Learning-Based Road Link Quality Estimation for Intelligent Alert-Message Dissemination

Raoua Chakroun, and Thierry Villemur

***Abstract*—Accurately assessing the quality of road links is essential for effectively sharing critical messages in dynamic vehicular network environments. Unfortunately, existing literature lacks models to estimate the quality of links between infrastructure and vehicles due to the complexity and variability of vehicular communication networks, including channel variations and interference patterns. To address this gap, we propose a prediction model based on supervised machine learning to estimate the Packet Reception Rate (PRR) on the road. Our model updates communication zones dynamically to align with traffic conditions. We train and evaluate our model using a dataset generated from a realistic mobility scenario simulated using NETSIM and SUMO. Our performance tests indicate promising results in terms of prediction accuracy. This work is an important step toward establishing an efficient and reliable scheme for disseminating alert messages, considering the fluctuations in traffic conditions and vehicular mobility.**

***Index Terms*—Vehicular communications, Quality of Service assessment, Machine Learning.**

I. Introduction

INTELLIGENT Transport Systems (ITS) [1] have been developed to provide communication capabilities to all types of vehicles and transport infrastructures. Specifically, Cooperative ITS (C-ITS) [2] aims to enable seamless communication between vehicles (Vehicle-to-Vehicle: V2V), vehicles and infrastructure (Vehicle-to-Infrastructure: V2I), and overall, vehicles and all their surroundings (Vehicles-to-everything: V2X). These systems strive to offer a safer (reducing the number of accidents on the road), more efficient (reducing travel time and pollution), and more comfortable (passenger entertainment: multimedia, infotainment, etc.) mode of transportation.

The integration of communication capabilities with the sensing and perception capabilities of vehicle sensors paves the way for the development of numerous Cooperative Intelligent Transport Systems (C-ITS) services and use cases. A wide variety of C-ITS services have been proposed in the literature, including dissemination of traffic i nformation and alert messages between vehicles, traffic guidance, and more. These services are designed to support the main objectives of C-ITS.

As a result, the upcoming generation of vehicles will be equipped with interfaces that allow them to connect and communicate with other elements of intelligent transportation systems, such as pedestrians, other vehicles, and infrastructure like Base Stations (BS), Roadside Units (RSU), and cloud. These communication capabilities make it possible for the next-generation connected vehicles to have centralized network control, and vehicles-to-infrastructure network connectivity that can support traffic control.

To make the most of the various access networks' capacities and efficiently manage the network resources, a hybrid multi-access vehicular network based on a software-defined network (SDN) is suggested in multiple works. This approach brings flexibility to the control and management of the network, which is essential for providing communication services that meet the requirements of C-ITS services. Therefore, this work considers SDN-based vehicular networks.

Furthermore, an SDN controller has the advantage of maintaining visibility of the vehicular network's current state. This is particularly important given that vehicle mobility and traffic conditions can cause connection issues. By identifying areas with poor or no network connection (gray zones), the SDN controller can proactively warn network entities to take action in order to provide the level of performance required for the various C-ITS services.

Nevertheless, to uphold the demands of the required C-ITS services that adhere to stringent Quality of Service (QoS) criteria, even in the face of swiftly changing traffic conditions, the network should possess the capability to foresee potential alterations associated with traffic evolution.

This paper presents an extension to our prior research [3], where we formulated an I2V link quality estimation method tailored for road applications. Employing Machine Learning algorithms, this technique gauges packet reception rates (PRR) within designated $40 \times 40m^2$ squares. This enables the assessment of reception quality in each square and the identification of areas where reception quality may be compromised (gray zones). Utilizing the predicted gray zone positions as input in the Q-learning-based rebroadcast zone placement algorithm as described in [4], enables the creation of an efficient alert message system, detailed in [4]. The approach relies on a centralized perspective constructed by the network controller within its coverage and control area, incorporating information

Manuscript received November 20, 2023; revised December 11, 2023. Date of publication January 31, 2024. Date of current version January 31, 2024.

R. Chakroun was with the Department of Services and Architectures for Advanced Networks, LAAS-CNRS, Toulouse, 31400 France (e-mail: raouachakroun1993@gmail.com).

T. Villemur is with the Department of Services and Architectures for Advanced Networks, LAAS-CNRS, Toulouse University, Toulouse, 31400 France (e-mail: thierry.villemur@laas.fr).

This paper was presented in part at the International conference on Software, Telecommunications and Computer Networks (SoftCOM) 2023.

Digital Object Identifier (DOI): 10.24138/jcomss-2023-0171

on road traffic, vehicle characteristics, and potentially their road trip details. We have devised a supervised learning model that integrates traffic information and "Hello" default exchange messages between vehicles and infrastructure, thus minimizing transmission overhead.

The remainder of the paper is structured as follows: Section II introduces the general motivations for the link quality estimation problem. Subsequently, Section III provides a synthesis of existing works in the scientific literature. Section IV offers a general overview of the proposed link quality estimation model, while Section V delves into the details of the proposed models. The subsequent section describes the dataset used. Moving on to Section VII, the focus is on the experimental part, initially presenting the metrics considered and then analyzing the evaluation results for the proposed model. Section VIII details the application framework of the link estimation model. Emerging discussion points are presented in section IX. Finally, the concluding section wraps up this work.

II. Link Quality Estimation: Motivation

Within vehicular networks, the radio signal propagation channel undergoes significant temporal and spatial variations, impacting the link quality on the road. To ensure robust and enduring performance in such networks, it becomes imperative to efficiently estimate the link quality on the road to guarantee certain dissemination techniques. This enables the adaptation of link parameters and the selection of relays, facilitating the choice of an alternative or more reliable route or area for data retransmission. In essence, the better the link quality is, the higher the successful reception rate and the more reliable the communication is. However, challenging factors like channel fluctuations, send/receive issues, and intricate interference patterns directly influence link quality, potentially resulting in unreliable connections.

While it remains challenging to integrate these dynamic factors into an analytical model, rendering such models less adaptable to realistic networks given the inherently unpredictable and dynamic nature of the design environment. Conversely, accurate link quality prediction holds the potential for substantial performance enhancements, including augmented network throughput through minimized packet loss, prolonged network lifespan by restricting retransmissions, mitigated topology outages, enhanced reliability, and more. Ultimately, fluctuations in link quality wield a considerable impact on the overall network connectivity. Hence, efficient estimation or prediction of link quality can identify the optimal link from a pool of candidates for data transmission [4].

In this paper, the link quality is defined by the Packet Reception Rate (PRR) of vehicles via the road infrastructure. This approach is employed to identify gray zones on the road that exhibit poor connectivity with the infrastructure.

III. Related Work

Improving vehicular networks for reliable communications is a hard challenge [5]–[7]. One of the first steps to solve it is to assess the wireless link quality. Unfortunately, the statistical channel models studied during the last decade do not predict wireless link quality with high accuracy, due to the highly dynamic nature of the vehicular environment [5].

Most link quality estimation techniques in vehicular networks are proposed to estimate reactively the quality of the V2V links [8]–[11]. These works select the next hop/broadcaster between the sender and its neighboring nodes. Node selection is based on the signal's strength or packet reception rates over a given link. It is used to characterize the quality of its forward link. However, such mechanisms have assumed a fixed communication range among the nodes, which is not realistic [12], [13]. In addition, the links' qualities of the broadcaster can considerably vary for a given node in time for several reasons, such as varying surrounding node densities and fading channel effects [14]. Our work completes the previous ones by allowing proactive estimations that can be periodically improved by reactive adjustments.

There is no consensus for defining link quality and using a standard unit of measure for the metric [15], [16]. In our work, we consider that link quality can be characterized by throughput or reliability parameters.

Machine Learning (ML) techniques [6] have recently emerged for predicting link quality in wireless environments. They supersede former techniques based on predefined models [17]–[19].

The ML-based algorithms developed in [20], [21] are used to predict Vehicle-to-Vehicle (V2V) path loss. They prove that the application of such models offers better performance than traditional analytical models based on log distance path loss.

To estimate V2V link reception quality, the Benrhaim and al. [22] method relies on periodic beacons exchanged between vehicles. They propose a Bayesian network-based scheme at different locations in the zone covered by the transmission range of the sender for the estimation. This is the only work that estimates the road links' quality. Estimation results show good accuracy. However, the sample of parameters considered in all the simulations remains small and limited.

Most of the works proposed in the literature to estimate the vehicular communication links' quality assume simplifications of vehicle mobility. All these works concern V2V links, where only one is interested in estimating V2V link quality on the road. Approaches based on machine learning techniques generally present the best performances for both problems. Our work is the first one that focuses on I2V wireless quality links on the road. The proposed method for estimating road links' quality excludes any vehicle mobility or communication range assumption [4].

IV. Link Quality Estimation Model Overview

This work proposes an intelligent link quality estimation algorithm to predict gray zones in vehicular communication networks. We have developed a prediction model based on machine learning techniques to estimate the Packet Reception Ratio (PRR) of Roadside Units (RSUs) messages for each zone in a predefined region, depending on the current traffic conditions. The considered region is divided into small squares of $40 \times 40m^2$ as shown in Fig. 2 and described in [23]. The squares are used to identify gray zones where the PRR is

less than 90% (a predefined threshold) [24]. The controller identifies these gray zones. Fig. 1 illustrates the proposed approach's network architecture and its key elements. Each region has an SDN controller, as detailed in [25], to manage the RSUs providing V2I wireless connectivity in the region. We assume all vehicles are furnished with a GPS module capable of transmitting information, including their position ($P(x, y)$) and the packet response to the "Hello" message received from the RSU during association and beacon message exchanges. This data is regularly gathered by each RSU, either stored in the cloud or directly shared with the network controller. These collected features are then utilized as input for the model M_{PRR} executed in the SDN controller, facilitating the estimation of PRR. Additionally, we assume that each RSU entity reports information regarding newly associated vehicles

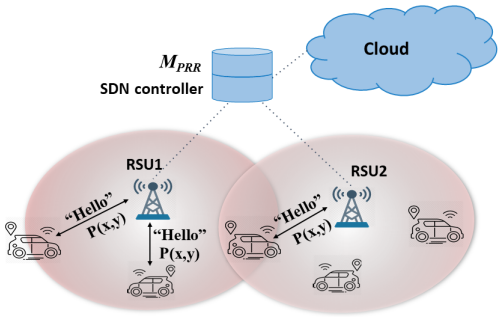

Fig. 1. Key elements of the proposed approach.

Fig. 2. Considered geographic map [4].

Therefore, This model helps to identify gray zone positions, which represent areas with weak communication links. The positions of these zones are input to a reinforcement learning-based technique for Vehicle-to-Vehicle (V2V) rebroadcast zone placement, which optimally adjusts the rebroadcast zone placement according to the evolution of gray zones caused by changing traffic conditions, as proposed in [23]. Finally, the combination of both techniques enables intelligent dissemination of Alert Messages (AMs) using the Location Alert Message Dissemination (LAMD) procedure proposed in [26].

The complete dissemination process is summarized in Fig. 3. First, the SDN controller identifies gray zones on a geographic map by using the "Road link quality estimation" algorithm (1). Next, it uses the location of these zones to place or update rebroadcast zones using the "Q-learning rebroadcast zone placement" technique (2). These zones are then shared with vehicles proactively during handover. Finally, when vehicles receive an alert message, they utilize the "LAMD procedure" (3) with rebroadcast zones as input.

V. Supervised Learning-based Road I2V Link Quality Estimation

A. Supervised Learning

A Machine Learning (ML) system differs from a regular computer program. It doesn't follow a set of instructions to perform tasks based on inputs. Instead, it learns the best actions to take, such as making decisions or predictions, usually by analyzing data or past experiences. These systems are broadly classified into three categories: supervised learning, unsupervised learning, and reinforcement learning. [27].

Our study focuses on supervised learning, which involves using a labeled dataset for training. In this process, a dataset D is defined as $D(x_1, y_1), ...(x_n, y_n)$. The objective is to train a model M that can establish the best correlation between the predictors, which are the input variables X, and the labels, which are the output variable y. The model should be able to predict the corresponding output $\widehat{y_n} = M(X_n)$ with high accuracy for new input data X_n whose outputs are unknown. We categorize supervised learning into two types: regression, where the predicted value is a continuous real number denoted as $y \subset \mathbb{R}$, and classification, wherein y is a member of a finite set $C = c_1, c_2, ..., c_n$ referred to as classes.

Moreover, we explore the so-called ensemble learning techniques as outlined in [28]. These techniques involve training multiple models, whether of the same or different techniques and aggregating their predictions. This approach stands out as one of the most popular and potent methods in supervised algorithms, facilitating the creation of a generalized model and mitigating overfitting. Specifically, we adopt the *Random Forest* technique [29], which involves training a collection of decision tree models.

This choice is motivated by several advantages. Firstly, it enables the handling of problems involving multiple classes, distinguishing itself from other techniques that concentrate solely on binary classification. This capability is referred to as multivariate classification. Additionally, it incorporates a feature for selecting the most influential predictors, known as "feature importance." In the trees constructed, the most critical features are prone to appear near the root, while others are typically situated closer to the leaves.

As outlined earlier, our focal issue revolves around estimating PRR. This predicament has been formulated as a regression problem, with the variable of interest being the PRR within each small zone (square) on the map under the coverage of the

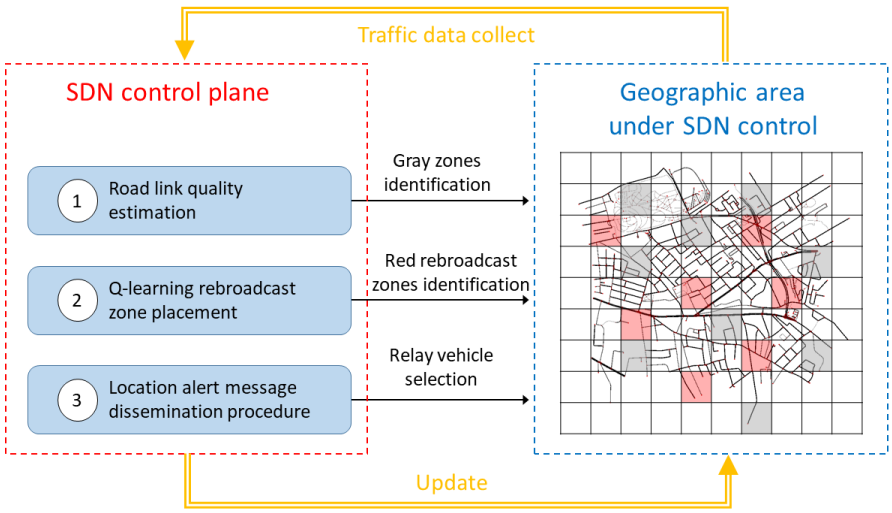

Fig. 3. Intelligent Link Quality Estimation for an Efficient Alert Message Dissemination Scheme

SDN controller. Subsequent sections delve into the variables designated for training (features) to construct our model. These features are crafted in alignment with specific objectives, namely, i) features necessitating minimal information from the vehicles, and ii) features independent of the used network technology. Following this, we elaborate on the techniques employed to train and fine-tune the model parameters.

B. Proposed Framework

Vehicular networks exhibit propagation channel characteristics that markedly distinguish them from other wireless systems. The physical environment within vehicular channels is prone to random variations induced by diverse factors, encompassing mobility patterns, swift fluctuations in traffic density, path loss effects, and environmental influences. The swift temporal variability and non-stationary nature of these channels warrant the development of a distinct framework for predicting link quality in vehicular networks. Our objective in this undertaking is to construct a ML model adept at predicting PRR on the road with minimal error. ML models are well-suited for addressing classification and pattern recognition challenges, making them an ideal choice for this endeavor.

Adhering to the conventional machine learning workflow outlined in [27] and harnessing the capabilities of SDN [30], we introduce the SDN-enabled machine learning framework for PRR prediction, as depicted in Fig. 4. The machine learning workflow encompasses six stages: Problem Formulation, Data Collection, Data Analysis, Model Construction, Model Validation, and the final stage involves Deployment and Inference.

The framework rests on two foundational pillars: SDN and the potency of well-suited ML algorithms. These algorithms are designed to glean insights from a historical dataset, utilizing the acquired knowledge to offer accurate estimations for new observations.

Based on Fig. 4, the workflow of the framework is as follows. Initially, the offline construction of the prediction model is done by training and fine-tuning historical data. This historical data may consist of a vast number of samples, where each sample represents a combination of values for different features and the associated target value. Since we are working in a supervised learning configuration, the collected data include the transmission power of RSUs (T), vehicle position (P) (to identify the zone identifier (Z_i) and calculate the distance between the vehicle and the RSU), the distance between the vehicle and RSU (D) (to identify the average distance between the concerned zone and the RSU), and packet status ($Status$) to calculate the target by zone (whether or not the vehicle received the ”Hello” messages from RSU). The features' description, collection, and processing will be elaborated in the following section.

The designed model is deployed (1) as the Inference Agent for PRR inference. Its purpose is to predict gray zones in the region and update rebroadcast zones for disseminating alert messages based on traffic needs and conditions. This proactive approach assumes that the controller has historical data on traffic conditions (e.g., traffic density at peak hours) and updates these zones accordingly on a set schedule (e.g., every two hours or three times a day).

After processing, each zone has an online input (2) composed of $(Z_i, T, P_{loss}, V_d, D)$, where V_d is the vehicle density in the zone Z_i, calculated according to the real-time number of vehicles in the zone/square, and P_{loss} is the packet loss in the zone calculated regarding real-time parameters and traffic conditions. When the controller launches the updates of gray zones for the geographic map under its coverage, this input is obtained. An inference of the PRR in each zone (small square) is made based on this input, so gray zones are identified (3). The Q-learning placement Agent uses this information as input to update rebroadcast zones, as described in [23].

Upon completion of the process, the resulting output is systematically gathered, and the historical dataset is efficiently updated with the newly acquired data (4). Maintaining an up-to-date database is crucial, enabling the consideration of

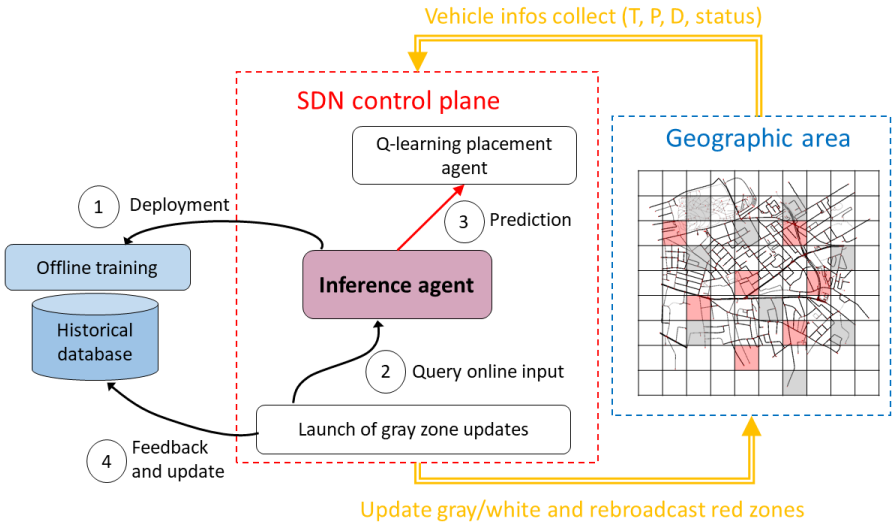

Fig. 4. Packet Reception Ratio (PRR) inference framework.

new dynamics stemming from changes in traffic patterns. The combination of historical data gathering and the real-time update of the dataset with newly collected information constitutes the foundation of our framework. Additionally, there is the possibility of enriching the historical data from cloud-based sources.

VI. Learning-based Modeling

In conformance with standard ML workflow in [27], this section begins with problem formulation. In the context of PRR inference, the target metric is a continuous variable, rendering its prediction a regression problem.

A. Dataset and Analysis

By following supervised learning principles, our model needs to undergo a training phase utilizing a dataset. To assess the performance of the proposed model in a practical VANET urban environment setting, we need to have a dataset that fulfills the following requirements:

- Network Coverage: Contemplate cells exhibiting diverse communication ranges achieved through the adjustment of RSUs' transmission power, encompassing both small and large coverage areas. Furthermore, it is imperative to ascertain the geographical positions of these entities for distance calculation.
- Road Traffic: Acquire data from vehicles, including their locations, traversing the majority of roads, both main and secondary, within the coverage area of a designated RSU.
- Size of the dataset: Gathering data over an extended period involves manipulating vehicle density, transmission powers, RSU positions, and path loss coefficients. This approach enables an exploration of the temporal variations in the measured metrics.

Within vehicular networks, the scientific community has conducted numerous data collection campaigns to generate datasets for studying network performance. However, only a few datasets are publicly accessible. Furthermore, these datasets often lack a focus on urban mobility and may not encompass all the parameters essential for our model.

As there is no existing dataset within the scientific community that entirely meets our requirements, we opted to create our own dataset. However, technical constraints and time limitations hinder the execution of a comprehensive data collection campaign. Consequently, we turn to an approach centered on simulation tools, mobility emulation, and VANET networks.

The dataset utilized in this study was generated through integrating the microscopic road traffic simulator SUMO with the event-based network simulator NETSIM, as detailed in [31]. NETSIM emulates the Dedicated Short Range Communication (DSRC) protocol stack, including signal strength, handover, and connectivity, while SUMO manages vehicle mobility. This combined framework delivers a realistic simulation of DSRC connectivity for vehicles. Our simulation setup consists of two primary components: the first one involves the implementation of the DSRC network, and the second one involves the simulation of vehicle mobility. For the DSRC network, we selected an area of $2\times2km^2$ in a European-like city, specifically Toulouse, France, extracted from Owen Street Maps (OSM). This area was chosen for its significance, being situated in the city center with high traffic densities (Urban environment), substantial buildings affecting signal quality on the road, and irregular road structures. The random trip application within the SUMO package was employed to automatically generate trips for vehicles within the specified map area. We assume all vehicles are equipped with DSRC wireless communication modules. Concerning the DSRC network, we strategically place four RSUs in the selected region, as illustrated in Fig. 5. The success of wireless transmission hinges on various factors, including distance, transmitter power, path loss, fading, and receiver sensitivity. Additionally, the transmission coverage of an RSU can vary significantly based on the environment

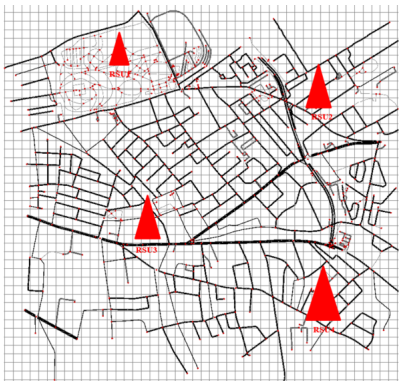

Fig. 5. Map with geographic locations of RSUs [4].

(Highway, Urban, Obstructions, Line of Sight), spanning from 100 to 700 meters for the same transmission power, as demonstrated in [32], [33].

For our network simulations, we conducted 196 simulations, where each RSU broadcasted a control message every 100 ms for 500 seconds (27 hours), varying transmission power, RSU positions, vehicle densities, and path loss coefficients in each run. Table I provides a listing of different manipulated parameter values. Path loss refers to the reduction in the power density of an electromagnetic wave during propagation and can result from various effects such as free reflection, aperture-medium coupling loss, and absorption. The path loss exponent fluctuates between 2 and 5, depending on the coherence bandwidth and Doppler shift of the surrounding environment. In our simulator-based approach, we directly utilize this parameter from the simulator, although in reality, it can be calculated based on several parameters.

An SDN controller, equipped with a global view and knowledge, can calculate this parameter in real-time using information shared by the RSUs and the cloud, considering factors like the distance between RSUs and zones, the environment (urban or other), weather conditions, etc. [34]. Following each packet transmission, we record the vehicle position (to identify the zone identifier), the vehicles that received the message, and the packet status (success or error). The simulations yield a dataset comprising 52,007 observations. Fig. 6 illustrates the number of samples generated for each PRR range. The dataset generated and collected after each simulation is outlined in Table II.

TABLE I
SIMULATION CONFIGURATION PARAMETERS

Parameter	Value
Vehicle Density	from 5 to 500 vehicles/zone(square)/hour
Path loss coefficient	from 2 to 5
Distance	from 0 to 2 km
Transmission power	from 10 to 50 dbm

We depict the geographical map as a 50×50 grid matrix of $40 \times 40m^2$ (see Fig. 5). Accordingly, our PRR prediction focuses on small road zones/squares, considering only those squares containing road areas. This approach is preferred over predicting the PRR for an entire road since the link quality

TABLE II
GENERATED AND COLLECTED DATA SET

Name	Feature Description
Packet Id	Sent packet identifier
RSU ID	Sender RSU identifier
Vehicle ID	Receiver vehicle identifier
P(x,y)	Vehicle position
Transmission power	Transmission power of the sender RSU
Path loss exponent	Power density of an electromagnetic wave
Packet status	”received” or ”not-received”

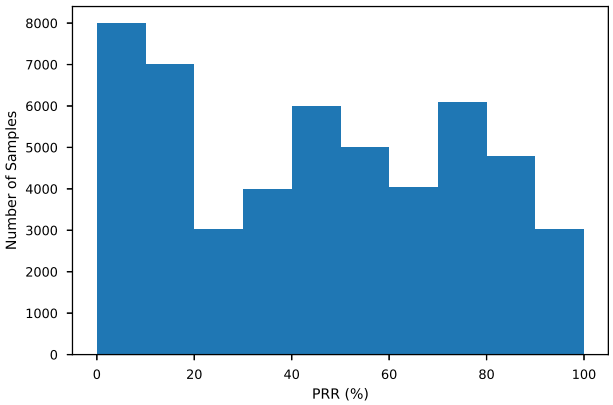

Fig. 6. Number of samples generated for each PRR range [4]

may vary from the beginning to the end and the center of the road. Predicting the PRR by zone (a specific segment of the road) enhances precision and accuracy.

To tailor our dataset for our model, we undertake specific data processing steps, enabling us to predict the PRR in each zone as detailed in Table III. Initially, we utilize the vehicle's position to identify the relevant zone. For each simulation, we calculate the vehicle density per zone per hour, the average distance between the zone and the sender, and the PRR (comparing the number of packets received by vehicles in the zone to the number sent by RSUs to those vehicles).

Our model operates as a supervised learning model, utilizing a labeled dataset in which the training data encompasses the desired solutions. The value of the predicted variable, PRR, is derived from the collected 'Packet Status' values (Received: 1, Not Received: 0). Therefore, the PRR represents the average reception packet status collected per vehicle within each small zone on the road. As the variable to be predicted is a continuous real number, we employ a regression technique.

TABLE III
PARAMETERS AND NOTATIONS [3]

Name	Feature Description
Z_i	Zone identifier
$V_d(i)$	Vehicle density per hour in a zone
$D(i)$	Average distance between the sender and the zone
T	Transmission power of the sender RSU
$P_{loss}(i)$	Path loss exponent
$\widehat{PRR}(Z_i)$	Packet Reception Rate in the concerned zone

B. Model Training

In the preceding section, we introduced the diverse learning variables employed by our model M_{PRR}. M_{PRR} primarily incorporates features such as vehicle density, distance, transmission power, and path loss exponent (Equation 1 [4]). These features are crafted based on two primary criteria: i) minimizing the need for extensive information from the vehicles, and ii) independence from the used network technology.

$$\widehat{PRR(Z_i)} = M_{PRR}(V_d(i), D(i), T, P_{loss}(i)) \quad (1)$$

Algorithm 1 encapsulates the input information and features used by the M_{PRR} model for predicting the packet reception ratio in each road zone within the purview of the SDN controller. It is pertinent to mention that the zone ID list and distance from the RSU are inherently stored in the controller database due to their fixed positions.

Algorithm 1: Road PRR Estimation

Input:
$Z_i, i \in 0, .., N$ List of road zone ID
$P_{loss}(i)$ per $zone$: path loss per zone
$T(i)$: Transmission power of the RSU that covers the zone
$V_d(i)$: Vehicle density in the concerned zone
Output:
$\widehat{PRR}(Z_i)$: the Packet reception ratio by zone

for $i = 0$ *to* $i = N$ **do**
  $D(i) = \sqrt{((x_{RSU} - x_i)^2 - (y_{RSU} - y_i)^2)}$ /* distance between the zone i of coordinates (x_i, y_i) and the RSU that covers it of coordinates (x_{RSU}, y_{RSU}) */
  $\widehat{PRR}(Z_i) = M_{PRR}\ (D(i),\ P_{loss}(i),\ T(i)\ ,\ V_d(i)\ )$

Using a dataset comprising various attributes and labeled with the desired results (PRR), offline training of the models is conducted to establish the optimal relationship between features and labels. As previously mentioned, our model is built upon the Random Forest (R.F) technique, which merges multiple Decision Tree models (D.T). In a decision tree, data are organized into a tree structure, and the model leverages this structure to make predictions for new data. Subsequently, based on the input data, predictions are generated by crossing the tree from its root to a terminal node, commonly referred to as a leaf. These terminal nodes encapsulate the values of the predictions, specifically the packet reception ratio (PRR) for the model M_{PRR}. In our scenario, which is a regression case, the average of the observation values within a node is employed as the prediction [4].

The tree construction forms the foundation for predictions and constitutes the primary objective of the training phase. Throughout this phase, the model establishes nodes, determines the number of observations (samples) per node, and formulates rules for each node. For every node, the model seeks the pair (k,t_k), where k denotes the attribute to be considered (such as distance, density, transmitter power, etc.), and t_k represents the value of this attribute. This pair is selected to minimize the Mean Square Error (MSE) by utilizing the cost functions outlined in equation 2.

$$j(k, t_k) = \frac{m_{left}}{m} MSE_{left} + \frac{m_{right}}{m} MSE_{right} \quad (2)$$

Where,

$$\begin{cases} MSE_{node} = \sum_{i \in node} (\hat{y}_{node} - y(i))^2 \\ \hat{y}_{node} = \frac{1}{m_{node}} \sum_{i \in node} y^{(i)} \end{cases} \quad (3)$$

$p^2_{i,k}$: denotes the percentage of observations belonging to class k among all training observations within the i^{th} node.,

$m_{left/right}$: represents the count of instances in the left/right subset.

Our Random Forest model undergoes training with a collection of decision tree models. Initially, each tree is trained using a randomly selected subset of the dataset (with the size specified as a parameter). Additionally, in the training process of each tree, the attribute k is randomly chosen when making a node split. Ultimately, the predictions generated by each tree are aggregated to yield an overall prediction. In the case of regression, the final prediction is determined by taking the average of the values estimated by all the trees.

The model establishes a set of parameters, referred to as hyper-parameters, to guide the construction of trees. Among these parameters, we highlight:

- *n_estimators*: It determines the number of trees trained by the model. Typically, a higher number enhances prediction accuracy, but it also increases processing costs, particularly in large datasets.
- *max_depth*: It establishes the maximum depth of the tree, representing the number of levels from the root to the last terminal leaf node. A greater depth enables the representation of more information from the dataset, but it also raises the risk of the model overfitting the data.
- *min_samples_leaf*: It sets the minimum number of observations needed for a terminal node, signifying the minimum size of leaves. A smaller size allows the model to capture noise in the data.
- *max_features*: It represents the number of elements in the list K from which the model selects the pair (k,t_k). It enables the control of model randomness. The maximum value is the number of features in the dataset (default value). A higher value introduces less randomness into the model, but it incurs an additional processing cost during training (for the selection of the pair (k,t_k)), particularly with a very large number of trees.

In general, these parameters control the growth of the trees. Opting for deeper trees with small leaves tends to offer a more comprehensive representation of the data, but there is a risk of the model overfitting the data. On the other hand, constraining the growth of these trees results in a more generalized model, yet small trees with large leaves may lead to underfitting, where the model excels only on training data but performs poorly on testing data, ultimately compromising model performance. Hence, selecting these parameters is a crucial step in model training to achieve a well-performing, generalized model.

The only way to adjust these parameters is to train multiple models with different values and select the combination that yields the highest accuracy.
To streamline the process of exploring multiple possibilities

and minimize processing overhead, we initially employ a randomized search using the *RandomizedSearchCV* technique. This helps identify initial values for each parameter from a broad range of input values. Subsequently, we utilize a grid search through *GridSearchCV* to explore all potential combinations within smaller ranges, bounding the values previously identified. This allows us to pinpoint the best combination with the highest accuracy. Both techniques rely on cross-validation, using the training set for both training and validation purposes. Specifically, the training subset is randomly divided into k distinct blocks. In each iteration (performed k times), the model reserves a different block for evaluation and undergoes training using the remaining parts (k-1 blocks). This method ensures that no dedicated portion of the dataset is exclusively used for validation (validation set) [4].

VII. Performance Evaluation

The objective is to assess the proficiency of the model (M_{PRR}) in predicting the packet reception ratio within each square to pinpoint the gray zones in the region. We scrutinize the outcomes through data visualization of the examined scenario, aiming to discern the strengths and limitations of the proposed model.

We allocate 75% of the dataset for model training and reserve the remaining 25% for testing, a commonly employed ratio. Subsequently, for each input i in the test set X, we calculate the corresponding output $\hat{y}_i = M(X_i)$ using the trained model M. We then compare this output with the actual value y_i. This process enables us to compute the prediction accuracy for each model, utilizing the performance metrics detailed below.

A. Performance Metrics

To assess the accuracy of the proposed ML performance models, we consider two evaluation metrics:

- The prediction **score** R^2 (Eq. 4): It signifies the portion of the variance in the dependent variable that can be anticipated from the independent variables. It reflects the proportion of accurately predicted samples. A highly accurate regression model would have a relatively high R squared, approaching 100 when expressed as a percentage. We will express the score in percentage terms.
- **Normalized Mean Absolute Error** ($NMAE$) (Eq. 5): It denotes the average of the absolute differences between the estimated and observed values of PRR. Our objective is to minimize the $NMAE$ as much as possible.

$$R^2(y,\hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)}{\sum_{i=1}^{n}(y_i - \bar{y})} \text{ , with } \bar{y} = \frac{\sum_{i=1}^{n} y_i}{n} \quad (4)$$

$$NMAE(y,\hat{y}) = \frac{\frac{\sum_{i=1}^{n}|y_i - \bar{y}|}{n}}{\bar{y}} \text{ , with } \bar{y} = \frac{\sum_{i=1}^{n} y_i}{n} \quad (5)$$

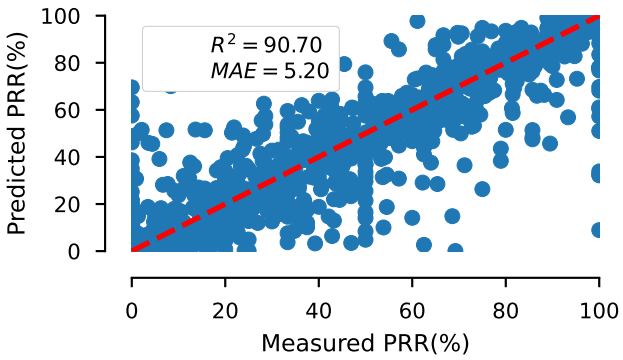

Fig. 7. Predicted vs. real observations (K = 10000) [4].

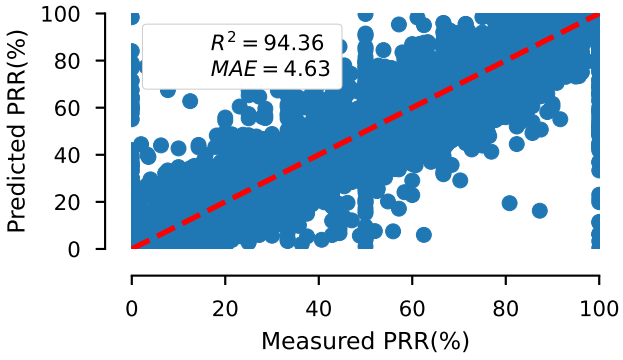

Fig. 8. Predicted vs. real observations (K = 52007 samples) [4].

B. Results

For a subset with 10000 samples, wherein 7500 samples are utilized for training the random forest model, we achieve a score of $R^2 = 90.70\%$ and $NMAE = 5.20\%$ on the remaining 2500 sets. Utilizing all the datasets (52,007 samples) generated through NETSIM simulations yields scores around 95%, with an associated $NMAE$ of approximately 5%. These outcomes underscore the capability of the ML-performance approach to deliver accurate predictions. Fig. 7 displays the actual observed test points alongside their corresponding predictions made by the random forest model for $K = 10000$ samples. However, it is evident that when the subset encompasses the entire initial dataset ($K = 52007$ samples), the prediction accuracy improves, with a score nearly reaching 0.95, as depicted in Fig. 8.

The performance of the proposed model aligns well with the requirements of proactive network control based on the estimation of I2V link quality on the road. This capability enables the periodic definition and updating of gray zones within the region, facilitating the adjustment of rebroadcast zones for efficient dissemination of alert messages.

Now, let's consider the scenario of a network control function computing routing paths. Some approaches in the literature consider link quality as a criterion for selecting routing paths. Errors in estimation (PRR larger or smaller than the estimated duration) can lead to inefficient link selection, thereby impacting the effectiveness of proactive routing. Many link quality-

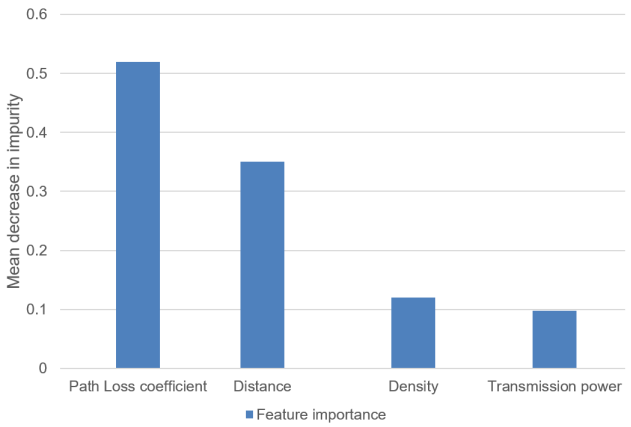

Fig. 9. Feature importance using Mean Decrease in Impurity (MDI).

based routing approaches aim to eliminate links with low PRR, necessitating accurate estimation of poor-quality links. The proposed model adequately addresses this requirement.

Fig. 9 illustrates the importance of features, using the Mean Decrease in Impurity (MDI) to calculate each feature's significance. MDI is computed as the sum over the number of splits (across all trees) that include the feature, proportionally to the number of samples it splits. The results indicate that distance and packet loss coefficient (which encompasses geographic area, urban/non-urban classification, size and presence of buildings in the area, weather, etc., influencing communication channels) are the most critical features. The absence or neglect of these features would considerably diminish the model's accuracy on the test set. For instance, removing path loss from the features could lead to a decrease in score accuracy of up to 55%.

VIII. Application

The intelligent PRR inference framework (refer to Fig. 4) can handle mixed PRR inference alongside Q-learning rebroadcast zone placement. The separation of the network's control plane from its data plane by SDN introduces flexibility in network management, enabling seamless map updates for gray and rebroadcast zones. Furthermore, it simplifies the integration of machine learning techniques into the management plane.

In this section, we applied the two proposed techniques, namely M_{PRR} and Q-learning placement, to periodically update the rebroadcast zones based on changes in traffic conditions. This, in turn, reflects the variations in road link quality and the emergence of gray zones.

A. Q-learning Rebroadcast Zones Placement

In this technique, the set of gray zones/squares is taken as input. This method assumes that the controller possesses both a prior and updated view of the link quality in each road segment, a capability facilitated by our proposed M_{PRR} model. The algorithm's initial step involves the random selection of N feasible rebroadcast points. Subsequently, in each iteration, t, the position of each rebroadcast zone/square $((x_i, y_i), \forall i \in N)$ undergoes an exploratory move with a probability of ϵ, or it selects the best-known action to date (highest Q_value) with a probability of $1 - \epsilon$. Throughout the learning phase, the algorithm explores different states within a fixed simulation/iteration run to identify the optimal policy that maximizes the expected action-value function (Q_value) and, consequently, the total coverage of gray zones. A gray zone is deemed covered by a rebroadcast zone if the distance between the center of this zone (square) and the center of the rebroadcast zone is less than a predefined threshold. For more details on the Q_learning algorithm, the reader is referred to [4], [23].

B. Application Scenario

We employ the same scenario as presented in [4], involving 8 RSUs (refer to Fig. VIII-B). In this scenario, the identification of gray zones in the considered map is achieved through simulation in the following manner. Initially, RSUs are configured to broadcast alert messages every 100 ms for a duration of 500 seconds. Subsequently, the PRR is computed for each square, and squares with a PRR below 90% are categorized as gray zones (see Fig. VIII-B). The same RSUs' positions, transmission power, path loss model, and traffic density are initially utilized to measure the PRR for defining the gray zones.

Firstly, we estimate the position of the gray zones using M_PRR, followed by the application of the Q-learning placement algorithm. Consequently, we observe the same number of rebroadcast zones (15 rebroadcast zones, resulting by real simulation shown in Fig. VIII-B), with a slight variation (1 to 2 squares of shift) in the position of six green-circled zones (refer to Fig. VIII-B). This discrepancy has practically negligible impact on the performance of the LAMD procedure (step (3) in Fig. 3), remaining within a few meters difference due to the 4.63% error prediction rate of our model. This underscores the success of our model in estimating the PRR, making it convenient for the SDN controller to update rebroadcast zone positions based on selected parameters (path loss coefficient, traffic density, and transmission power) whenever necessary [4].

IX. Discussion

Estimating link quality on the road lays the foundation for intelligent and effective network control. In our proposed approach, we primarily utilized the identification of the road packet reception ratio as the main learning variable for our model. Specifically, timing this identification during the update of the rebroadcast zones ensures the efficient and dependable dissemination of alert messages, allowing these zones to adapt to changing traffic and mobility conditions [4]. Importantly, this process does not add any extra load on the network. Performance tests yielded outstanding results in the majority of cases.

The model undergoes offline training utilizing data collected under diverse traffic conditions, aiming to approximate real-world mobility and varying traffic scenarios across different hours each day. However, trends captured by the models

Initial RSU placement | Q-learning without M_{PRR} | Q-learning with M_{PRR}

Fig. 10. Rebroadcast Zones placement.

during real-world training may experience further variations with the introduction of new installations and reconstructions in the area, such as new buildings, facades, and parking areas. These changes can impact parameters like the communication medium and path loss. Consequently, it becomes imperative for the controller to receive updates reflecting these changes, ensuring the timely recalibration of road zones and efficient recalculation of path loss for each zone.

Conversely, the service provider ITS can adjust network parameters to optimize its network, such as modifying cell coverage or adding/deleting a cell. Such modifications can impact the performance of the model. Re-training becomes a consideration if the prediction error surpasses a predefined threshold (established based on the service using the predictions) when incorporating new data and accounting for these altered conditions [4].

X. Conclusion

This paper introduces a machine-learning-based I2V quality link estimation technique on the road, specifically, the Packet Reception Ratio (PRR) in each small zone to identify gray zones and dynamically update rebroadcast zones based on changing traffic conditions. The model was trained and evaluated using a dataset primarily generated through the NETSIM framework. The results demonstrate a high prediction accuracy rate, enabling the timely adjustment of rebroadcast zones in response to regular fluctuations in traffic conditions. This ensures the reliable dissemination of alert messages.

Due to the lack of datasets in conformance with the specifications of our study environment and incorporating the features considered by our model, we opted to create our dataset. This dataset is derived from a realistic mobility scenario. The training and evaluation of our models are conducted using this generated dataset, and the results of performance tests show promising levels of prediction accuracy.

The current study focuses on a particular urban setting (Toulouse, France) and relies on simulated data. The subsequent phase involves extending the scope to encompass additional urban areas and real-world scenarios. The inclusion of real data is crucial to enhance the model's applicability. Fine-tuning network parameters, such as the calculation of path loss exponent, will be guided by feedback from real-world data.

Acknowledgment

The initial presentation of this work is outlined in the PhD thesis titled "An Efficient Emergency Information Dissemination Scheme for Emerging Infrastructure-based Vehicular Networks."

We would like to express our gratitude to Slim Abdellatif for his invaluable assistance. His insightful input and support significantly contributed to our progress.

References

[1] K.N. Qureshi, and H. Abdullah, "A Survey on Intelligent Transportation Systems, " Middle-East Journal of Scientific Research, Vol. 15 No 5, pp. 629–642, 2013.

[2] J. Lukkien, "Introduction to Cooperative Intelligent Transportation Systems, " In: Automotive Systems and Software Engineering, Y. Dajsuren, M. van den Brand (eds), Springer, 2019.

[3] R. Chakroun, T. Villemur, and B. Nougnanke, "Learning-Based Infrastructure To Vehicle Link Quality Estimation, " 31th International Conference on Software, Telecommunications and Computer Networks (SoftCOM), Split, Croatia, September 21-23 2023.

[4] R. Chakroun, "an Efficient emergency information dissemination scheme for emerging infrastructure-based vehicular networks," PhD. thesis, University of Toulouse, France, Nov. 2022.

[5] L. Tang, K.C. Wang, Y. Huang, and F. Gu, "Channel characterization and link quality assessment of IEEE 802.15. 4-compliant radio for factory environments, " IEEE Transactions on industrial informatics, Vol. 3, No. 2, pp. 99–110, 2007.

[6] G. Cerar, H. Yetgin, M. Mohorčič, and C. Fortuna, "Machine learning for wireless link quality estimation: A survey, " IEEE Communications Surveys & Tutorials Vol. 23 No. 2, pp. 696–728, 2021.

[7] S. Krug, H. Toepfer, T. Hutschenreuther, and J. Seitz, ''Towards Robust Communications of Wireless Sensor Networks in Vehicular Environments: A Case Study, "Journal of Communications Software and Systems, Vol. 13 No. 4, pp. 157–164, January 2018.

[8] R. Bauza, J. Gozalvez, and M. Sepulcre, "Power-aware link quality estimation for vehicular communication networks., "IEEE Communications Letters, Vol. 17 No. 4, pp. 649–652, 2013.

[9] H. Okada, A. Takano, and K. Mase, "A proposal of link metric for next-hop forwarding methods in vehicular ad hoc networks, " 6th IEEE Consumer Communications and Networking Conference, pp. 1–5, 2009.

[10] X. Cai, Y. He, C. Zhao, L. Zhu, and C. Li, "LSGO: Link state aware geographic opportunistic routing protocol for VANETs, " EURASIP Journal on wireless communications and networking, Vol. 2014, No. 1, pp. 1–10, 2014.

[11] H. Wang, G. Tan, and J. Yang, "An improved VANET intelligent forward decision-making routing algorithm, " Journal of Networks, Vol. 7 No. 10, pp. 1546—1553, 2012.
[12] A. Amoroso, G. Marfia, and M. Roccetti, "Going realistic and optimal: A distributed multi-hop broadcast algorithm for vehicular safety, " Computer networks, Vol. 55 No. 10, pp. 2504–2519, 2011.
[13] C.F. Wang, Y.P. Chiou, and G.-H. Liaw, "Nexthop selection mechanism for nodes with heterogeneous transmission range in VANETs, " Computer Communications, Vol. 55, pp. 22–31, 2015.
[14] J. Rak, "LLA: A new anypath routing scheme providing long path lifetime in VANETs, " IEEE communications letters, Vol. 18 No. 2, pp. 281–284, 2013.
[15] N. Baccour, A. Koubâa, L. Mottola, M. A. Zúijiga, H. Youssef, C. A. Boano, and M. Alves. "Radio link quality estimation in wireless sensor networks: A survey, " ACM Transactions on Sensor Networks (TOSN), Vol. 8, No. 4, pp. 1–33, 2012.
[16] C. J. Lowrance, and A. P. Lauf, "Link quality estimation in ad hoc and mesh networks: A survey and future directions, " Wireless Personal Communications, Vol. 96, No. 1, pp. 475–508, 2017.
[17] 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 vehicular technology magazine, Vol. 13, No. 2, pp. 94–101, 2018.
[18] N. Mohammedali, T. Kanakis, A. Al-Sherbaz and M. Agyeman, "Management and Evaluation of the Performance of end-to-end 5G Inter/Intra Slicing using Machine Learning in a Sustainable Environment, " Journal of Communications Software and Systems, Vol. 19 No. 1, pp. 91–102, March 2023.
[19] I. A. Bartsiokas, P. K. Gkonis, D. I. Kaklamani and I. S. Venieris, "ML-Based Radio Resource Management in 5G and Beyond Networks: A Survey, " IEEE Access, Vol. 10, pp. 83507–83528, 2022.
[20] B. Turan, A. Uyrus, O. Nuri Koc, E. Kar, and S. Coleri, "Machine Learning Aided Path Loss Estimator and Jammer Detector for Heterogeneous Vehicular Networks, " IEEE Global Communications Conference (GLOBECOM), pp. 1–6, 2021.
[21] P. M. Ramya, M. Boban, C. Zhou, and S. Stańczak., "Using learning methods for v2v path loss prediction, " IEEE Wireless Communications and Networking Conference (WCNC), pp. 1-6, 2019.
[22] W. Benrhaiem, and A. Senhaji Hafid, "Bayesian networks based reliable broadcast in vehicular networks, " Vehicular Communications, Vol. 21, pp.100–181, 2020.
[23] R. Chakroun, S. Abdellatif and T. Villemur, "Q-Learning Relay Placement for Alert Message Dissemination in Vehicular Networks, " 19th International Conference on Mobile Systems and Pervasive Computing (MobiSPC), Niagara Falls, Canada, pp. 222–230, August 9-11 2022.
[24] R. Meireles, M. Boban, P. Steenkiste, O. Tonguz, and J. Barros, "Experimental study on the impact of vehicular obstructions in VANETs, "IEEE Vehicular Networking Conference, pp. 338–345, 2010.
[25] S. Toufga, S. Abddellatif, H.T. Assouane, P. Owezarski, and T. Villemur, "Towards Dynamic Controller Placement in Software Defined Vehicular Networks," Sensors, Vol. 20, No. 6, pp. 1701, March 2020.
[26] R. Chakroun, S. Abdellatif and T. Villemur, "LAMD: Location-based Alert Message Dissemination scheme for emerging infrastructure-based vehicular networks, " Internet of Things , Vol. 19, Article 100510, 2022.
[27] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, "Machine learning for networking: Workflow, advances and opportunities, " IEEE Network, Vol. 32, No. 2, pp. 92–99, 2017.
[28] O. Sagi, and L. Rokach, "Ensemble learning: A survey, " Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2018.
[29] L. Breiman, "Random forests, " Machine learning, Vol. 45, No. 1, pp. 5 – 32, 2001.
[30] N. Foster, N. McKeown, J. Rexford, G. Parulkar, L. Peterson, and O. Sunay, "Using deep programmability to put network owners in control, "ACM SIGCOMM Computer Communication Review, Vol. 50, No 40, pp 82—88, 2020.
[31] J. S. Weber, M. Neves, and T. Ferreto, "VANET simulators: an updated review, " Journal of the Brazilian Computer Society, Vol. 27, No. 1, pp. 1–31, 2021.
[32] J. Gozálvez, M. Sepulcre, and R. Bauza, "IEEE 802.11 p vehicle to infrastructure communications in urban environments, " IEEE Communications Magazine, Vol. 50 No. 5, pp. 176–183, 2012.
[33] A. Böhm, K. Lidström, M. Jonsson, and T. Larsson, "Evaluating CALM M5-based vehicle-to-vehicle communication in various road settings through field trials, " IEEE Local Computer Network Conference, pp. 613–620, 2010.
[34] Y. A. Zakaria, E.K.I. Hamad, A.S. Abd Elhamid, and K.M. El-Khatib, "Developed channel propagation models and path loss measurements for wireless communication systems using regression analysis techniques," Bulletin of the National Research Centre, Vol. 45, No. 54, 2021.

Raoua Chakroun got her Ph.D. in Computer Science from the University of Toulouse, France. As of 2023, she serves as an assistant professor at INSA Toulouse. Her ongoing research interests span Vehicular Networks, Software Defined Networks, Machine Learning, Intelligent Transport Systems, and Data Science.

Thierry Villemur T. Villemur is full professor at University of Toulouse (University Institute of Technology Blagnac), France, and researcher at LAAS-CNRS Laboratory in networking and communication technologies. In former years, he was contributing in distributed systems and middleware layers, more precisely in collaborative systems and group communications. Now, he focuses on networking researches. His current research topics include adaptive software architectures, software programmable networks, and vehicular communications.