Agile Calibration Process of Full-Stack Simulation Frameworks for V2X Communications

Ioannis Mavromatis, Andrea Tassi, Robert J. Piechocki, and Andrew Nix
Department of Electrical and Electronic Engineering, University of Bristol, UK
Emails: {Ioan.Mavromatis, A.Tassi, R.J.Piechocki, Andy.Nix}@bristol.ac.uk

Abstract—Computer simulations and real-world car trials are essential to investigate the performance of Vehicle-to-Everything (V2X) networks. However, simulations are imperfect models of the physical reality and can be trusted only when they indicate agreement with the real-world. On the other hand, trials lack reproducibility and are subject to uncertainties and errors. In this paper, we will illustrate a case study where the interrelationship between trials, simulation, and the reality-of-interest is presented. Results are then compared in a holistic fashion. Our study will describe the procedure followed to macroscopically calibrate a full-stack network simulator to conduct high-fidelity full-stack computer simulations.

Keywords—Connected Autonomous Vehicles, V2X, IEEE 802.11p/DSRC, Full-Stack Simulator, VEINS, Validation process.

I. INTRODUCTION

Connected Autonomous Vehicles (CAVs) are enhanced daily with new autonomous features, leading gradually to fully autonomous means-of-transports [1]. Being part of the ecosystem of Intelligent Transportation Systems (ITSs), CAVs will require an agile interconnecting framework [2], [3], providing a constant service and optimal system behavior. To optimize and further enhance the performance of this framework, initial experimental evaluation of real-world trials and simulation results is required.

In this work, we aim to establish a connection between simulated and trial-based results for a vehicular network. Introducing the procedure followed, we will describe how inconsistencies during the experiments are identified and excluded from the evaluation. We will later present a calibration framework for the fine-tuning of the imperfect simulation results to enhance their behavior and achieve high-fidelity "real-world" results.

Simulations are approximated models of the physical world. However, they are easily and inexpensively conducted using an appropriate network simulator achieving near-perfect results. For example, the number of vehicles within a network can be easily scaled up to increase the network congestion (as in [4]). Furthermore, they offer a high degree of flexibility. Using different configurations and isolating particular parameters, we can examine the behavior of a system under specific conditions. This is the case of [5] where the impact of different beacon intervals has been investigated with respect to the end-to-end delay, for different road networks and city-wide scenarios.

On the contrary, real-world trials are based on a "perfect" model. For example, authors in [6] experimentally analyzed the performance of a vehicular network based on data coming from off-the-shelf IEEE 802.11p devices. The disadvantages of this experimental evaluation are the cost, the required time and the inability to reproduce – since it is affected by physical parameters varying over time and it cannot be easily isolated and ignored.

A very complex real-world system is harder to model or might require increased resources to be simulated. Trial results can aid the design of a system by abstracting various parameters and introducing them as a priori knowledge in a simulation. For instance, authors in [7], designed a simulator for geometry-based efficient propagation models for Vehicle-to-Vehicle (V2V) communications. Their simulator was based on an extensive trial campaign, in order to identify parameters such as the path loss exponent and the small-scale signal deviation for different distances and environments. Trial-based results though may vary between different devices, being related with the quality of the equipment used and the software accompanying it.

For the above reasons, it is obvious that various approximations, random and systematic errors are introduced during a system performance evaluation. A direct comparison between the simulated and real-world results will end up in a performance difference. To increase the accuracy of this scientific evaluation, in this paper we will establish a framework where real-world trials and simulations co-exist. Sharing knowledge between them, we will fine-tune a full-stack network simulator, enhancing the accuracy of our results and giving us the leverage of more precise experimentation later. For our trials, an open-source testbed will be used, consisting of single-board devices equipped with different wireless Network Interface Controllers (NICs) designed to be IEEE 802.11p compliant. The simulated results will be acquired using the VEINS network simulator [8]. This is a vehicular networking framework based on Omnet++ [9] and is compliant with the IEEE 802.11p/Wireless Access in Vehicular Environments (WAVE) standards.

The rest of the paper is organized as follows. In Sec. II, the hierarchical framework and the interrelationship between simulations and trials are described. The different entities of this framework, the relationship between them and the practical issues of the validation process are discussed. In Sec. III, the procedure to fine-tune VEINS is analyzed,

starting from an initial calibration isolating various parameters and moving towards a full-stack system optimization. Individually analyzing each scenario, their fine-tuned performance evaluation is examined in Sec. IV. Finally, Sec. V concludes the paper and provides future research avenues.

II. HIERARCHICAL FRAMEWORK: TRIALS AND SIMULATIONS

Consider an ITS consisting of a number of CAVs and Road Side Units (RSUs) on a road network. Different kinds of data are exchanged with respect to the safety- or infotainment-related applications and services running on the ITS. For example, WAVE Short Messages (WSMs) [10], are safety-critical messages encapsulating core information about CAVs (e.g. position, velocity, size, etc.). These messages are either broadcast every 100 ms or are triggered to announce road hazards. They are relatively short (~300-800 B) and a high delivery rate and a low one-hop end-to-end delay are regarded as their Quality-of-Service (QoS) constraints.

Apart from WSMs, safety-critical applications will be key in future ITSs. For example, video-assisted overtaking or traffic monitoring applications are tested on CAVs [11]. These applications require the transmission of video streams encapsulated within UDP packets. Increased data rate and low jitter are their main QoS requirements, with a more forgiving bit error rate (BER) performance due to the adoption of Forward Error Correction (FEC) codes and the new generation higher efficiency video encoders.

Following the content-related QoS requirements, each application behaves differently under various physical environments. Three different environments can be found in vehicular networks (urban, suburban, rural) [12]. For example, urban environments are affected by blockages from the buildings, significantly attenuating the signal. Urban canyon behaviors can be introduced, under specific circumstances, waveguiding the signal. On the other hand, foliage is the main form of blockage in rural areas while vehicles tend to move faster introducing a higher Doppler Shift effect. Each scenario should be approached differently when simulated, adapting accordingly the various channel model characteristics.

A. Co-operation and Co-existence of Trials and Simulations

Cooperation between trials and simulations is mandatory to increase the accuracy of a system performance validation. Exchanging information between them can enhance the outcome, maximize the time utilization and minimize the cost. Establishing a framework between the *reality-of-interest*, i.e. the part of the real world (e.g. a city, a neighborhood or a road) that we are interested in investigating, the *trials* conducted and the *simulation models* used, will help us better understand the requirements and the limitations of each one. The interrelationship between these three entities can be found in Fig. 1.

The assessment of the simulation accuracy can be divided in two phases, the *validation* and the *verification* [13]. The verification (Fig. 1 – a) is the confirmation that a model
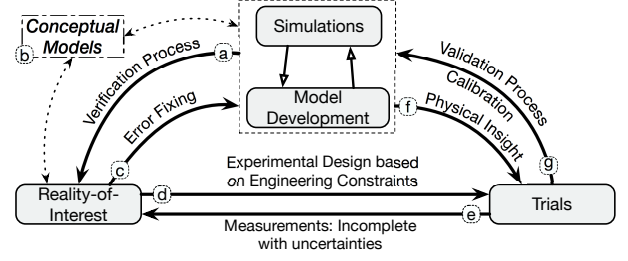

Fig. 1. The interrelationship between the experiments, the simulations and the real world as well as the processes that connect them.

is correctly implemented and reflects the real world. This is confirmed by conceptual models – abstractions of a system that characterize and standardize a network function, e.g. the OSI model (Fig. 1 – b). Verifying a model, the existing errors can be determined and fixed to assure that it matches specifications and assumptions with respect to the initial concept (Fig. 1 – c). For this work, we will not discuss the concept of the model verification for VEINS any further. For more, we kindly refer the reader to [14].

A trial-based evaluation is limited by the engineering constraints (Fig. 1 – d). For example, a NIC might support decreased transmission power compared to the standard specifications, limiting the operational range of a device. What is more, trials suffer from uncertainties (e.g. the attenuation of the signal with respect to the weather is unpredictable, not easily measured and environment-dependent) affecting the reliability and the validity of the experiment. The replicability of the experiment is also a big concern. All roads are not the same and the devices have different specifications, so replicating an experiment is difficult. This leads to incomplete results (Fig. 1 – e) as it is impossible to validate all possible combinations.

The validation of a theoretical model assesses the fidelity that a model reproduces the state and behavior of the real world (Fig. 1 – f). Verification usually precedes the validation of a model. The models can be validated with simple experiments isolating the external factors affecting the performance. To achieve a meaningful representation of the real world, a simulator should be fine-tuned at first (Fig. 1 – g) using inputs from measured results (e.g. path loss exponent) or applying weights at the output to minimize the divergence error. A direct comparison between the absolute values of the experimental results should be avoided. The validation process should focus on the trends of performance between the different experiments (e.g. both the simulated and trial-based results have a relative degradation when one parameter is changed).

B. Hierarchical Validation of a Simulation Model

Differences between trial and simulated results can arise from various reasons. Typical examples are *measurement errors* (e.g. calibration errors, noise or data acquisition methods, etc.), *formulation errors* (e.g. incorrect channel models) or *numerical errors* (e.g. overflow of integers, subtraction of
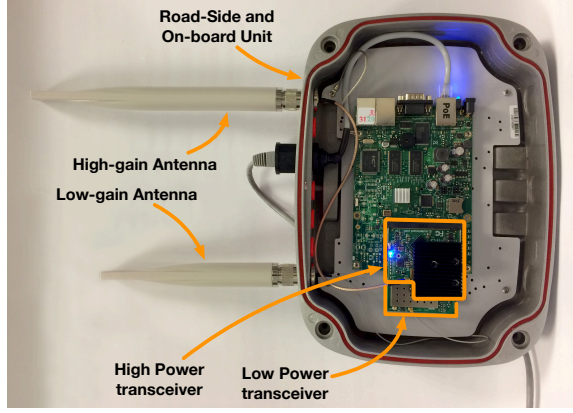
Fig. 2. Low-latency Linux Kernel implementation of IEEE 802.11p/DSRC units. The figure shows units designed for RSU and OBU deployment.

floating points, etc.). Two different error categories exist. Firstly, the *random* errors, affecting the relative precision of a model or a measurement. Secondly, the *systematic* errors affecting the absolute value of a result, being repeatable though and therefore, easily predicted.

In order to achieve the required level of accuracy, the validation procedure should be carried out throughout the entire development process following a hierarchical approach [15]. Fragmenting the problem into smaller entities and solving them individually, the necessary level of precision can be achieved without increasing the complexity. Generally, the trials should test crucial features of the simulation models, such as the impact of the considered assumptions or the simplifications. On the other hand, simulations allow incremental validation towards a "real-world-like" system.

III. Fine-Tuning VEINS Network Simulator

Designing a real-world ITS solution, all the above should be taken into account as well as the initial conditions, the boundaries and the trends of the performance. Isolating the characteristics that disruptively affect the performance we can isolate the systematic and the random errors approaching an "ideal-like" system. Using a network simulator, we can further validate different scenarios based on our initial configuration. To do so, the fine-tuning process of VEINS requires a detailed study on the available hardware, the performance metrics used, an insight into how the simulator operates and what are the differences of the real-world. This study will be discussed in the next subsections.

A. Description of the Experimental Setup

For our experimental validation, we prototyped an open-source IEEE 802.11p/DSRC testbed (Fig. 2) meeting the following requirements:

- *Easily customisable.*
- *Low cost* with the potential to be massively deploy later.
- *Open-source operating system* providing enough flexibility for future developments.
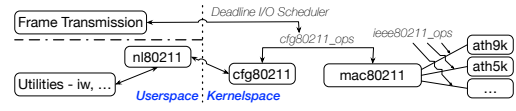

Fig. 3. Linux Kernel Modules that have been modified to implement a IEEE 802.11p/DSRC stack.

- *Dual-operation as RSUs and On-Board Units (OBUs).*
- *Weatherproof.*

To meet these requirements, each device was equipped with a Mikrotik RB433 single-board computer (CPU 300 MHz, 64 MB RAM, 64 MB storage space, x3 Ethernets, x3 MiniPCI slots) [16]. Two IEEE 802.11a NICs were used for redundancy in the wireless links. The first one was a Mikrotik R52H [17], regarded in this work as a low-power (LP) NIC, with transmission power of up to 25 dBm and connected to a dipole antenna of 7 dBi gain at 5.9 GHz. The second model was a Mikrotik R5SHPn [18], operating as the high-power (HP) transceiver in our experiments (29 dBm maximum transmission power). This NIC was connected to a 9 dBi-gain dipole antenna.

A low-latency OpenWRT Linux distribution was used[1] as the operating system. The different Atheros chipsets of each transceiver (AR5414 for the LP and AR9220 for the HP) required the use of two different Atheros drivers (ath5k for the LP and ath9k for the HP). Both were modified accordingly to enable IEEE 802.11p compatibility. The Linux kernel modules that we modified have been summarized in Fig. 3. The software modules *cfg80211* and *nl80211* act as interfaces between the user and kernel space, *mac80211* is the general driver framework, and *iw* is the NIC configuration utility. Furthermore, *cfg80211_ops* and *ieee80211_ops* define the operations and the callbacks between the different blocks. The Outside the Context of a BSS (OCB) mode was enabled in the MAC layer, allowing the NICs to operate without being associated with an access point and the *iw* utility was modified accordingly to include the new commands for using OCB mode. Finally, the 5.9 GHz band was added in the regulatory domain.

Each device can be connected with a GPS dongle via its USB interface. Finally, a beaconing interface was developed that was able to acquire viable vehicle information (e.g. length, position, emissions, etc.) from an Engine Control Unit (ECU) and broadcast them to surrounding devices.

B. Initial V2X Calibration Scenario

Consider a scenario with an ideal channel (no reflections, free-space path loss) between two ideal stationary vehicles (isotropic antennas, zero hardware attenuation). As this ideal system does not exist in the real world, the experimental setup was initially demonstrated inside an anechoic chamber (8.5 m×4.5 m×5 m) using both the HP and LP transceivers. A number of experiments was conducted at a distance of ~6.5 m for each transceiver and each MCS using

---

[1]OpenWRT Barrier Breaker Release no. 14.07 - https://openwrt.org/

two devices, one acting as a RSU and the other as an OBU. A UDP data stream, transmitted from the OBU to the RSU and generated using iPerf traffic generator[2], and a periodic beacon every 100 ms, were used to saturate the channel. As known, the network level performance is affected by the signal-to-interference-plus-noise ratio (SINR) and the sensitivity levels for each MCS. The SINR degrades based on the disruptive characteristics of the channel (e.g. distance attenuation, multipath, antenna misalignment, etc.) and the devices (e.g. thermal noise, etc.). Using an anechoic chamber, we conducted the experiments under near-optimal conditions (SINR greater than the sensitivity level) and therefore the optimal performance was achieved.

The same scenario was designed in VEINS as well. By partitioning the design process into smaller problems, we managed to achieve the required level of similarity. Firstly, we considered the IEEE 802.11p Physical Layer (PHY) frame, which consists of three fields [19]: (i) The *Preamble* marks the beginning of the PHY frame, is responsible for the appropriate antenna selection and corrects the timing and frequency offsets, (ii) the *Signal* field (SIG) specifies the frame rate and length and (iii) The *Data* field consisting of the Physical layer Service Data Unit (PSDU) that encapsulates the MAC frame, the Physical Layer Convergence Procedure (PLCP) Service, and a Tail field. The Data field can also be padded with extra bits so its length is a multiple of the coded bits in an OFDM symbol. The above are transmitted using BPSK ¹/₂ Modulation and Coding Scheme (MCS). The length of each field can be found in Tab. I.

In VEINS, the duration of the Preamble and SIG is controlled by the parameter *preambleDuration*, while the bit length of the Data field is set equal to *headerBitLength* (see Fig. 4). The simulated PHY bitrates are governed by the wireless interface operational mode – namely, *opMode* that has been set to "*p*", in this case. Each MCS should be manually configured for each individual simulation and matched with the appropriate PHY bitrate using pairs of the simulation parameters *bitrate* and *modulation*. Other PHY parameters that should be set within VEINS are the channel *bandwidth*, the *carrierFrequency*, the *antennaType* (*ConstantAntennaGain* in this case) and the *gain*.

The multi-channel operation introduced in the WAVE 1609.4 standard [10] was not considered in order to identify the maximum performance under saturation conditions. Furthermore, the RTS threshold – namely *rtsThreshold*, was set to a value greater than the frame size. This ensured that the RTS/CTS procedure was disabled. The MAC layer backoff times are drawn from a Contention Window (CW) starting from $CW_{\min}$ (*cwMinData* and *cwMaxData*). The values chosen for our setup (Tab. I) where proven to be optimal for vehicular communications [20]. The length of the MAC TX queue size is capped by the driver (in the ath5k case), so the same value was considered in the simulation as well (*maxQueueSize* parameter within VEINS).

TABLE I
SIMULATION AND EXPERIMENTAL PARAMETERS.

Parameter	Value	
Experiment/Simulation Time	10	s
Carrier Frequency	5.9	GHz
Bandwidth	10	MHz
MTU	1500	kB
UDP Packet Length	8192	kB
Beacon Length	500	B
Beacon Interval	100	ms
Preamble Duration	32	μs
SIG Duration	8	μs
PLCP Service Length	16	bit
Tail Length (Data Field)	6	bit
$CW_{\min}, CW_{\max}$	$[15, 1023]$	
TX MAC queue size	50	
Background Noise	$\mathcal{N}(-110, 3)$	dBm
Connector and Cable Losses	3	dBm

The VEINS parameter *sentInterval* sets the interval between the generation of two consecutive UDP packets. In order to saturate the channel, a very precise interval was chosen for each MCS. Suitable values were found with a trial and error method to fully utilize the channel without having packets discarded from the MAC TX queue. All the simulation and experimental parameters can be found in Tab. I and Sec. III-A.

The results for the above calibration scenario are shown in Figs. 5 and 6. The central rectangle is the interquartile region (IQR) between the first and the third quartile, while the line within represents the median. The whiskers constitute the maximum and the minimum values and the asterisks show the outliers. A value is regarded as an outlier if it outside $\pm 2.7\sigma$ (99.3% percentage coverage of the normally distributed samples).

Fig. 5 shows the reception throughput measured at the transport layer. As mentioned in Sec. II-B, a meaningful comparison should focus on the trends. Therefore, a trend can be seen in the performance whereby the simulation results are slightly better for some MCSs (e.g. QPSK ³/₄) while for others (e.g. BPSK ¹/₂) they are almost identical. The deviation in the mediam is of the order of up to ~0.5 Mbps for the HP device and ~1 Mbps for the LP transceiver. Overall, we observed that the LP transceiver has a worse throughput performance (median values decreased by ~5%) compared to the HP one; following the same trend for all MCSs. This difference is due to the operation of the different drivers used.

Fig. 6 compares the inter-arrival jitter performance. The jitter, as defined in RFC 1889 [21], is the statistical variance of the inter-arrival time between packets. Comparing the absolute values of the results, we see a huge difference between the trials and the simulations. However, comparing the relative variation in the jitter performance for the different MCSs, it is shown that in both cases, the results follow a similar trend starting with an increased jitter for the lower MCSs, and having a better performance as the bitrate

Initial Calibration Process

Real World Representation

Design Phase | *Testing Phase* | *Design Phase* | *Testing Phase*

PHY Layer
Preamble Duration
Header Bit Length
Operational Mode
Bitrate   MCS

MAC Layer
MTU length
Beacon Length
Beacon Interval
TX Queue Size
Contention Window
MAC Address: AUTO

Applications
Safety Related
Beacons   UDP Streams

Link Layer
ARP requests
IP Address: AUTO

Transport Layer
UDP Packet Length
Transmission Interval

Initial Calibration within the Anechoic Chamber

Random Errors   Systematic Errors

Knowledge from the Initial Calibration Phase

Scenarios
Complex Scenarios

Real Device Char.
TX Power
RX Sensitivity
SINR Threshold
Electrom. Field Effects   Cosmic Background Noise
Thermal Noise
Background Noise

Channel Model
Propagation Model
Path Loss Model
Obstacle Loss Model
Signal Representation

Comparison with Experimental Results

Random Errors   Systematic Errors

Feedback Line: Step back to fix the errors or apply weights to the outcome.
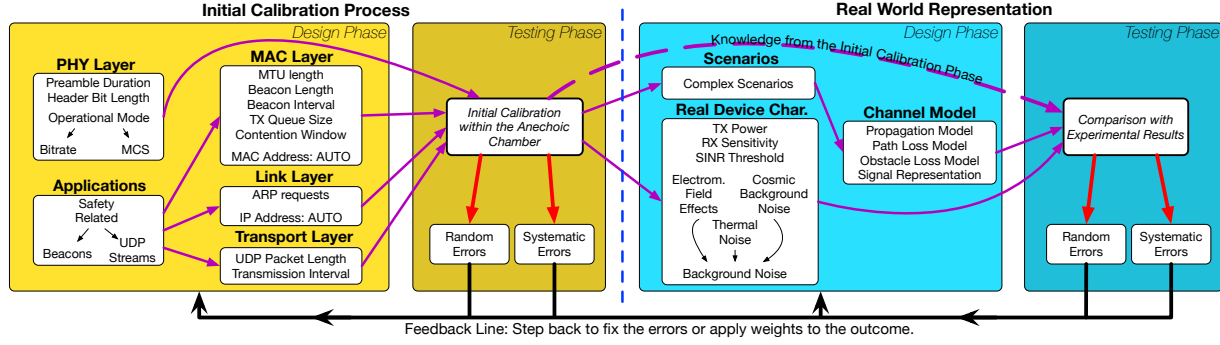
Fig. 4. Hierarchical Validation Process. Each entity is individually fine-tuned to achieve high-fidelity results after the calibration of the entire system. The fine-tuning is a two-phase process starting from an initial calibration and moving to a more complex real-world representation.

is increased. To that extent, the jitter values measured with VEINS have been multiplied by 56000 in order to get the same order of magnitude with the ones obtained by the LP and HP transceivers. The huge difference in the absolute values was somehow expected as our devices are built upon a single-core CPU, which executes tasks with the same priority according to the Linux *Deadline I/O Scheduler* – thus the CPU cannot fetch/push data streams towards the transceivers at a constant I/O rate. Since in VEINS this issue is not present, the simulated and measured jitter performance may vary significantly.

With regards to the framework introduced in Sec. II, we fragmented our system into smaller problems, fine-tuning individually each of them and managed to build an ideal subsystem able to achieve similar performance with minor deviations between the trial and the simulated results. In the next sections, we will introduce more complex entities such as the real device profiles and the channel behavior approaching a better representation of the real world.

C. Moving towards a Realistic Representation of the World

The surrounding environment plays a key role in the performance of vehicular networks. Each environment has its own characteristics, therefore a theoretical analysis is required for the signal degradation due to the different channels. The channel behavior within VEINS lies beneath the *physical medium model*. This model is further split into different submodules (Fig. 4).

At first, the propagation model (*propagationType*) describes the propagation time within the channel. Considering *constantSpeed* as our propagation model, the propagation time is proportional to the distance travelled. To accurately represent the signal and its fluctuation, a *DimensionalAnalogModel* was utilized, meaning that the signal power deviation is represented over both time and frequency.

The long-term signal degradation in the real-world depends on the distance, the carrier frequency, the device positioning, etc. In VEINS, the simulation parameter

pathLossType describes the path loss model that is responsible for computing the power reduction based on the traveled distance, the velocity factor, the carrier frequency and the path loss exponent for each environment. The short-term signal degradation, affected by the multipath distortion caused by the surrounding buildings can be described within VEINS using small-scale fading models (e.g. Nakagami, Rician, etc.) and fine-tuning their individual parameters (shape-factor for Nakagami, K-factor for Rician, etc.) accordingly. An obstacle loss model can be added to an existing path loss model. If so, the parameter *ObstacleLossType* specifies the material absorption when a ray is traveling within an object. The physical obstacles can be listed using an XML file processed by the *physical environment model* within VEINS.

D. Integration of Real Device Profiles in VEINS

After configuring the parameters related to the signal propagation, it is key to consider the different characteristics of each simulated NIC. Unfortunately, commercial-off-the-shelf (COTS) devices have frequently to be treated as a "black box" as many of their physical level performance characteristics are unknown and not easily measurable. Therefore, we will base some of our simulation parameters on speculations based on the datasheet of each considered transceiver.

At first, the SINR can fluctuate from random effects such as the thermal noise, the cosmic background noise, electromagnetic field effects, etc. These effects are not predictable and do not particularly come from a specific source to be isolated. VEINS represents this signal variation with a background noise model (*backgroundNoiseType*), configured using the *backgroundNoise* parameter following a Normal distribution. Also, the cables and the connectors in a system introduce a systematic attenuation described as the *systemLoss* that was measured within our laboratory.

According to the manufacturer datasheet, we have access to the transmission power given at 20 MHz of channel bandwidth, for each MCS. However, the IEEE 802.11p bandwidth is equal to 10 MHz. From the energy-per-symbol-to-noise
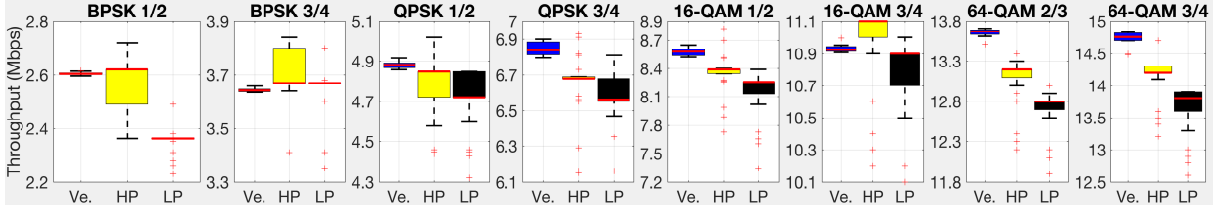
Fig. 5. Values of throughput obtained from VEINS (initial calibration), the HP and LP transceivers, for different MCSs.
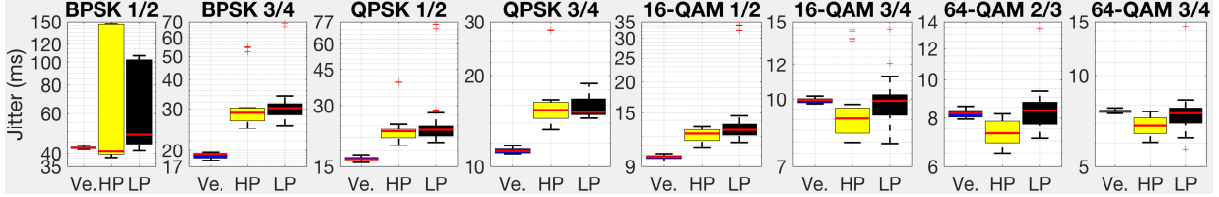

Fig. 6. Values of jitter associated to the UDP stream. Obtained from VEINS (initial calibration), the HP and LP transceivers for different MCSs.

power spectral density equation it follows that:

$$\frac{E_s}{N_0} = \frac{C}{N} \frac{B}{f_s} \tag{1}$$

where E_s is the energy per symbol, N_0 is the noise power, C/N is the carrier-to-noise ratio, B is the channel bandwidth and finally f_s is the symbol rate. In our case, the only non-constant variable is B. Therefore, for a 10 MHz channel, the E_s/N_0 ratio is expected to be twice as much as that measured using a 20 MHz channel. As N_0 is measured per unit of bandwidth (per MHz), it follows that E_s is doubled. Finally, knowing the number of bits per symbol of each MCS, we can infer the maximum transmission power for a 10 MHz channel. These values are summarized in Tab. II.

With regards to the sensitivity of the receiver, from the Minimum Operational Sensitivity (MOS) relation, we know that:

$$\text{MOS} = \frac{SINR_{\text{thr}} \, k \, T_\alpha \, B \, (NF)}{G_{\text{rx}}} \tag{2}$$

where $SINR_{\text{thr}}$ is the minimum SINR needed to process (not just detect) a signal, NF is the noise figure, k is Boltzmann's Constant, T_α is the effective noise temperature referred at the input of the receiver, and G_{rx} is the isotropic antenna gain. Obviously, $SINR_{\text{thr}}$ depends not just on the NIC but also on the MCS in use. Authors in [22] measured the $SINR_{\text{thr}}$ under a V2I scenario for two different antenna heights. Their lower antenna configuration was very similar to ours, so their $SINR_{\text{thr}}$ results will be utilized for our scenarios. Knowing the $SINR_{\text{thr}}$, the only variable in (2) is B. Therefore, halving B, the MOS will be doubled. Finally, the antenna gain values were taken directly from the manufacturer datasheet. All the values are presented in Tab. II.

IV. PERFORMANCE EVALUATION AND MACROSCOPIC VIEW

A. Scenarios and channel analysis

In Sec. III, the full-stack calibration process for VEINS was discussed. As mentioned, the different environments signif-

TABLE II
SIMULATION PARAMETERS BASED ON THE MANUFACTURER DATASHEET.

Modulat.	TX power		RX sensitivity		$SINR_{\text{thr}}$ [22]		Units
	LP	HP	LP	HP	1/2 MCS	3/4 MCS	
BPSK	27	28	−93	−93	10	15	dBm
QPSK	26	27	−85	−88	10	15	dBm
16-QAM	25	26	−80	−84	17	17	dBm
64-QAM	24	24	−73	−80	20	25	dBm

icantly change the behavior of the system. To that extent, three different scenarios were designed and evaluated. As in the initial calibration scenario, we considered one RSU and one OBU devices, stationary during the experiments having their performance being evaluated for different distances, MCSs and for both the HP and LP transceivers. Again a UDP stream and a periodic beaconing transmitted from the OBU to the RSU saturate the channel to evaluate the network throughput (as described in Sec. II and III-B). The two different transceiver configurations were simulated within VEINS using the parameters from Tabs. I and II based on the analysis preceded in the previous sections.

The first scenario (see Fig. 7 – A) is an urban road with buildings on both sides. The devices are positioned on the pavement and there is always a Line-of-Sight (LOS). The buildings surrounding the devices cause multipath distortion. However, since the devices are always in LOS and the experiment was conducted at a relatively short distance, a Rician fading model was considered with a K-factor $k = 3.36$ dB and a path loss exponent $\alpha = 2.3$ [23].

The second scenario (see Fig. 7 – B) is a suburban area on a bent and sloppy road with foliage in between the devices and a few buildings on one side of the road. LOS existed between the devices for the first ~50 m. For the rest of the experiment, the RSU was hidden behind the road slope and the vegetation. This scenario can be split into two different subscenarios. Up to 50 m, we considered a Rician fading
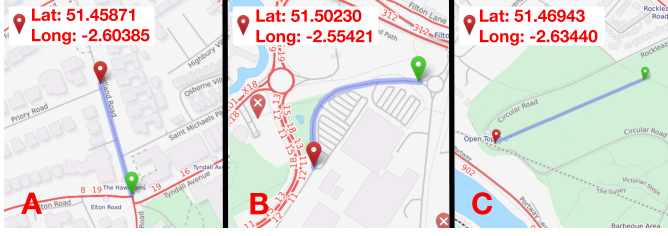
Fig. 7. The three different scenarios were conducted around the city of Bristol, UK. a) the urban, b) the suburban, c) the rural scenario.

model with $k = 2.45$ dB and $\alpha = 2.3$ [23]. For the NLOS part, we refer to Rayleigh fading model with $\alpha = 2.5$.

The third scenario refers to a rural environment (see Fig. 7 – C). Both the RSU and OBU are always in LOS, and no high buildings or other objects were surrounding the devices apart from some foliage. Therefore, the impact of multipath was minimum. As such, a Rician fading model was considered, with $k = 8$ dB and $\alpha = 2.2$ [12]. For this scenario, only the HP transceiver was used.

B. Performance Evaluation

During the experiments carried in scenarios A, B and C, the RSU and OBU were fitted on a tripod at ~1.8 m height. A consistent setup has been simulated in VEINS. During both the trials and simulations, each performance metric we measured is the result of an average of multiple experiments. The ARP probe was disabled by manually inserting the addresses of the devices in their respective ARP tables. Due to limited space, the most meaningful results for each distance and MCS will be shown while the rest will be described within the text.

With regards to the urban scenario, Fig. 8 shows the communication throughput that can be sustained by the OBU as a function of distance between RSU and OBU, for each MCSs. For lower modulations (BPSK, QPSK) and all distances, the same trend and performance were seen as in the calibration process (Fig. 5). Again, the difference in the performance between the transceivers, not observed within VEINS, is due to the different drivers used. Increasing the MCS and the distance separating the devices, even though the median values remain similar to what shown in Fig. 5, the introduced multipath distortion leads to a larger number of outliers. For 16-QAM 1/2 and 3/4, it was seen that as the distance is increased, the SINR drop starts being observed within VEINS for the LP configuration (e.g. 110 m) having a different behavior compared to the trials. Finally, for 64-QAM 2/3 and 3/4, the signal received from the LP transceiver within VEINS is significantly attenuated. The performance degradation is about 1 Mbps compared to the calibration scenario (50 m) reaching up to 3 Mbps at 110 m.

Fig. 9 shows the communication throughput measured in the case of the suburban scenario, for different MCSs. For distances of 30 m and 60 m, we observe that the results follow the same trend as in the urban case (see Fig. 8). In particular, despite the RSU being hidden after 50 m due

to the road slope, the overall communication throughput was not severely impacted. For greater distances and the HP scenario, VEINS behaves slightly worse compared to the actual device. However, for the LP scenario, the actual device achieved less throughput compared to the simulated result. Especially for 64-QAM 2/3 and 3/4 and a distance of 200 m, our LP transceiver achieved zero throughput during the trials whereas the VEINS result is around 3 Mbps and 1.2 Mbps respectively. This is due to the BER calculation within VEINS that is approximated based on a Gaussian error function not exactly reflecting the reality.

Fig. 10 refers to the rural scenario. Again, for lower modulation schemes (BPSK, QPSK), the same trend was observed as before. For higher MCS, VEINS again exhibits a sharp performance degradation when the distance increases. This is clearer at 64-QAM 2/3 and 3/4 where VEINS outperforms the trial performance at 550 m as expected from the trend observed. However, it is significantly worse at 700 m.

V. CONCLUSIONS, LESSONS LEARNED AND FUTURE WORK

In this paper, we described the process to follow in order to calibrate a full-stack network simulator for vehicular communications. In our study, we considered VEINS as the network simulator of interest. At first, we introduced an initial calibration process where an "ideal-like" scenario was evaluated. Two different transceivers were used to demonstrate the difference emanating from dissimilar hardware and how it can impact the experimental results. Specifically, we focused on result trends and identified the reasons that they exist.

To that extent, a trend in the performance was shown for both the real and the simulated scenario. For some MCSs (e.g. BPSK 1/2, QPSK 1/2, etc.) the network throughput is almost identical whereas for the others there is a slight deviation of up to ~1 Mbps (e.g. 64-QAM 3/4, etc.). The different drivers and hardware introduced a dissimilarity between the HP and the LP transceivers, something not observed within VEINS as the differences in the hardware cannot be easily simulated. A similar trend was seen in the jitter performance evaluation as well. The huge difference between the absolute values due to the Linux scheduling algorithm serves as a proof of capacity that the direct comparison between simulated and actual trial results should be avoided.

After the initial calibration, VEINS was further fine-tuned to achieve high-fidelity results for more complex scenarios. Three different vehicular environment scenarios were evaluated with respect to the network throughput. The trends identified, showed that for the HP transceiver VEINS has a sharper performance degradation when the distance is increased whereas, for the LP one, it behaves smoother. As discussed, some of the simulation parameters are based on various assumptions and therefore this difference in the performance is introduced. Finally, we concluded that the proposed tuning process allows VEINS to deliver high-fidelity simulation results of IEEE 802.11p/DSRC links,
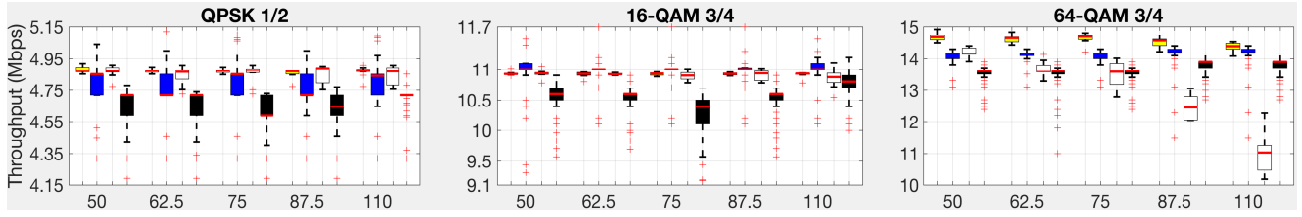
Fig. 8. Values of throughput as a function of the distance between RSU and OBU. Results refer to an urban scenario. Each quartet represents the results for a single position with the order (from left to right): 1) VEINS-HP, 2) Trials-HP, 3) VEINS-LP, 4) Trials-LP.
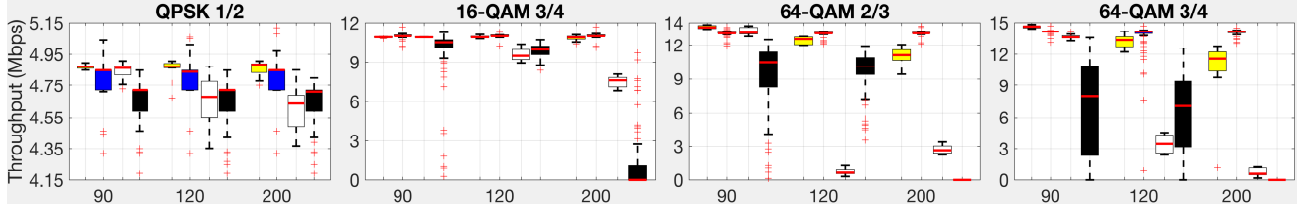

Fig. 9. Values of throughput as a function of the distance between RSU and OBU. Results refer to a suburban scenario. Each quartet follows the same order as in Fig. 8
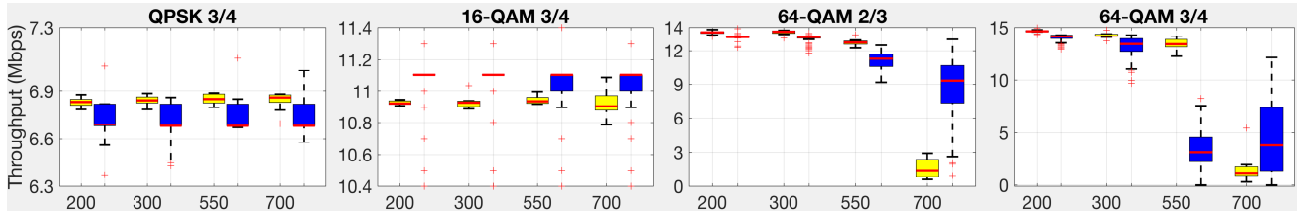

Fig. 10. Values of throughput as a function of the distance between RSU and OBU. Results refer to a rural scenario. Each boxplot pair is: 1) VEINS-HP, 2) Trials-HP.

which will be pivotal in city-scale simulation scenarios – scenarios impossible to be calibrated otherwise.

REFERENCES

[1] J. Levinson *et al.*, "Towards Fully Autonomous Driving: Systems and Algorithms," in *IEEE Intel. Veh. Symp. IV*, Jun. 2011, pp. 163–168.

[2] P. Demestichas *et al.*, "Intelligent 5G Networks: Managing 5G Wireless/Mobile Broadband," *IEEE Veh. Technol. Mag.*, vol. 10, no. 3, pp. 41–50, Sep. 2015.

[3] A. Tassi *et al.*, "Modeling and Design of Millimeter-Wave Networks for Highway Vehicular Communication," *IEEE Trans. Veh. Technol.*, Aug. 2017.

[4] C. Han, M. Dianati, and M. Nekovee, "Effective Decentralised Segmentation-based Scheme for Broadcast in Large-scale Dense VANETs," in *IEEE WCNC 2016*, Apr. 2016, pp. 1–6.

[5] C. Sommer, O. K. Tonguz, and F. Dressler, "Adaptive Beaconing for Delay-Sensitive and Congestion-aware Traffic Information Systems," in *Proc. IEEE VNC 2015*, Dec. 2010, pp. 1–8.

[6] F. A. Teixeira *et al.*, "Vehicular Networks Using the IEEE 802.11p Standard: An Experimental Analysis," *Vehicular Communications*, vol. 1, no. 2, pp. 91 – 96, 2014.

[7] M. Boban, J. Barros, and O. Tonguz, "Geometry-Based Vehicle-to-Vehicle Channel Modeling for Large-Scale Simulation," *IEEE Trans. Vehic. Techn.*, vol. 63, no. 9, pp. 4146–4164, Nov. 2014.

[8] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Trans. Mobile Comput.*, vol. 10, no. 1, pp. 3–15, Jan. 2011.

[9] "Omnet++ Discrete Event Simulator." [Online]. Available: https://omnetpp.org

[10] S. A. M. Ahmed, S. H. S. Ariffin, and N. Fisal, "Overview of Wireless Access in Vehicular Environment (WAVE) Protocols and Standards," *Indian Journal of Science and Technology*, vol. 6, no. 7, 2013.

[11] E. Belyaev *et al.*, "The Use of Automotive Radars in Video-Based Overtaking Assistance Applications," *IEEE Trans. Intell. Transp. Syst*, vol. 14, no. 3, pp. 1035–1042, Sep. 2013.

[12] S. Zhu *et al.*, "Probability Distribution of Rician K-Factor in Urban, Suburban and Rural Areas Using Real-World Captured Data," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 7, pp. 3835–3839, Jul. 2014.

[13] R. G. Sargent, "An Introductory Tutorial on Verification and Validation of Simulation Models," in *2015 Winter Simulation Conference (WSC)*, Dec. 2015, pp. 1729–1740.

[14] F. Klingler, F. Dressler, and C. Sommer, "Ieee 802.11p Unicast Considered Harmful," in *IEEE VNC 2015*, Dec. 2015, pp. 76–83.

[15] P. J. Roache, *Verification and Validation in Computational Science and Engineering*. Hermosa Albuquerque, NM, 1998, vol. 895.

[16] "Mikrotik RB433 Data Sheet." [Online]. Available: https://mikrotik.com/product/RB433

[17] "Mikrotik R52H Data Sheet." [Online]. Available: https://mikrotik.com/product/R52H

[18] "Mikrotik R5SHPn Data Sheet." [Online]. Available: https://routerboard.com/R5SHPn

[19] A. Abdelgader and W. Lenan, "The Physical Layer of the IEEE 802.11p WAVE Communication Standard: The Specifications and Challenges," in *Proc. of WCECS 2014*, vol. 2, Oct. 2014.

[20] R. Reinders *et al.*, "Contention Window Analysis for Beaconing in VANETs," in *2011 7th International Wireless Communications and Mobile Computing Conference*, Jul. 2011, pp. 1481–1487.

[21] H. Schulzrinne *et al.*, "RTP: A Transport Protocol for Real-Time Applications," United States, 2003.

[22] A. Paier, D. Faetani, and C. F. MecklenbrÃd'uker, "Performance Evaluation of IEEE 802.11p Physical Layer Infrastructure-to-Vehicle Real-World Measurements," in *2010 3rd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL 2010)*, Nov. 2010, pp. 1–5.

[23] O. Renaudin *et al.*, "Wideband MIMO Car-to-Car Radio Channel Measurements at 5.3 GHz," in *2008 IEEE 68th Vehicular Technology Conference*, Sep. 2008, pp. 1–5.