VeCEn: A Data Acquisition Framework for Heterogeneous Vehicular Networks

Sagar Kavaiya*, Dhaval K Patel†, Mukesh Zaveri‡, Mehul S Raval§, S.N. Merchant¶
*†§School of Engineering and Applied Science, Ahmedabad University, India.
‡ Department of Computer Science and Engineering SVNIT, Surat, Gujarat, India
¶ Department of Electrical Engineering IIT Bombay, Powai Mumbai, India.
Email: *†§{sagar.k, dhaval.patel, mehul.raval}@ahduni.edu.in
‡mazaveri@coed.svnit.ac.in ¶merchant@ee.iitb.ac.in

***Abstract*—Real-time simulation and emulation are crucial for the deployment of the intelligent transportation system (ITS). Also, traffic data acquisition through experiments is a costly proposition. Therefore, it is important to synthesize data using open-source traffic simulators and use it in crucial ITS studies using network simulators. The paper proposes a data acquisition framework and integrator named Vehicles in Connected Environment (VeCEn). It helps to store the network parameters (like received signal strength, and delay) and physical parameters (like speed, and acceleration) by integrating network (OMNET++) and traffic simulators (SUMO - Simulation in Urban Mobility). The paper highlights the internal architecture and working of VeCEn and proves its stability by evaluating performance metrics, and physical and network parameters. We believe that such an integrated environment will enhance our understanding of traffic patterns, user activities, vehicle density estimation, and the impact of velocity on ITS problems.**

***Index Terms*—Data acquisition, network simulator, simulation of urban mobility, traffic simulator, vehicular networks.**

I. Introduction

Vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication have been reported as key factors to enable the quality of service (QoS) in the intelligent transportation system (ITS). With the rise of cooperatives, ITS complexity in research and societal problems related to vehicles have increased. The quality of service (QoS), broadcasting of safety messages, and impact of environmental factors on traffic systems are becoming major concerns for policymakers. The problems related to resource allocations in V2V communications are tackled through multiple ways to satisfy the latency requirement in emergency and infotainment services [1]. Nowadays problems related to resource allocation in transport engineering, which mainly includes low V2V and V2I payload, and sum rate are tackled with deep reinforcement learning (DRL). The DRL with extrinsic reward mechanism is adopted to allocate the resources through minimizing latency constraints [2]. Numerous traffic simulators [3], [4] are available to simulate and emulate ITS. Mainly, the traffic simulators are focusing on vehicular dynamics, traffic density, and route planning [5].

Real-time communication between connected vehicles is emerging as a growing area and has led to the development of many self-learning environment-based techniques. This helps in providing each other with safety warnings and traffic information. The V2V communication share omnidirectional messages about speed and location among the vehicles. Due to this one can have a 360-degree environment surrounding the vehicle, making drivers aware of the nature of the neighboring vehicles. This enables the transfer of safety messages with location, and speed and helps to reduce the number of accidents [5]. The V2I communication enables the transfer of details about road conditions, traffic congestion, and accidents on the roadway. The V2V and V2I communication would need effective spectrum sensing techniques [6]–[8]. Thus, the proposed work considers the data acquisition framework and its applicability by extracting information like primary user (PU) activity statistics, and total error rate. Therefore, this work enables data acquisition by combining the network and traffic simulator.

A. Motivations and Novelties

The existing simulators have limited functionality and Table I summarizes the comparison with existing simulators and the motivation to use the VeCEn integrator. From a traffic generation point of view, simulation of urban mobility (SUMO) [9] can initiate the simulation by generating traffic but the parameters of the traffic experiment like the number of vehicles, total route length, total lane length, and a number of collisions are not being extracted at the same time while an experiment is running. In general, the information on such parameters is not stored so the proposed integrator will extract such data from the SUMO. Similarly, when the mobility traces are being imported into the discrete event simulator like OMNET++ [10], the networking facility between the vehicles is enabled. Based on the limitations of the existing platforms and the requirements of heterogeneous vehicular networks, the VeCEn enhances the QoS in simulation-based environments and data acquisition systems. In the context of the above motivations, our contributions are summarized below:

1) Developing a mechanism for connecting the traffic simulator, and the network simulator. The integrator follows the IEEE vehicular standards (802.11p) and continuously generate episodes of the required scenes. This creates a restore point in a local machine to save the traces of the generated episodes and it will provide the exchange of information of the saved traces to the object-oriented pointer which will decide the task of extraction of the data.

Table I: State of the art simulators features and their improvisation by VeCEn.

Name of Platform	Type of Platform	Used for	Available Features	Limitations of Existing Platforms	Requirements in Heterogeneous Networks	How VeCEn overcomes the limitations?
NGSIM [4]	Traffic simulator	Testing the existing mobility models	- OSM facility - mobility model generator	- Unable to load ad-hoc standards	- Needs an interface of IEEE 802.11p and higher	- Provides interface with IEEE 802.11p using OMNET++
MOBYSIM [5]	Hybrid simulator	Generating Artificial traffic	-OSM and route facility	- Unable to store the route traces	- Needs to store the current traces	- Can store the information of traces in csv file via invoker
Paramics [7]	Driving simulator	Identifying the driving patterns	- MOVE facility	-Unable to extract the route information	-Needs to identify the route	- Can identify the current vehicle route
VISSIM [8]	Traffic simulator with networking facility	Generating artificial traffic and connecting vehicles via ad hoc protocols	- Integration of vehicles in networks simulation (VEINS) and traffic simulator	- Unable to load data from VEINS	- Needs an interface which can load data from networking library	- Can extract the networking parameters from OMNET++
AIMSUN [9]	Traffic + Driving simulator	Generating the artificial traffic and identifying driving patterns	- OSM and Route facility	-Unable to load route traces	- Needs to store the current traces	- Can load the data in a data reservoir

2) As a use case, the primary user's activity statistics are derived from the available data. It also estimates the shape and scale parameters of the distributions of idle and busy periods can be done. Information about estimated parameters helps to calculate the probability about the activeness or inactiveness of the PUs.
3) The stability of the integrator has been rigorously tested in terms of repeat values per scene, empty value per scene and number of sample points per episode.

The rest of the paper is organized as follows: Section II describes the connected environment setup which is to be generated in the traffic simulator. Section III describes the data acquisition technique of the proposed VeCEn. Section IV covers the discussions on results and Section V concludes the paper.

II. Connected Environment Setup

The roadside units (RSUs) and vehicles act as PU and passengers in a car act as SU. As per the scenarios mentioned for Dedicated Short Range Communication (DSRC) based communications in [11], the primary network shares the resource with the secondary network. The passengers in a car or pedestrians using an unlicensed band allow finding opportunistic access to the spectrum. Similar to the [12], [13], we assume that PUs are the licensed TV/DSRC band holders and SUs are unlicensed Wi-Fi/Digital Television (DTV) band holders. The DSRC channels are used for sharing spectrum awareness. While PUs channels are idle, a secondary user (SU) can utilize the channels to communicate with each other. According to the wireless access in vehicular environment (WAVE) MAC layer protocol, the control channel is used to set up the communication link. The PU's protection radius is R. In the model, vehicular users in a cognitively connected environment are mobile and PUs are stationary. According to that, the speed of the vehicle is a relative velocity between them. The proposed integrator is responsible to store the physical parameters (like the velocity of the vehicle, total trip time, and total road length) and network parameters (like received signal strength, delay, and path loss). During integration, the distance between the PU node and the SU node is a function of their relative speed, the PU's protection range, and the sensing range of the vehicle user. The channel model considers all entities specific to a vehicular environment, such as multi-path fading, Doppler shift, and scattering, which can be mathematically expressed as:

$$h(\tau, t) = \sum_{k=0}^{P-1} h_k(t) e^{-j2\pi f_c \tau_k(t)} \delta[\tau - \tau_k(t)], \tag{1}$$

where τ_k is the path delay of k^{th} path, t is time variable, f_c is the carrier frequency, $h_k(t)$ is the channel gain, and δ is the impulse function. The spectrum sensing operation performs a binary hypothesis test as follows:

$$H_0 : r(t) = \sum_{t=0}^{\infty} n_r(t) \tag{2}$$

$$H_1 : r(t) = \sum_{t=0}^{\infty} h(\tau, t) x(t - \tau) + n_r(t), \tag{3}$$

where $r(t)$ is the received signal, $x(t-\tau)$ is the transmitted signal, and $n_r(t)$ is the noise. The hypothesis of the absence and presence of the PU are H_0 and H_1, respectively. When the hypothesis H_0 is true then either there is no PU within the sensing range or it is inactive within the sensing range s. Thus the PU 'absent' event depends on the α and the probability of the α event is Pr(A). When a channel is used by the primary user and the SU detects a PU signal in a given channel (i.e., hypothesis H_1), this PU 'busy' event depends on β with probability as Pr(B). Note that (Pr(A)= 1 - Pr(B)). The α and the β are the primary activity statistics parameters obtained from the distributions of the data for idle and busy periods. The probability of missed detection under the impact of primary activity statistics depends on the sensing range of the vehicular user, the protection range of PU, the velocity of the vehicular user, and the energy detection threshold given by [14, Eq. 16]:

$$\begin{aligned} Pr(miss)_{PAS} = {} & Pr(\Delta \le \lambda | H_1, B) Pr(B) Pr_{ON} + \\ & Pr(\Delta \le \lambda | H_0, A) Pr(A) Pr_{OFF} \end{aligned} \tag{4}$$

where $Pr(miss)_{w/oPAS}$ is the missed detection without activity statics. $Pr(\Delta \leq \lambda | H_1, B)$ is the missed detection over small scale fading. Δ is the test statistics and λ is detection threshold. The Pr_{ON} and Pr_{OFF} can be derived from the shape parameter α and scale parameter β obtained from the data. The α and β parameters can be obtained by [15, Eq. 5, 6] for single and double-lane traffic. Moreover, under the impact of mobility, the channel gain is explained as [16] :

$$h_k(t) = \frac{g_x}{\sqrt{1 + d_X^{\zeta}}}, \tag{5}$$

where ζ denotes the path loss exponent, g_x is the distribution of small-scale fading and d_X is the distribution of distances of SU from PU over dense and medium traffic.

III. Proposed VeCEn Integrator

This section describes the internal architecture of the proposed integrator. The states involved in generating the traffic and network simulation are highlighted.

A. Data Acquisition through Vehicles in Connected Environment (VeCEn) Integrator

We analyze the performance of the proposed integrator by using the SUMO [9], OMNET++, and $GEMV^2$. $GEMV^2$ [17] is a computationally efficient propagation model for V2V communications, which explicitly accounts for surrounding objects. The model considers different link types (Line of Sight (LOS), non-LOS due to static objects, and non-LOS due to vehicles) to calculate large-scale signal variations. Additionally, $GEMV^2$ statistically determines small-scale signal variations using a simple geometry-based model that takes into account the surrounding static and mobile objects. By implementing the proposed approach in $GEMV^2$, we show how it behaves in realistic propagation conditions, including time-varying LOS environments that affect the path loss as well as highly dynamic network topology. The experimental traffic data is generated in SUMO [9] for a 2-km^2 region around the Ahmedabad university area, Gujarat, India, and used as an input to $GEMV^2$ to generate the traces. The proposed framework utilizes the randomtraffic.py file of SUMO for traffic generation and shows its integration into the network file of OMNET++. This integration enables an understanding of traffic patterns, and activity statistics for vehicular networking. The entire process is divided into five parts which are discussed below:

1) Traffic State - : SUMO: Road networks can be either generated using an application named "netgen" or generated by importing a digital road map. The road network importer "netconvert" allows to read networks from other traffic simulators such as VISUM [18], Vissim [19], or MATsim [20]. The desired location of the Ahmedabad university area is incorporated as an open street map (OSM) file. Along with the OSM file, the route and network files are generated for the same locations with the help of the commands. The outcome is the project file which is containing information on traffic parameters like the number of vehicles, speed, and initial location. We restricted our area of experiment with the proposed network model highlighted in [21] to match the map file with the predefined proposed schemes. This is due to validate the performance of the proposed VeCEn with the previously proposed distributions of distances.
2) Network State - OMNET ++ : The network simulator also known as a discrete event simulator is responsible to provide the ability to traffic as connected via vehicular ad-hoc network protocols. For that, we customized the binary class file to add the recent and widely adopted roadside units to work as PUs. The network simulator enables the experiment for a time period of 100ms to 1 minute. Then, the vehicles are provided access to the spectrum.
3) Trace State - $GEMV^2$: The process of the network simulation is quite challenging to visualize. We generate the traces as a provided resources link with the color code varying with the dBm to make it manually visible. This enables us to know whether the vehicles are able to access the resource or not while considering real-time non-LOS and LOS environments. Note that for mobility the traces are updated with respect to time.
4) Integration State - VeCEn : Till this stage, the process of traffic and network simulation are performed independently. However, in vehicular communications, the traffic and network utilization should be analyzed in parallel. Therefore, we proposed a framework that enables the connectivity between the network and traffic simulator. The VeCEn is responsible to unload the data of traffic files from SUMO and to load it in OMNET++ continuously. Simultaneously, it is also responsible to extract the performance of the network simulator as a project class file which is further restored as a data file.
5) Storage and Restore State: The acquired data file is the project class file which is usually not readable as CSV files. To overcome reading limitations we also prepare the storage and restore stage in the VeCEn which provides the data in CSV form.

Figure 1 is the internal architecture of the integrator. One can notice that the network state is enabled after generating the traffic from SUMO. The interface between the traffic state and VeCEn is done through the mobility models. Similarly, the interface between the network state and VeCEn is done through the networking library. The implementation libraries are available on a repository citelab. After the network state, the integrator is invoked through a file that uses location to store the data of the selected parameters. The variable supply file containing the functions which eliminate or replace the missing values with the mean, median, and mode of the selected parameter is used to remove the repeated values and to increase the accuracy. Since the experiment is running, there are discrepancies in the extracted data at the first level, so the error filter of the integrator will identify the number of errors in the extracted parameters. This process continues until the scene ends. The error filter will reduce the number of mistakes and dump the extracted information into the final python simulation. The last python simulation file is integrated with the extraction facility in CSV and JSON.

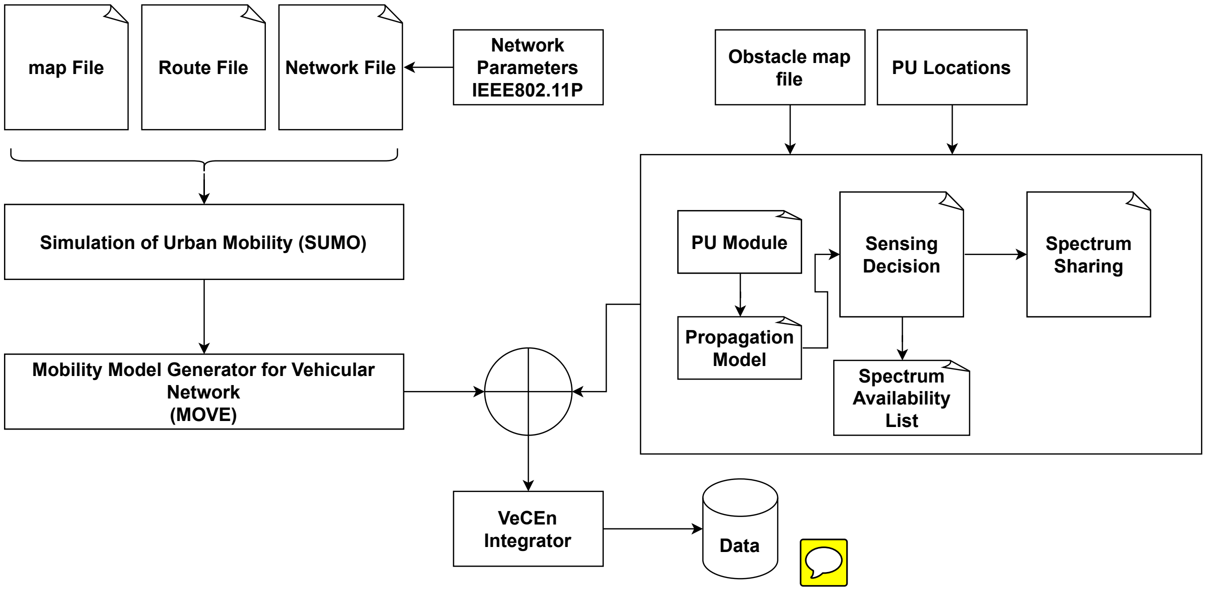

Figure 1: Integration of network and traffic simulator through VeCEn.

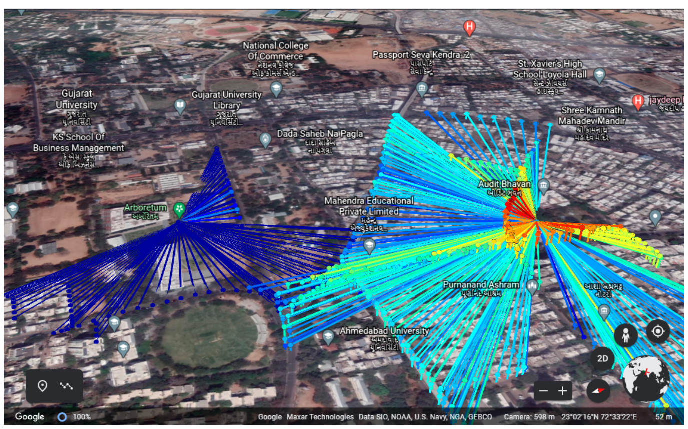

Figure 2: VeCEN integrator output V2I links.

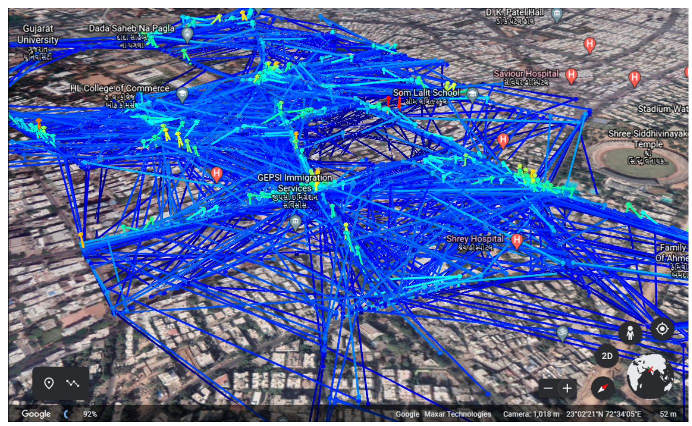

Figure 3: VeCEN integrator output V2V links.

B. Explanation of Acquired Data sets and Use Case for Total Error Rate

The performance of the considered cognitive vehicular network (CVN) is evaluated at the fusion center (FC) in terms of total error rate (TER). The SU broadcast the binary decision which highlights the absence and presence of PU at each cognitive user. The decision rule is performed as per given in [22]. The decision is then sent to the FC which is affected by the imperfectness of the reporting channel. As a result, N binary decisions are obtained from N SUs. Due to the imperfection in the reporting channel, the total error rate Q can be obtained as [22] :

$$Q \triangleq \beta E_f + (1-\beta)E_m, \tag{6}$$

where E_f is false alarm probability and E_m is the missed detection probability at the FC. The β and (1 - β) in [22] are the probabilities of the prior information of hypotheses H_0 and H_1 respectively. The equiprobable hypothesis i.e., $\beta = 0.5$ reflects the total error rate of the considered dense traffic CVN model.

$$\begin{aligned} E_f &= 1 - [(1-P_f)(1-\varepsilon) + \varepsilon P_f]^N, \\ E_m &= [P_m(1-\varepsilon) + \varepsilon(1-P_m)]^N \end{aligned} \tag{7}$$

We consider $N = 2$. The E_m and E_f to evaluate the TER given as [22, Eq.13] and [22, Eq.14] respectively. The ε is considered as the error probability.

IV. Experiments and Discussions

We analyze the performance of the proposed data acquisition within a VANET environment using the $GEMV^2$ V2V propagation simulator and MATLAB. The experimental traffic data is generated in SUMO for a 2-km^2 region around Ahmedabad university with the single-lane and double-lane scenarios. However, the VeCEn is not limited to the considered area only. The acquired data with steps is available on [23].

Based on methods provided in [24], we compute the shape

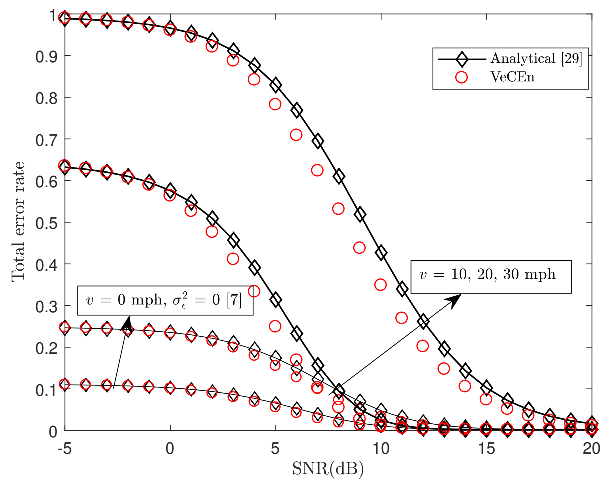

Figure 4: Total error rate against SNR [29].

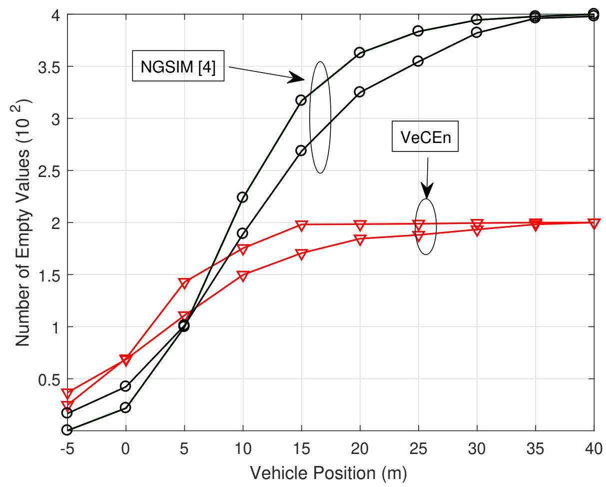

Figure 6: Stability Check 1: Number of empty values against vehicle position.

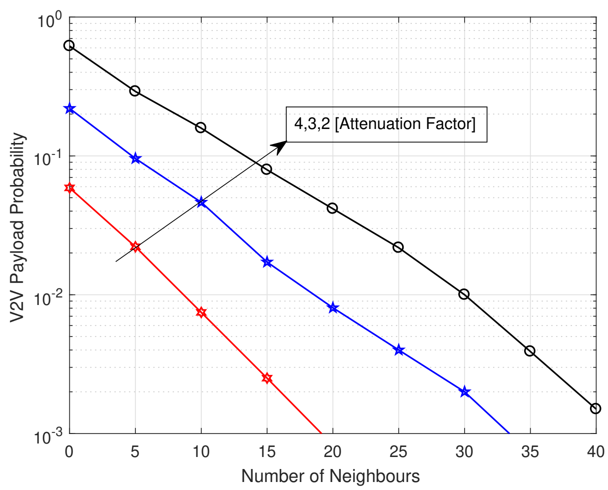

Figure 5: V2V payload against several neighbors for 35000 episodes.

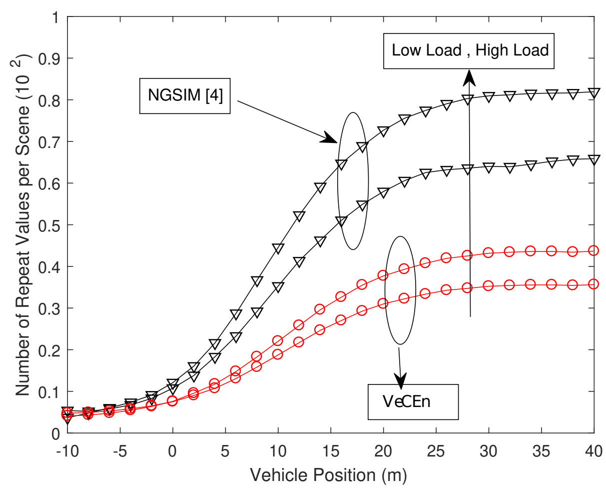

Figure 7: Stability Check 2: Number of repeat values against vehicle position.

and scale parameters of the distributions of the idle and busy periods. We conclude that a single lane follows the exponential distribution with α = 2.82 and β = 0.60. From that parameters, we compute the probabilities for which the PU is idle and busy. In this section, based on the parameters, we compute the various values of ON and OFF period probabilities. For the present research study, the performance evaluation compared the following aspects of existing works and the simulator. Figure 2, and Figure 3 present the received power distribution under the V2I and V2V scenarios. In difference from the [25], the VeCEn is capable to store and present information regarding the RSSI under the mobility and acceleration impact. The red color shows the minimum RSSI with the highest shadowing effect while the shades of blue color show the connected link with various RSSI.

It is important to note that for each collected measurement data point, we calculate the mean and standard deviation using the per-packet received power difference between VeCEn and measurements provided in [3]. Furthermore, we use the same value of relative permittivity (r = 1.003) in all environments to calculate the reflection coefficient for the ground reflection (i.e., we do not fit the value to a given dataset).

Vehicular user is considered to be mobile and each vehicular user has its sensing range. The residential base stations or access points are considered to be PUs and each PU has its protection range where secondary users are not permitted to use PUs' active channels. First, we import the OpenStreetMap file in SUMO to obtain the medium and dense traffic scenes. We utilize the randomtrips.py file to distinguish imported maps in the single and double lanes. First, we plotted the variation of TER provided in [22] for single-lane medium traffic without activity statistics as shown in Figure 4. The maximum sensing range of the secondary user is considered to be the same as the maximum range in IEEE 802.11p DSRC standard for vehicular networks as provided in [21].

We observed that the error rate decreases with the SNR as shown in Figure 4. Then we plotted the channel gain obtained from the network simulator project file concerning the distance of vehicle users as a function of SNR in Figure 5. It implies that for the LOS and V2V links, the strength of the link is more than NLOS which is intuitive but it focuses on the validation of the proposed integrator. The one-to-one correspondence among vehicles in both simulation, thus obtained by preserving their memory between two successive time steps because of the constant velocity between the two vehicles. Then, for each vehicle in the surroundings of the interactive vehicle, it is checked whether it existed in the previous simulation step. In case of existence, the vehicle in the traffic simulation is assigned to the vehicle with the same visual features in the driving simulation; otherwise, a new vehicle in the driving simulation is assigned to it. Further, as a stability check of the integrator, we check its ability to give the instantaneous values of the corresponding experiment by plotting the number of empty values, several repeat values, and several extracted scenes per vehicle position in Figure 6, and Figure 7 respectively. The two different plots show the runs of each integrator. The negative value indicates that the vehicle is on either side of the base station. Finally, all the vehicles are simulated by the traffic-simulator model until they enter the surroundings of the interactive vehicle. From that point on, the vehicles are simulated by the traffic module. The integrator can be used to determine the requirements of deploying ITS in real-time for practical applications.

V. Conclusion

In this paper, we have proposed the data acquisition framework by integrating the network and traffic simulators. For a validation purposes, the obtained data from the integrator is utilized to plot the various performance metric like missed detection probability, detection probability, and total error rate. We conclude that in contrast with the existing works, our proposed integrator can restore the data from both the network and traffic simulator together which can open a new dimension to the applicability of machine learning, non-linear optimization, and numerical analysis methods.

VI. Acknowledgement

The authors thank the School of Engineering and Applied Science, Ahmedabad University for providing the infrastructural support. This work is sponsored by the University research board under Grant URBSEASI22A3 and the Department of Science and Technology-Gujarat Council of Science and Technology (DST-GUJCOST) under Grant GUJCOST/STI/2021-22/3916.

References

[1] L. Liang, H. Peng, G. Y. Li, and X. Shen, "Vehicular communications: A physical layer perspective," *IEEE Trans. on Veh. Technol.*, vol. 66, no. 12, pp. 10 647–10 659, 2017.

[2] H. Ye, G. 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, 2019.

[3] L. Yang, J. H. Yang, E. Feron, and V. Kulkarni, "Development of a performance-based approach for a rear-end collision warning and avoidance system for automobiles," in *Proc. of IEEE Intelligent Vehicles Symposium. (Cat. No. 03TH8683)*, 2003, pp. 316–321.

[4] C.-Y. Chang and Y.-R. Chou, "Development of fuzzy-based bus rear-end collision warning thresholds using a driving simulator," *IEEE Trans. on Intel. Transportation Systems*, vol. 10, no. 2, pp. 360–365, 2009.

[5] M. van Noort, B. van Arem, and B. Park, "Mobysim: An integrated traffic simulation platform," in *Proc. of IEEE Intelligent Transportation Systems*, 2010, pp. 1301–1306.

[6] X. Chen, C. Wu, T. Chen, H. Zhang, Z. Liu, Y. Zhang, and M. Bennis, "Age of information aware radio resource management in vehicular networks: A proactive deep reinforcement learning perspective," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2268–2281, 2020.

[7] H. Zhou, W. Xu, Y. Bi, J. Chen, Q. Yu, and X. S. Shen, "Toward 5g spectrum sharing for immersive-experience-driven vehicular communications," *IEEE Wireless Commun.*, vol. 24, no. 6, pp. 30–37, 2017.

[8] A. M. Vegni and D. P. Agrawal, *Cognitive Vehicular Networks*. Boca Raton, FL, USA: CRC Press, Inc., 2016.

[9] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility: an overview," in *Proc. of Third International Conference on Advances in System Simulation*. ThinkMind, 2011.

[10] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, 2008, pp. 1–10.

[11] D. Jiang, V. Taliwal, A. Meier, W. Holfelder, and R. Herrtwich, "Design of 5.9 ghz dsrc-based vehicular safety communication," *IEEE wireless commun.*, vol. 13, no. 5, pp. 36–43, 2006.

[12] B. Aygun and A. M. Wyglinski, "A voting-based distributed cooperative spectrum sensing strategy for connected vehicles," *IEEE Trans. on Veh. technol.*, vol. 66, no. 6, pp. 5109–5121, 2016.

[13] K. D. Singh, P. Rawat, and J.-M. Bonnin, "Cognitive radio for vehicular ad hoc networks (cr-vanets): approaches and challenges," *EURASIP journal on wireless communications and networking*, vol. 2014, no. 1, p. 49, 2014.

[14] D. B. Rawat, R. Alsabet, C. Bajracharya, and M. Song, "On the performance of cognitive internet-of-vehicles with unlicensed user-mobility and licensed user-activity," *Computer Networks*, vol. 137, pp. 98–106, 2018.

[15] D. K. Patel, S. Kavaiya, Z. Ding, Y. L. Guan, and S. Sun, "Impact of primary user activity statistics in cognitive vehicular networks," *IEEE Transactions on Vehicular Technology*, vol. 71, no. 3, pp. 2859–2873, 2022.

[16] Y. Ai, M. Cheffena, A. Mathur, and H. Lei, "On physical layer security of double rayleigh fading channels for vehicular communications," *IEEE Commun Lett.*, vol. 7, no. 6, pp. 1038–1041, Dec 2018.

[17] M. Boban, J. Barros, and O. K. Tonguz, "Geometry-based vehicle-to-vehicle channel modeling for large-scale simulation," *IEEE Trans. Veh. Technol.*, vol. 63, no. 9, pp. 4146–4164, 2014.

[18] A. A. Ahmed, A. A. Alkheir, D. Said, and H. T. Mouftah, "Cooperative spectrum sensing for cognitive radio vehicular ad hoc networks: An overview and open research issues," in *Electrical and Computer Engineering (CCECE), 2016 IEEE Canadian Conference on*. IEEE, 2016, pp. 1–4.

[19] "PTV Vissim new," Dec. 2020. [Online]. Available: https://www.ptvgroup.com/en/solutions/products/ptv-vissim/

[20] M. Di Felice, R. Doost-Mohammady, K. R. Chowdhury, and L. Bononi, "Smart radios for smart vehicles: Cognitive vehicular networks," *IEEE Vehicular Technol. Mag.*, vol. 7, no. 2, pp. 26–33, 2012.

[21] S. Zhu, C. Guo, C. Feng, and X. Liu, "Performance analysis of cooperative spectrum sensing in cognitive vehicular networks with dense traffic," in *Proc. VTC Spring*, 2016, pp. 1–6.

[22] L. Gahane, P. K. Sharma, N. Varshney, T. A. Tsiftsis, and P. Kumar, "An improved energy detector for mobile cognitive users over generalized fading channels," *IEEE Trans. Commun.*, vol. 66, no. 2, pp. 534–545, Feb 2018.

[23] "MICxN - Overview." [Online]. Available: https://github.com/MICxN

[24] M. López-Benítez and F. Casadevall, "Time-dimension models of spectrum usage for the analysis, design, and simulation of cognitive radio networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 5, pp. 2091–2104, 2013.

[25] M. Boban, J. Barros, and O. K. Tonguz, "Geometry-based vehicle-to-vehicle channel modeling for large-scale simulation," *IEEE Trans. Veh.Technol.*, vol. 63, no. 9, pp. 4146–4164, 2014.