely positive answer. Mode 4 standard is therefore enhanced with a mechanism to predict when CAMs will be generated and when to reserve radio resources on the time-frequency grid of LTE.

In detail, the current study proposes to interpret the aperiodic CAM sequence as a series of sub-sequences that are periodic over a short time scale, and to rely on ML to forecast the sub-sequence length and periodicity. Then, the idea is to tailor Mode 4 radio resource reservations so as to fit the period and length of the single sub-sequence forecast by ML, reducing the risk of future collisions. Additionally, the identification of free radio resources performed by Mode 4 (and analogously by Mode 2 in NR-V2X) is modified and made more effective. The main outcomes of this work are the following:

- ML achieves excellent accuracy in predicting the temporal patterns of CAMs;
- the new, ML-enhanced scheduling of resources outperforms LTE-V2X Mode 4 under all points of view, warranting higher rates of packet delivery, fewer collisions and better channel utilization.

The rest of the paper is organized as follows. In Section II, the main features of LTE-V2X are recalled, and the challenges that Mode 4 faces in the dissemination of aperiodic traffic are outlined. The generation rules of CAMs and their intrinsic aperiodicity are also discussed. In Section III, the new ML-based policy to accommodate aperiodic CAM traffic is illustrated in detail. In Section IV, the performance metrics are introduced. In Section V, the simulation results are presented and the conclusions are drawn in Section VI.

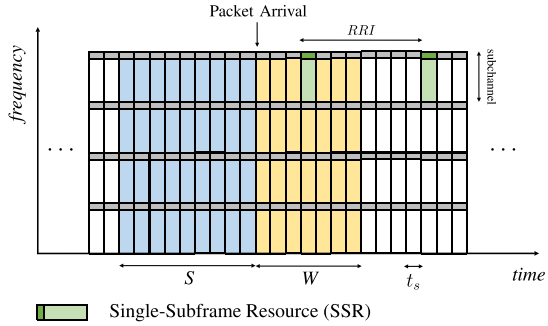


Fig. 1. LTE-V2X time-frequency resource grid.

II. C-V2X Communications

A. LTE-V2X in Release 14

The LTE-V2X solution for vehicular communications has been standardized by the Third Generation Partnership Project (3GPP) in Release 14. Also known as Cellular Vehicle-to-Everything (C-V2X), this technology was mainly designed to disseminate CAMs, Decentralized Environmental Notification Messages (DENMs) and Basic Safety Messages (BSMs), and therefore to allow the development of a first, fundamental set of safety applications. In order to support vehicular communications in both in-coverage and out-of-coverage scenarios, LTE-V2X introduced two different resource allocation schemes known as Mode 3 and Mode 4. Mode 3 delegates the selection of collision-free radio resources to the evolved Node B (eNB), which coordinates the assignment of resources to all vehicles under cellular coverage. However, safety-critical applications cannot depend on the availability of the cellular infrastructure; hence, Mode 4 has been designed to allow vehicles to select resources via an autonomous and distributed approach that requires no eNB assistance.

In LTE-V2X Mode 4, vehicles communicate over a 10 or 20 MHz wide channel located in the 5.9 GHz Intelligent Transport System (ITS) band. At physical layer, Orthogonal Frequency Division Multiplexing (OFDM) is employed with a fixed 15 kHz subcarrier spacing, and transmission resources are arranged over the time-frequency resource grid exemplified in Fig. 1. The time unit is the subframe, whose duration is $t_s = 1$ ms, whereas the basic frequency unit is the Resource Block (RB), 180 kHz wide. A group of adjacent RBs within the same subframe is called a subchannel. In LTE-V2X, every packet is encapsulated within a Transport Block (TB), whose transmission requires a different number of subchannels, depending on the TB size. Moreover, the transmission of each TB is complemented by the corresponding Sidelink Control Information (SCI), which contains decoding-critical information and is transmitted over two RBs, which are frequency-adjacent to the associated TB.

In Release 14, the Mode 4 resource allocation mechanism has been mainly tailored to serve periodic traffic. This is manifest in the Sensing-based Semi-Persistent Scheduling (SSPS) algorithm that the vehicles adopt for the distributed selection of transmission resources. The outcome of the SSPS mechanism is the selection – and reservation – of a collision-free Single-Subframe Resource (SSR), defined as the set of subchannels able to accommodate the transmission of the TB and of its associated SCI. Let us indicate the vehicle that needs to transmit a message and runs the SSPS algorithm as the ego-vehicle. The steps that it goes through are the following:

1) *List Creation:* in the first phase, the ego-vehicle focuses on the Candidate Single-subframe Resources (CSRs) included within the selection window, W. As Fig. 1 indicates, the selection window is the interval that goes from the time the packet is ready for transmission up to its latency deadline, dependent on the Packet Delay Budget (PDB). The ego-vehicle exploits the channel status information collected during the previous 1000 subframes, the so-called sensing window S, to learn which resources in W are already reserved by other vehicles. The ego-vehicle, therefore, builds a list, L_1, removing from the selection window the CSRs that satisfy the following two conditions: *(i)* the ego-vehicle has received an SCI indicating that the CSR will be used by another vehicle; *(ii)* the Reference Signal Received Power (RSRP) averaged over the RBs of the examined CSR is higher than a given threshold. Such threshold is a configurable parameter and its value is iteratively increased by 3 dB until list L_1 includes at least 20% of the initial CSRs. Last, the ego-vehicle builds a second list, L_2, including the top 20% of the CSRs in L_1 with the lowest average Received Signal Strength Indicator (RSSI). The RSSI value is averaged in a periodic fashion over the 10 previous occurrences of the examined CSR, equally spaced of 100 ms.
2) *Resource Selection and Reservation:* in the second phase, the ego-vehicle randomly selects an SSR from list L_2 and also randomly sets the reselection counter C_{resel} in $[C_{\min}, C_{\max}]$, indicating the consecutive number of times the resource will be reserved. For a packet periodicity $T \geq 100$ ms, $C_{\min} = 5$ and $C_{\max} = 15$ [28]. The time interval between consecutive reservations is termed Resource Reservation Interval (RRI), and it matches the packet generation period T, $RRI = T$. After each transmission, the reselection counter is decremented by one; when it expires, the SSPS algorithm is invoked again with probability $1 - P$, $P \in [0, 0.8]$ as indicated in [29].

Once the SSR has been selected, the ego-vehicle broadcasts the TB and the SCI, the latter including the RRI value. Neighboring vehicles are informed that the ego-vehicle intends to employ the same SSR for the next transmission after RRI ms, and avoid using that resource. If the ego-vehicle does not maintain the current reservation when the reselection counter expires, it notifies others by setting the RRI in the SCI equal to 0. Fig. 1 visually summarizes the relevant elements of the SSPS algorithm.

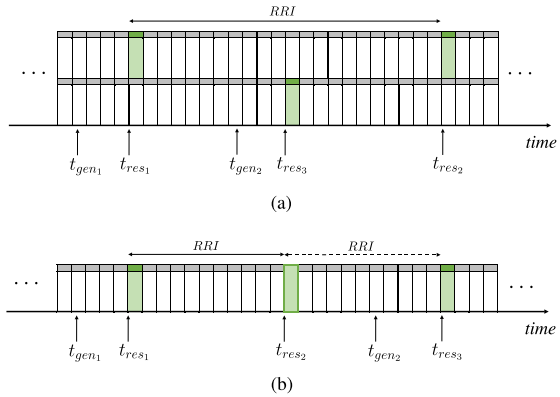


Fig. 2. Aperiodic traffic effects on LTE-V2X Mode 4 reservations. (a) Latency reselection. (b) Unused reservation.

B. Impact of Aperiodic Traffic on Mode 4

When periodic traffic is examined, the RRI setting is a simple task, as the RRI has to match the packet generation period T. Depending on the value of the reselection probability P, Mode 4 is forced to select new resources only when C_{resel} expires; following the vocabulary in [9], this is an event termed *counter reselection* throughout this work. Note that the number of counter reselections a vehicle performs depends on the reselection probability P and on the average C_{resel} value.

However, when aperiodic traffic is considered, additional and unforeseen resource reselections can be triggered. Specifically, when resources are reserved with an RRI larger than the current packet inter-arrival time, then the so-called *latency reselections* [9] occur.

The situation is exemplified in Fig. 2(a): here, it is assumed that at t_{gen_1} an incoming packet triggers a counter reselection: the next two SSRs are reserved at t_{res_1} and t_{res_2}, $t_{res_2} = t_{res_1} + RRI$. Then, let next packet be generated at t_{gen_2}, $t_{gen_2} < t_{res_2}$, but $t_{res_2} - t_{gen_2} > PDB$; it follows that the reserved resource is not able to cope with the packet latency deadline. Therefore, a latency reselection is triggered at t_{gen_2}, and a new set of subchannels is selected and reserved at time t_{res_3} replacing the original reservation. Latency reselections should be avoided as much as possible, as they increase the probability of packet collisions.

Aperiodic traffic is also responsible for the phenomenon of *unused reservations* [9], which are observed when resources are reserved with an RRI lower than the current packet inter-arrival time. This circumstance is illustrated in Fig. 2(b), where the packet generated at t_{gen_1} triggers the reservation of resources at t_{res_1}, t_{res_2} and t_{res_3}, with $t_{res_2} = t_{res_1} + RRI$ and $t_{res_3} = t_{res_1} + 2RRI$. However, the second packet is generated at $t_{gen_2} > t_{res_2}$, hence leaving the reservation at t_{res_2} idle. Unused reservations negatively affect Mode 4 performance in two different ways: first, a fraction of the overall system capacity is wasted, as the reserved resources are not utilized by either the ego-vehicle or the neighboring vehicles. Second, as shown in Fig. 2(b), the unused reservation at t_{res_2} does not allow the

ego-vehicle to broadcast the corresponding SCI and announce the next reservation at t_{res_3}; the resources employed by the ego-vehicle at t_{res_3} are therefore sensed free from nearby users, increasing the risk of packet collisions.

To summarize, the RRI configuration is a key element for the proper operation of Mode 4 SSPS mechanism. Ideally, the RRI value should match the time pattern of the traffic profile, therefore varying over time. However, this task cannot be accomplished when aperiodic traffic is considered, and the inevitable mismatch can severely affect Mode 4 communication effectiveness. In this regard, the authors in [7] and [9] showed that the performance of LTE-V2X is degraded to a remarkable extent, when aperiodic messages are considered.

To the authors' knowledge, NR-V2X has not identified a solution to cope with aperiodic traffic. The question of how to accommodate such traffic type therefore remains open and it is addressed by the current work in the case of aperiodic CAM dissemination. To this aim, the next Section will elaborate on CAMs; the goal is to substantiate that CAMs are aperiodic, but their occurrence pattern can be forecast.

C. ETSI-Generated CAM Sequences

CAMs are facility-layer packets devised to regularly broadcast and exchange information among vehicles, and between vehicles and the roadside infrastructure. They represent the fundamental elements to build road safety and traffic efficiency applications [19]. When initially investigating LTE-V2X performance, CAM occurrences were modeled as periodic packets [30], a choice that perfectly suits the use of Mode 4. However, the standard algorithm for the generation of CAMs released by ETSI [19] indicates that the inter-arrival time between consecutive messages, T_{GenCAM}, is variable. Its duration strongly depends on the vehicle dynamics: if the vehicle modifies its trajectory, if its speed or acceleration/deceleration are sufficiently high, then T_{GenCAM} shortens and CAMs become more frequent. In greater detail, the ETSI algorithm defines the upper and lower bounds for T_{GenCAM}, namely:

$$T_{GenCAM_{\text{Min}}} \leq T_{GenCAM} \leq T_{GenCAM_{\text{Max}}} \tag{1}$$

where $T_{GenCAM_{\text{Min}}} = 100$ ms and $T_{GenCAM_{\text{Max}}} = 1000$ ms, the latter also representing the default value for the generation period. Within such limits, CAMs are triggered depending on the transmitting vehicle dynamics, which have to be sampled every $T_{CheckGenCAM}$ milliseconds, $T_{CheckGenCAM} \leq T_{GenCAM_{\text{Min}}}$. The typical setting is $T_{CheckGenCAM} = T_{GenCAM_{\text{Min}}} = 100$ ms. Specifically, a new CAM shall be immediately generated every time one of the following conditions is met [19]:

- the absolute difference between the current heading and the heading included in the previous CAM is greater than 4°;
- the distance between the current position and the position included in the previous CAM is greater than 4 m;
- the absolute difference between the current speed and the speed included in the previous CAM is greater than 0.5 m/s;
- the time elapsed since last CAM generation is equal to or greater than $T_{GenCAM_{\text{Max}}}$.

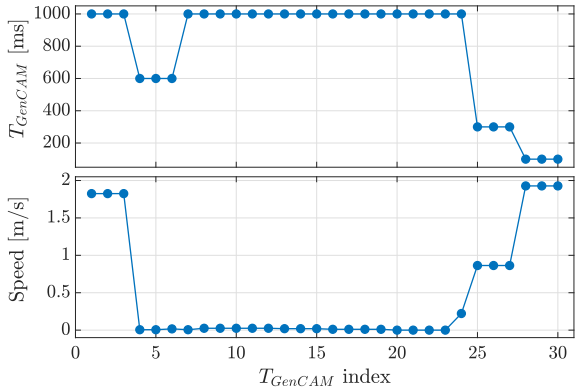


Fig. 3. Correlation between vehicular speed and T_{GenCAM}.

Besides T_{GenCAM} specifications, [19] details the mandatory and optional fields in a CAM, allowing for variable size messages. The rules of the standard lead to CAM traffic which in most of the cases exhibits aperiodic inter-arrival times and variable CAM sizes. The last remark is well documented by the experimental survey in [31], which offers an analysis of CAMs collected during actual test-drives. The study reveals that CAM inter-arrival time often changes from one message to the next, that its distribution is very diverse, and heavily dependent on the drive scenario (urban, suburban, or highway). Similar conclusions hold for the size variability of CAMs. The correlation between CAM inter-arrival times and the vehicle behavior is exemplified in Fig. 3, where the temporal sequence of T_{GenCAM} values, i.e., a CAM trace, and the vehicle speed are reported as a function of the T_{GenCAM} sample index. These patterns refer to a vehicle moving along a straight trajectory, that initially decelerates until a complete stop, at T_{GenCAM} index $= 4$, and then starts to accelerate again from T_{GenCAM} index $= 23$, causing T_{GenCAM} to accordingly vary. Fig. 3 shows that when the vehicle decelerates (accelerates) in the first (last) portion of the CAM trace, CAMs are more frequently issued. On the other hand, T_{GenCAM} settles at 1000 ms when the vehicle stops, in the central portion of the trace. Here, variations in heading, position, or speed are not sufficiently large for generating a CAM before the timeout condition, $T_{GenCAM_{\text{Max}}} = 1000$ ms, occurs. Such a simple, yet exemplary instance is extracted from a wider measurement campaign we performed in different settings [32].

The strong correlation between CAM inter-arrival times and vehicle dynamics suggests that the adoption of ML can be beneficial to predict the temporal evolution of CAM sequences and in turn, lead to an effective reservation policy of radio resources. Indeed, a carefully chosen set of input features, that the vehicle locally retrieves, can be used to feed an ML algorithm, producing the desired outcome, i.e., when next CAMs are likely to occur.

The following Section will therefore illustrate a novel approach to deliver aperiodic CAMs, removing the intrinsic inefficiencies that plague the original Mode 4.

III. The Predictive Reservation Framework

Subsection II-B highlighted the mismatch between aperiodic traffic and the periodic reservation strategy of Mode 4, causing

the undesired phenomena of latency reselections and unused reservations. Moreover, Sec. II-C dwelled on the aperiodicity of actual CAM sequences, suggesting that their temporal evolution can be successfully predicted.

The key proposal of this paper is therefore the following: adopt ML to forecast what the next T_{GenCAM} value will be, and how many occurrences of it will appear. Next, exploit ML prediction to set: (i) the resource reservation interval RRI; (ii) the reselection counter C_{resel}, whose value is no longer randomly chosen, rather, it exactly matches the number of occurrences forecast by ML.

Additionally, the current study significantly intervenes in the list creation phase of the original SSPS algorithm. As better explained in the next subsection, it builds a more reliable list of available candidate resources than the one produced by the legacy SSPS.

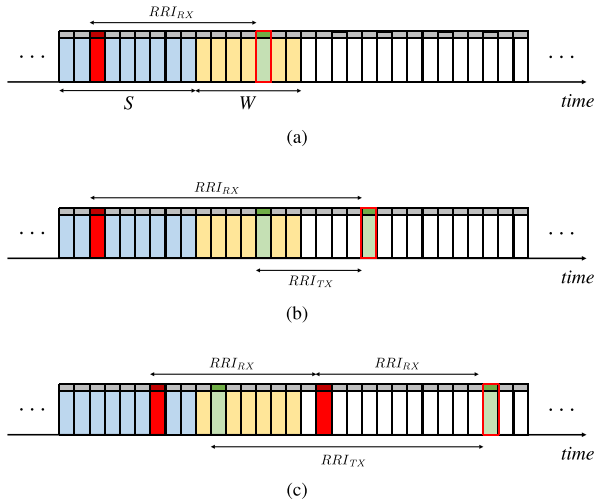


Fig. 4. SSPS detection of potential collisions for different RRI_{TX} and RRI_{RX} relations.

A. Modified SSPS Implementation

In our proposed solution, resources are drawn from list L_1, as opposed to list L_2. As a matter of fact, in the presence of aperiodic traffic, L_2 is not as meaningful as when vehicles periodically generate packets. It is not a case that NR-V2X will no longer use L_2 [33]. Moreover, our proposal sets the selection window $W = 100$ ms, the minimum CAM inter-arrival time, to avoid broadcasting out-of-sequence messages. To better understand the last statement, recall that CAM inter-arrival times can take on any value in the $[100, 200, \ldots, 1000]$ ms set; hence, if the selection window W is wider than 100 ms, the $(j+1)$-th CAM might be transmitted before the j-th, an event that has to be prevented.

An additional and meaningful modification concerns the list creation phase of the original SSPS algorithm. Given the CAM selection window W is 100 ms wide and that the RRI is dynamically determined via ML, observe that not all ongoing reservations fall within W and are spotted by the ego-vehicle. It follows that the original SSPS list creation mechanism loses effectiveness, increasing the risk of packet collisions. For this reason, we propose a new version of the SSPS process leading to the creation of list L_1, that we name *look-ahead* SSPS version. This SSPS reworking requires that the SCI also includes the current C_{resel} value, in addition to RRI. It is a minimal modification with respect to the choice of the legacy algorithm, necessitating very few bits. Yet, it remarkably extends the collision-avoidance capability of the original SSPS algorithm, as the numerical results will show.

As a matter of fact, if the ego-vehicle exploits the knowledge of the reselection counter of nearby vehicles, it can detect potential collisions for any possible combination of the resource reservation intervals used by itself and by its neighbors. To further clarify such enhanced capability of identifying collisions, Fig. 4(a)–(c) exemplify the SSPS operation in three different scenarios. In these figures, the candidate resources examined by the ego-vehicle are represented in green, the subchannels in use by the generic neighboring vehicle are indicated in red, and the selection window in yellow. Furthermore, RRI_{TX} indicates the resource reservation interval adopted by the ego-vehicle, whereas RRI_{RX} represents the reservation interval of the generic nearby vehicle, heard by the ego-vehicle in the SCI it receives. In Fig. 4(a), the candidate resource examined by the ego-vehicle is immediately excluded, as it coincides with the reservation placed by the nearby vehicle inside the selection window. The collision is avoided in the case exemplified in Fig. 4(b) too, as the ego-vehicle also verifies if any of its future reservations outside of the selection window coincides with the very next resource reserved by the nearby vehicle. Fig. 4(c) portrays an instance where the reservation heard by the ego-vehicle is not included within its selection window and RRI_{RX} is lower than RRI_{TX}. In this case, the original SSPS algorithm cannot detect the future collision, as the ego-vehicle is exclusively aware of the first reservation from the neighboring vehicle, after RRI_{RX} ms, and therefore does not exclude the examined resource. Here, the future collision would be spotted only if the ego-vehicle additionally knew the remaining number of ongoing reservations, i.e., the current C_{resel} value of the nearby vehicle, in addition to the periodicity of ongoing transmissions. Our *look-ahead* version of SSPS proposes to exploit the C_{resel} knowledge and performs this further check. Therefore, it creates a smaller, yet more reliable L_1 list, detecting and avoiding all the potential collisions exemplified in Fig. 4(a)-(c).

B. Machine Learning to Predict CAM Sequences

When the proposed strategy enters the resource selection and reservation phase, the first step that the ego-vehicle accomplishes is to forecast through ML the very next T_{GenCAM} value, as well as the length of the next sequence of identical T_{GenCAM} inter-arrival times. To do so, ML explores a large set of CAM traces to identify correlation patterns between some user-defined input features and the CAM traces. Then, such knowledge is leveraged to anticipate future CAM inter-arrival times [34]. In this work, the set of input features taken into account are:

Algorithm 1: Predictive Reservation.

Input : KNN input features
Output: RRI, C_{resel}

$i = 1$;
T_{GenCAM_i} = `Predict` (Input features, i);
RRI = T_{GenCAM_i};
$C_{resel} = 1$;
while $i \leq N$ **do**
  $i = i + 1$;
  T_{GenCAM_i} = `Predict` (Input features, i);
  if T_{GenCAM_i} = RRI **then**
    $C_{resel} = C_{resel} + 1$;
  else
    break;
  end
end
if $C_{resel} > 3$ **then**
  $C_{resel} = random[3, C_{resel}]$
end

- trajectory, current speed, and position of the ego-vehicle;
- current speed and position of the vehicle immediately preceding the ego-vehicle.

We choose to predict the next CAM inter-arrival time through the k-Nearest Neighbors (KNN) ML algorithm, an instance-based learning technique used for both regression and classification problems. KNN simply stores the training data without attempting to infer a general structure out of them. Moreover, KNN is inherently designed for multi-class problems and its classification consists in assigning the input features the most common label, i.e., the next predicted T_{GenCAM} value, among the k nearest neighbors. The second action of the ego-vehicle is to dynamically set the (RRI, C_{resel}) pair employed by the SSPS strategy in accordance to the ML forecast.

In greater detail, whenever SSPS triggers the resource selection and reservation phase, Algorithm 1 is invoked. The algorithm exploits KNN to predict the very next T_{GenCAM} value, T_{GenCAM_1}, and sets RRI equal to it, that is, $RRI = T_{GenCAM_1}$, while C_{resel} is initially set equal to 1. Then, as long as the next predicted inter-arrival time $T_{GenCAM_{i+1}}$ coincides with the previous T_{GenCAM_i}, the algorithm keeps incrementing the estimate of the reselection counter C_{resel}. Furthermore, when KNN outcome indicates that more than 3 consecutive CAM inter-arrival times will display the same value, the actual reselection counter value is randomly selected within the $[cntr_{min}, cntr_{max}]$ interval, where $cntr_{min} = 3$ and $cntr_{max}$ is the current C_{resel} estimate. This expedient avoids repeated packet collisions on resources reserved by different vehicles, a circumstance that might occur when vehicles generate CAMs with the same periodicity, e.g., in a congested intersection. The output of Algorithm 1 is finally used to set RRI and the reselection counter C_{resel} that indicates how many times the selected resource is reserved. Note that there is a maximum allowed value for C_{resel}, indicated by N. Moreover, observe that inequality $C_{resel} \geq 1$ reveals that at least one reservation has to be placed.

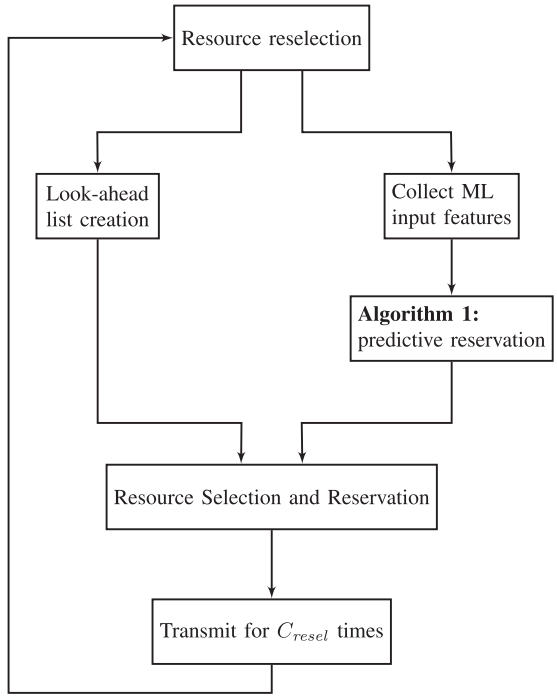


Fig. 5. Flowchart showing the proposed KNN-look ahead solution.

The overall flowchart of the proposed solution, termed KNN-look ahead from now onward, is reported in Fig. 5.

IV. Key Performance Indicators

When assessing the performance of a vehicular radio access solution, there are several Key Performance Indicators (KPIs) that are worth being considered.

One of the most widely adopted is the Packet Reception Ratio (PRR). Its definition relies on the notion of distance slice; the i-th distance slice is defined as the set of transmitter-receiver distances that fall within the $(a_i, b_i]$ range, $a_i = i \cdot 20$ m and $b_i = (i + 1) \cdot 20$ m. For the i-th slice, the PRR is defined as [30]:

$$PRR = \frac{\sum_{j=1}^{M} X_i^j}{\sum_{j=1}^{M} Y_i^j} \tag{2}$$

where X_i^j, indicates the number of vehicles within the i-th slice that successfully received the j-th packet, Y_i^j is the number of vehicles within the i-th slice when the j-th packet was transmitted and M denotes the number of packets generated during the simulation. The PRR is a reliability indicator, quantifying the probability that the message being broadcast by a vehicle can be heard at a given distance slice.

An additional standard-compliant reliability indicator is the Packet Inter-Reception (PIR). For a given transmitter-receiver pair, the PIR is defined as the time between two consecutive successful receptions of packets belonging to the same application flow, assuming the transmitter-receiver distance is within the $(0, D]$ range. Usually, its Cumulative Distribution Function (CDF) is provided, considering all transmitter-receiver pairs involved in the simulation.

Two additional KPIs are the Propagation Losses Ratio (PLR) and the Collision Losses Ratio (CLR). For the i-th slice, the PLR is defined as

$$PLR = \frac{PL_i}{PL_i + CL_i + SR_i} \tag{3}$$

and similarly, the CLR value is determined as

$$CLR = \frac{CL_i}{PL_i + CL_i + SR_i} \tag{4}$$

where:

- PL_i is the number of packets that were lost due to poor propagation conditions within the i-th slice, i.e., the packets that did not collide but experienced a Signal-to-Noise Ratio (SNR) not sufficient for the correct decoding of either the TB or its associated SCI;
- CL_i is the number of packets lost within the i-th slice because of a collision, i.e., the packets that were caught in a collision and whose reception failed because the Signal-to-Interference-plus-Noise Ratio (SINR) did not allow a correct decoding of either the TB or the SCI.
- SR_i is the number of successfully received packets within the i-th slice.

In the following, subscript i will be omitted, unless strictly necessary.

We observe that the PLR measures the fraction of radio resources that could not be successfully employed because of errors introduced by lousy propagation conditions. As such, it is influenced by the choices performed at physical layer, the channel model adopted in the geographical area that is being examined, and the CAM size. On the contrary, the CLR indicates to what extent harmful collisions could not be avoided, and it is therefore dictated by the radio resource assignment strategy.

A parameter also worth being monitored is the Channel Busy Ratio (CBR), which is defined as follows: given the n-th subframe, the CBR is the fraction of subchannels whose RSSI exceeds a given threshold over subframes $[n-100, n-1]$. The CBR is relevant to understand the load currently insisting on the radio channel. Additional metrics specific to LTE-V2X are [9]:

- the Latency Reselections Ratio (LRR), defined as the fraction of message transmissions that triggered a latency reselection over the total number of transmitted messages;
- the Unused Reservations Ratio (URR), defined as the fraction of unused reservations over the total number of resource reservations that were performed;
- the Counter Reselection Ratio (CRR), defined as the fraction of message transmissions that triggered a resource reselection due to the depletion of the reselection counter over the total number of transmitted messages.

V. Numerical Results

A. Physical and Medium Access Control Layer Configuration

As regards the Physical (PHY) and Medium Access Control (MAC) layers, this work relies on the custom ns-3 C-V2X module first introduced by the authors in [6] and finalized in [7]. The development of the simulator adheres to 3GPP Release 14 and Release 16 specifications and features all the elements that characterize Mode 4 communications. Vehicles have been configured to transmit their messages over the 10 MHz wide channel located in the 5.9 GHz ITS band, with 15 kHz subcarrier spacing. The 10 MHz channel is divided into 4 subchannels that consist of 12 RBs each, assuming adjacent transmission of the TB and of its associated SCI. The size of CAM messages, indicated by X, is fixed to either 190 or 470 bytes, which are the smallest and the largest statistically relevant sizes of CAMs [31]. Vehicles transmit their packets using QPSK modulation with 0.7 code rate, therefore mapping the 190 and 470 B-long packets into 1 and 2 subchannels, respectively. The transmission power is set to 23 dBm and the receiver sensitivity to -90.4 dBm. As in [9], the RSRP threshold is -140 dBm. The PHY layer impairments introduced by the radio channel are captured using the 5G-compliant error model presented by the authors in [7]. In greater detail, shadowing is modeled via a lognormally distributed random variable and small-scale fading is evaluated using two different Clustered Delay Lines (CDL) corresponding to the Line-Of-Sight (LOS) and Non-Line-Of-Sight (NLOS) scenarios, as detailed in [30]. The Packet Error Rate (PER) curve for the TB carrying the actual CAM and the associated SCI are reported in [7].

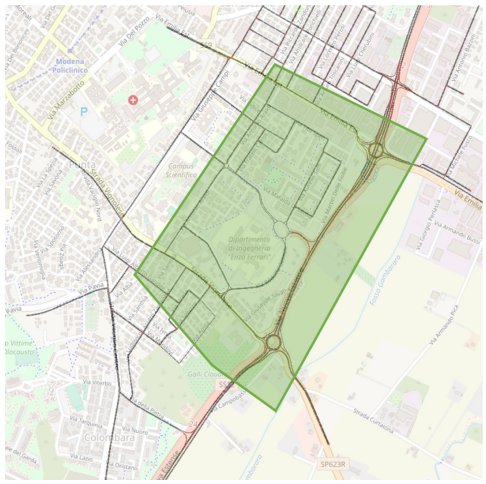


Fig. 6. The examined suburban road topology.

B. Outcomes

1) Suburban Setting: The first set of results refer to some outskirts of the Italian city of Modena, that we classify as a suburban setting example. Here, microscopic vehicular mobility has been simulated through SUMO [35]. The examined road topology is reported in Fig. 6 and it has been imported in SUMO using Open Street Map [36]. The area is approximately 2.5 km wide and 3 km long. Vehicles have been randomly generated at the area edges and have been assigned random trajectories that traverse the entire topology. The average vehicular density is 42 vehicles/km and the vehicles speed varies in the [50,100] km/h interval, depending on traffic conditions and on the vehicle *speedFactor*, a SUMO parameter that defines the maximum velocity of each vehicle as a function of the lane speed limits.

We have additionally developed a set of custom tools based on the SUMO Traffic Control Interface (TraCI) [37], to extract the elements that characterize the behavior of every vehicle, namely, heading, position, and speed; the periodicity for their collection was coincident with $T_{GenCAM_{\text{Min}}} = 100$ ms. They have allowed us to generate CAM messages in accordance with the rules set by the ETSI algorithm recalled in Sec. II-C. For every car, we also recorded the position and speed of the preceding vehicle, to complete the set of input features used by ML. As requested by Algorithm 1, these features fed a real-time implementation of the KNN algorithm, to predict the longest sequence of T_{GenCAM} values with the same periodicity. The number k of KNN nearest neighbors was taken equal to 3.

The dataset of CAM traces was collected from a total of 6800 vehicles during 20 minutes of SUMO simulation. The least represented T_{GenCAM} values in the dataset were oversampled using the Synthetic Minority Oversampling TEchnique (SMOTE) [38]. All input features were further normalized using min-max normalization, i.e., their range of values was re-scaled between 0 and 1. The training set and the test set were generated employing a $70-30$ split ratio.

First, Figs. 7(a)-(c) delve into the ability of the KNN algorithm to predict the upcoming sequence of CAM messages, reporting the confusion matrix for three different values of the T_{GenCAM} index i, $i = 1, 5$ and 10. The confusion matrix is a two-dimensional matrix indexed with the true and predicted class labels, and it is commonly used to visualize the performance of an algorithm. Fig. 7(a) reports the confusion matrix of KNN for $i = 1$, that is, when KNN forecasts next T_{GenCAM} value. Figs. 7(b) and (c) show the confusion matrix when KNN predicts the fifth ($i = 5$) and the tenth ($i = 10$) T_{GenCAM} value, respectively. These figures reveal that KNN is able to accurately forecast T_{GenCAM_1} value, and that the degradation in predicting T_{GenCAM_5} and $T_{GenCAM_{10}}$ values is modest.

The goodness of the prediction outcomes is further highlighted by Fig. 8, which reports the accuracy (black curve, circle markers) and the macro-F1 score (red curve, square markers) of KNN as a function of i, $i = 1, 2 \ldots, N$, with $N = 10$. For a given T_{GenCAM} index i, the accuracy measures the fraction of correctly predicted T_{GenCAM} values over the total number of samples, and the macro-F1 score is the mean of class-wise F1-scores, where the F1-score is a common metric that combines precision and recall [34]. As could have been expected, KNN performance deteriorates for larger values of i, as the CAM inter-arrival time to be forecast is increasingly distant in time. However, both indicators settle on fairly high levels, greater than 0.9, even for $i = 10$.

In the next set of figures, the focus shifts to the performance of the proposed KNN-look ahead solution. First, Fig. 9 shows the propagation losses ratio PLR as a function of the distance D between the transmitting and the receiving vehicle. Solid lines refer to $X = 470$, dashed lines to $X = 190$ bytes. Recall that the PLR measures the fraction of packets that were lost because of scarce propagation conditions over the total; as a matter of fact, these curves do not depend on the resource assignment strategy but are exclusively determined by the PHY layer choices and by the CAM size. So, when the radio propagation environment is more hostile (e.g., greater D values) and the CAM size is longer, the PLR increases. For these curves, as well as for the results shown next, a proper number of simulations has been executed to obtain sufficiently tight 95% confidence intervals. To avoid border effects, the results have been collected only from the central area of the setting; this corresponds to the green-shaded area in Fig. 6.

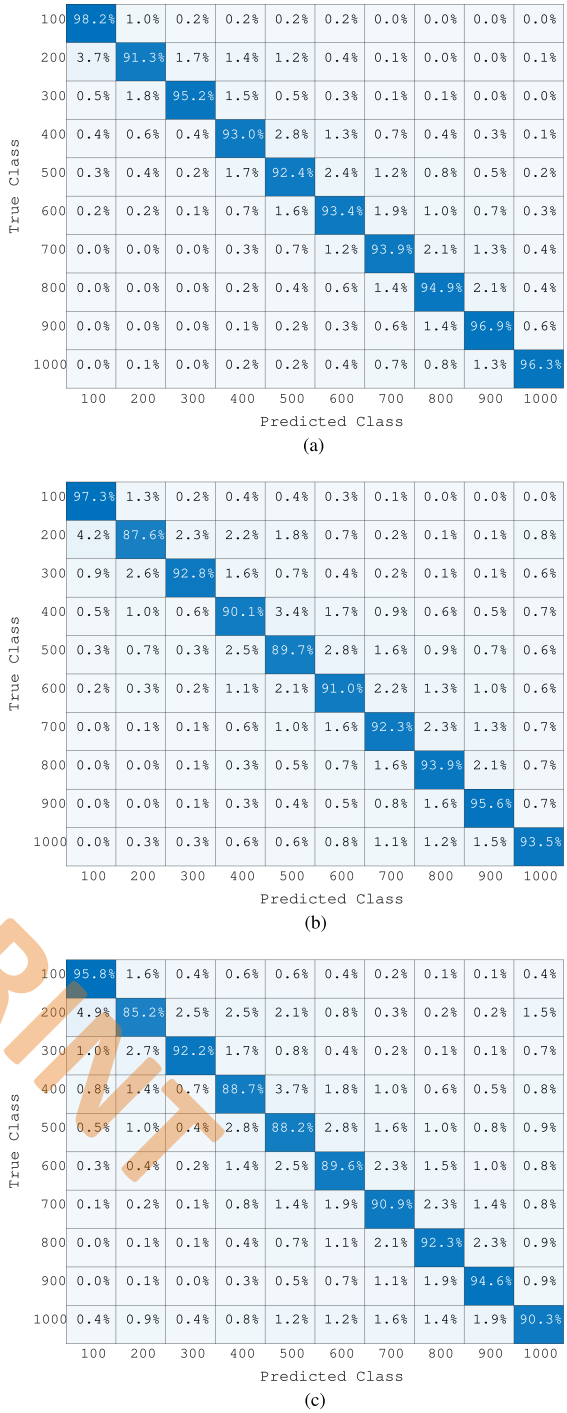


Fig. 7. Confusion matrix for the prediction of the T_{GenCAM_i} values. (a) T_{GenCAM_1}. (b) T_{GenCAM_5}. (c) $T_{GenCAM_{10}}$.

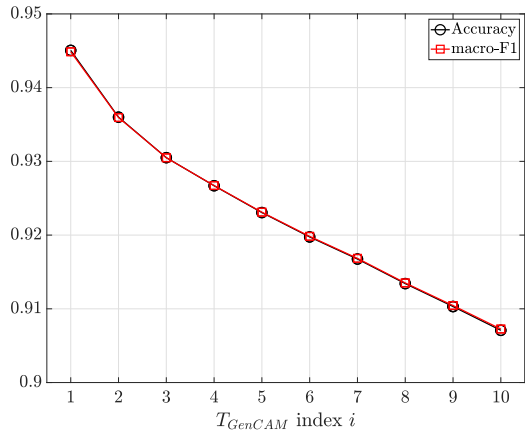


Fig. 8. Accuracy and macro-F1 score as a function of the T_{GenCAM} index.

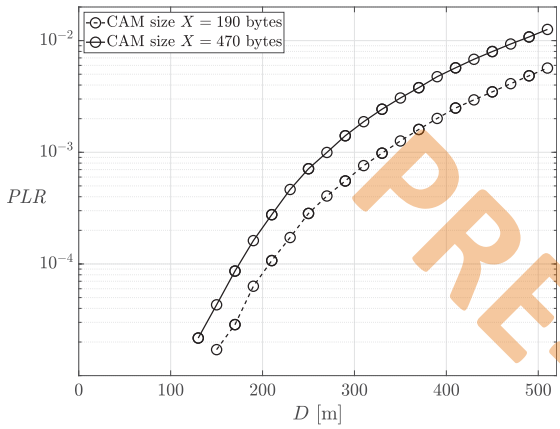
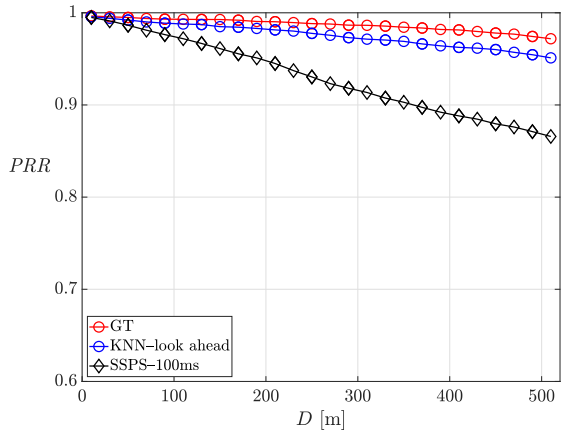
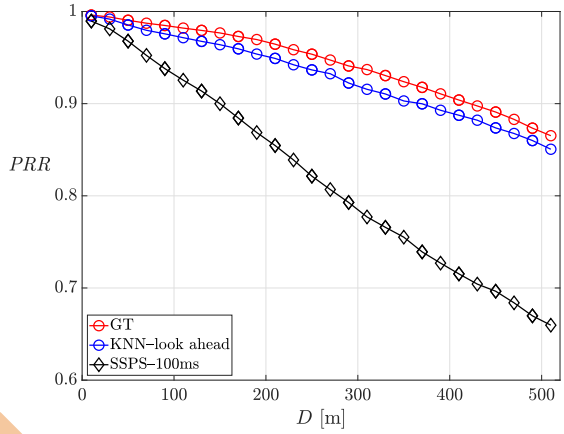


Fig. 9. PLR as a function of the Tx-Rx distance D.



(a)



(b)

Fig. 10. PRR as a function of the Tx-Rx distance D, suburban scenario. (a) CAM size $X = 190$ bytes. (b) CAM size $X = 470$ bytes.

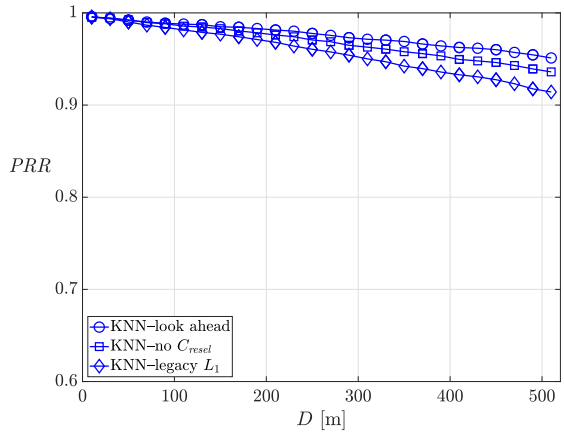
In the following figures, the proposed approach is confronted against the original SSPS algorithm with $RRI = T_{GenCAM_{\text{Min}}} = 100$ ms; the latter is a convenient setting, as it guarantees that CAMs gain access to the channel without generating any latency reselections. Adhering to the standard, our SSPS implementation randomly chooses the actual C_{resel} value in [5,15]. In accordance with [2], we set $P = 0$, that is, every time the counter expires, the vehicle has to select a new transmission resource with probability $1 - P = 1$.

To quantify how effective the KNN choice is within the ML domain, the Ground Truth (GT) benchmark is considered: this benchmark exploits the *a priori* knowledge of the CAM sequences to assign radio resources and place reservations that perfectly match the actual CAM sequences.

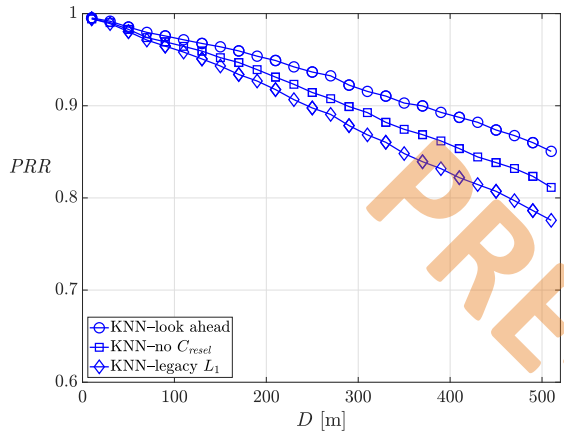
Fig. 10(a) and (b) report the PRR curves for the original SSPS mechanism with $RRI = 100$ ms (black curve, diamond markers), the curves obtained when the KNN-look ahead proposal is adopted (blue curve, circle markers), as well as the PRR values corresponding to the ideal GT benchmark (red line, circle markers). When the CAM size is 190 bytes, Fig. 10(a) indicates that our proposal guarantees an attractive improvement, and Fig. 10(b) reveals that the gain becomes significant when a larger size ($X = 470$ bytes) is considered, that is, when the load on the radio channel increases. Both figures also reveal that the KNN-look ahead approach attains a performance that is very close to the GT benchmark, i.e., to the ideal performance.

Fig. 11(a) and (b) quantify the effects that different mechanisms for the creation of the L_1 list have on the KNN resource selection process, hence on PRR. In both figures, the lowest PRR curve refers to the solution that relies on the original L_1 list (solid curve, diamond markers); the intermediate curve refers to the alternative where the L_1 creation additionally concentrates on perspective collisions that might occur outside of the selection window W without exploiting the knowledge of C_{resel} (dashed line, square markers); the third, upper curve, to the proposed KNN look-ahead solution (dot-dashed, circle markers). These two figures indicate that the design choices summoned in our proposal consistently lead to the best-performing approach.

Fig. 12(a) and (b) offer a further insight, displaying the CLR curves for the same choice of parameters as in Fig. 10(a) and (b). Coherently, the proposed KNN-look ahead strategy displays the lowest CLR values. These figures additionally reveal that the CLR values of the GT benchmark are not zero for all D values. The existence of a CLR "floor" is justified by observing that, even if every vehicle were able to perfectly forecast its

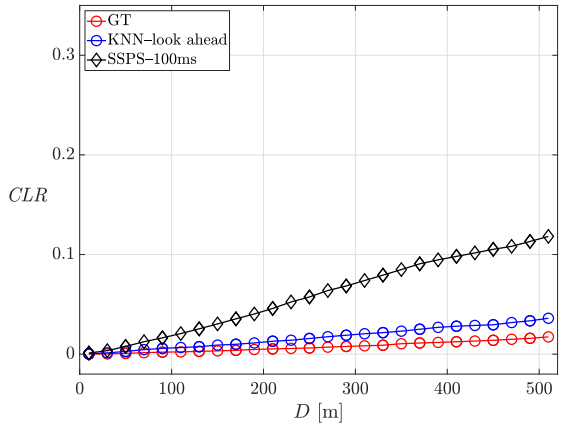


(a)

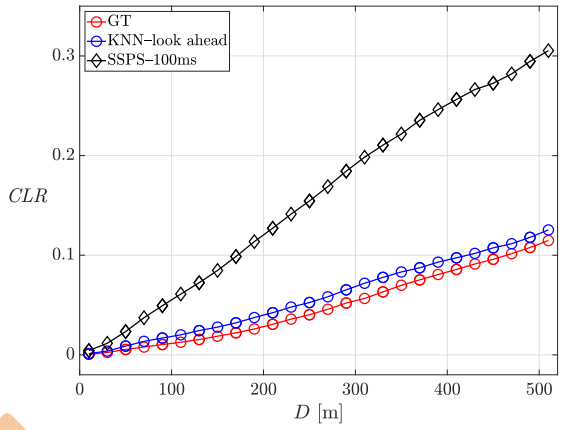


(b)

Fig. 11. PRR comparison for different L_1 lists, suburban scenario. (a) CAM size $X = 190$ bytes. (b) CAM size $X = 470$ bytes.



(a)



(b)

Fig. 12. CLR as a function of the Tx-Rx distance D, suburban scenario. (a) CAM size $X = 190$ bytes. (b) CAM size $X = 470$ bytes.

TABLE I
SUBURBAN SCENARIO: URR, LRR AND CRR VALUES

	URR	LRR	CRR
SSPS, RRI=100 ms	0.61	0.0	0.18
KNN-look ahead	0.12	0.10	0.22
Ground Truth	0.0	0.0	0.34

CAM transmission requirements over time and select resources accordingly, its selection could nevertheless coincide with the choice performed by other vehicles. This phenomenon is intrinsic to the distributed nature of the channel access mechanism and cannot be further reduced, unless a total redesign of the radio access technique is undertaken.

The effectiveness of the KNN-look ahead approach is further evidenced by the values provided in Table I, where the Latency Reselections Ratio LRR and the Unused Reservations Ratio URR of the proposed solution are compared against the values of these ratios for the original SSPS mechanism with $RRI =$ 100 ms and for the GT benchmark.

The Table shows that SSPS with $RRI = 100$ guarantees no latency reselections, as it respects the most stringent delay requirement, but the URR climbs to 0.61. At the other end of the scale, the GT benchmark perfectly eliminates unused reservations and reselections. The proposed solution lies in between, being able to significantly reduce the unused reservations ratio from 0.61 to 0.12. However, this improvement is achieved at the expense of a non-zero fraction of latency reselections. It is worth noting that, as long as they are not prevalent, latency reselections do not have the same negative impact on communication reliability as unused reservations. For the sake of completeness, the Counter Reselection Ratio CRR is also reported in the last column of the table: as expected, its value increases for the proposed solution and even more for the GT benchmark, as reselections become more frequent to track T_{GenCAM} variability.

Next, Fig. 13 reports the Probability Mass Function (PMF) of the T_{GenCAM} values observed in the suburban scenario. It is interesting to note that the PMF mainly concentrates around two values, 200 ms and 300 ms. As they are not integer multiples, the SSPS algorithm with RRI set equal to 100 ms is not very effective in detecting potential collisions. This explains why we observed fairly low PRR values for it.

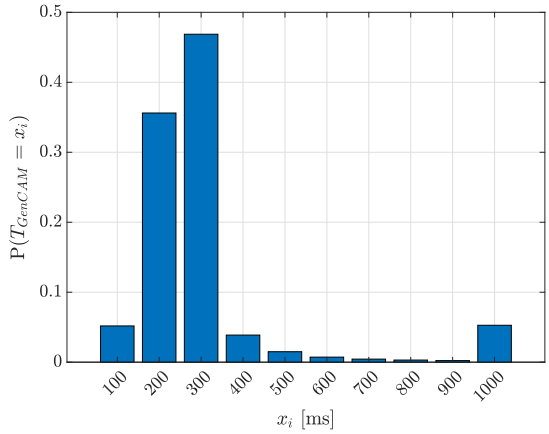


Fig. 13. T_{GenCAM} PMF, suburban scenario.

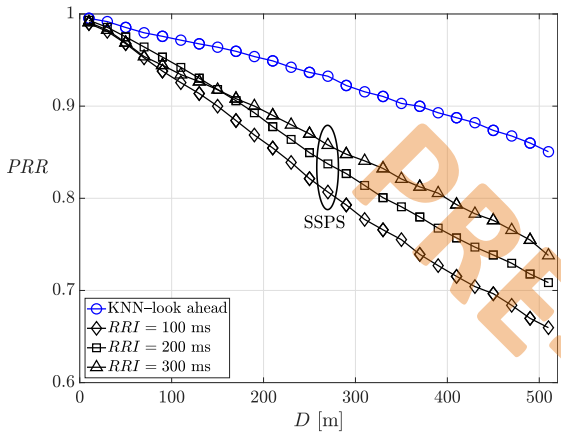


Fig. 14. PRR comparison against SSPS with different RRI settings, suburban scenario.

TABLE II
SUBURBAN SCENARIO: CBR VALUES

	$X = 190$	$X = 470$
SSPS, RRI=100 ms	0.24	0.4
KNN-look ahead	0.26	0.46
Ground Truth	0.27	0.47

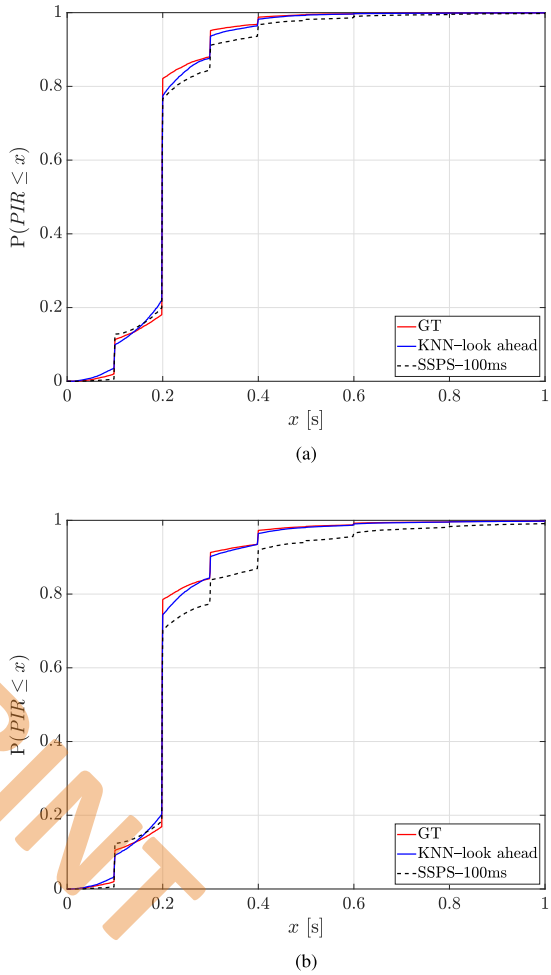


Fig. 15. PIR CDF, suburban scenario. (a) CAM size $X = 190$ bytes. (b) CAM size $X = 470$ bytes.

Given the *a posteriori* knowledge that the PMF reported in Fig. 13 provides, Fig. 14 shows the PRR attained by the legacy SSPS strategy when its reservation periodicity RRI is set so as to match the first or the second most frequently observed T_{GenCAM} value; that is, $RRI = 300$ ms (dashed black line) and $RRI = 200$ ms (dot-dashed black line). Such PRR values are further confronted against the baseline performance provided by SSPS with $RRI = 100$ ms (solid black line), and against the performance that the proposed KNN-based look ahead solution attains (blue line). Tuning the reservation periodicity improves the PRR of the legacy SSPS algorithm: unfortunately, the most proper RRI selection would be possible only if the T_{GenCAM} PMF were *a priori* known. Instead, KNN – or any alternative ML choice – does not necessitate such knowledge and yet, provides far higher PRR levels.

To further complete the assessment picture, Table II reports the CBR levels observed in the suburban scenario. The CBR of the generic vehicle has been computed every 0.2 seconds, the values have been time-averaged over the central portion of the simulation time and finally averaged over all vehicles. The RSSI threshold to discriminate between a busy and an idle channel is set to a value 0.5 dB greater than the receiver sensitivity level, therefore to -89.9 dBm. The CBR values reported in Table II reveal the magnitude of the channel load increase due to a larger packet size. Moreover, the CBR is not only useful for assessing the amount of traffic insisting on the communication channel. Given a specific setting, the CBR also reflects the effectiveness of the adopted access strategy: a more accurate scheduling mechanism maximizes the use of the available transmission resources, resulting in larger CBR values. This is the case encountered here, where the KNN-look ahead approach achieves higher CBR values than SSPS with 100 ms.

Fig. 15(a) and (b) show the PIR Cumulative Distribution Function (CDF) when $D = 520$ m, for the GT benchmark (red curve), for the proposed KNN-look ahead solution (blue curve)

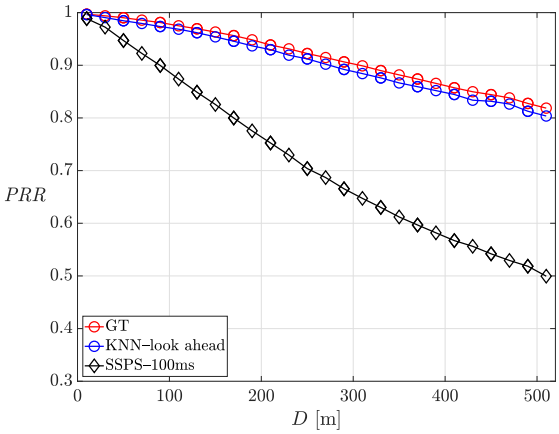


Fig. 16. PRR as a function of the Tx-Rx distance D, highway scenario, CAM size $X = 470$ bytes.

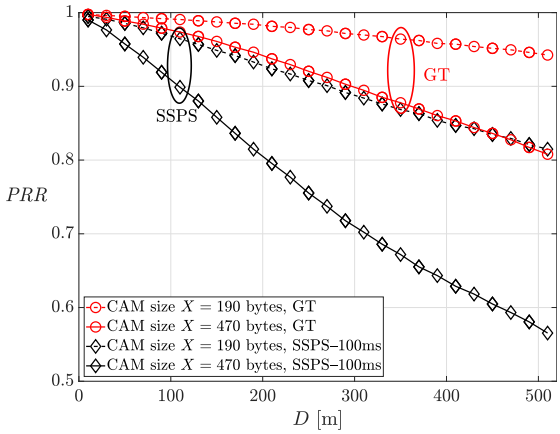


Fig. 17. PRR as a function of D, highway scenario, CAM trace Markov model.

and for the legacy SSPS (black curve). When $X = 470$ bytes, Fig. 15(b) indicates that the probability of observing PIR values lower than 200 ms and 300 ms, the two most frequent T_{GenCAM} values, is 0.74 and 0.90 for the KNN-look ahead solution. This is an improvement with respect to the values of the original SSPS implementation, equal to 0.70 and 0.84, respectively. Moreover, the discrete nature of the T_{GenCAM} values in the [100, 1000] ms range reflects in the step behavior of the PIR CDF.

2) Highway Setting: We also considered a second setting, termed *highway*, represented by a 5 km-long highway trunk, where six 4-meter wide lanes are deployed. Adhering to the specifications in [30], the vehicles' speed is 70 km/h and the vehicular density is 120 vehicles/km. For these numerical choices, Fig. 16 compares the PRR of the proposed KNN-look ahead solution to the PRR of the SSPS algorithm with $RRI = 100$ ms and to the GT upper bound, for the most demanding CAM size $X = 470$ bytes. The figure shows that the KNN-look ahead approach (blue line, circle markers) leads to a remarkable improvement in the PRR performance with respect to the original SSPS mechanism (black line, diamond markers), achieving PRR levels very close to the GT benchmark (red line, circle markers). It is however known that SUMO reveals some limits in the highway set-up: the constant speed and the nearly straight vehicular trajectories lead to an almost constant CAM inter-arrival time, $T_{GenCAM} = 300$ ms. The same behavior was observed when the vehicular speed varies within the [70,140] km/h range: here too, T_{GenCAM} is nearly constant and equal to 200 ms. We have overcome this simulation hurdle by employing one of the empirical models for the generation of CAM messages that were proposed in [39]. These models are derived from real-world traces of CAM traffic collected on a highway trunk [31], for different implementations of the ETSI algorithm by two Original Equipment Manufacturers (OEMs), Volkswagen and Renault. They consist of m-th order Markov sources that model: (i) CAM size and T_{GenCAM} variability; (ii) CAM size variability only; (iii) T_{GenCAM} variability only. We adopted the model that seizes CAM temporal variability, drawn from the Volkswagen CAM traces, setting $m = 5$. For this model, the average T_{GenCAM} value is 330 ms,

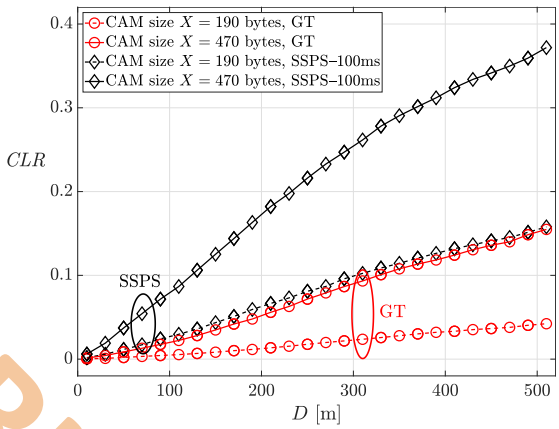


Fig. 18. CLR as a function of D, highway scenario, CAM trace Markov model.

close to the constant T_{GenCAM} value characterizing the SUMO implementation at 70 km/h constant speed. With the help of this analytical tool, we associated every vehicle with a specific CAM trace. Unfortunately, such empirical models have no notion of vehicle dynamics, so they do not provide the input features the KNN algorithm requires. Nonetheless, the reproduction of highway CAM traces allows to determine the GT performance, and therefore to assess the maximum improvement that ML achieves. In this respect, Fig. 17 concentrates on the PRR performance considering two different packet sizes, $X = 190$ bytes (dashed lines) and $X = 470$ bytes (solid lines). Adopting the same choice of colors and markers of Fig. 10(a)-(b), the black curves correspond to the original SSPS implementation with $RRI = 100$ ms, whereas the red curves refer to the GT benchmark, identifying the PRR upper bound. The significant improvement achieved by the GT solution with respect to the original SSPS mechanism is evident and becomes remarkable when $X = 470$ is considered. The original SSPS performance drops below 0.6 when $D \geq 450$ m, whereas the GT sets at $PRR = 0.85$. Fig. 18 is the counterpart of Fig. 17 on the (CLR, D) plane. This figure further highlights the enhanced

TABLE III
HIGHWAY SCENARIO: CBR VALUES

	$X = 190$	$X = 470$
SSPS, RRI=100 ms	0.34	0.49
Ground Truth	0.39	0.61

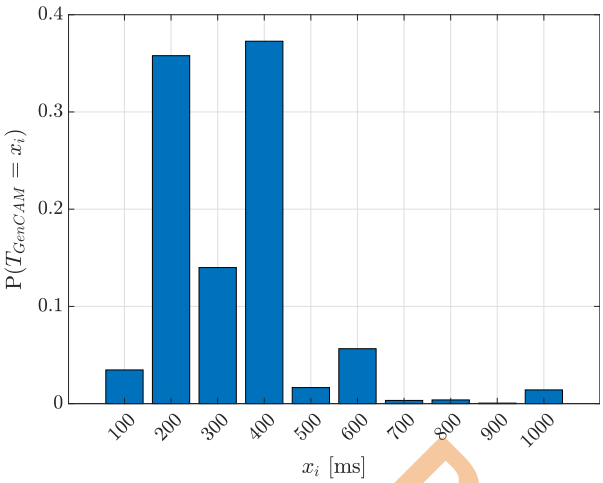


Fig. 19. T_{GenCAM} PMF, highway scenario.

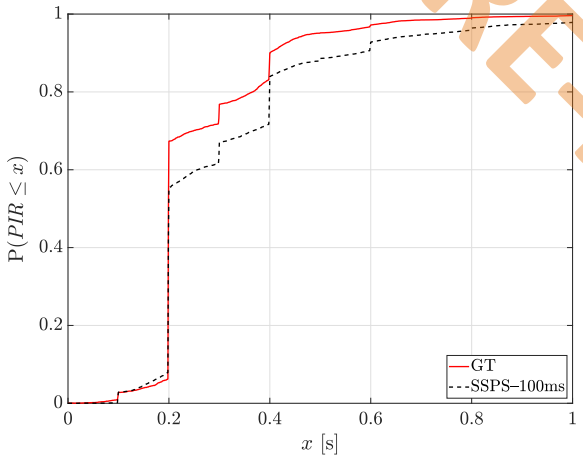


Fig. 20. PIR CDF, CAM size $X = 470$ bytes, highway scenario.

collision-avoidance capability of the ML-based strategy with respect to the standard-compliant solution, that increases for increasing packet sizes. Its superiority is substantiated by the CBR values reported in Table III. The first column of the Table refers to $X = 190$ bytes: the CBR increases from 0.34 to 0.39 when moving from SSPS with $RRI = 100$ ms to the GT benchmark. Likewise, the CBR rises from 0.49 to 0.61 in the second column that refers to $X = 470$ bytes, once more highlighting the significant impact of T_{GenCAM} predictions on the selection of collision-free resources.

The PMF of the T_{GenCAM} samples generated in the highway scenario is shown in Fig. 19. As in the suburban setting, the PMF mainly condenses around two values, 200 ms and 400 ms. Finally, Fig. 20 reports the PIR CDF when $D = 520$ m and $X = 470$ bytes. Here too, the GT benchmark provides an upper bound to the PIR achievable performance, highlighting the maximum amount of improvement with respect to the original SSPS reservation strategy. It is worth observing that the implementation of the proposed approach on an actual vehicle is feasible, as the input features that KNN employs can be easily retrieved. The ego-vehicle position can be obtained via the Global Navigation Satellite System (GNSS), its speed can be measured by in-vehicle sensors, and the use of onboard lidars and radars can offer accurate estimates of the position and speed of the preceding vehicle. The vehicle trajectory prediction is a widely investigated topic in the industrial and the academic world, and algorithms like the one reported in [40] can estimate the ego-vehicle trajectory in an accurate manner.

As regards the introduction of ML, we showed that a simple technique such as KNN leads to a remarkable performance improvement with respect to the original LTE-V2X Mode 4. The selection of a more sophisticated ML algorithm, although possible, would only lead to incremental improvements and to unnecessary complexity.

VI. CONCLUSION

This paper has proposed an ML-based solution termed KNN-look ahead to distribute aperiodic CAMs to vehicles. The approach relies on a limited set of features, which each vehicle employs to forecast its next CAM generation times. The ML outcome is combined with a careful selection and reservation of the available radio resources. The simulation results have demonstrated that the proposed strategy achieves an excellent performance, consistently outmatching the legacy LTE-V2X Mode 4 approach.

REFERENCES

[1] A. Bazzi, G. Cecchini, A. Zanella, and B. M. Masini, "Study of the impact of PHY and MAC parameters in 3GPP C-V2V mode 4," *IEEE Access*, vol. 6, pp. 71685–71698, 2018.
[2] M. Gonzalez-Martín, M. Sepulcre, R. Molina-Masegosa, and J. Gozalvez, "Analytical models of the performance of C-V2X mode 4 vehicular communications," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1155–1166, Feb. 2019.
[3] A. Bazzi, C. Campolo, A. Molinaro, A. O. Berthet, B. M. Masini, and A. Zanella, "On wireless blind spots in the C-V2X sidelink," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9239–9243, Aug. 2020.
[4] F. Abbas, P. Fan, and Z. Khan, "A novel low-latency V2V resource allocation scheme based on cellular V2X communications," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 6, pp. 2185–2197, Jun. 2019.
[5] V. V. Chetlur and H. S. Dhillon, "Coverage and rate analysis of downlink cellular vehicle-to-everything (C-V2X) communication," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1738–1753, Mar. 2020.
[6] L. Gibellini and M. L. Merani, "Out-of-coverage multi-hop road safety message distribution via LTE-A cellular V2V (C-V2V)," in *Proc. IEEE 88th Veh. Technol. Conf.*, Chicago, IL, USA, 2018, pp. 1–6.
[7] L. Lusvarghi and M. L. Merani, "On the coexistence of aperiodic and periodic traffic in cellular vehicle-to-everything," *IEEE Access*, vol. 8, pp. 207076–207088, 2020.
[8] S. Bartoletti, B. M. Masini, V. Martinez, I. Sarris, and A. Bazzi, "Impact of the generation interval on the performance of sidelink C-V2X autonomous mode," *IEEE Access*, vol. 4, pp. 35121–35135, 2021.
[9] R. Molina-Masegosa, J. Gozalvez, and M. Sepulcre, "Comparison of IEEE 802.11p and LTE-V2X: An evaluation with periodic and aperiodic messages of constant and variable size," *IEEE Access*, vol. 8, pp. 121526–121548, 2020.
[10] L. Lusvarghi and M. L. Merani, "MoReV2X - a. new radio vehicular communication module for ns-3," in *Proc. IEEE 94th Veh. Technol. Conf.*, 2021, pp. 1–7.

[11] C. Campolo, V. Todisco, A. Molinaro, A. Berthet, S. Bartoletti, and A. Bazzi, "Improving resource allocation for beyond 5 G V2X sidelink connectivity," in *Proc. 55th Asilomar Conf. Signals, Syst., Comput.*, 2021, pp. 55–60.
[12] F. Hussain, R. Hussain, A. Anpalagan, and A. Benslimane, "A new block-based reinforcement learning approach for distributed resource allocation in clustered IoT networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 3, pp. 2891–2904, Mar. 2020.
[13] H. Xiang, M. Peng, Y. Sun, and S. Yan, "Mode selection and resource allocation in sliced fog radio access networks: A reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 69, no. 4, pp. 4271–4284, Apr. 2020.
[14] H. Ye, Y. Li, and B.-H. F. Juang, "Deep reinforcement learning based resource allocation for V2V communications," *IEEE Trans. Veh. Technol.*, vol. 68, no. 4, pp. 3163–3173, Apr. 2019.
[15] H. Ye, L. Liang, G. Ye Li, J. Kim, L. Lu, and M. Wu, "Machine learning for vehicular networks: Recent advances and application examples," *IEEE Veh. Technol. Mag.*, vol. 13, no. 2, pp. 94–101, Jun. 2018.
[16] F. A. Ghaleb *et al.*, "Deep kalman neuro fuzzy-based adaptive broadcasting scheme for vehicular Ad Hoc network: A context-aware approach," *IEEE Access*, vol. 8, pp. 217744–217761, Dec. 2020.
[17] J. Aznar-Poveda, E. Egea-López, and A. J. García-Sanchez, "Cooperative awareness message dissemination in EN 302 637-2: An adaptation for winding roads," in *Proc. IEEE 91st Veh. Technol. Conf.*, 2020, pp. 1–5.
[18] O. Amador, I. Soto, M. Uruena, and M. Calderon, "GoT: Decreasing DCC queueing for CAM messages," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2974–2978, Dec. 2020.
[19] European Telecommunications Standards Institute (ETSI), "Intelligent transport systems (ITS); vehicular communications; basic set of applications; specification of cooperative awareness basic service," EN 302 637–2 V1.4.1, Apr. 2019.
[20] M. Sepulcre, J. Gozalvez, G. Thandavarayan, B. Coll-Perales, J. Schindler, and M. Rondinone, "On the potential of V2X message compression for vehicular networks," *IEEE Access*, vol. 8, pp. 214254–214268, Dec. 2020.
[21] A. Bazzi, A. Zanella, and B. M. Masini, "Optimizing the resource allocation of periodic messages with different sizes in LTE-V2V," *IEEE Access*, vol. 1, pp. 43820–43830, Mar. 2019.
[22] A. Bazzi, C. Campolo, B. M. Masini, A. Molinaro, and A. Zanella, "Enhancing cooperative driving in IEEE 802.11 vehicular networks through full-duplex radio," *IEEE Trans. Wireless Commun.*, vol. 17, No 4, pp. 2402–2416, Apr. 2018.
[23] N. Lyamin, A. Vinel, M. Jonsson, and B. Bellalta, "Cooperative awareness in VANETs: On ETSI EN 302 637-2 performance," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 17–28, Jan. 2018.
[24] G. Giambene, M. S. Rahman, and A. Vinel, "Analysis of V2V sidelink communications for platoon applications," in *Proc. IEEE Int. Conf. Commun.*, 2020, pp. 1–6.
[25] S. Roger *et al.*, "Low-latency Layer-2-Based multicast scheme for localized V2X communications," *IEEE Trans. Int. Transp. Syst.*, vol. 20, no. 8, pp. 2962–2975, Aug. 2019.
[26] B. Coll-Perales, J. Gozalvez, and M. Gruteser, "Sub-6GHz assisted MAC for millimeter wave vehicular communications," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 125–131, Mar. 2019.
[27] Z. Li, L. Xiang, X. Ge, G. Mao, and H. -C. Chao, "Latency and reliability of mmWave multi-hop V2V communications under relay selections," *IEEE Trans. Veh. Technol.*, vol. 69, no. 9, pp. 9807–9821, Sep. 2020.
[28] 3GPP, "TS 38.321 NR; medium access control (mac) protocol specification (v16.4.0, Release 16)," *Tech. Spec.*, Mar. 2021.
[29] *3GPP*, "TS 38.331 NR; radio resource control (RRC) protocol specification (v16.3.1, Release 16)," *Tech. Spec.*, Jan. 2021.
[30] *3GPP*, "TR 37.885; Study on evaluation methodology of new vehicle-to-everything (V2X) use cases for LTE and NR (v15.3.0, Release 15)," *Tech. Spec.*, Jun. 2019.
[31] CAR 2 CAR Communication Consortium, "survey on ITS-G5 CAM statistics," TR2052, V1.0.1, Dec. 2018.
[32] L. Zucchini, "Vehicular safety communications," B. S. thesis, Dept. Eng, "Enzo Ferrari," University of Modena and Reggio Emilia, Modena, Italy, Jul. 2020.
[33] M. H. C. Garcia *et al.*, "A tutorial on 5 G NR V2X communications," *IEEE Commun. Surv. Tut.*, vol. 23, no. 3, pp. 1972–2026, Jul.–Sep. 2021.
[34] P. Harrington, *Machine Learning in Action*, Shelter Island, NY, USA: Manning Publications, Apr. 2012.
[35] P. A. Lopez *et al.*, "Microscopic traffic simulation using SUMO," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Maui, HI, USA, 2018, pp. 2575–2582.
[36] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org," Accessed: Feb. 1, 2020. [Online]. Available: https://www.openstreetmap.org/
[37] A. Wegener, M. Piorkowski, M. Raya, H. Hellbruck, S. Fischer, and J. Hubaux, "TraCI: An interface for coupling road traffic and network simulators," in *Proc. 11th Commun. Netw. Siml. Symp.*, 2008, pp. 155–163.
[38] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jan. 2002.
[39] R. Molina-Masegosa, M. Sepulcre, J. Gozalvez, F. Berens, and V. Martinez, "Empirical models for the realistic generation of cooperative awareness messages in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 69, no. 5, pp. 5713–5717, May 2020.
[40] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao, "Vehicle trajectory prediction based on motion model and maneuver recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Tokyo, 2013, pp. 4363–4369.

Luca Lusvarghi (Graduate Student Member, IEEE) received the master's degree (*summa cum laude*) in electronics engineering in July 2019 from the University of Modena and Reggio Emilia, Modena, Italy where he is currently working toward the Ph.D. degree with the International ICT Doctorate School of the Department of Engineering, Enzo Ferrari. His research interests include 5G and 6G technologies, particularly on cellular vehicle-to-everything communications, and non-orthogonal multiple access techniques.

Maria Luisa Merani (Senior Member, IEEE) is currently an Associate Professor with the Department of Engineering, University of Modena and Reggio Emilia, Modena, Italy. In October 2009 Cambridge University Press published her textbook *Hands on Networking: From Theory to Practice* and in the same year she was a Coauthor of the Springer book *Handbook of P2P Networking*. Her research focuses on area of wireless networking. She was the Technical Program Co-Chair for the 2nd IEEE International Symposium on Wireless Communication Systems 2005 (ISWCS'05) and IEEE Globecom, 2007 and 2009 editions. In 2010, she was the General Chair of the fifth edition IEEE International Symposium on Wireless Pervasive Computing. Dr. Merani was the Editor of IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS.