POLITECNICO DI TORINO
Repository ISTITUZIONALE

VaN3Twin: the Multi-Technology V2X Digital Twin with Ray Tracing in the Loop

Original
VaN3Twin: the Multi-Technology V2X Digital Twin with Ray Tracing in the Loop / Pegurri, R., Gasco, D., Linsalata, F., Rapelli, M., Moro, E., Raviglione, F., Casetti, C.. - In: IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. - ISSN 1536-1276. - ELETTRONICO. - 25:(2026), pp. 16812-16827. [10.1109/TWC.2026.3685710]

Availability:
This version is available at: 11583/3003129 since: 2026-05-10T09:45:03Z

Publisher:
IEEE

Published
DOI:10.1109/TWC.2026.3685710

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright
IEEE postprint/Author's Accepted Manuscript

©2026 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collecting works, for resale or lists, or reuse of any copyrighted component of this work in other works.

(Article begins on next page)

17 November 2026

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

VaN3Twin: the Multi-Technology V2X Digital Twin with Ray Tracing in the Loop

Roberto Pegurri*, Diego Gasco†, Francesco Linsalata*,
Marco Rapelli†, Eugenio Moro*, Francesco Raviglione‡, Claudio Casetti†

*Department of Electronics, Information and Bioengineering, Politecnico di Milano, Italy
†Department of Control and Computer Engineering, Politecnico di Torino, Italy
‡Department of Electronics and Telecommunications, Politecnico di Torino, Italy

{roberto.pegurri, francesco.linsalata, eugenio.moro}@polimi.it
{diego.gasco, marco.rapelli, francesco.raviglione, claudio.casetti}@polito.it

***Abstract*—This paper presents VaN3Twin—the first open source, full-stack Network Digital Twin (NDT) framework for simulating the coexistence of multiple Vehicle-to-Everything (V2X) communication technologies with accurate physical-layer modeling via ray tracing. VaN3Twin extends the ms-van3t simulator by integrating Sionna Ray Tracer (RT) in the loop, enabling high-fidelity representation of wireless propagation, including diverse Line of Sight (LoS) conditions with a focus on LoS blockage due to other vehicles' meshes, Doppler effect, and site-dependent effects—e.g., scattering and diffraction. Unlike conventional simulation tools, the proposed framework supports realistic coexistence analysis across Dedicated Short Range Communication (DSRC) and Cellular-V2X (C-V2X) technologies operating over a shared spectrum. A dedicated interference tracking module captures cross-technology interference at the time-frequency resource block level and enhances Signal to Interference plus Noise Ratio (SINR) estimation by eliminating artifacts such as the bimodal behavior induced by separate LoS/NLoS propagation models. Compared to field measurements, VaN3Twin reduces application-layer disagreement by 50% in rural and over 70% in urban environments with respect to current state of the art simulation tools, demonstrating its value for scalable and accurate digital twin–based V2X coexistence simulation.**

***Index Terms*—V2X, Ray Tracer, Vehicular networks, Sionna RT, ms-van3t, ns-3, co-channel coexistence, NDT**

I. Introduction

In 2016, Toyota was the first car manufacturer to release commercial vehicles with Vehicle-to-Everything (V2X) communication capabilities, using Dedicated Short Range Communications (DSRC) based on IEEE 802.11p for the Japanese market [1]. Shortly thereafter, similar deployments appeared in the U.S. and Europe [2], [3], with General Motors and Volkswagen Group respectively. Today, the V2X market continues to grow, with an increasing number of vehicles capable of communicating with thousands of Road Side Units (RSUs) deployed along major highways [4]. From a technological perspective, however, the landscape has grown increasingly complex. Since the introduction of LTE-V2X in 3GPP Rel. 14 (2017), Toyota has steered the Asian V2X market toward Cellular-V2X (C-V2X) technology. In contrast, Europe continues to rely primarily on IEEE 802.11p for V2X communication. Nevertheless, major auto-makers such as Audi, Mercedes, and Ford have frequently announced plans for future deployments leveraging C-V2X technologies (i.e., LTE-V2X and the emerging NR-V2X standard) [5]–[7]. As a result, vehicles from different manufacturers—each potentially using distinct V2X technologies—will soon coexist on the road, creating significant challenges for interoperability, standardization, and coordinated deployment. In anticipation of this scenario, and as evidence of its growing relevance, many On Board Unit (OBU) and RSU manufacturers have started equipping their products with support for multiple V2X technologies. However, the coexistence of these technologies remains a significant challenge. In the United States, the FCC reallocated 45 MHz of the 5.9 GHz safety band to unlicensed services and reserved the remaining 30 MHz exclusively for C-V2X, with follow-up rules adopted in December 2024 instructing legacy DSRC installations to migrate. Instead, in Europe both C-V2X and IEEE 802.11p operate within the same 5.8–5.9 GHz frequency band, therefore the issue of co-channel interference has emerged as a critical area of research in the V2X field. Indeed, numerous researchers and vehicular consortia are actively investigating these challenges [8]–[10], which have emerged as some of the most pressing and widely discussed topics in the V2X domain. Predicting the performance of C-V2X and DSRC technologies in real-world coexistence scenarios is highly complex, especially in large-scale urban environments with fleets of vehicles equipped with either or both communication technologies. Although recent advancements in emulation frameworks for vehicular field testing [11] have significantly enhanced evaluation capabilities, conducting a comprehensive coexistence and co-channel interference study in real-world scenarios would require a large fleet of V2X-equipped vehicles capable of supporting one or more V2X Radio Access Technologies (RATs). However, current OBU offerings from vendors typically support only a single RAT at a time—e.g., either DSRC or C-V2X, but not both simultaneously. This limitation makes large-scale field testing for coexistence scenarios logistically, economically, and technologically challenging. As a result, advanced simulation tools are essential to analyze the complex, large-scale scenarios expected in future vehicular networks. Moreover, accurate interference modeling is crucial to support research on agile

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

spectrum utilization and effective spectrum sharing across heterogeneous radio technologies. Beyond traditional simulation goals, high-fidelity modeling is also a cornerstone of the emerging paradigm of Network Digital Twin (NDT) [12]. An NDT aims to create an accurate replica in real time of a network, requiring simulation models to maintain a high level of precision in all layers of the communication stack [13]. This requirement is particularly critical in the context of V2X wireless networks, where operating conditions evolve rapidly due to high mobility, complex propagation environments—especially in urban scenarios—and finally, multi-RAT coexistence. In such settings, and depending on the final scope of the simulation, precision becomes indispensable to ensure that the digital replica faithfully mirrors its physical counterpart. As is often the case, the more realistic the simulator, the more sophisticated and reliable the resulting models. However, to date, few vehicular simulation tools incorporate realistic channel models. For instance, simulators based on Network Simulator Version 3 (ns-3) [14] often rely on stochastic channel models that omit critical features, such as deterministic shadowing and multipath effect characterization and comprehensive propagation loss models. Another critical limitation, extremely relevant in V2V scenarios, is their failure to account for the impact of surrounding vehicles on the received signal. Furthermore, only a limited number of vehicular simulators have native support for full-stack implementation of multiple communication technologies, including IEEE 802.11p, LTE-V2X, and NR-V2X, and integration with the higher layers standardized by ETSI. These limitations have hindered the development of robust co-channel interference studies using vehicular simulations. Addressing these gaps is crucial for advancing our understanding of coexistence scenarios and improving the reliability of simulation-based research.

In this article, we present VaN3Twin[1], a framework based on the integration of Sionna Ray Tracer (RT) [15], an open source[2] ray tracing engine for high-fidelity wireless channel modeling, into the ms-van3t V2X simulator [16], resulting in a reliable and NDT-ready framework for full-stack and multi-RAT V2X simulations. VaN3Twin also has built-in support for modeling the coexistence of multiple communication technologies and intrinsic consideration for both static (buildings, trees, and roadside infrastructure) and dynamic objects (vehicles, pedestrians, etc.) in the simulated propagation environment. The ms-van3t simulator represents the state of the art among open source V2X simulators built on the ns-3 platform. It supports the design, simulation, and hardware-in-the-loop emulation of ETSI-compliant V2X applications. However, as ns-3 native simulator, ms-van3t traditionally relies on probabilistic models to simulate wireless channel effects. Although effective for large-scale simulations, these models often oversimplify physical-layer phenomena, limiting accuracy in dynamic and interference-prone environments. By using ray tracing, VaN3Twin overcomes this limitation, bridging the gap between high-level vehicular simulation and realistic modeling.

Our main contributions are summarized as follows.

- Realization of a first-of-its-kind open source, full-stack, and multi-RAT V2X NDT named VaN3Twin, overcoming the limitations of traditional stochastic and frequency-dependent channel models with ray tracing. While [13] focused exclusively on the integration of Sionna RT into the ns-3 loop, the present work introduces a complete V2X NDT, focusing on accurate simulation of scenarios of multiple V2X technologies sharing the same spectrum, thereby explicitly considering their coexistence;
- Development of an adaptable and modular framework, allowing the ray tracing component to run on a dedicated GPU-equipped server, while the rest of VaN3Twin may be executed on a separate machine. This division not only optimizes computational resources but also increases scalability for high-fidelity, city-scale V2X simulations;
- Validation of VaN3Twin as a NDT, performed against field measurements in real-world rural and urban testbeds. The results demonstrate significant improvements in fidelity at both the propagation and packet reception levels, with VaN3Twin reducing the disagreement with field data by 50% in rural and over 70% in urban environments when compared to state of the art simulation tools;
- Development of a unified coexistence modeling framework capable of simulating heterogeneous deployments with one or more RATs simultaneously within a shared spectrum portion. This includes a dedicated VaN3Twin coexistence module that incorporates the effects of co-channel coexistence into Signal to Interference plus Noise Ratio (SINR) estimation;
- Demonstration of improved SINR estimation accuracy in coexistence scenarios, showing that ray-based simulations produce smoother SINR distributions and eliminate the artificial bimodality introduced by the use of separate propagation models for Line of Sight (LoS) and Non-Line of Sight (NLoS) conditions.

The remainder of the article is organized as follows. Section II details VaN3Twin, highlighting its flexible architecture, the integration of a ray tracer, data exchange mechanisms, and mobility synchronization strategies. Section III validates the accuracy of the proposed framework's ray-based propagation model by comparing simulation results with real-world vehicular measurements in rural and urban environments, highlighting improved alignment in both Received Signal Strength Indicator (RSSI) values and packet reception performance. Section IV introduces the coexistence module for multi-RAT interference management in ms-van3t and VaN3Twin. It presents a simulation comparing SINR estimation under 3GPP-based and ray-based models in co-channel scenarios, with particular emphasis on NLoS conditions caused by vehicular obstructions, and evaluates the resulting impact on cumulative and punctual packet-level performance. The section also provides an analysis on VaN3Twin scalability. Section V reviews related work on V2X coexistence simulation and ray tracing integration. Finally, Section VI concludes the paper.

II. VaN3Twin Framework

This section presents the proposed architecture, which integrates Sionna RT into the existing ms-van3t workflow,

[1]https://github.com/DriveX-devs/VaN3Twin
[2]https://github.com/nvlabs/sionna

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

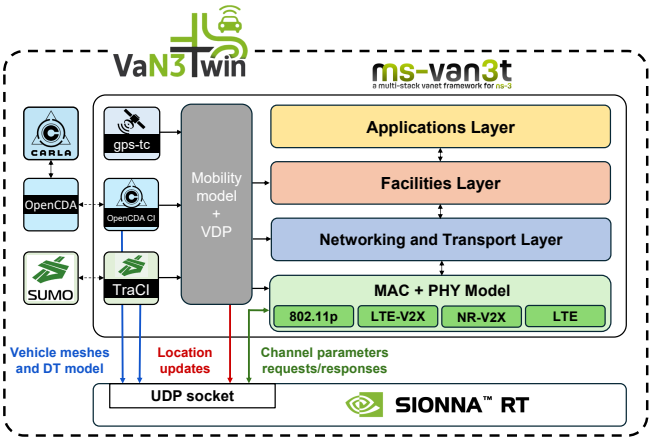


Figure 1: An overview of the proposed VaN3Twin framework: ms-van3t high-level architecture with Sionna RT integration.

as illustrated in Figure 1. Specifically, the figure displays the complete ms-van3t stack, spanning from the Application Layer to the Physical Layer—the latter expanded to show all the available RATs in ms-van3t. Figure 1 also outlines the mobility interfaces of ms-van3t, which can interact with either SUMO [17] or CARLA [18] mobility simulators or with pre-recorded vehicular traces. When using SUMO or CARLA, the mobility state of each vehicle is retrieved through standardized V2X message exchange interfaces, namely the Traffic Control Interface (TraCI) [19] for SUMO and OpenCDA for CARLA. Mobility data are eventually received by the ms-van3t Mobility model and subsequently distributed to the appropriate layers via the Vehicle Data Provider (VDP) module. Finally, at the bottom of Figure 1, the integration with Sionna RT is shown. This integration is achieved through data exchange via a UDP socket, enabling a fast and modular architecture. As a result, the ray tracer can operate either on the same local machine as ms-van3t or on a dedicated remote server with enhanced computational capabilities, as modularity is a key feature of VaN3Twin. Furthermore, by relying on ms-van3t and its underlying ns-3 engine, every module of VaN3Twin can be either replaced, extended, or re-implemented from scratch, including protocol stacks, channel models, mobility patterns, and performance metrics. Moreover, the employed ray tracing engine (Sionna RT) is open source and fully customizable. Through a dedicated adapter interface, it can be entirely substituted without affecting the integration logic. This design choice maximizes flexibility for secondary development—following standard ns-3 extension practices, such as models, helpers, and attributes—and facilitates the adoption of future improvements in ray tracing technology. Documentation[3] and APIs are provided to guide developers, thereby supporting secondary development and experimentation. As for the information exchange between ms-van3t and the ray tracer engine, the communication is structured into three distinct data flows, each identified by matching color codes in Figures 1 and Figure 2, where Figure 2 offers a more detailed representation of the interaction workflow. Specifically, the three data flows comprise: (i) the initialization of the Digital Twin (DT) including buildings and roadside infrastructure and the mapping of vehicle meshes, represented in the figures by blue arrows; (ii) the transmission of dynamic location updates, indicated in red; (iii) the request and response exchange of channel parameters, depicted in green. Each of these flows is analyzed in greater detail in the following subsections.

[3]https://www.nsnam.org/documentation

A. DT model initialization and vehicle mesh mapping

The ray tracer operates within a Three Dimensional (3D) scenario that includes both static and dynamic objects, each represented by a 3D mesh. The geometry of each mesh defines the physical shape of the object, while its material properties (e.g., metal, glass, or plastic) govern its electromagnetic characteristics, thereby influencing the interaction of electromagnetic waves with the object's surface. Static objects (such as buildings, road signs, and roadside units) remain fixed throughout the simulation and constitute our DT model. To support the simulation of vehicular traffic within the modeled environment, the scenario topology—including polygons for buildings—was extracted from OpenStreetMap (OSM) [20] and imported into the SUMO traffic simulator. This integration enables the construction of a complete DT model of the infrastructures in our scenario (as illustrated in Figure 2a). Conversely, dynamic objects represent mobile entities such as vehicles. For this purpose, various vehicle meshes (e.g., passenger cars, sedans, trucks, buses, etc.) were selected from the supported mobility simulators. Each vehicle mesh was further segmented into distinct components—i.e., vehicle body, lights, windows, rims, etc.—in order to assign appropriate radio material properties to each part. These vehicle mesh types were employed to model the dynamic elements of the simulation scenario. The process begins by defining the set of vehicles to be included in the overall simulation, defined as $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$. To distinguish vehicles participating in the simulation at time t, we define the subset of active vehicles $\mathcal{V}_{\text{active}}(t) \subseteq \mathcal{V}$. The set of 3D car meshes (shown in Figure 2b) is denoted by $\mathcal{M} = \{M_{v_1}, M_{v_2}, \ldots, M_{v_N}\}$, where each mesh $M_{v_n} \in \mathcal{M}$ represents the geometry and material properties of vehicle v_n. Each active vehicle $v_n \in \mathcal{V}_{\text{active}}(t)$ is equipped with two antennas, $(a_{v_n,\text{Tx}}, a_{v_n,\text{Rx}})$, one for transmission and

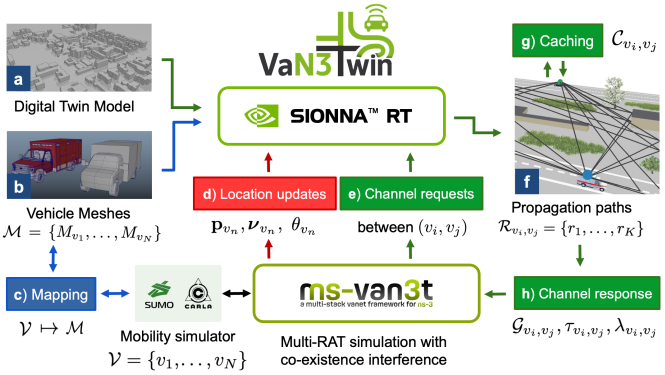


Figure 2: Flowchart of the entire VaN3Twin framework.

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

one for reception, mounted on the mesh M_{v_n} at positions:

$$\mathbf{p}_{\text{ant}}(t') = \mathbf{p}_{v_n}(t') + \mathbf{p}_{\text{dis}}, \tag{1}$$

where $\mathbf{p}_{\text{dis}} = [x_{\text{dis}}, y_{\text{dis}}, z_{\text{dis}}]$ denotes the displacement of the antenna relative to reference position $\mathbf{p}_{v_n}(t')$ of vehicle v_n at time $t' < t$, which is the time denoting the last update applied to the mesh M_{v_n}. Each vehicle $v_n \in \mathcal{V}$ is associated with a unique 3D mesh $M_{v_n} \in \mathcal{M}$ in the ray tracing scenario through a one-to-one mapping $\phi : v_n \mapsto M_{v_n}, \forall v_n \in \mathcal{V}$ (see Figure 2c). Both the DT initialization and the vehicle meshes mapping ϕ are initialized once at the beginning of the simulation and remain constant throughout, incurring no runtime computational cost. Whenever a vehicle v_n becomes active—i.e., $v_n \in \mathcal{V}_{\text{active}}(t)$—the mesh M_{v_n} is updated to reflect its physical state and its dynamic location updates.

B. Dynamic location updates

At simulation time t, each vehicle $v_n \in \mathcal{V}$ is characterized by a position vector $\mathbf{p}_{v_n}(t) = [x_{v_n}(t), y_{v_n}(t), z_{v_n}(t)]$, a velocity vector $\boldsymbol{\nu}_{v_n}(t) = [\nu_{x,v_n}(t), \nu_{y,v_n}(t), \nu_{z,v_n}(t)]$, and a heading angle $\theta_{v_n}(t)$. Moreover, at each simulation time step, information regarding the active vehicles—i.e., vehicle classes, vehicle IDs, positions, and orientations—is retrieved from the Mobility model of ms-van3t. To ensure a coherent and time-consistent representation of moving objects, dynamic location updates generated by the mobility simulator are transmitted from the Mobility model of ms-van3t to the ray tracer (Figure 2d). Specifically, whenever an updated mobility state becomes available for a vehicle v_n at time t, a location update message is sent to the ray tracer containing the tuple $\{\mathbf{p}_{v_n}(t), \boldsymbol{\nu}_{v_n}(t), \theta_{v_n}(t)\}$ related to vehicle v_n. Upon reception, this information is integrated into the ray tracing scenario and used to update the radio propagation environment by repositioning the vehicle meshes according to the received positions and orientations, and by setting the corresponding vehicle configurations accordingly. Thus, at each update, the variables $\mathbf{p}_{v_n}(t)$, $\boldsymbol{\nu}_{v_n}(t)$, and $\theta_{v_n}(t)$ are updated to reflect the current state of each active vehicle $v_n \in \mathcal{V}_{\text{active}}(t)$ and each vehicle mesh M_{v_n} is synchronized accordingly. The velocity $\boldsymbol{\nu}_{v_n}(t)$ is provided to allow for Doppler shift—and consequent channel coherence time—computation. This synchronization ensures that the propagation environment faithfully reflects the time-varying geometry and kinematics of the vehicular network, which is essential for the realistic modeling of wireless channel dynamics under mobility. To limit the computational cost of ray tracing scene updates, synchronization is performed only if the vehicle has moved beyond a minimum threshold distance Δd_{min}, defined as:

$$\Delta d_{\text{min}} = \max\left(\Delta d_0,\ \nu_{v_n(t)} \cdot \tau_{\text{coh}}\right), \tag{2}$$

where Δd_0 is a base distance threshold, τ_{coh} is the channel coherence time, and $\nu_{v_n(t)}$ the current velocity of the vehicle. Since a Doppler-based threshold may lead to excessive updates, vehicular simulations typically favor the use of larger Δd_0, with its numerical value being scenario-dependent and determined empirically. Clearly, this simple approach may be extended into more advanced and efficient mechanism, also possibly linked to the specific geometry of the considered scenario. Dynamic location updates are applied only if the following condition holds:

$$\|\mathbf{p}_{v_n}(t) - \mathbf{p}_{v_n}(t')\| > \Delta d_{\text{min}}, \tag{3}$$

where $t' < t$ denotes the time of the last update applied to the mesh M_{v_n}. This condition is always true if a location update procedure for M_{v_n} has never been completed before.

C. Channel parameter requests and responses

This final data exchange, represented by the green arrows in Figure 1 and Figure 2, takes place between the ray tracing component and the Physical Layer of ms-van3t. Specifically, when the RAT model within ms-van3t detects an imminent transmission between two vehicles $(v_i, v_j) \in \mathcal{V}_{\text{active}}(t) \subseteq \mathcal{V}$ at simulation time t, a channel parameter request is issued to the ray tracer (see Figure 2e), bypassing the native channel computation mechanisms of ns-3. Upon receiving a channel parameter request, the ray tracer computes the set of propagation paths $\mathcal{R}_{v_i,v_j}$ (as illustrated in Figure 2f) between the antennas mounted on the corresponding vehicle meshes M_{v_i} and M_{v_j}. Although initially triggered by a specific vehicle pair, the ray tracing engine computes all pairwise propagation paths among the vehicles in $\mathcal{V}_{\text{active}}(t)$ to avoid redundant computations. This strategy, in conjunction with carefully selected trade-offs aimed at reducing ray tracing complexity while maintaining accuracy [21], amortizes the cost of scene analysis by leveraging GPU parallelization and distributing initialization overhead across multiple channel queries. The resulting set of admissible propagation paths, under the assumption of channel reciprocity, is:

$$\mathcal{R}_{v_i,v_j} = \mathcal{R}_{v_j,v_i} = \{r_1, r_2, \ldots, r_K\}. \tag{4}$$

Each path r_k is characterized by a complex gain α_k, a delay τ_k, direction of departure $\boldsymbol{\psi}_k$, direction of arrival $\boldsymbol{\vartheta}_k$, and a Doppler shift φ_k. Only those paths that satisfy relevant physical and geometric constraints (e.g., interaction type, angular resolution, and bounce count) are included in $\mathcal{R}_{v_i,v_j}$. Depending on the characteristics of the environment, both LoS and NLoS components may be present in the computed propagation paths. In addition to the set of propagation paths $\mathcal{R}_{v_i,v_j}$, the channel impulse response $\mathbf{h}_{v_i,v_j}(t)$ between a transmitting vehicle v_i and a receiving vehicle v_j is computed, derived using the impulse response model for multipath channels:

$$\mathbf{h}_{v_i,v_j}(t) = \sum_{k=1}^{K} \alpha_k\, e^{j2\pi\varphi_k t}\, \mathbf{a}_{v_i}(\boldsymbol{\vartheta}_k)\mathbf{a}_{v_j}^{\mathrm{T}}(\boldsymbol{\psi}_k)\, g\left(t - \tau_k\right), \tag{5}$$

where α_k denotes the complex path gain of the k-th ray, which depends on the underlying propagation mechanisms (i.e., reflection, diffraction, scattering), φ_k represents the Doppler shift associated with the k-th path, resulting from the relative motion between v_i, and v_j, $\mathbf{a}_{v_i}(\boldsymbol{\vartheta}_k) \in \mathbb{C}^{N_T \times 1}$ and $\mathbf{a}_{v_j}^{\mathrm{T}}(\boldsymbol{\psi}_k) \in \mathbb{C}^{N_R \times 1}$ correspond to the transmit and transposed receive array response vectors, respectively, where $\boldsymbol{\psi}_k$ and $\boldsymbol{\vartheta}_k$ denote the directions of arrival and departure of the ray, $g(t-\tau_k)$ denotes the pulse shaping filter, with τ_k being the propagation delay associated with the k-th ray. Among the various parameters

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

involved in the channel impulse response $\mathbf{h}_{v_i,v_j}(t)$ described above, a selected subset can be extracted and included in the channel parameter response sent to the Physical Layer of ms-van3t (see Figure 2h). Specifically, each channel parameter response is composed of the tuple $\{\mathcal{G}_{v_i,v_j}, \tau_{v_i,v_j}, \lambda_{v_i,v_j}\}$, where $\mathcal{G}_{v_i,v_j}$ denotes the total path gain, defined as:

$$\mathcal{G}_{v_i,v_j} = \left| \sum_{k=1}^{K} \alpha_k \right|^2 ; \tag{6}$$

τ_{v_i,v_j} denotes the propagation delay of the communication, defined as the minimum among all individual path delays, i.e.:

$$\tau_{v_i,v_j} = \min\{\tau_1, \tau_2, \ldots, \tau_K\} ; \tag{7}$$

where, τ_{v_i,v_j} is used solely to capture the end-to-end propagation delay, whereas the full set of multipath components is preserved in the channel characterization and fully incorporated in the subsequent receiver processing.

λ_{v_i,v_j} indicates the LoS condition, and is defined as:

$$\lambda_{v_i,v_j} = \begin{cases} 1, & \text{if } \exists r_k \in \mathcal{R}_{v_i,v_j} \text{ that is LoS,} \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

For computation efficiency reasons, the set of propagation paths $\mathcal{R}_{v_i,v_j}$ computed with ray tracing is stored in a cache $\mathcal{C}$ for potential reuse (shown in Figure 2g). Each cache entry corresponding to a vehicle pair (v_i, v_j) is denoted by $\mathcal{C}_{v_i,v_j}$. When a channel parameter request is issued at time t, two cases may arise:

- *Cache miss*: If $\mathcal{C}_{v_i,v_j} = \emptyset$. In this case, the ray tracing engine computes $\mathcal{R}_{v_i,v_j}$ for all $(v_i, v_j) \in \mathcal{V}_{\text{active}}(t)$, extracts the requested parameter, and stores the result: $\mathcal{C}_{v_i,v_j} \leftarrow \mathcal{R}_{v_i,v_j}$;
- *Cache hit*: If $\mathcal{C}_{v_i,v_j} \neq \emptyset$. In this case, the stored result is used without additional computation.

Whenever any vehicle v_n undergoes a state change that results in an update of its associated mesh M_{v_n}, the entire cache for every vehicle pair $(v_i, v_j) \in \mathcal{V}_{\text{active}}(t)$ is invalidated: $\mathcal{C} \leftarrow \emptyset$. This conservative strategy ensures cache consistency with the current simulation state, fully reflecting the dynamic impact of mobile scatterers (other vehicles) on propagation. While this mechanism may appear stringent, it is crucial in vehicular networks, where transient changes in the environment can substantially affect multipath structure, received power, and signal quality. In the following section, we validate the ray tracing propagation model of VaN3Twin by comparing it against real-world vehicular field measurements.

III. VaN3Twin Validation through Real-data

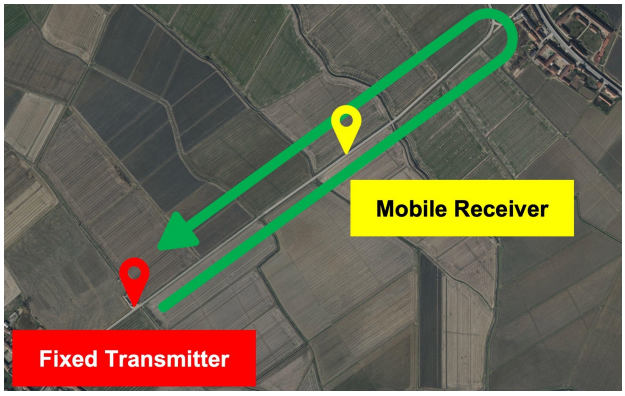
To assess how accurately VaN3Twin can replicate the wireless propagation in a V2X network, two measurement campaigns were conducted: one in a rural setting in Sali Vercellese, Vercelli, Italy (Figure 3a) and the other in the city center of Turin, Italy (Figure 3c). The field tests were conducted using a dedicated setup, thoroughly detailed in [11] and illustrated in Figure 4. In particular, Figure 4a shows an APU2E4 embedded board from PC Engines [23]—the red box in the figure—and an ArduSimple simpleRTK2B Fusion board [24]—the small black box. The OBU runs the latest version of OpenWrt-V2X [25] as its operating system, along with the most recent release of the Open Stack for car (OScar) framework [11], which provides a complete ETSI C-ITS protocol stack. The ArduSimple board enables Global Navigation Satellite System (GNSS) positioning through Real-Time Kinematic (RTK) services, achieving centimeter-level accuracy. Finally, Figure 4b displays the MobileMark SMWG-313 antenna [26], a magnetic, automotive-grade, multi-band antenna designed to support both the GNSS receiver and two Multiple Input, Multiple Output (MIMO) antenna elements for transmission and reception at V2X frequencies. An identical setup, as previously described, was deployed on two vehicles: one was parked at the roadside and served as the transmitter T_x, periodically sending vehicular Cooperative Awareness Messages (CAMs) at fixed time intervals; the other was in motion and acted as the receiver R_x. Two sets of field tests were conducted with T_x and R_x—shown in Figure 3a and Figure 3c—during which a range of parameters was measured. Upon packet reception, R_x recorded the payload, the UNIX timestamp of the event, the average measured RSSI, and its current GPS coordinates. The DTs for ray tracing in VaN3Twin—Figure 3b and Figure 3d—were modeled in Blender using the Blosm add-on, which imports 2.5D representations of the propagation environment from OpenStreetMap. Following ITU-R P.2040 recommendations [27], buildings were modeled as concrete structures. When present, trees were modeled as fully wooden objects, while the ground as concrete. Vehicles were included as full 3D meshes with metallic bodies to account for their scattering contribution. The ray tracing engine simulates specular reflection, diffuse reflection, and refraction, with a maximum of 5 interactions (bounces) per ray. Transmitting and receiving antennas were modeled as isotropic, which is a valid approximation of the antennas employed in the drive test. Simulation parameters used in the VaN3Twin ray tracer are summarized in Table I. Although this work focuses on low-rate CAM traffic, VaN3Twin supports a wide range of application-layer profiles, including high-rate and latency-sensitive workloads such as cooperative perception and sensor sharing. CAM traffic was selected to enable direct validation against real-world measurements obtained using hardware limited to such traffic generation. Extending the evaluation to additional traffic profiles is straightforward, as the underlying ns-3 simulator supports arbitrary traffic patterns, including

Table I: VaN3Twin parameters for ray tracing.

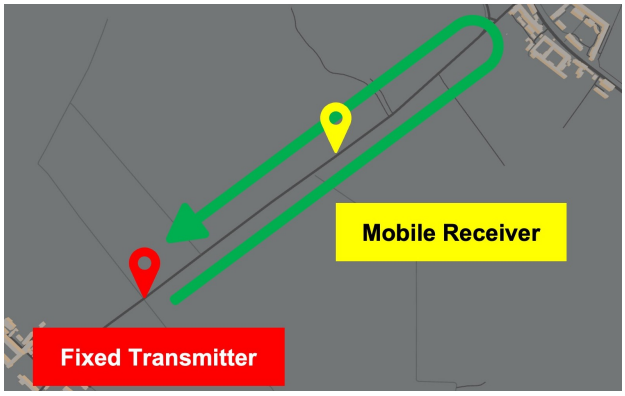
Ray tracing parameters	
Carrier Frequency [GHz]	5.89
Buildings and Roads Material	Concrete [27]
Trees Material	Wood [27]
Cars Material	Metal [27]
Reflections	Specular, Diffused
Refractions	Yes
Maximum Interactions	5
Type of Antennas	Isotropic
Polarization	Vertical
T_x **and** R_x **Height** [m]	1.8

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

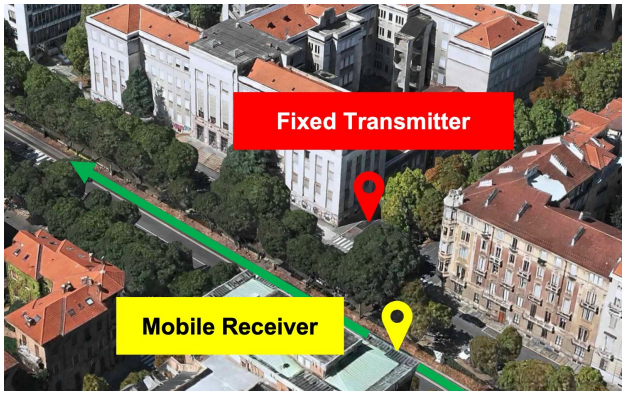
This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710



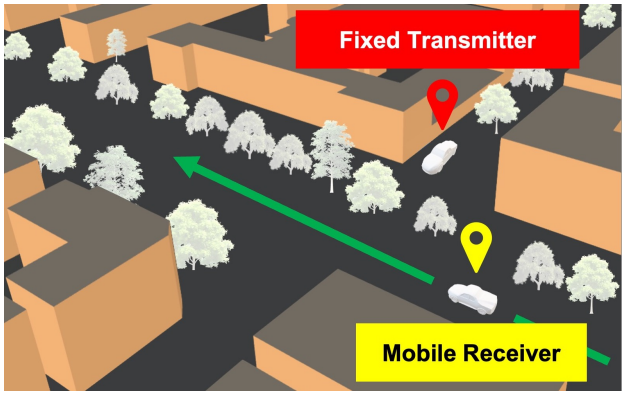
(a) Satellite view of the rural scenario [22].



(b) Replica for the VaN3Twin ray tracer of the rural scenario.



(c) Satellite view of the urban scenario [22].



(d) Replica for the VaN3Twin ray tracer of the urban scenario.

Figure 3: Areas of the two measurement campaigns in Sali Vercellese, Vercelli, Italy (a) and its DT (b) and Turin, Italy (c) and its DT (d), rural and urban scenarios, respectively. In both DTs, the static T_x antenna is represented as the red marker, while the R_x is yellow and moves along the green trajectory.

for example adaptive video streaming [28]. Validation was carried out at two levels: (i) propagation, by comparing real R_x RSSI measurements against their virtual counterparts; and (ii) packet reception, by assessing alignment between packet receptions in the virtual R_x and the physical dataset. For (ii), we introduce a dedicated metric, the Disagreement Ratio (DR), to assess the extent to which VaN3Twin enhances overall replication accuracy. The DR quantifies the level of disagreement between VaN3Twin and a baseline in terms of binary decision outcomes. Let $\mathcal{R}$ and $\mathcal{M}$ denote the sets of decision outcomes, positive or negative, by VaN3Twin and by the baseline, respectively. The symmetric difference $\mathcal{R}\Delta\mathcal{M}$ then represents the set of outcomes classified as positive or negative by one, but not by the other, i.e., the instances where the two disagree on the decision. $\mathcal{R}\cup\mathcal{M}$ represents the totality of the decisions. The DR is defined as:

$$DR = |\mathcal{R}\Delta\mathcal{M}| \,/\, |\mathcal{R}\cup\mathcal{M}| \,. \tag{9}$$

A lower value of DR indicates a stronger alignment between the considered entities, whereas higher values highlight greater discrepancies in their decision outcomes. To enable a comprehensive evaluation of the differences between stochastic models (as adopted by current V2X state of the art simulation tools), a ray-based propagation model, and real-world measurements, the same simulation was also conducted using the vanilla version of ms-van3t. In this configuration, ms-van3t relies on a stochastic propagation model based on 3GPP TR 36.885 [29] specifications, which incorporates distance-dependent, probabilistic LoS status determination. More specifically, the LoS probability is:

$$P_{LoS} = \min\left(\frac{18}{d}, 1\right)\left(1 - e^{-d/36}\right) + e^{-d/36}, \tag{10}$$



(a) OBU and ArduSimple GNSS devices.



(b) Magnetic MobileMark antenna.

Figure 4: Vehicle setup for field tests.

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

where d is the distance between the transmitter and the receiver. Unlike ray tracing, this model has no tunable parameters, as it is rigidly defined by 3GPP specifications. Notably, this model is well aligned with the conditions of both scenarios. The rural campaign featured a full LoS between transmitter and receiver, closely matching theoretical assumptions. Meanwhile, in the urban scenario, the transmitter was placed at a height comparable to that of a typical cellular microcell, precisely the conditions for which the model was designed to operate. These represent ideal conditions for the model's intended use, making the comparison particularly relevant. In conclusion, both the real-vehicle field tests and the simulations with VaN3Twin and ms-van3t used the IEEE 802.11p communication stack.

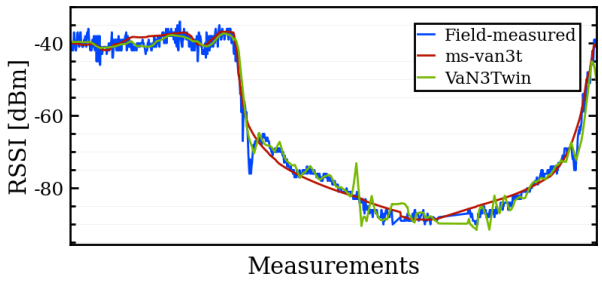
A. LoS rural scenario

This measurement campaign was carried out in a completely open and flat environment near Sali Vercellese, in the countryside surrounding Vercelli, Italy. The environment is free from obstacles such as buildings or trees. The transmitting vehicle traveled along a straight road, performing a round trip following the trajectory shown in Figure 3a and Figure 3b. The moving vehicle, R_x, maintained an uninterrupted LoS condition with the transmitting vehicle, T_x, which remained parked and stationary. This scenario serves as a necessary baseline for calibrating more complex experiments, such as those conducted in urban environments, where the presence of buildings and other obstacles significantly affects signal propagation. Figure 5 presents a comparison of signal propagation across the different models. Specifically, the values obtained from field measurements are shown in blue, those generated by VaN3Twin in green, and the ones with ms-van3t in red. In Figure 5a, which depicts the RSSI values, it can be observed that the measured signal closely follows the trend reproduced by the VaN3Twin simulation, with consistently aligned values along the entire trajectory of the moving vehicle R_x. As expected, ms-van3t also aligns well with the measured data, given the simplicity of the scenario. However, it is worth noting that the RSSI trend produced by ms-van3t appears overly smooth and lacks local fluctuations, an inherent limitation of its theoretical formulation which does not account for small-scale propagation effects. This observation is further supported by Figure 5b, where the Probability Density Functions (PDFs) of the three datasets exhibit overall consistent trends.

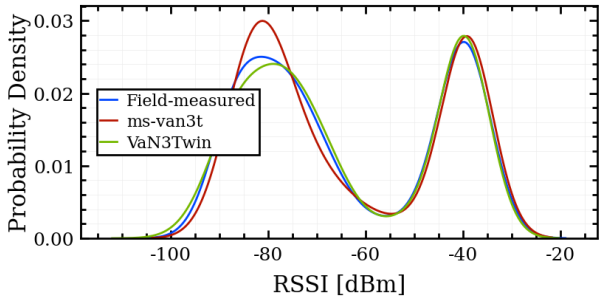
Regarding the Packet Reception DRs, a disagreement of 12.82% between the real-world measurements and the simulation results is obtained using ms-van3t. Despite the simplicity of the propagation environment, the use of VaN3Twin nearly halves this discrepancy, reducing it to only 6.5%.

B. NLoS urban scenario

Compared to the previously considered rural case, the urban scenario presents a significantly more complex propagation environment. These measurements were conducted in a particularly busy area near the city center of Turin, Italy, which is characterized by multiple trees, parked cars, and various other obstacles, causing a considerable increase in the number



(a) Comparison of the field-measured RSSI (blue) with the estimated value using VaN3Twin (green) and ms-van3t (red).



(b) Probability Density Function of the field-measured RSSI (blue), the estimated value with VaN3Twin (green) and with ms-van3t (red).

Figure 5: Measured and simulated RSSI values (a) and their Probability Density Function (b) for the rural measurement campaign in Sali Vercellese, Vercelli, Italy.

of interactions of the transmitted signal with its surroundings affecting the RSSI. Unlike the scenario in Sali Vercellese, the visibility conditions in this case vary depending on the relative positions of the vehicles within the environment. At the beginning of the measurement, the parked transmitting vehicle T_x and the moving receiver R_x are in a NLoS condition. As illustrated in Figure 3c and Figure 3d, the transmitter follows a straight trajectory from right to left. Around the midpoint, a brief LoS interval occurs before the vehicles return to a NLoS configuration, which persists until the end of the test. This pattern replicates a typical urban scenario, in which two vehicles located on perpendicular roads attempt to communicate with one another. Similarly to Figure 5, Figure 6 presents a comparison of signal propagation across the different models for the NLoS urban scenario. The results in Figure 6a show a reasonable alignment between the RSSI values obtained from field measurements and those extracted from VaN3Twin. Although the match is less precise than in the quasi-ideal rural case—due to the increased complexity of the urban environment—the overall trends remain consistent. It is important to note that while VaN3Twin offers a high-fidelity reproduction of realistic channel conditions, it cannot guarantee a perfect match in all situations. In this case, for instance, augmenting the level of detail of the virtual scene with additional elements such as parked vehicles, trees, and street furniture would likely reduce the residual gap and further improve simulation accuracy. Conversely, the output of vanilla ms-van3t reveals evident limitations in this context. The graph

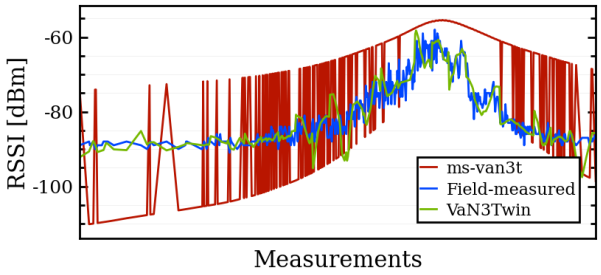
Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

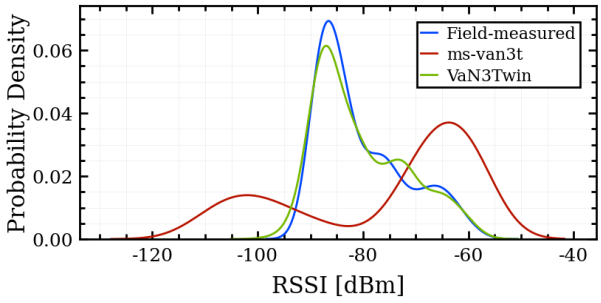
Table II: Packet Reception DRs of ms-van3t and VaN3Twin with measured data.

Type of Scenario	ms-van3t	VaN3Twin	Difference
Rural (Sali Vercellese)	12.82%	6.5%	49.3%
Urban (City of Turin)	27.9%	8.16%	70.8%

shows abrupt and continuous RSSI fluctuations, results of the rigid alternation between LoS and NLoS propagation models as per equation (10). This behavior is further highlighted in Figure 6b, where the PDF generated by VaN3Twin closely mirrors the measured distribution, accurately capturing both the mean and the variability of the RSSI values. In contrast, the vanilla ms-van3t model yields a distinctly bimodal distribution which leads to unrealistic clustering and fails to reflect the smooth variations observed during the field test. The computed for Packet Reception DR in the NLoS urban scenario reveals a substantial disagreement of 27.9% between real-world data and the simulation performed using the ms-van3t model. Also in this case, VaN3Twin significantly reduces the discrepancy, achieving a DR of just 8.16%. This result highlights that, although the 3GPP-based stochastic model in ms-van3t is specifically calibrated for urban scenarios with NLoS conditions, VaN3Twin achieves an accuracy level that is more than three times higher. In conclusion, given its deterministic and ray-based propagation model, VaN3Twin is capable of reproducing even complex, urban environments, leading to a higher punctual accuracy at the applicative level



(a) Comparison of the field-measured RSSI (blue) with the estimated value using VaN3Twin (green) and ms-van3t (red).



(b) Probability Density Function of the field-measured RSSI (blue), the estimated value with VaN3Twin (green), and with ms-van3t (red).

Figure 6: Measured and simulated RSSI values (a) and their Probability Density Function (b) for the urban measurement campaign in Turin, Italy.

when compared to the stochastic model in vanilla ms-van3t. Table II summarizes the computed Packet Reception DRs using ms-van3t and VaN3Twin to reproduce real-world measurements, showing decreases of 49.3% in the rural scenario and 70.8% in the urban scenario. Building on this capability, the following section introduces the VaN3Twin coexistence module, which accounts for interference generated by other transmitting vehicles operating in parallel with different RATs, thereby further increasing the realism and completeness of the simulation.

IV. VaN3Twin for Multi-RAT Coexistence

Given the pivotal role of the ray tracing engine in VaN3Twin for reconstructing a consistent and realistic channel model, we leveraged this capability to develop a dedicated *coexistence module*. While the vanilla version of ms-van3t could simulate scenarios where vehicles using different RATs coexist, it did not account for the interference that transmissions from one RAT might cause to others. To address this limitation, we designed a module to explicitly model inter-RAT interference, thereby enabling accurate evaluation of communication performance in multi-RAT scenarios. This section introduces the underlying logic of the VaN3Twin module developed to manage multi-RAT coexistence, accounting for partial or full spectrum overlap between concurrent transmissions across different vehicular communication technologies. Finally, we compare three configurations: the vanilla ms-van3t model without coexistence support, the ms-van3t propagation model coupled with the coexistence module, and the coexistence module integrated with the accurate channel modeling of VaN3Twin.

A. Proposed coexistence implementation

The vanilla version of ms-van3t includes different default propagation models for each RAT. However, the LTE-V2X and NR-V2X modules typically rely on 3GPP-based propagation models [29], [30] that are incompatible with those used by the IEEE 802.11p module, and vice versa. Furthermore, as previously discussed, the default channel models in ms-van3t employ probabilistic LoS determination, limiting their accuracy in complex scenarios. To partially mitigate these limitations, ms-van3t allows manual adjustment of the channel condition through external inputs. This can be achieved either by directly injecting the LoS status—e.g., from a ray tracing engine—or by generating a 2D building map within the simulator for automatic LoS/NLoS classification. However, the latter approach is both time-consuming and imprecise, as it represents buildings using basic box geometries. In contrast, VaN3Twin leverages ray tracing to deliver significantly improved modeling accuracy and a unified channel abstraction that is consistent across different RATs. This unified approach enables consistent and realistic channel estimation, as previously formalized in equation (5). Such consistency is especially critical for multi-RAT coexistence scenarios, where realistic modeling of shared or partially overlapping spectrum is required. In a V2X simulation scenario, let $\mathcal{W}$ denote the set of simulated RATs for which the coexistence

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

module is enabled. Each $w \in \mathcal{W}$ operates over a frequency band of bandwidth $B^{(w)}$, centered at carrier frequency $f_c^{(w)}$, and utilizes an Orthogonal Frequency Division Multiplexing (OFDM) waveform with $L^{(w)}$ subcarriers spaced by $\Delta f^{(w)}$. The corresponding sub-channel grid is given by:

$$\mathcal{F}^{(w)} = \left\{ f_c^{(w)} + l \cdot \Delta f^{(w)} \right\} \text{ with}$$

$$f_c^{(w)} + l \cdot \Delta f^{(w)} \in \left[f_c^{(w)} - \frac{B^{(w)}}{2},\ f_c^{(w)} + \frac{B^{(w)}}{2} \right], \quad (11)$$

where $l \in L^{(w)}$ is an integer. To evaluate inter-RAT interference, a global resource grid must be defined that encompasses all technologies in the set $\mathcal{W}$. For each $w \in \mathcal{W}$, the corresponding grid is defined as:

$$\mathcal{Z}^{(w)} = \mathcal{F}^{(w)} \times \mathcal{S}^{(w)}, \quad (12)$$

where $\mathcal{S}^{(w)}$ is the set of corresponding time slots. The overall resource grid is defined as:

$$\mathcal{Z} = \mathcal{F} \times \mathcal{S}, \text{ with } \mathcal{F} = \bigcup_{w \in \mathcal{W}} \mathcal{F}^{(w)}, \quad \mathcal{S} = \bigcup_{w \in \mathcal{W}} \mathcal{S}^{(w)}. \quad (13)$$

Given a resource block $z \in \mathcal{Z}$, let $\mathcal{W}(z) \subseteq \mathcal{W}$ denote the set of RATs actively using it. To monitor potential inter-RAT interference, the coexistence module continuously observes the communication channel and tracks all transmissions from each $w \in \mathcal{W}(z)$.[4] To support this functionality, a set of tracking objects $\mathcal{H}$ is introduced. For each transmitting vehicle $v_n \in \mathcal{V}_{\text{tx}}(t) \subseteq \mathcal{V}_{\text{active}}(t)$ at time t, a corresponding tracking object $H_{v_n} \in \mathcal{H}$ is defined as:

$$H_{v_n} = \left(w \in \mathcal{W},\ f_c^{(w)},\ B^{(w)},\ T,\ S(f) \right), \quad (14)$$

where T is the transmission duration and $S(f)$ the Power Spectral Density (PSD). Let us now consider two vehicles transmitting using technologies w and w', both belonging to $\mathcal{W}(z)$, with $w \neq w'$. The SINR on resource block z for the transmission using technology w is computed as:

$$\text{SINR}_{w,z} = \frac{P_{\text{Rx},w,z}}{N_{0,z} + P_{\text{int},w} + \sum_{w' \neq w} P_{\text{Rx},w',z}}, \quad (15)$$

where $P_{\text{Rx},w,z}$ denotes the received power of the desired signal, $N_{0,z}$ the noise power on z, $P_{\text{int},w}$ the intra-RAT interference from other vehicles using w, and $P_{\text{Rx},w',z}$ the interfering power from technology w'. The contribution of $P_{\text{Rx},w',z}$ in equation (15) is computed as follows: let v_{int} be an interfering vehicle using technology $w' \in \mathcal{W}(z)$, with an associated tracking object $H_{v_{\text{int}}}$. To discard negligible interference, a threshold on the total path gain $\mathcal{G}_{\text{thr}}$ is introduced. For a receiving vehicle v_r using technology $w \neq w'$, interference from v_{int} is ignored if the total path gain $\mathcal{G}_{v_{\text{int}},v_r}$ is less than the minimum threshold, i.e., $\mathcal{G}_{v_{\text{int}},v_r} < \mathcal{G}_{\text{thr}}$. Otherwise, the received interference power is calculated based on the PSD S and the total path gain $\mathcal{G}_{v_{\text{int}},v_r}$. The overall SINR for technology w is obtained by computing a bandwidth-weighted average over all resource blocks $z \in \mathcal{Z}$:

$$\text{SINR}_w = \frac{1}{|\mathcal{Z}|} \sum_{z \in \mathcal{Z}} \omega_z \cdot \text{SINR}_{w,z}, \quad (16)$$

[4]This feature enables concurrent simulation of IEEE 802.11p and NR-V2X.

Algorithm 1 SINR evaluation with inter-RAT contributions

Require: Set of active RATs $w \in \mathcal{W}$
Require: Sub-channels $\mathcal{F}^{(w)}\ \forall\ w \in \mathcal{W}$
Require: Time slots $\mathcal{S}^{(w)}\ \forall\ w \in \mathcal{W}$
for all $w \in \mathcal{W}$ **do**
  Set $\mathcal{F} \leftarrow \mathcal{F} \cup \mathcal{F}^{(w)}$
  Set $\mathcal{S} \leftarrow \mathcal{S} \cup \mathcal{S}^{(w)}$
end for
$\mathcal{Z} \leftarrow \mathcal{F} \times \mathcal{S}$
Track each $v_n \in \mathcal{V}_{\text{tx}}(t) \subseteq \mathcal{V}_{\text{active}}(t)$
for all blocks $z \in \mathcal{Z}$ **do**
  Identify set of active RATs $\mathcal{W}(z) \subseteq \mathcal{W}$
  for all RATs $w \in \mathcal{W}(z)$ **do**
    Initialize interference to only intra-RAT contributions
    Set $P_{\text{int,tot}} \leftarrow P_{\text{int},w}$
    for all $w' \in \mathcal{W}(z),\ w' \neq w$ **do**
      for all tracked interferers v_{int} using w' **do**
        if total path gain $\mathcal{G}_{v_{\text{int}},v_r} \geq \mathcal{G}_{\text{thr}}$ **then**
          $P_{\text{int,tot}} \leftarrow P_{\text{int,tot}} + P_{\text{Rx},w',z}$
        end if
      end for
    end for
    Compute $\text{SINR}_{w,z}$
  end for
end for
for all $w \in \mathcal{W}$ **do**
  Compute average SINR_w weighted by ω_z
end for

where ω_z is a weight related to the bandwidth of block z. For clarity and ease of understanding, the SINR computation process described above is also summarized in Algorithm 1.

Furthermore, to illustrate the capabilities of the coexistence module, a simulation of a vehicular scenario with multi-RATs communication is presented and analyzed in the following subsection. The goal is not to conduct an exhaustive co-channel interference study for vehicular multi-RAT simulations, but to evaluate how SINR computation and its impact at the application level differ when interference from a coexisting technology is modeled using VaN3Twin versus ms-van3t. This comparison quantifies the benefits of integrating accurate physical-layer modeling within an NDT framework.

B. Multi-RAT simulation with coexistence module

To validate the coexistence mechanism under realistic interference conditions, a multi-RATs scenario is considered in which 20 vehicles traverse an urban environment containing buildings and obstacles, resulting in both LoS and NLoS propagation conditions. The DT was created following the same procedure introduced in Section III while the ray tracing parameters are the ones in Table I. Each vehicle periodically transmits CAMs and Collective Perception Messages (CPMs) following ETSI transmission rules, using a common carrier frequency of 5.9 GHz and a bandwidth of 10 MHz. The vehicles are evenly split between two technologies: 10 use IEEE 802.11p and 10 use NR-V2X. Table III and Table IV

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

summarize the main simulation parameters for NR-V2X and IEEE 802.11p communication technologies, respectively. The setup is implemented in VaN3Twin with an ns-3 simulation configured to define the communication environment. The configuration process begins by reading the .xml route file from SUMO, which defines the vehicles to be generated. Next, the ns-3 nodes are instantiated, with each node representing a C-ITS station. Each node is equipped with the necessary Facilities, such as the Cooperative Awareness (CA) Basic Service, followed by the creation of the PHY and MAC layers for each communication technology in use. Once all components are properly initialized, the simulation starts, and synchronization between ns-3, SUMO, and Sionna RT proceeds as previously described. Since all vehicles operate on the same frequency band, transmissions from one technology can act as interference for the other, depending on spectral overlap and propagation conditions. In order to establish a baseline representative of the current state of the art, the simulation is first conducted in ms-van3t without the coexistence module. Subsequently, the same simulation is repeated with the coexistence module enabled, using both ms-van3t and VaN3Twin. This procedure allows for the assessment of how the different propagation modeling approaches—3GPP-based in ms-van3t and ray-based in VaN3Twin—affect the estimated SINR for each received packet, the identification of potential interferers, and ultimately the outcome of the packet decoding process. To ensure identical propagation conditions for both models, LoS determination in ms-van3t was injected using information from VaN3Twin's ray tracer, enabling a fair and meaningful comparison. Figure 7 shows the SINR PDFs produced by all three considered simulators. While the distribution obtained using VaN3Twin exhibits a Gaussian-like trend, the ones generated by the two versions of ms-van3t both display two prominent peaks. This bimodal characteristic stems from the rigid model switching governed by the binary LoS/NLoS classification of the communication link, as already observed during the validation phase—see Figure 6. In real-world propagation environments, however, the transition between LoS and NLoS conditions is not discrete but rather gradual. Even under LoS conditions, surrounding obstacles can cause signal refraction and attenuation, leading to a degradation in reception quality. Such effects are not captured by the discrete switching logic of ms-van3t, but are instead considered in the ray tracer behavior of VaN3Twin. More generally, Figure 7 highlights a tendency of simulators using 3GPP-based propagation models to overestimate SINR values, even if the deterministic LoS state information from the ray tracing engine is injected. The absence of coexistence-aware modeling in the vanilla ms-van3t configuration leads to significantly higher SINR estimates.

Another critical aspect in vehicular communication systems is the impact of surrounding vehicles, which can give rise to Non-Line of Sight due to Vehicles (NLoSv) conditions. These scenarios cannot be neglected in realistic modeling. For example, millicar [32], a V2V module for ns-3 specifically designed for FR2 communications, addresses this issue through a probabilistic approach that models both the likelihood of vehicular obstructions and the statistical distribution of their physical dimensions. As discussed in Section II-C, the ray tracing engine provides a binary output λ_{v_i,v_j} indicating the LoS status between two vehicles (v_i, v_j). This determination is based solely on the presence or absence of at least one LoS path within the set of admissible propagation rays $\mathcal{R}_{v_i,v_j}$. Moreover, the currently available models in ms-van3t do not support NLoSv conditions. Consequently, such cases are conservatively treated in ms-van3t as NLoS due

Table III: VaN3Twin parameters for NR-V2X.

NR-V2X Parameters	
Carrier Frequency [GHz]	5.9
Channel Bandwidth [MHz]	10
Subchannel Bandwidth [MHz]	10
Numerology	2
Modulation and Coding Scheme	14
Sidelink Threshold For PSSCH (RSRP) [dB]	21
Resource Blocks For PSSCH	10
Number Of TX Per Resource Block	3
Reservation Period [ms]	20
PHY Modulation	OFDM
Subcarrier Spacing [kHz]	60
Tx Power [dBm]	30
Sensitivity [dBm]	-93
SINR Threshold [dB]	10
Resource Selection Method	Random [31]
Propagation Loss Model	3GPP-based [30]
Sidelink Mode	PC5 Mode 4

Table IV: VaN3Twin parameters for IEEE 802.11p.

IEEE 802.11p Parameters	
Carrier Frequency [GHz]	5.9
Channel Bandwidth [MHz]	10
PHY Modulation	OFDM
Tx Power [dBm]	30
Sensitivity [dBm]	-93
SINR Threshold [dB]	10
Data Rate [Mb/s]	3
Carrier Access Protocol	CSMA/CA
Propagation Loss Model	3GPP-based [29]

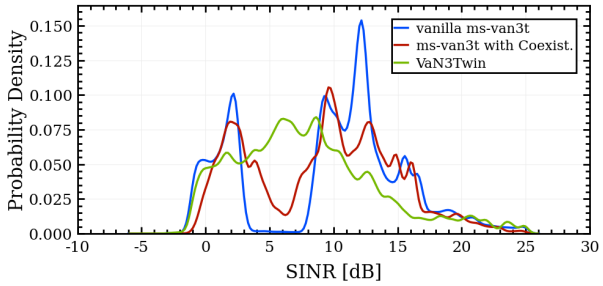


Figure 7: Probability Density Functions of the measured SINRs for all the simulated V2V links computed using vanilla ms-van3t without coexistence module (blue), ms-van3t with coexistence module (red), and VaN3Twin (green).

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

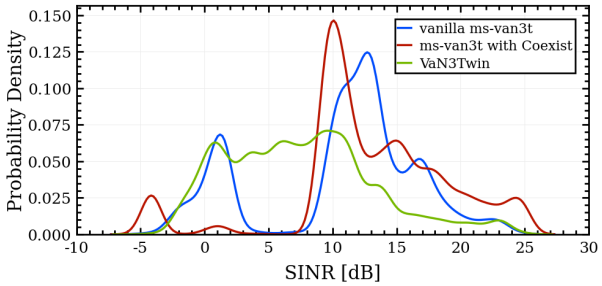


Figure 8: Probability Density Functions of SINRs for V2V links with frequent NLoSv scenarios computed using vanilla ms-van3t without coexistence module (blue), ms-van3t with coexistence module (red), and VaN3Twin (green).

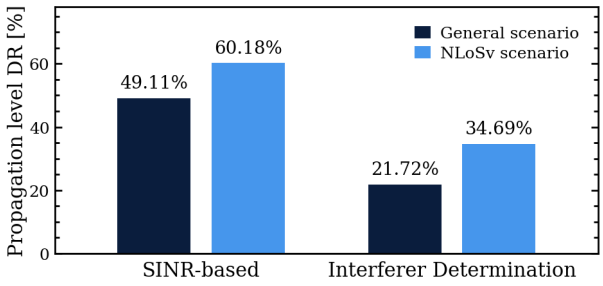


Figure 9: Propagation level Disagreement Ratios in interferer detection and SINR-based packet decodability decisions for a general scenario and for the ones with frequent NLoSv.

to large obstacles as buildings. This limitation is evident in Figure 8, which presents the SINR PDF for a subset of vehicles frequently experiencing NLoSv conditions during the simulation. Here, the distribution obtained with VaN3Twin retains the previously observed Gaussian-like profile. Conversely, the two versions of ms-van3t notably overestimate SINR—irrespective of whether NLoSv conditions are treated as NLoS—both showing a pronounced peak around 10 dB. In conclusion, the Gaussian-like distributions observed in the VaN3Twin results of Figures 7 and 8 reflect a more realistic propagation behavior, as discussed in Section III. As detailed in Section IV-A, identifying nearby transmitters using other RATs as potential sources of interference directly shapes the resulting SINR. This estimation, ultimately, determines whether an ongoing transmission is decodable at the receive or not. The combination of these mechanisms allows VaN3Twin to form a decision about the expected reception outcome, which may differ from that of ms-van3t. Figure 9 reports two Propagation level DRs for the two previously considered sets of simulated vehicles. The DR for interferer detection between ms-van3t with coexistence enabled and VaN3Twin is 21.72%, reaching 34.69% in scenarios with frequent NLoSv conditions. This indicates that in up to 35% of the cases, the ms-van3t incorrectly discards an interfering signal—treating it as negligible—whereas VaN3Twin correctly identifies it as having a significant impact on the transmission. These incorrect assumptions, combined with the higher modeling accuracy of VaN3Twin, also propagate to the final decision on packet decodability, further increasing the divergence between the two models. In the general case, the DR reaches 49.11%, meaning that ms-van3t and VaN3Twin disagree on packet reception in nearly half of the instances. This disagreement rises to 60.18% under frequent NLoSv conditions, highlighting how ms-van3t is unable to capture transient obstructions. For example, for the communication between two NR-V2X vehicles shown in Figure 10, ms-van3t and VaN3Twin yield differences in the estimated interfering power of up to 17.53 dBm, resulting in a corresponding SINR deviation of 4 dB. As for the application layer performance, let us first consider the impact of the coexistence module on the overall Packet Reception Ratio (PRR) for the NR-V2X communication, which measures the final number of successfully processed packets. Using the vanilla version of ms-van3t, the resulting PRR is 97%, decreasing to 93% if the coexistence module is used. On the other hand, VaN3Twin computes a lower PRR of 84%. The overall impact of inter-RAT interference on the PRR is then limited, primarily due to the small payload sizes of CAMs and CPMs and the robust retransmission mechanisms provided by NR-V2X. However, such aggregated indicators fail to capture discrepancies at the single packet level, crucial in a NDT. Let us consider the Packet Reception DR between ms-van3t with coexistence enabled and VaN3Twin. In the general case, we observe a DR of 30.56%, revealing a substantial mismatch between the two simulators. In the LoS-dominant scenario with frequent NLoSv conditions, the DR is slightly lower at 18.43%. This reduction is primarily attributed to the higher likelihood of successful packet delivery at 5.9 GHz when vehicles are in close proximity, as is typical in such scenarios.

In conclusion, the overall impact of coexistence on aggregated packet delivery metrics appears limited. However, this apparent agreement at the application layer is somewhat misleading, as high packet reception DRs were observed despite underlying modeling inaccuracies. This undermines the suitability of ms-van3t and, in general, of ns-3 as a NDT component, which requires not only statistically accurate outcomes but also a faithful reproduction of real-world communication dynamics. Furthermore, significant discrepancies were observed in SINR estimation, especially during transient obstructions. These differences are expected to have a greater impact in more demanding use cases, where larger data payloads and higher spectral efficiency requirements intensify

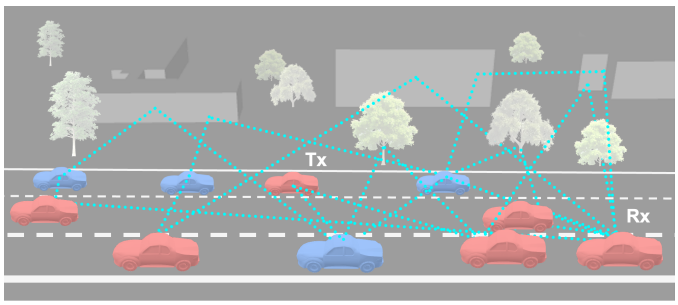


Figure 10: Simulation snapshot of an NR-V2X communication between Rx and Tx (red) interfered by 802.11p vehicles (blue) in a NLoSv-prone environment. Selected propagation paths from interferers to the receiver Rx are shown in blue.

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

Table V: VaN3Twin scalability analysis.

Vehicles	**Cache Hit**	**Speedup**	**Mean Cost** [ms]	**P95** [ms]
5	92.65%	13.5×	168.56	204.14
10	95.97%	24.7×	196.30	253.25
20	98.46%	64.1×	232.98	295.65
40	99.02%	150.6×	252.03	468.66
60	99.58%	237.1×	446.25	847.61

the need for accurate channel modeling, particularly at higher frequencies. Consequently, in scenarios involving advanced communication tasks—e.g., shared vision, cooperative perception, or platooning—the divergence between the two models is likely to become more pronounced. These use cases inherently demand higher SINR levels to ensure lower Block Error Rate (BLER) values and to maximize the achievable transmission rate, thereby further exposing the limitations of ms-van3t under realistic vehicular conditions.

C. Runtime and scalability analysis

The practical deployment of VaN3Twin on large-scale scenarios depends critically on the runtime cost of ray tracing computations. To quantify scalability, we recorded the cost—in terms of wall-clock time required to perform each channel parameter computation—with 5, 10, 20, 40, and 60 active vehicles, the maximum value allowed by the considered setup. Experiments were conducted on a workstation running Ubuntu 22.04.5 equipped with an NVIDIA GeForce RTX 3090 GPU with 25 GB of VRAM, 16 Intel Core i9-10980XE CPUs at 3.00 GHz, and 256 GB of RAM. The same scenario considered in Section IV-B was used with ray tracing parameters for Sionna RT v1.2.0 as per Table I except for the maximum number of interactions, which is set to 3. VaN3Twin propagation paths caching subsystem presented in Section II-C was enabled across all runs. The collected wall-clock times exhibit a strong bimodality introduced by the caching mechanism and therefore, in order to avoid biases introduced by the dominant reuse of cached propagation paths, we evaluate VaN3Twin scalability by jointly considering (i) the computational cost of full ray tracing computations between every active vehicle and (ii) the overall cache hit ratio. Observed statistics are summarized in Table V. As the number of vehicles increases, two effects emerge. First, the mean cost of a full ray tracing snapshot increases predictably with the number of active vehicles, rising from 168.56 ms (5 vehicles) to 446.25 ms (60 vehicles), with the 95th percentile increasing from 204.14 ms to 847.61 ms. This behavior reflects the growth in scene complexity but remains limited thanks to the grouped computation of channels between active vehicle pairs and the GPU parallelization capabilities of Sionna RT. Second, the cache-hit ratio increases monotonically, reaching 99.58% with 60 vehicles. In dense scenarios, each ray tracing snapshot produces channel information for a larger number of vehicle pairs, rapidly populating the cache and yielding a higher proportion of subsequent (almost) zero-cost operations, even though cache invalidation occurs more frequently. A critical factor is the overall system-wide acceleration, with the joint application of VaN3Twin's tightly coupled grouped channel computations and propagation path caching yielding a 237× speedup relative to the baseline. As full ray tracing executions become comparatively rare at higher densities, their contribution to the amortized per-request cost is substantially reduced. This observation also suggests that increasing the age of the twin—i.e., balancing output accuracy with reduced update frequency from the physical network—combined with trajectory prediction to proactively schedule ray tracing snapshot computations, can further approach real-time operation. Within the tested range, VaN3Twin thus exhibits sublinear growth in amortized runtime and favorable reuse dynamics. These results indicate that high-fidelity ray tracing in large-scale V2X scenarios is computationally attainable, provided that propagation path reuse mechanisms are appropriately designed and dimensioned, together with the possibility to deploy such system as a federation of smaller, edge-located NDTs operating over limited spatial areas.

V. Literature Review

This section reviews the state of the art in accurate V2X simulations and NDTs, with particular emphasis on co-channel coexistence and the integration of ray tracing. We begin by analyzing prior work on coexistence studies and the extent to which open source simulators support them. We then examine the use of ray tracing as a key enabler for PHY-layer-accurate simulations in complex urban environments. Finally, we highlight the novelty of our approach compared to existing open source solutions, besides providing an overview on vehicular DTs, to which VaN3Twin belongs. Table VI summarizes the differences between VaN3Twin and the main frameworks for V2X simulation and emulation currently available.

A. Simulation of co-channel coexistence

Vehicular communications are a cornerstone for higher levels of driving automation, particularly at SAE levels 4 and 5 [45]. In this context, simulation plays a crucial role in the NDT development, enabling the testing of new services in a safe and cost-effective manner [12], [46]. To this end, accurate simulation of the PHY and MAC layers, especially their interaction, is essential. While various studies have assessed each technology independently, both through simulation [16], [47], [48] and field testing [49], [50], research on the coexistence of multiple technologies operating on the same channel remains scarce. For example, the study in [51] analyzes the interference that next-generation 6G networks may cause to passive sensing satellite systems. As for V2X communications, studies like Maglogiannis et al. [52], consider separate channels, thereby excluding interference effects. In [53], Elloumi et al. propose a learning-based resource allocation framework for spectrum sharing in Vehicle-to-Infrastructure (V2I) communications. However, their work does not implement a full-stack approach, focusing instead on interference-aware scheduling within a single RAT. In particular, the coexistence of IEEE 802.11p and C-V2X Mode 4 [54] remains an open challenge. Both technologies share the congested 5.8–5.9 GHz spectrum, and their simultaneous deployment can degrade performance due

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

Table VI: Comparison of V2X frameworks with VaN3Twin, focusing on coexistence and ray tracing integration.

Framework/study	**Supported RATs**	**Coexistence** DSRC and C-V2X	**Ray tracing**	**V2X Protocols** Full-stack implementation	**Open source**
WiLabV2XSim [33]	IEEE 802.11p, LTE-V2X, NR-V2X	✓	*X*	*X*	✓
LTEV2VSim [34]	IEEE 802.11p, LTE-V2X	*X*	*X*	*X*	✓
Jayaweera et al. [35]	IEEE 802.11p, LTE-V2X	∼ (MATLAB-based numerical results)	*X*	*X*	*X*
Veins [36]	IEEE 802.11p, LTE-V2X [37]	*X*	*X* (can be potentially integrated as in [38])	✓(ETSI stack requires Artery [39])	✓
Rusca et al. [40]	Access technology-agnostic single RAT	*X*	✓(MATLAB RT)	*X*	✓
Gaugel et al. [41]	Access technology-agnostic single RAT	*X*	∼ (offline RT traces)	*X*	*X*
ns3sionna [42]	IEEE 802.11	*X*	✓(Sionna RT in the loop)	*X*	✓
DT-CoVeSS [43]	C-V2X	*X*	✓(customized Sionna RT)	∼ (support for SAE J3161 and IEEE 1609.2)	✓
ms-van3t [16], [44]	IEEE 802.11p, LTE, LTE-V2X, NR-V2X	*X*	*X*	✓(ETSI stack with support to multiple versions [16])	✓
VaN3Twin	**IEEE 802.11p, LTE, LTE-V2X, NR-V2X**	✓	**✓(Sionna RT in the loop)**	**✓(ETSI stack with support to multiple versions [16])**	✓

to mutual interference [55]. In [55], a possible solution is proposed involving an enhanced preamble for C-V2X, tested via the WiLabV2XSim simulator [33], which is based on LTEV2VSim [34]. This MATLAB-based tool supports both IEEE 802.11p and C-V2X and has also been used in [56] to assess different channel allocation strategies. Recognizing the importance of coexistence, Jayaweera et al. [35] propose an optimization-based approach for dynamic technology and channel selection, though their simulations rely on MATLAB and do not involve a full-featured V2X stack. While these tools can model different technologies, standard versions of simulators like ns-3 do not support mutual interference between Wi-Fi and cellular-based protocols [14]. Extensions for C-V2X [57], [58] and native IEEE 802.11p support exist, but precise coexistence modeling remains challenging. The ms-van3t framework, based on ns-3, SUMO [17], and CARLA [18], supports multiple technologies and ETSI basic services, but prior to this work, detailed modeling of mutual interference was lacking. OMNeT++ [59], another popular network simulator, has been extended for V2X via frameworks like Veins [36] and OpenCV2X [37]. However, it does not offer an integrated approach to simulate coexistence between cellular and Wi-Fi-based technologies and presents a more complex architecture for real-world deployment compared to ns-3. Before our work, WiLabV2XSim [33] was the only open source simulator capable of naively modeling co-channel interference between multiple V2X technologies.

B. Integration of V2X simulators with ray tracers

While simulators like ns-3, OMNeT++, and MATLAB-based tools differ in coexistence modeling, they commonly rely on stochastic or analytical channel models—e.g., Winner [60]—which provide averaged metrics but struggle with accuracy in complex environments. These models often fail to capture interference at the packet level or the impact of environmental obstacles causing reflection, diffraction, and shadowing. To overcome these limitations, integrating ray tracing into system-level vehicular simulations has been explored. For instance, Ruz Nieto et al. [38] integrated the Opal ray tracer [61]—an open source ray tracing engine based on NVIDIA OptiX, using a Shooting and Bouncing Rays (SBR) method—with OMNeT++ to simulate LoRaWAN communications. While promising, the study does not include mobility or standardized V2X protocols. In [40], Rusca et al. developed a MATLAB-based RT framework using real mobility data to compute precise channel parameters. More specifically, the framework computes parameters such as path gain, phase shift, and angle of arrival, focusing on the PHY layer and supporting advanced wireless channel emulation, but lacks a standard-compliant V2X stack and requires commercial MATLAB licenses—an issue common to all MATLAB-based simulators using the built-in RT engine [62]. Gaugel et al. [41] proposed a more precise PHY model in ns-3 by incorporating ray tracer output, enabling bit-level simulation of packet reception. However, the integration is offline and does not support realistic mobility or a complete vehicular stack. A more integrated solution, named *ns3sionna*, was recently presented by Zubow et al. [42], who coupled Sionna RT with ns-3 to simulate IEEE 802.11ac. Their approach relies on ZeroMQ for inter-process communication and focuses on indoor and outdoor WLAN scenarios. In contrast, our work integrates Sionna RT into ms-van3t using lightweight UDP sockets for low-latency IPC as introduced by some of the authors of this paper in [13]. This approach enables accurate simulations that support not only PHY and MAC layers but the entire V2X stack, including support for multiple RATs and

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

realistic vehicle mobility via SUMO, CARLA or real GPS coordinates gathered from the physical world. The proposed framework is uniquely suited for high-fidelity studies of multiple technologies and additionally enables their simultaneous simulation, supporting coexistence analyses in dynamic, multi-technology vehicular environments. Another example specifically targeting V2X communications was recently presented by Twardokus et al. [43], who conducted a security evaluation of C-V2X leveraging a high-fidelity DT using Sionna RT. However, their work does not account for the coexistence of DSRC vehicles operating alongside C-V2X in the same environment.

C. Comparison between existing ray tracers

The computational efficiency of the selected ray tracing engine is critical for large-scale wireless simulations, especially in dynamic vehicular scenarios. Among the methods for ray tracing performance evaluation, benchmarking—i.e., measuring wall-clock execution time on representative inputs—offers a practical alternative to theoretical complexity analysis, as it does not require access to the internal algorithmic structure, not available for closed source commercial software. Following this empirical approach, Zhu et al. [63] compared three ray tracing engines: Remcom Wireless InSite [64], Sionna RT [15], and MathWorks' Ray Tracing Toolbox [62]. Tests were performed on two scenarios: a simplified urban layout with low-resolution meshes, and a photorealistic digital twin generated via CARLA with complex, high-resolution geometry. Results show that Sionna RT consistently achieved the lowest execution times across both scenarios, with especially strong performance in complex scenes when configured to leverage on GPU acceleration and parallelization. Its runtime also scaled well with scene complexity, highlighting its robustness for realistic simulations. However, speed alone does not guarantee accuracy. Therefore, the adoption of Sionna RT within VaN3Twin is not only based on performance alone, but also on the validation of the overall framework against real-world measurements (see Section III). Furthermore, Sionna RT is the only open source option evaluated, which makes it preferable in terms of reproducibility and extensibility.

D. Vehicular digital twins

In recent years, the concept of Digital Twin has gained significant attention as a way to define platforms able to model realistic scenarios and analyze their evolutions through simulated processes. While DTs can be created for many physical systems—e.g., a DT of the mechanical part of a vehicle—they are increasingly adopted to model the behaviors of networks, including vehicular ones with their related mobility patterns. In this context, real-world data are continuously provided to the the V2X DT allowing its models to process them and make decisions subsequently fed back in closed-loop to the real vehicular network. The V2X DT can also be employed to model in advance real-world scenarios—either from environment and mobility models or from previously collected data—to study system evolution under different configurations, for instance to support pre-deployment evaluations of new technologies. Moreover, hybrid approaches are possible, where the V2X DT emulates additional entities that communicate in real time with real-world components, effectively enabling virtual hardware integration while reducing experimental costs. This specific capability is also supported by the VaN3Twin native emulation mode. For a V2X DT to operate in the loop, parallel processing is often required to ensure near-real-time performance. A representative example is the RAVEN [65] platform, which combines vehicle data ingestion with contextual information to predict vehicle positions and channel conditions, thereby managing interference and maximizing the capacity of multi-hop mmWave vehicular networks. Another notable case is the Vehicle-to-Cloud ADAS framework [66], where a cloud-based DT performs ADAS decision processes that are subsequently used as actuation guidance. Additional V2X DT frameworks include TuST [67], [68], a large-scale mobility model replicating the entire city of Turin, Italy, from processed real traffic data. Similarly, Rusca et al. [40] employ recorded mobility traces to construct accurate digital representations of vehicular environments for wireless emulation, while [69] presents a DT of a metro station reproducing the mobility dynamics of a complete indoor pedestrian scenario.

VI. Conclusion

This work introduces VaN3Twin[5], a scalable V2X NDT, capable of replicating the coexistence of heterogeneous technologies with unprecedented accuracy. By integrating the open source ray tracing engine Sionna RT into the full-stack ms-van3t simulator, we enable high-fidelity propagation modeling, scalable simulation across multiple communication technologies, and accurate interference tracking at the time-frequency level. Comparative simulations demonstrate that classical stochastic models produce oversimplified and often misleading results—e.g., up to 49% mismatch in SINR-based reception decisions compared to the ray-based model in Van3Twin. The lack of consideration for the NLoSv scenarios in ms-van3t further broadens this mismatch, leading to 60% disagreement in cases prone to vehicular blockage. Validation against real-world measurements confirms the improved propagation and application level fidelity of VaN3Twin. The use of ray tracing reduces the Packet Reception DR by more than 50% in rural scenarios and over 70% in complex urban environments. These improvements position VaN3Twin as a valuable asset for site-specific validations, large-scale coexistence testing, and standardization support including, for example, the safe pre-deployment evaluation of new features introduced in 3GPP Rel. 18 [70] such as ML-based scheduling, enhanced NR-V2X sidelink, and improved positioning, alongside AI model training with VaN3Twin-generated datasets and support in network planning operations. Future work will offload additional PHY functions—e.g., waveform generation and multi-antenna systems—to GPUs, improving simulation speed and potentially accuracy by reducing the reliance on abstractions. Additionally, it will focus on systematically analyzing the

[5]VaN3Twin and the simulation configuration files and their ray tracing scenarios used for gathering the results presented in this article are available at this GitHub repository: https://github.com/DriveX-devs/VaN3Twin

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

trade-off between computational performance and simulation accuracy for example by tuning ray tracing parameters such as interaction types and maximum interaction depth, and by adapting the NDT update rate to the dynamics of different environments. Another a viable direction for future work is assessing whether Universal Software Radio Peripheral (USRP) hardware and related software (such as OpenAirInterface or srsRAN) can be leveraged for the aim of performing additional field tests with C-V2X to further compare the outputs of VaN3Twin with real measured data. These tests will also be performed with dedicated C-V2X chipsets as compatible hardware will reach the market. Finally, future work can also involve a deeper investigation on the real-time capabilities of VaN3Twin, also leveraging its emulation mode to interface with real-world connected vehicles.

ACKNOWLEDGMENTS

This publication is part of the project PNRR-NGEU which has received funding from the MUR - DM 117/2023. This article was supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART", Structural Project 6GWINET). This study was also carried out within the MOST – Sustainable Mobility National Research Center and received funding from the European Union Next-GenerationEU (PIANO NAZIONALE DI RIPRESA E RESILIENZA (PNRR) – MISSIONE 4 COMPONENTE 2, INVESTIMENTO 1.4 – D.D. 1033 17/06/2022, CN00000023).

REFERENCES

[1] S. Abuelsamid. (2018) Toyota Has Big Plans To Get Cars Talking To Each Other And Infrastructure In The U.S. [Online]. Available: https://www.forbes.com/sites/samabuelsamid/2018/04/16/toyota-launches-aggressive-v2x-communications-roll-out-from-2021/

[2] B. M. Masini, A. Bazzi, and A. Zanella, "A survey on the roadmap to mandate on board connectivity and enable V2V-based vehicular sensor networks," *Sensors*, vol. 18, no. 7, p. 2207, 2018.

[3] 5G Automotive Association, "An assessment of LTE-V2X (PC5) and 802.11 p direct communications technologies for improved road safety in the EU," *5G Automotive Association, Tech. Rep., December*, 2017.

[4] C-Roads Platform. (2024) Webinar: From Vision to Reality - Operational C-ITS Services. [Online]. Available: https://www.c-roads.eu/platform/about/news/News/entry/show/webinar-from-vision-to-reality-operational-c-its-services.html

[5] M. Lucarelli. (2023) Auto, cycling and tech innovators launch Coalition for Cyclist Safety based on V2X deployments. [Online]. Available: https://media.audiusa.com/releases/586

[6] Euro NCAP. (2022) Mercedes-Benz Car-to-X Communication.

[7] Ford Media Center. (2019) Ford Accelerates Connectivity Strategy in China; Targets Production of First C-V2X-Equipped Vehicle in 2021.

[8] A. Bazzi, A. Zanella, I. Sarris, and V. Martinez, "Co-channel coexistence: Let ITS-G5 and sidelink C-V2X make peace," in *2020 IEEE MTT-S International Conference on Microwaves for Intelligent Mobility (ICMIM)*. IEEE, 2020, pp. 1–4.

[9] P. Roux and V. Mannoni, "Performance evaluation for co-channel coexistence between ITS-G5 and LTE-V2X," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*. IEEE, 2020, pp. 1–5.

[10] CAR 2 CAR Communication Consortium, "White paper on Additional Investigation of ITS-G5 and Sidelink LTE-V2X Co-Channel Coexistence Methods," White Paper, Tech. Rep., 2022.

[11] M. Rapelli, F. Raviglione, and C. Casetti, "OScar: An ETSI-Compliant C-ITS Stack for Field-Testing with Embedded Hardware Devices," in *2024 22nd Mediterranean Communication and Computer Networking Conference (MedComNet)*. IEEE, 2024, pp. 1–4.

[12] S. Roongpraiwan, Z. Li, T. Yu, and K. Sakaguchi, "Digital twin-enabled blockage-aware dynamic mmwave multi-hop v2x communication," 2025. [Online]. Available: https://arxiv.org/abs/2503.03590

[13] R. Pegurri, F. Linsalata, E. Moro, J. Hoydis, and U. Spagnolini, "Toward digital network twins: Integrating sionna RT in ns-3 for 6G Multi-RAT networks simulations," in *IEEE INFOCOM WKSHPS: Digital Twins over NextG Wireless Networks (DTWIN 2025) (INFOCOM DTWIN 2025)*, London, United Kingdom (Great Britain), May 2025, p. 5.88.

[14] G. F. Riley and T. R. Henderson, *The ns-3 Network Simulator*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 15–34. [Online]. Available: https://doi.org/10.1007/978-3-642-12331-3_2

[15] J. Hoydis, S. Cammerer, F. Ait Aoudia, A. Vem, N. Binder, G. Marcus, and A. Keller, "Sionna: An Open-Source Library for Next-Generation Physical Layer Research," *arXiv preprint*, Mar. 2022.

[16] F. Raviglione, C. R. Carletti, M. Malinverno, C. Casetti, and C. Chiasserini, "ms-van3t: An integrated multi-stack framework for virtual validation of V2X communication and services," *Computer Communications*, vol. 217, pp. 70–86, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366424000227

[17] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic traffic simulation using sumo," in *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018. [Online]. Available: https://elib.dlr.de/124092/

[18] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "Carla: An open urban driving simulator," in *Conference on robot learning*. PMLR, 2017, pp. 1–16.

[19] "Traci," https://sumo.dlr.de/docs/TraCI.html.

[20] OpenStreetMap contributors, "Planet dump retrieved from https://planet.osm.org ," https://www.openstreetmap.org, 2017.

[21] M. Lecci, P. Testolina, M. Polese, M. Giordani, and M. Zorzi, "Accuracy versus complexity for mmwave ray-tracing: A full stack perspective," *IEEE Transactions on Wireless Communications*, vol. 20, no. 12, pp. 7826–7841, 2021.

[22] Google Maps Data, ©2025 Airbus, Maxar Technologies. [Online]. Available: https://www.google.com/maps

[23] PC Engines, "APU2E4 System board schema." [Online]. Available: https://www.pcengines.ch/apu2e4.htm

[24] ArduSimple, "simpleRTK2B Fusion description and specifications." [Online]. Available: https://www.ardusimple.com/product/simplertk2b-fusion/

[25] F. Raviglione, M. Malinverno, and C. Casetti, "Open source platform for IEEE 802.11 p NICs evaluation," in *2019 IEEE 20th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM)*. IEEE, 2019, pp. 1–3.

[26] MobileMark, "MGW-313 description and specifications." [Online]. Available: https://www.mobilemark.com/product/smwg-313-2xdsrc-gns/

[27] ITU-R P.2040-1, "Effects of building materials and structures on radiowave propagation above about 100 MHz," Jul. 2015.

[28] G. Liu and L. Kong, "Simulation of video streaming over wireless networks with ns-3," 2023. [Online]. Available: https://arxiv.org/abs/2302.14196

[29] "3GPP TR 36.885: Study on LTE-based V2X Services," 3rd Generation Partnership Project (3GPP), Technical Report 36.885, 2016, version 14.0.0. [Online]. Available: https://www.3gpp.org

[30] "3GPP TR 38.901: Study on channel model for frequencies from 0.5 to 100 GHz," 3rd Generation Partnership Project (3GPP), Technical Report 38.901, 2017, version 14.0.0. [Online]. Available: https://www.3gpp.org

[31] Z. Ali, S. Lagén, L. Giupponi, and R. Rouil, "3gpp nr v2x mode 2: Overview, models and system-level evaluation," *IEEE Access*, vol. 9, pp. 89 554–89 579, 2021.

[32] M. Drago, T. Zugno, M. Polese, M. Giordani, and M. Zorzi, "Millicar: An ns-3 module for mmwave nr v2x networks," in *Proceedings of the 2020 Workshop on ns-3*, 2020, pp. 9–16.

[33] V. Todisco, S. Bartoletti, C. Campolo, A. Molinaro, A. O. Berthet, and A. Bazzi, "Performance analysis of sidelink 5g-v2x mode 2 through an open-source simulator," *IEEE Access*, vol. 9, pp. 145 648–145 661, 2021.

[34] A. Bazzi, G. Cecchini, M. Menarini, B. M. Masini, and A. Zanella, "Survey and perspectives of vehicular wi-fi versus sidelink cellular-v2x in the 5g era," *Future Internet*, vol. 11, no. 6, 2019. [Online]. Available: https://www.mdpi.com/1999-5903/11/6/122

[35] S. Jayaweera, S. Manosha, N. Rajatheva, and M. Latva-aho, "Coexistence of its-g5 and c-v2x at an urban road intersection," in *2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, 2021, pp. 253–258.

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Wireless Communications. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TWC.2026.3685710

[36] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011.

[37] B. McCarthy and A. O'Driscoll, "Opencv2x mode 4: A simulation extension for cellular vehicular communication networks," in *2019 IEEE 24th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2019, pp. 1–6.

[38] A. Ruz-Nieto, E. Egea-Lopez, J.-M. Molina-Garcıa-Pardo, and J. Santa, "A 3D simulation framework with ray-tracing propagation for LoRaWAN communication," *Internet of Things*, vol. 24, p. 100964, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542660523002871

[39] R. Riebl, H.-J. Günther, C. Facchi, and L. Wolf, "Artery: Extending veins for vanet applications," in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, 2015, pp. 450–456.

[40] R. Rusca, F. Raviglione, C. Casetti, P. Giaccone, and F. Restuccia, "Mobile RF Scenario Design for Massive-Scale Wireless Channel Emulators," in *2023 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit)*, 2023, pp. 675–680.

[41] T. Gaugel, L. Reichardt, J. Mittag, T. Zwick, and H. Hartenstein, "Accurate simulation of wireless vehicular networks based on ray tracing and physical layer simulation," in *High Performance Computing in Science and Engineering'11: Transactions of the High Performance Computing Center, Stuttgart (HLRS) 2011*. Springer, 2012, pp. 619–630.

[42] A. Zubow, Y. Pilz, S. Rösler, and F. Dressler, "Ns3 meets Sionna: Using Realistic Channels in Network Simulation," arXiv, cs.NI 2412.20524, 12 2024.

[43] G. Twardokus and H. Rahbari, "DT-CoVeSS: Advancing NextG C-V2X Security Evaluation through High-Fidelity Digital Twinning," in *IEEE INFOCOM WKSHPS: Digital Twins over NextG Wireless Networks (DTWIN 2025) (INFOCOM DTWIN 2025)*, London, United Kingdom (Great Britain), May 2025.

[44] M. Malinverno, F. Raviglione, C. Casetti, C.-F. Chiasserini, J. Mangues-Bafalluy, and M. Requena-Esteso, "A multi-stack simulation framework for vehicular applications testing," in *Proceedings of the 10th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, ser. DIVANet '20. New York, NY, USA: Association for Computing Machinery, 2020, p. 17–24. [Online]. Available: https://doi.org/10.1145/3416014.3424603

[45] B. Williams, *Automated Driving Levels*, 2021, pp. 19–41.

[46] L. Cazzella, F. Linsalata, M. Magarini, M. Matteucci, and U. Spagnolini, "A multi-modal simulation framework to enable digital twin-based v2x communications in dynamic environments," in *2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*, 2024, pp. 1–6.

[47] V. Vukadinovic, K. Bakowski, P. Marsch, I. D. Garcia, H. Xu, M. Sybis, P. Sroka, K. Wesolowski, D. Lister, and I. Thibault, "3gpp c-v2x and ieee 802.11p for vehicle-to-vehicle communications in highway platooning scenarios," *Ad Hoc Networks*, vol. 74, pp. 17–29, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S157087051830057X

[48] R. Molina-Masegosa, J. Gozalvez, and M. Sepulcre, "Comparison of IEEE 802.11p and LTE-V2X: An Evaluation With Periodic and Aperiodic Messages of Constant and Variable Size," *IEEE Access*, vol. 8, pp. 121 526–121 548, 2020.

[49] E. Moradi-Pari, D. Tian, M. Bahramgiri, S. Rajab, and S. Bai, "Dsrc versus lte-v2x: Empirical performance analysis of direct vehicular communication technologies," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 5, pp. 4889–4903, 2023.

[50] F. Raviglione, M. Malinverno, S. Feraco, G. Avino, C. Casetti, C. F. Chiasserini, N. Amati, and J. Widmer, "Experimental assessment of ieee 802.11-based v2i communications," in *Proceedings of the 18th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks*, ser. PE-WASUN '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 33–40. [Online]. Available: https://doi.org/10.1145/3479240.3488506

[51] P. Testolina, M. Polese, J. M. Jornet, T. Melodia, and M. Zorzi, "Modeling Interference for the Coexistence of 6G Networks and Passive Sensing Systems," *IEEE Transactions on Wireless Communications*, vol. 23, no. 8, pp. 9220–9234, 2024.

[52] V. Maglogiannis, D. Naudts, S. Hadiwardoyo, D. van den Akker, J. Marquez-Barja, and I. Moerman, "Experimental v2x evaluation for c-v2x and its-g5 technologies in a real-life highway environment," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1521–1538, 2022.

[53] M. Elloumi, M. Z. Hassan, and G. Kaddoum, "Spectrum sharing in internet-of-vehicles networks: Digital twin-empowered proactive interference management approach," *IEEE Transactions on Network and Service Management*, pp. 1–1, 2025.

[54] V. Maglogiannis, D. Naudts, S. Hadiwardoyo, D. van den Akker, J. Marquez-Barja, and I. Moerman, "Experimental v2x evaluation for c-v2x and its-g5 technologies in a real-life highway environment," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 1521–1538, 2022.

[55] A. Bazzi, S. Bartoletti, A. Zanella, and V. Martinez, "Performance analysis of ieee 802.11p preamble insertion in c-v2x sidelink signals for co-channel coexistence," *Vehicular Communications*, vol. 45, p. 100710, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214209623001407

[56] M. H. Manshaei, M. Krunz, and A. Yousseef, "Performance evaluation of dsrc and c-v2x coexistence in the 5.895-5.925 ghz spectrum," in *2024 International Conference on Computing, Networking and Communications (ICNC)*, 2024, pp. 862–867.

[57] F. Eckermann, M. Kahlert, and C. Wietfeld, "Performance Analysis of C-V2X Mode 4 Communication Introducing an Open-Source C-V2X Simulator," in *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, 2019, pp. 1–5.

[58] Z. Ali, S. Lagén, L. Giupponi, and R. Rouil, "3gpp nr v2x mode 2: Overview, models and system-level evaluation," *IEEE Access*, vol. 9, pp. 89 554–89 579, 2021.

[59] 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*, ser. Simutools '08. Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008.

[60] P. Kyösti, J. Meinilä, L. Hentila, X. Zhao, T. Jämsä, C. Schneider, M. Narandzic, M. Milojević, A. Hong, J. Ylitalo, V.-M. Holappa, M. Alatossava, R. Bultitude, Y. Jong, and T. Rautiainen, "WINNER II channel models," *IST-4-027756 WINNER II D1.1.2 V1.2*, 02 2008.

[61] E. Egea-Lopez, J. M. Molina-Garcia-Pardo, M. Lienard, and P. Degauque, "Opal: An open source ray-tracing propagation simulator for electromagnetic characterization," *PLOS ONE*, vol. 16, no. 11, pp. 1–19, 11 2021. [Online]. Available: https://doi.org/10.1371/journal.pone.0260060

[62] The MathWorks, Inc. (2024) Ray Tracing for Wireless Communications. [Online]. Available: https://www.mathworks.com/help/comm/ug/ray-tracing-for-wireless-communications.html

[63] M. Zhu, L. Cazzella, F. Linsalata, M. Magarini, M. Matteucci, and U. Spagnolini, "Toward real-time digital twins of em environments: Computational benchmark for ray launching software," *IEEE Open Journal of the Communications Society*, 2024.

[64] Remcom Wireless InSite. [Online]. Available: https://www.remcom.com/wireless-insite-em-propagation-software

[65] M. Elloumi, G. Kaddoum, M. Zoheb Hassan, and B. Selim, "Digital-twin-empowered interference management for multihop internet of vehicles networks over millimeter wave bands," *IEEE Internet of Things Journal*, vol. 12, no. 11, pp. 17 807–17 827, 2025.

[66] Z. Wang, X. Liao, X. Zhao, K. Han, P. Tiwari, M. J. Barth, and G. Wu, "A digital twin paradigm: Vehicle-to-cloud based advanced driver assistance systems," in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*, 2020, pp. 1–6.

[67] M. Rapelli, C. Casetti, and G. Gagliardi, "Tust: from raw data to vehicular traffic simulation in turin," in *2019 IEEE/ACM 23rd International Symposium on Distributed Simulation and Real Time Applications (DS-RT)*, 2019, pp. 1–8.

[68] ——, "Vehicular traffic simulation in the city of turin from raw data," *IEEE Transactions on Mobile Computing*, vol. 21, no. 12, pp. 4656–4666, 2022.

[69] M. Rapelli, G. S. Pannu, F. Dressler, and C. Casetti, "Content sharing in pedestrian-based micro clouds," in *2022 IEEE 95th Vehicular Technology Conference: (VTC2022-Spring)*, 2022, pp. 1–6.

[70] 3rd Generation Partnership Project. (2022) 3GPP Release 18. [Online]. Available: https://www.3gpp.org/release18

Authorized licensed use limited to: Politecnico di Torino. Downloaded on May 10,2026 at 09:43:21 UTC from IEEE Xplore. Restrictions apply.
© 2026 IEEE. All rights reserved, including rights for text and data mining and training of artificial intelligence and similar technologies. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.