Evaluating the Impact of a 3D Simulation Model on the Performance of Vehicular Networks

Enrico Zanotto, Leonardo Maccari
Dept. of Environmental Sciences, Informatics and Statistics
Ca' Foscari University of Venice
Venice, Italy
887566@stud.unive.it, leonardo.maccari@unive.it

***Abstract*—A crucial aspect of vehicular networks is the wireless communication between a vehicle and the road infrastructure, which is strongly influenced by various three-dimensional factors, including the height of buildings, the presence of natural and artificial obstacles, the altitude, and the positioning of the antennas. 2D simulators fail to capture these influences fully, as they simplify and flatten the terrain. This paper presents the extension of the Veins vehicular network simulators to take into account the third dimension. We describe the key algorithms we implemented, the limited increased simulation time, and most of all the improved accuracy in evaluating the coverage of a mobile vehicular network.**

***Index Terms*—Vanet, vehicles coverage, mobile networks, simulation, omnetpp**

I. Introduction

Vehicular networks are a largely studied topic in the networking research area, but their features are still far from being fully understood. One factor that limits the progress of research is the lack of precise and efficient software tools to simulate every detail of a realistic vehicular network. Much has been done in recent years, for instance, open source tools like SUMO and Veins [1], [2] allow to study a network using realistic vehicles movement patterns and detailed wireless communication models, however, the more the technology improves, the higher is the complexity it introduces and the requirements for the simulator.

As an example, consider the increase in the communication frequency that started with 5G and continues in the current discussion on 6G. 5G opened the way to the use of mmWave frequencies around 27GHz, and 6G is expected to support the sub-THz spectrum. While on the one hand this allows to access a large unused portion of the spectrum, on the other it poses significant challenges due to the harsh propagation conditions. *Non Line-Of-Sight* (NLOS) communication is greatly penalized compared to *Line-Of-Sight* (LOS) communication, calling for a denser deployment of the *Base Station*s (BSs)[1]. This in turn opens new research challenges related to estimating network coverage [3] or to the introduction of repeaters and intelligent surfaces [4].

To correctly model modern vehicular networks, simulators need to catch up with this complexity, even at the cost of requiring larger computational resources. One feature that has lagged is the availability of models that correctly represent the third dimension in vehicular networks. This is due to the added algorithmic complexity that is introduced when the scenario includes three-dimensional objects, but also to the scarce availability of easily usable open data to model the 3D shapes of urban areas, and finally, to the computational overhead that a 3D simulation introduces. However, the necessity to include the third dimension in network simulators has been recently verified [5] and the availability of open data is increasing, together with the familiarity of researchers with its use to correctly represent modern networks [6], [7].

This paper contributes to this research direction by extending the well-known Veins simulator to support the simulation of 3D obstacles. Veins already includes a path loss model that takes into consideration 2D maps, and it is able to compute an additional loss due to the wireless communication passing through an obstacle. This approach can be used to model vehicle-to-vehicle communication, that happens between devices whose antenna is generally placed between 1 and 2 meters from the ground but fails to represent a 3D scenario in which the BS is placed in a higher position, and thus, the ray that ideally represents the communication can simply pass on top of an obstacle. With the enhancement proposed in this paper Veins can exploit the availability of open maps and open altitude data-sets to model a realistic urban environment. We show that there is a striking difference between the use of the original 2D simulator and our 3D simulator with a limited penalty in the computational overhead. We quantify this difference in terms of frames correctly delivered to the vehicles, and in the possibility of BSs to communicate directly from one to the other, an approach that has attracted interest to reduce the costs of wired backhauling [8].

The rest of the paper is organized as follows: in section II we introduce the state of the art, and we position our contribution in the recent trend of developments of 3D simulators for vehicular networks; in section III we describe the current state of Veins, and motivate the improvements we propose; in section IV we give an overview of the algorithms used; in section V we explain the simulation scenario; in section VI we illustrate the numerical results. Section VII

[1]As we don't want to target a specific technology we use the generic BS term to indicate the infrastructure nodes, instead of specific terms introduced by any communication standard.

ISBN: 978-3-903176-61-4

© 2024 International Federation
for Information Processing (IFIP).
ISBN: 978-3-903176-61-4

concludes the paper.

II. Background

Simulating mobile networks using a three-dimensional approach is a studied subject, as it is intuitive that the performance of a mobile network is strongly influenced by the presence of obstacles. However the available open source tools are scarce and thus, the literature that provides reproducible results is limited. We start mentioning some works based on the Network Simulator open source software. Black et al. presented a visualizer for the NS-3 simulator that can represent 3D scenarios in order to more easily debug the simulation [9] while Regis et al. introduced a 3D mobility model for UAV, but did not focus on the signal propagation [10].

More relevant to our context are the works based on the Omnet++ simulator that includes the Veins package, a detailed, open source vehicular networks simulator [2]. Veins introduces LOS/NLOS estimation using a 2D representation of the scenario and an object library, that can assess if a ray (the ideal line between the sender and the receiver radio) intercepts an obstacle. Obstacle shapes can be extracted from an OpenStreetMap map.

A set of works sharing the same core group of authors advanced Veins in recent years. The necessity of the third dimension was first introduced by Brummer et al. [5], later on, Deinlein et al. proposed 5G-Sim-V2I/N that introduced very detailed models for estimating 5G coverage in urban areas. The simulator extends Veins introducing several 3D-related improvements, but still uses the Veins 2D obstacle intersection detection library [11]. An improved version of the simulator was introduced by the same authors later on, in which the Veins 2D model was coupled with a simplified 3D topology representation, used to determine if LOS was present or not in order to apply the corresponding LOS/NLOS ITU channel model [12]. Finally, the challenges of 3D vehicular network simulators not only lie in LOS/NLOS estimation due to buildings, but also in the use of accurate path loss models, the impact of mobile obstacles, and the use of realistic antenna patterns. These challenges were recently summarized and exposed [7].

Our approach is complementary to the previous ones and can enrich Veins furthermore. In order to realistically estimate the impact of 3D obstacles, a triangulation algorithm that supports generic shapes is needed, that is what we propose in this paper. We improve the 2D obstacle detection of Veins to consider prisms: three-dimensional polyhedrons with parallel and congruent bases. The bases are extracted by OpenStreetMap maps and are attributed a certain elevation, thus creating the prism. We then tessellate the prism dividing all its faces into triangles that are used to perform ray-tracing. The result of the algorithm application is not only a LOS/NLOS estimation, but, for each ray, the number of intersections with any building face, and the fraction of the length of the ray that passed inside an object. These data feed the Veins path loss model that uses a different decay exponent for the portion of ray that passed through free space or outside it. In essence, we are extending the approach of Veins to a truly 3D approach.

Our contribution is compatible with other simulation modules based on Veins, like the mentioned works by Deinlein et al. [7], [12], that provide complementary features. Once merged all together, these results can be used to improve the works that try to estimate the cost, effectiveness, or robustness of the deployment of a mobile network for vehicles or pedestrians [3] improving the realism of their results.

It is important to note that we limit our analysis to prisms not because our approach can not be used on generic polyhedrons, in fact, it can be used on any 3D object tessellated with triangles. However, reconstructing a generic 3D shape from available open data is not an easy task, and would require an effort beyond the scope of this paper.

III. Goals: From 2D Veins to 3D Veins

Veins incorporates a model of IEEE 802.11p, and the upper-layer communication stack of the DSRC/WAVE [13]. Veins allows for the periodic transmission of beacons, such as Wave Service Announcements (WSA), Basic Safety Messages (BSM), or Cooperative Awareness Messages (CAM). The obstacle model in Veins has been developed to accurately represent the impact of buildings on radio signal propagation in vehicular networks, particularly crucial in suburban and urban environments. The model was introduced starting from Veins 1.99.2 and can be activated by adding an obstacle control module (ObstacleControl) in the configuration file[2]. Veins adopts a representation of obstacles, including buildings, through two-dimensional polygons defined by a set of vertices in space, accurately outlining their perimeter. This simplification of obstacle geometry, while maintaining an adequate level of realism, enables easier and faster calculations during simulations. Obstacles are provided in a specifically formatted XML compatible with the OpenStreetMap standards.

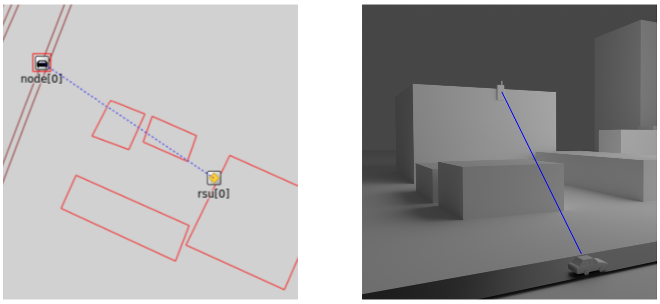
The path loss model is a classical two-ray ground in which the height of the two antennas can be specified, with an additional loss for rays that pass through obstacles. What is needed to know is the number of walls the ray passes through and the fraction of its length that is included inside any number of buildings. Veins introduces two parameters to evaluate the path loss through an obstacle:

- *dbm-per-cut*: This parameter represents the power loss in dBm of the radio signal for each time the ray enters or exits a building section. Regardless of the section's length, the simulator consistently reduces the received power by this quantity.
- *dbm-per-meter*: This parameter is proportional to the distance traveled by the signal inside the building and changes the exponent of the path loss function.

The general idea that motivates the need for a 3D simulator is that a 2D scenario is perceived to be less accurate than a

[2]See https://veins.car2x.org/documentation/modules/

ISBN: 978-3-903176-61-4



(a) (b)

Fig. 1: Example scenario: the path loss in the 2D representation (a) is higher than the one in the 3D representation (b).

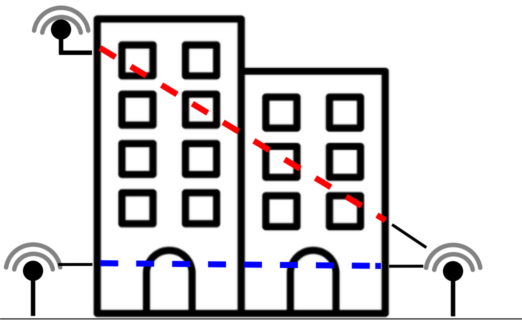


Fig. 2: The red segment is longer than the blue one, so the 3D simulator will compute a higher path loss than the 2D simulator.

3D one, as shown in fig. 1. This is particularly relevant for the infrastructure-to-vehicle communications as we expect the antenna on the BS to be higher than the one on the vehicle.

In some cases, however, the simplified models based on a 2D scenario could produce a better result than a 3D scenario, as depicted in fig. 2. This is due to the fact that the 3D scenario can precisely estimate the fraction of the 3D ray that passes through the building, while the 2D scenario can not. Even using the two-ray ground model that considers the height of the antennas, there is no correct approximation that can be used to estimate the fraction of the ray that passes inside the building without using a fully 3D model. The goal of the paper is to extend the logic used by Veins in the 2D context to 3D simulations and evaluate the difference between the two models. We will populate a realistic urban scenario with base stations and vehicles, generate periodic packets from the base stations, and report several measures that outline the difference between the two models.

Extending the 2D model to a 3D one requires two main additional features: the generation of a 3D mesh, and a ray-tracing algorithm.

A. *Generating a 3D mesh*

A 3D mesh is a three-dimensional representation of objects or structures that defines their surface through a grid of finite elements, essentially triangles. This triangle grid is used to subdivide the three-dimensional surface of the object into a series of triangular faces, each of which represents a portion of the overall surface. Meshes provide a discrete representation of the obstacle and are created by extruding the polygon that represents the building floor plan.

To implement this functionality, ObstacleControl will be tasked not only with calculating Path Loss but also with generating three-dimensional meshes through specific algorithms.

B. *Intersection and Path Loss Calculation*

With the introduction of three-dimensional meshes, it becomes necessary to compute intersections between line segments representing radio signals and the triangles constituting these meshes. Calculating intersections between segments and volumes requires a higher level of detail and complexity compared to simple 2D segment intersections. The result is a more realistic representation of path losses.

IV. The 3D mesh Creation Algorithms

The generation of 3D meshes involves creating a three-dimensional representation of the buildings themselves. This process mainly consists of two phases: the extrusion of the base polygon and the tessellation of the generated surfaces. Extrusion is the process by which a two-dimensional surface, such as a building floor plan, is vertically extended to create a three-dimensional representation. During extrusion, the heights and overall shape of the buildings are taken into account. This process generates the vertical facades of the buildings, creating a basic structure for the three-dimensional mesh. Our modified version of Veins uses the polygons obtained by OpenStreetMap as the base of the 3D building.

The next phase is surface tessellation, during which the extruded surfaces are subdivided into smaller triangles. This subdivision is necessary to discretely represent the three-dimensional surfaces and enable precise calculations within the simulation. The choice of the tessellation algorithm may depend on the specific simulation requirements and the complexity of the involved buildings. In this work, we implemented the Sweep Line algorithm for the triangulation of monotone polygons used in the domain of computational geometry [14].

A monotone polygon in geometry is defined in relation to a line L. A polygon P is said to be monotone with respect to a line L if every line orthogonal to L intersects the boundary of P at most twice. Figure 3 shows a monotone and non-monotone polygon with respect to the line L. A monotone polygon can be easily tessellated with triangles, scanning it from left to right, and connecting each vertex that is matched, with the the previous ones that were not connected yet. The figure shows how tessellation is realized by connecting in sequence point 3 to point 2, point 4 to point 3, point 5 to point 3, and so on. A non monotone polygon instead can not be easily tessellated with the same algorithm, for instance, point 3 in the non monotone polygon of fig. 3 can

ISBN: 978-3-903176-61-4

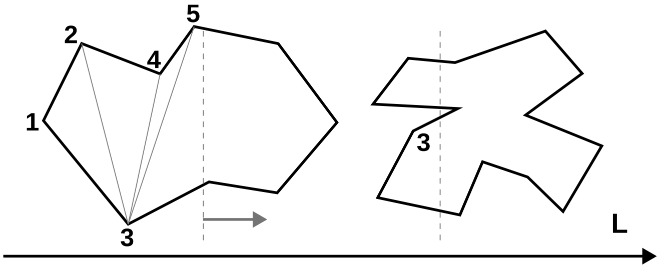


Fig. 3: Two polygons, the left one is monotone with respect to line L, while the right one is not.

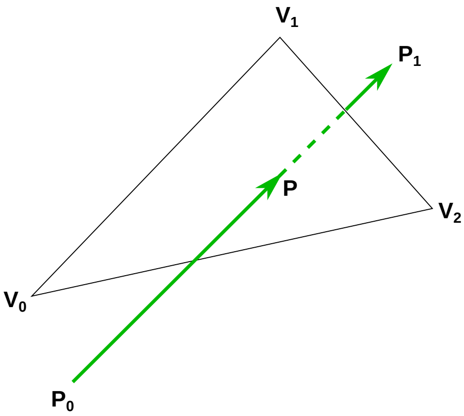


Fig. 4: The intersection between a ray and a triangle.

not be connected with the points that lie on its left because the connecting segment would lie outside of the polygon itself. The solution to this problem is to use an algorithm that identifies the vertices that create split and merges in the polygon and use them to partition a non monotone polygon into smaller monotone ones. The details of the algorithm are out of the scope of this paper and can be found in the related literature [14], its complexity is $O(n \log(n))$ where n is the number of vertices.

We note that the bases of the prism could be non monotone, while the sides are always rectangles orthogonal to the base, and thus are monotone by construction. Once the whole scenario has been tessellated, we can use this spatial representation to detect the intersections with a ray representing a communication link. Given the 3D coordinates of one triangle, as shown in fig. 4, verifying if the point P exists and obtaining its coordinates can be done in constant time. Repeating the same process on all the triangles that tessellate a building we can obtain the list of all the intersections with the triangles representing a prism and thus, the fraction of the ray that passes across the building. Note that if the base is non monotone, a ray can intersect a prism more than two times, and we will consider all the intersections. Once we have the data, during the simulations we use the *dbm-per-cut* and *dbm-per-meter* parameters to compute the path loss.

V. Evaluation Set-Up

We consider a 4 square km portion of the city of Erlangen depicted in fig. 5, that is one of the base scenarios already included in Veins. The city map is extracted from OpenStreetMap, that does not provide the building heights. While in previous works we were able to estimate the height of buildings using open data [3], in this work we decided to use a simpler approach, as our goal is to verify the impact of the 3D model and not to optimize the coverage for a specific city. We then used the open data for the city of Milan (Italy) to estimate a distribution of the building height and we applied the same distribution to our scenario. Considering the center of Milan, we mapped the building height to a normal distribution with $\mu = 18.404621$ and $\sigma = 10.221049$[3]. This makes the results easy to replicate.

In the chosen map we positioned 50 BSs, using two approaches. In the first one, we positioned the BSs in the middle of the crossroads, increasing the height from 5 to 20 meters with 5-meter steps (as said, in the 2D model the antenna height is considered in the two-ray ground model). We refer to this configuration as the *crossroads scenario*. These heights are compatible with the the *International Telecommunication Union* (ITU) recommendations that suggest a BS height between 10 and 25 meters above the ground for urban micro and macro cells respectively [15]. This is a case in which we expect the 2D simulation to behave close to the 3D simulation, as street crossings are the locations that have the best LOS visibility to the points on the street ground where vehicles are moving. We manually placed the BSs in order to achieve an even coverage of the whole area.

In the second approach, we positioned the BSs on the facade of the 50 tallest building, at a roughly one meter distance from the building, and the height of the BS is determined by the building height. In this second approach, we expected the 3D simulation to perform better than the 2D simulation. We refer to this configuration as the *buildings scenario*.

We simulated 2000 seconds, every second a new vehicle is added to the simulation in a random position, with a random destination. The vehicles then follow the realistic dynamics implemented by the SUMO simulator. Every second every BS generates a beacon frame, and after the computation of the path loss, the Veins 802.11p radio model evaluates if the packet is correctly received or not. In the crossroads scenario we generate roughly between 1.7 and 2.8 million received frames, while in the scenario using building facades we generate between 1.2 and 3.6 million received frames. The BSs use a transmission power of 1W, the frequency is the standard 5.9GHz used in 802.11p.

[3]Note that given a digital surface model it is not easy to distinguish the building altitude from the ground altitude. In our approach we divided the area in squares of 500m of side and estimated the building height as the average elevation of the roof points with the lowest point in the square.

ISBN: 978-3-903176-61-4

A. The Evaluation Metrics

We evaluate the simulations using metrics that capture different aspects of wireless communications. The first metric is the number of received frames by the vehicles in the whole simulation, varying the height of the antennas.

The second metric is the cumulative distribution of the number of frames received by each car in intervals of 1 second. More formally, we divide the simulation time into discrete intervals of one second length indexed with t_j, we consider vehicle i and we call $r_{i,j}(t_j, t_{j+1})$ the number of frames correctly received by i in the interval starting at time t_j. For each vehicle, for each interval we compute $r_{i,j}$ and then report the cumulative distribution of the set of values $\{r_{i,j}\ \forall i, j\}$. We also evaluate the *Received Signal Strenght Indicator* (RSSI) of the received frames (even if for brevity we report it only for the buildings scenario).

As a further evaluation metric, we report the feasibility of links between the BSs themselves. We add this metric in light of the recent renewed interest in wireless backhauls based on a mesh network model generated by the introduction of *Integrated Access and Backhaul* (IAB) in 5G. A 5G BS (gNodeB, in the 5G terminology) can be fiber-connected to the core network or it can use beamforming to create a wireless link to some other BS, creating a mesh topology that reaches some fiber-connected BS. This is made easier by the use of mmWave communications that allow to have very narrow beams. Using wireless backhaul links can reduce the number of fiber-connected BSs and thus the overall cost of the network. This is an active field of research [16] that requires precise 3D modelling, as the availability of LOS strongly impacts high-frequency links.

This concept can not be easily ported to lower frequencies (like the standard 5.9GHz used for 802.11p) but it is indeed interesting to evaluate the impact of a real 3D simulator in the estimation of the feasible links. Thus, for each BSs we verify if communication is possible or not (that is, frames are received by the radio) and we report the number of potential links that could be used to create a mesh backhaul among the BSs.

Finally, we report the time needed for the simulations on a 16 core Intel Xeon Gold 6348 CPU @ 2.60GHz, to account for the increased complexity of the 3D model. A correctly received frame must be managed by the upper layers in each node, and thus it triggers a higher number of events compared to one that is discarded by the radio. Thus, alongside the plain simulation time, we also report the simulation time divided by the number of correctly received frames.

B. Limitations

It is worth to also mention the limitations of this work, in the first place, due to time restrictions we were able to run the simulation using only one 3D scenario. Our simulations involve thousands of cars with different trajectories and millions of received frames, however, in future works we will extend the results with more maps and a better estimation of the results confidence. Initial results with multiple runs confirm the observation of the paper. We are also aware that the LOS/NLOS factor is only one among the many that influence communication: reflections, and Fresnel zone occupation are others. In light of this we will try to integrate[4] our code with other existing modules that take into account other factors, and that can benefit from our development [7].

Fig. 5: The map used in the simulation, with the positions of the BSs.

VI. Results

A. Received Frames in the Crossroads Scenario

Figure 6 reports the number of received frames in the crossroads scenario, for the 2D and 3D simulations and an increasing BS height. It can be clearly noticed that with the 3D scenario, the number of received frames is significantly higher at all BS heights. For BSs that are placed as low as 5 meters above the ground, we measured a 5.5% increase, and the gap increases when we position the BS at a higher level, up to +51.4% when BS are placed at 20 meters above the ground. As said, the ITU suggests useing a reference height between 10 and 25 meters, so the chosen range is a realistic one. A second, less visible trend is the slight decrease in the number of received frames in the 2D model, probably due to the longer distance from source to destination when the BS height is increased and the two-ray ground model is used. This first set of results shows that the 3D simulation, as expected, delivers more frames than the 2D simulation. This is however true on average, not for all the BSs, there are cases in which the 2D simulation delivers more frames than the 3D simulation, so both (the ameliorative and pejorative effect of 3D simulations) are present with a higher impact of the first one.

The second result we present is the cumulative distribution of the number of frames received per vehicle per time interval of 1 second. This is particularly relevant if we assume a

[4]The code is currently available at https://github.com/Zanotto-Enrico/veins-3d.

ISBN: 978-3-903176-61-4

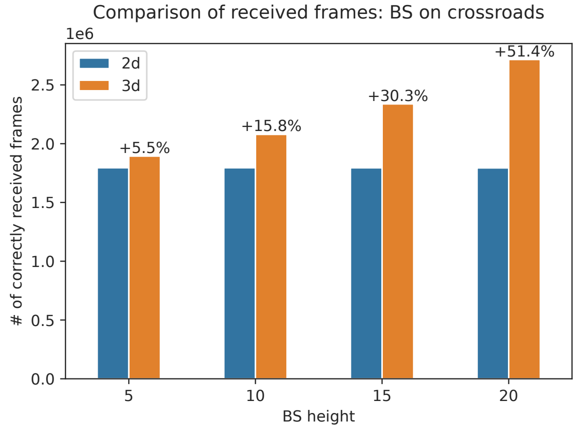


Fig. 6: The number of received frames, for the 2D and 3D simulations with increasing BG height

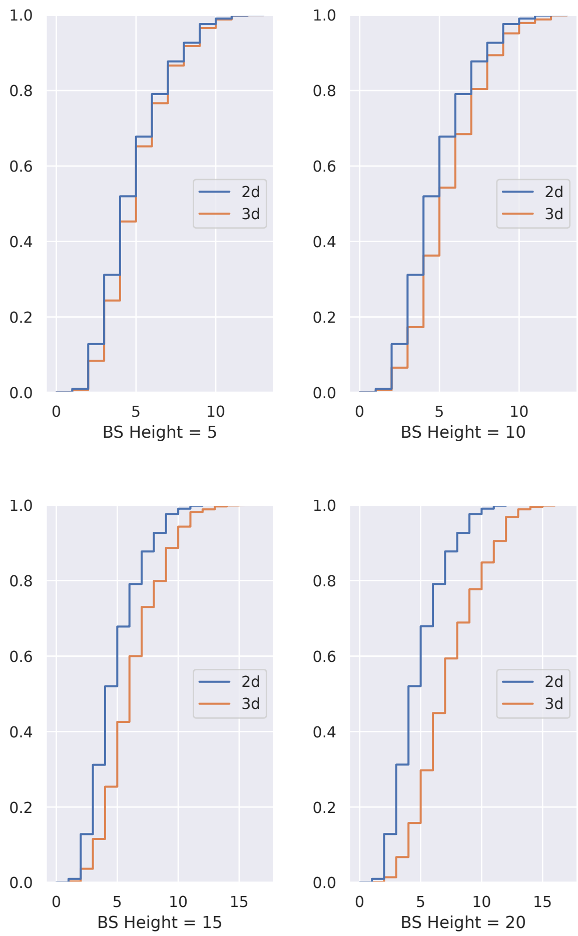


Fig. 7: The ECDF of the number of received frames per second, increasing the BS height.

road safety application, in which the vehicles should receive periodic updates on the status of the road, and modify their behavior accordingly. The overall average number of received frames in fact does not capture the variability due to different locations while the time-sampled values are more meaningful to show the fairness of the number of received frames. Figure 7 reports the whole CDF for all vehicles, all BS heights and both simulators. We can clearly observe that the CDF for the 3D simulator shifts right starting from the very beginning of the curve. This is important because it shows that the increased number of delivered frames happens not only in a small fraction of vehicles and time intervals but in the whole distribution, with the 3D simulation having also a higher maximum value. In practical terms, this means that the average we have shown in fig. 6 is not due to a small number of vehicles that receive a disproportionate number of frames, but is distributed along all time intervals and vehicles. We should recall that our simulations do not consider the interference due to obstacles that are not the buildings themselves, so it is surely optimistic in this sense. In a real-world scenario, the probability of failed delivery is higher and, for instance, the vehicles that receive two or less frames per second are at risk of being shadowed by other obstacles. If we consider, for instance, a simulation whose goal is to evaluate the positioning of BSs for reliable delivery of safety information then the effect of using 3D simulations appears evident from the very beginning of the curve, providing a more optimistic evaluation compared to the results of the 2D simulations in the critical, initial part of the distribution.

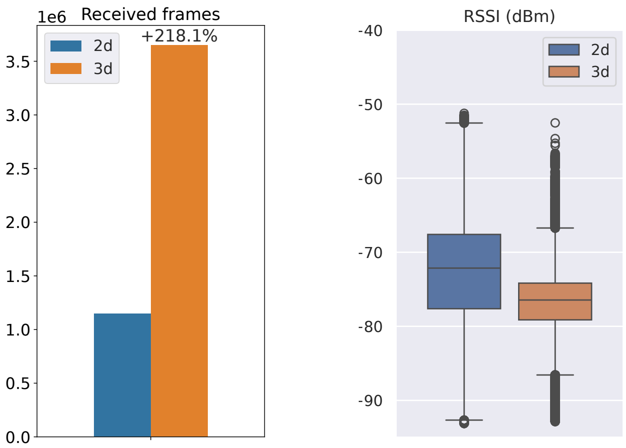
B. BS placed on Building Facades

Here we present the same metrics that we presented for the crossroads scenario once the BSs are placed on building facades. This is a configuration in which we expect the 3D simulations to deliver even more frames than the 2D equivalent, as explained in section V.

Figure 8a reports the comparison of the number of received frames for both simulations and shows a remarkable difference between the two, with the 3D simulator delivering more than three times the number of frames than the 2D one. As a further comparison, fig. 8b reports the box plots of the received signal strength for each frame. We notice that the 3D simulator brings a small decrease in the median value (about 4 dBs) and a more compact distribution. These effects could be due to several factors that need further investigation. Among some possible reasons, we mention that an increased LOS probability makes it possible to connect nodes that are further away from the BS and thus, could lower the median RSSI, also the effect represented in fig. 2 can lower the RSSI on links that are present in both the 2D and 3D simulation.

Figure 9 reports the distribution of the number of frames received per car per second, that obviously confirms the results we already commented for the crossroads scenario, but in a largely more evident way.

ISBN: 978-3-903176-61-4



(a) (b)

Fig. 8: The number of received frames (a), and a box plot of the the RSSI (b) for the 2D and 3D simulations with BS placed on building facades.

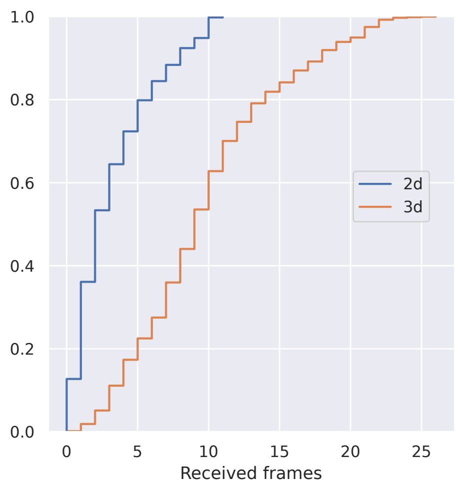


Fig. 9: The ECDF of the number of frames received by each car in each 1-second interval, BSs placed on building facades.

C. Potential Backhaul Links

As already mentioned, the presence of potential links among the BSs is an interesting element to keep in consideration, in light of the current trends that push the frequencies of mobile networks up to mmWave and beyond. Figure 10 reports the number of potential backhaul links in the crossroads scenario, showing an impressive increase in their number (from 8.7% up to 221.7% depending on the BS height). The availability of such a potentially dense mesh network would make a huge difference in planning a mixed wireless-wired backhaul, thus lowering the total cost of the infrastructure. Again we see how 3D simulations are a game

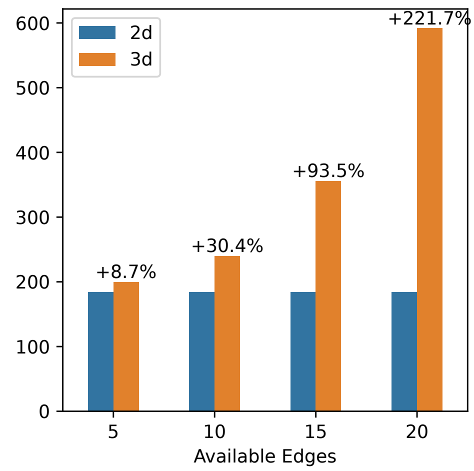


Fig. 10: Potential number of backhaul edges.

changer in the fidelity of the produced results. This effect is exacerbated in the building facade scenario, in which the backhaul link passes from 84 for the 2D simulations to 1332 for the 3D simulations.

D. Simulation Time

Finally, we report the time necessary to run the simulations. Table I shows the absolute time necessary to run the simulations, which is pretty stable when increasing the BS height, and tells that a 2D simulation roughly takes 2/3 of the time needed for a 3D simulation. The time per received frame instead shows a decreasing trend for the 3D simulator, reaching values that are even lower than the 2D one. While this may seem a positive outcome, it actually shows that the increased elaboration time depends on the number of frames that are evaluated, and not the ones that are correctly delivered. In other words, the time overhead due to the events in the upper layers is marginal, and all the overhead is due to the 3D simulator.

If we compare the time penalty with the increase in the accuracy of the metrics listed so far, we can conclude that this is a relatively low price to pay to achieve more realistic results.

VII. Conclusions

The communications between a mobile network infrastructure and the vehicles is still an open research topic, and the more mobile networks become complex, the higher the need for reliable software tools to model their performance. The existing obstacle shadowing model in Veins supports only 2D surfaces and is thus intrinsically limited when the goal of the simulation is to study the communications with a fixed mobile infrastructure, that is generally made of BSs placed tens of meters above the ground, and the vehicles themselves. This paper presents the extension of Veins to take into account the third dimension. Our results show that with a limited penalty in the execution time, we are able

ISBN: 978-3-903176-61-4

simulation	5m	10m	15m	20m
Absolute simulation time				
2d	626	631	633	621
3d	914	916	921	932
2d on buildings	588			
3d on buildings	931			
μs per received frame				
2d	347	351	352	346
3d	482	440	393	343
2d on buildings	512			
3d on buildings	254			

TABLE I: The execution time of each simulation run (seconds) and the time per received frame (micro-seconds).

to produce results that are largely different from the ones obtainable with the 2D simulator. In particular, we have shown that the number of received frames by the vehicles increases substantially, and the number of potential links between couples of nodes in the backhaul network also largely grows.

Based on these results we plan to integrate our code with the mainline Veins code base and with the derived projects to further investigate the performance of current and future mobile networks as a carrier of critical services for vehicles and mobile users in general. As a further enhancement, we consider quantizing the points in the ground with a certain precision and to pre-calculate the path loss from the position of the BSs to all the points in the ground. This could be done before the start of the simulation using Cuda libraries on a GPU to obtain massive parallelization, and thus, it would completely remove the delay introduced by the path loss computation during the simulation.

REFERENCES

[1] 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 *21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

[2] C. Sommer, D. Eckhoff, A. Brummer, D. S. Buse, F. Hagenauer, S. Joerer, and M. Segata, "Veins: The open source vehicular network simulation framework," in *Recent Advances in Network Simulation: The OMNeT++ Environment and its Ecosystem*. Springer International Publishing, 2019.

[3] G. Gemmi, M. Segata, and L. Maccari, "Vehicles or pedestrians: On the gNB placement in ultradense urban areas," in *18th Wireless On-Demand Network Systems and Services Conference (WONS)*, 2023.

[4] M. Segata, M. Lestas, P. Casari, T. Saeed, D. Tyrovolas, G. Karagiannidis, and C. Liaskos, "Enabling Cooperative Autonomous Driving through mmWave and Reconfigurable Intelligent Surfaces," in *Wireless On-Demand Network Systems and Services Conference (WONS)*, 2023.

[5] A. Brummer, R. German, and A. Djanatliev, "On the Necessity of Three-Dimensional Considerations in Vehicular Network Simulation," in *Wireless On-demand Network Systems and Services Conference (WONS)*, 2018.

[6] E. Moro, F. Linsalata, M. Magarini, U. Spagnolini, and A. Capone, "Advancing O-RAN to Facilitate Intelligence in V2X (preprint)," 2023. [Online]. Available: https://arxiv.org/abs/2307.01029

[7] A. Brummer, R. German, and A. Djanatliev, "Methodology and Performance Assessment of Three-Dimensional Vehicular Ad-hoc Network Simulation," *IEEE Access*, vol. 11, 2023.

[8] M. Polese, M. Giordani, T. Zugno, A. Roy, S. Goyal, D. Castor, and M. Zorzi, "Integrated Access and Backhaul in 5G mmWave Networks: Potential and Challenges," *IEEE Communications Magazine*, vol. 58, no. 3, 2020.

[9] E. Black, S. Gamboa, and R. Rouil, "NetSimulyzer: A 3D Network Simulation Analyzer for Ns-3," in *Proceedings of the Workshop on Ns-3, (WNS3)*, 2021.

[10] P. A. Regis, S. Bhunia, and S. Sengupta, "Implementation of 3D Obstacle Compliant Mobility Models for UAV Networks in Ns-3," in *Proceedings of the Workshop on Ns-3 (WNS3)*, 2016.

[11] T. Deinlein, R. German, and A. Djanatliev, "5G-Sim-V2I/N: Towards a Simulation Framework for the Evaluation of 5G V2I/V2N Use Cases," in *European Conference on Networks and Communications (EuCNC)*, 2020.

[12] T. Deinlein, A. Brummer, R. German, and A. Djanatliev, "On the Impact of Buildings on the LoS Evaluation in System-Level V2I/N Simulations," in *Vehicular Technology Conference (VTC2021-Fall)*, 2021.

[13] Y. Li, "An overview of the DSRC/WAVE technology," in *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness, (QShine)*.

[14] M. Berg, O. Cheong, M. Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed. Springer, 2008.

[15] ITU-R, "Guidelines for evaluation of radio interface technologies for IMT-2020," Report ITU-R M.2412-0. [Online]. Available: https://www.itu.int/pub/R-REP-M.2412-2017

[16] G. Gemmi, R. L. Cigno, and L. Maccari, "On the Properties of Next Generation Wireless Backhaul," *IEEE Transactions on Network Science and Engineering*, vol. 10, no. 1, 2023.

ISBN: 978-3-903176-61-4