Accelerating Vehicle Network Simulations in Urban Scenarios through Caching

Sergio M. Tornell, Carlos T. Calafate, Juan-Carlos Cano and Pietro Manzoni
Universitat Politècnica de València
Camino de Vera, s/n, 46022 Valencia, Spain
sermarto@upv.es, {calafate, jucano, pmanzoni}@disca.upv.es

***Abstract*—When designing new Vehicular Network (VN) protocols, the cost and time required for deploying real prototypes to run tests and evaluate new proposals becomes excessive. Consequently, simulation plays a key role in Vehicular Networks development. Simulation models and tools have been improved to better mimic the different factors that affect communications in real world scenarios. In terms of the communications channel, models have evolved from simplistic distance-based models to elaborated stochastic models that take into account both distance and the obstacle layout. However, increasing the complexity of the models also increases the time needed to run simulations. In this paper, we present an optimization applicable to the obstacle model included in VEINS, a well-known VN simulator, which relays on a cache table to accelerate ray-tracing calculations at the physical layer. Our results show that the proposed optimization can reduce by up to 75% the simulation time with minimal differences in terms of simulation results.**

***Index Terms*—VANET, vehicular networks, simulation models, shadowing, propagation**

I. Introduction

In the last decade, Vehicular Networks (VNs) have been one of the hot topics in the networking research community. VNs aim at enabling vehicles to communicate with other vehicles (V2V) as well as with Road Side Units (RSUs) such as traffic lights, information panels, or specifically deployed equipment (V2I). VN technologies enable a plethora of new services for drivers and passengers, and new opportunities for road management entities, both of which have been previously addressed by several authors [1][2].

Due to the high cost of deployment, simulations have become the most common method for VN protocols and applications evaluation [3][4]. The research community has intensively focused on the development of new models. One of the milestones in this evolution towards better models was the release of the Vehicular Network Simulator (VEINS) [5]. Besides other important improvements to mobility, MAC layer, or application layer models, VEINS includes an empirical model to compute 802.11gnp radio shadowing effects in urban environments [6][7], which is critical to guarantee that obtained results are meaningful [8]-[9].

Although accurate models significantly improve the quality of simulations results, we have found that their utilization introduces a very high overhead. In most cases the amount of vehicles we were able to simulate within a reasonable time was limited to a few dozens. To solve this problem, and to enable researchers to run long-lasting simulations and having more vehicles, we have optimized the VEINS' obstacles-based shadowing model. The results we present in this paper show that the new implementation reduces the execution time by up to 75% while maintaining the correctness of the final results. The cost to be paid is just some increase in terms of memory resources consumed.

The rest of this paper is organized as follows: In the next section we present some of the previous proposals in this area. In section III we shortly introduce the shadowing model and its implementation in VEINS; in section IV we present our proposed optimization; in section V we present the results that demonstrate the superior performance of our proposal; Finally, in section VI, we conclude the paper and refer to some open areas for future improvements.

II. Related Work

In this section we briefly summarize some of the most relevant propagation models proposed for VN simulation.

Historically, there have been two different types of propagation models: global models, and local models. The former ones represent the effects of attenuation over the whole simulation area as a stochastic process, while the second group of models take into account the effects of the existing obstacles between transmitter and receiver to predict the attenuation.

Concerning global models, the first proposed models were the Free Space and the Two-Ray-Ground propagation models; both are deterministic propagation models, and both define the communication range as a perfect circle [10]. Since this approximation is far from being real, the research community gradually introduced more accurate models. In 2008 a list of probabilistic propagation models were introduced in the Inet framework for the Omnet++ [11] simulator. The implemented models were: the log-normal shadowing model, the Rice fading model, the Rayleigh fading model [10], and the Nakagami propagation model [12]. All these models are typically available in other networks simulators (e.g., ns-2 [13]).

Although the aforementioned models are widely used by the cellular networking industry to predict aggregated statistics under urban and rural environments, they are not suitable for microscopic simulations. The research community soon understood that problem, and started developing more advanced models that consider attenuation effects due to the obstacles in each destination environment. Most of those models are based on Ray-Tracing [14], [15]. However, Ray-Tracing techniques are too CPU-intensive and do not scale well when the number of nodes in the simulation increases. More recently, researchers proposed several radio propagation models that simplify the Ray-Tracing methodology by simply adding a certain attenuation value according to the number of traversed obstacles. As an example, in [9] the authors present a propagation model that limits the communication to Line Of Sight (LOS). In 2010, an obstacle propagation model was introduced in VEINS and in the INET framework [6], both freely available for the Omnet++ simulator [16]. However, the overhead imposed by this model limits the number of nodes we are able to simulate in a reasonable time.

In this paper we present an optimization to the implementation of VEINS' obstacle model, which has became widely used due to its rigour and accuracy.

III. The Propagation Model

The propagation model is one of the most important parts of network simulators. It is in charge of calculating the received power at the receiver location. This value is then used by the radio model to run a stochastic process which decides whether the data were correctly received.

The propagation model must take into account several effects of the propagation process, namely, path loss, shadowing, and small scale effects. The path loss component includes propagation losses due to the natural expansion of the radio wave, while the shadowing refers to the effects associated to the presence of mountains or big buildings in the surrounding area of the transmission. Finally, small scale effects refers to the Rayleigh effect or the Doppler effect.

According to the propagation model the received power can be calculated as follows:

$$P_{rx}[dBm] = P_{tx}[dBm] + G_{tx} + G_{rx} - L_{pl} - L_s - \sum L_x \quad (1)$$

where P_{rx} and P_{tx} are the received and the transmitted power, respectively, G_{rx} and G_{tx} are the gain of the antennas, L_{pl} is the attenuation due to path loss in dB, L_s is the attenuation due to shadowing effects, and $\sum L_x$ is the total attenuation due to all small scale effects [10]. In the next subsection we will describe how the shadowing effects are modeled and implemented in VEINS. The discussion of other components of the propagation model are outside the scope of this paper.

A. VEINS' Shadowing Model

The shadowing model currently used by VEINS is based on obstacles. Every time a new transmission takes place the model identifies which obstacles are traversed by the communication line connecting transmitter and receiver, and adds the attenuation corresponding to every wall along the path to the total attenuation experienced by the transmission. The location, shape, and attenuation of walls of all the obstacles are defined in an *xml* file provided at initialization time.

Fig. 1: Screenshot from the VEINS simulator. Stars represent the intersection of the transmission line with the obstacles' walls.

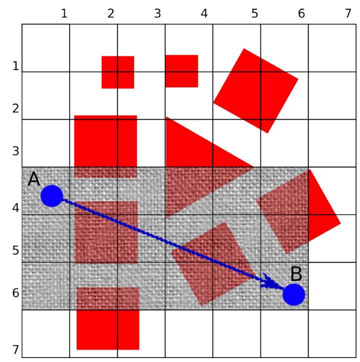


Fig. 2: For each source-destination pair, the model only iterates over obstacles inside the rectangle defined by locations A and B.

Figure 1 shows the transmission straight line traversing several obstacles, as well as the points of intersection with the walls.

The shadowing attenuation due to obstacles is calculated as follows:

$$L_s = \beta n + \gamma d_m \quad (2)$$

Where n is the number of traversed walls, d_m is the distance traveled inside the obstacle by the signal, and β and γ are coefficients obtained from the obstacles definition.

B. Implementation Details

When identifying the obstacles traversed by a transmission, the model needs to iterate over the whole set of obstacles, checking if any of the segments of each obstacle shape intersects the transmission line towards a specific receiver. This operation requires a high computation time since it involves not only the iteration process, but also some costly geometric calculations.

To reduce the CPU cost associated to those calculations, the developers of VEINS decided to implement two different mechanisms:

Transmission cache: due to the discrete nature of the simulation time, it is very common that two nodes experience several consecutive transmissions while maintaining a constant location. To save resources, a cache stores the result of a transmission between two given coordinates.

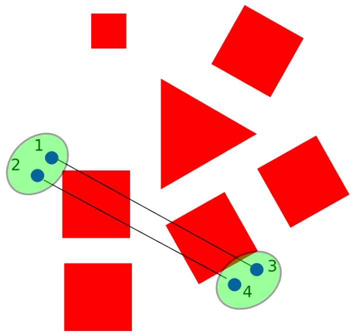


Fig. 3: A transmission from node 1 to node 3 will experience the same attenuation than a transmission from node 2 to node 4.

Obstacle Grid: To avoid iterating over the whole set of obstacles, a virtual grid is defined on top of the simulation area. Each obstacle is associated to the grid cell it belongs to. Therefore, when a transmission occurs, the model only needs to iterate over the grid cells inside the rectangle defined by the location of both nodes. Figure 2 illustrates this mechanism. Assuming that node A initiates a transmission to node B, the model will iterate only over those cells inside the rectangle defined by (X_a, Y_a) and (X_b, Y_b), being (X_i, Y_i) the location of the nodes. An *already iterated* list of obstacles complements this mechanism to avoid iterating several times over the same obstacle when it belongs to more than one grid cell.

IV. Our Proposal

In this section we detail our proposed optimization to the obstacle shadowing model implementation included in VEINS.

A. Definitions and Assumptions

The attenuation suffered by a propagated signal is the difference between the transmitted and the received power. Solving from equation 1:

$$A = P_{tx} - P_{rx} \tag{3}$$

$$A = G_{tx} - G_{rx} + L_{pl} + L_s + \sum L_x \tag{4}$$

From now on, we focus only on L_s, which does not depend on the transmitted power.

Our proposal does the following three assumptions: *i)* persistent obstacles, *ii)* local correlation, and *iii)* symmetry.

Persistent obstacles: We assume that the attenuation due to obstacles experienced by a transmission between two specific points is constant during the simulation time, *i.e.* obstacles (buildings) do not move or vary their attenuation characteristics throughout the simulation time.

Local correlation: We assumed that, given two small enough areas, A_a and A_b, the attenuation due to obstacles experienced by a transmission between a given node located in any point $(X_a, Y_a) \in A_a$ and another node located in $(X_b, Y_b) \in A_b$ can be assumed to be constant, *i.e.* we assume that two different transmission events will experience the same attenuation if their respective sources and destinations are close enough. Figure 3 illustrates this assumption.

Symmetry: According to equation 2, the attenuation due to obstacles experienced by a transmission between two points is symmetrical, *i.e.* it does not matter which node is the transmitter.

B. Details and Implementations

Our optimization is based on a cache table that stores the attenuation due to obstacles for transmissions between two points A and B. However, due to the continuous nature of coordinates, the probability that two independent transmissions between any two independent nodes occurs on exactly the same coordinates can be assumed to be zero. Therefore, applying the second assumption, we defined a cell grid and assumed that the attenuation experienced by transmissions between two given cells of that grid is constant. If the size of the cells is small, the introduced error is almost zero. Moreover, we have observed that, in the original model, the attenuation due to obstacles can be reduced in most cases to a binary variable that determines whether two nodes located at certain coordinates can communicate. Using the third assumption, we can reduce the size of the cache table to half. The critical parameter of our optimization is the cell size of the grid. If cells are too big, we will introduce greater errors, while if cells are too small, there will be no impact on time performance (few cache hits). In section V we will discuss about this trade-off by performing a wide variety of simulation scenarios.

Our optimization can be downloaded from our github account[1].

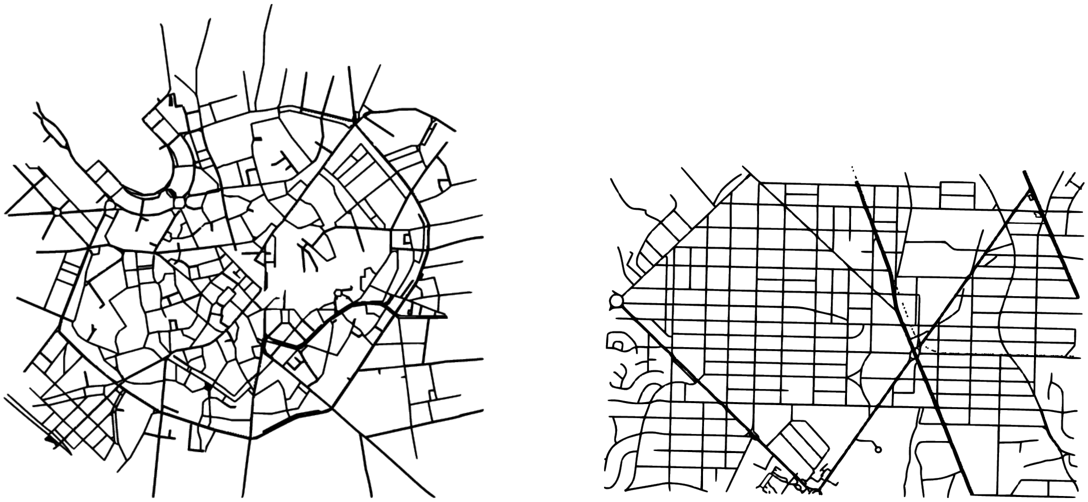
V. Validation of our Proposal

In this section we describe several experiments we have done to test the validity of our optimizations. We have performed two different sets of experiments: the first set to determine the one-hop communication impact, and a second set to determine the multi-hop communication impact. This section is divided into three subsections: in the first one, we present the common scenario, while in the second and the third subsections we present the results for the one-hop and the multi-hop scenarios, respectively.

A. Common Scenario

In our simulations, vehicles' mobility is driven by the microscopic traffic simulator *Simulation of Urban MObility* (SUMO) [17]. To achieve more diversity in our results, we have repeated our simulations using two different road maps: a $7.2km^2$ area from downtown Milan, and a $6km^2$ area of a residential district belonging to the city of Washington. The first map represents a typical old European city, where the streets are short and narrow. The second map, represents a typical residential area, with long and wide avenues. Figures 4b and 4a shown their respective layouts. According to the default configuration, the location of the vehicles is updated every simulated second. The number of vehicles in both scenarios is fixed to 500. To ensure that the number of vehicles is constant during the simulation we have used VACaMobil

[1] https://github.com/GRCDEV/inet/tree/newObs



(a) Road layout of Milan. (b) Road layout of Washington.

Fig. 4: Maps used in our simulations.

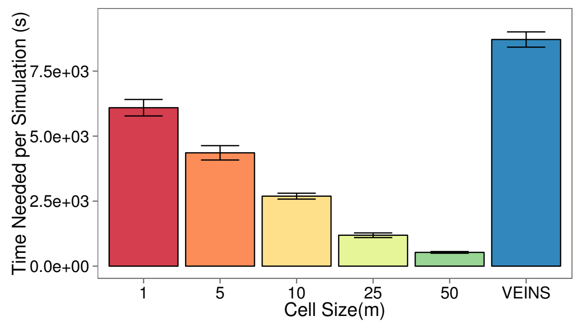
[18]. When using VACaMobil, vehicles are removed from the simulation when they arrive to their destinations. Therefore, every time a vehicle disappears a new vehicle with a random route is introduced in the network.

Concerning the propagation model, we used the model presented in [8]. Besides the effects of obstacles, this model adds to the VEINS shadowing model the effects of multi-path, as well as other small-scale effects.

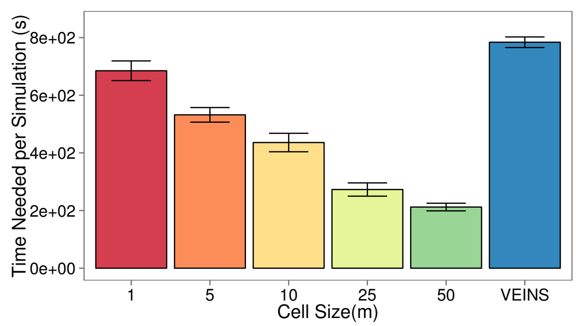
To evaluate the impact of the grid cell size in our optimization we have repeated all the simulations using different sizes for the grid cell of our cache, namely: $1m$, $5m$, $10m$, $25m$ and $50m$. We compared the obtained results against the original VEINS obstacle model. To provide statistical guarantees in terms of data similarity, we have repeated every experiment 50 times. In total, we run 1200 simulations.

We used two metrics commonly used in VN protocol evaluations: the number of received packets and the number of detected collisions at the mac layer. To test the validity of our optimizations we will show not only the aggregated average results, but also their distributions across the obtained values for every experiment to visually evaluate the similarity with the original shadowing model included in VEINS.

We have performed our simulations in a server cluster available in our university. This cluster has 72 nodes, each with 2 *Intel Xeon E5-2450* 8-core processors and 64GB of DDR3 RAM.



(a) Washington



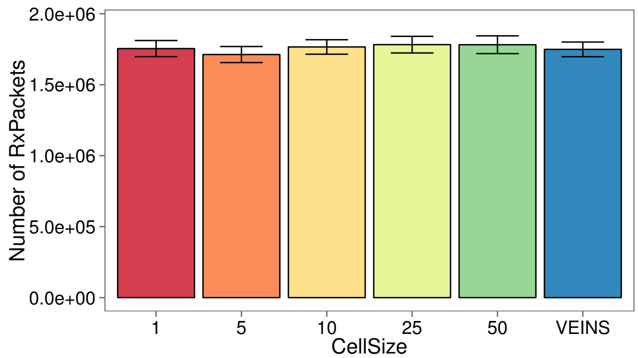
(b) Milan

Fig. 5: Average time needed per simulation for the one-hop scenario.

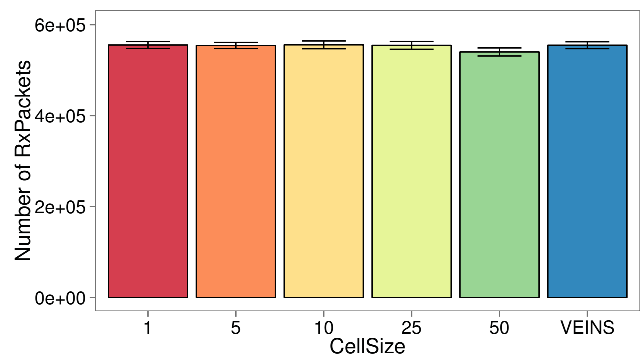
B. Impact of Caching in One-Hop Communication

In this scenario, all the vehicles in the simulation send a 520Bytes beacon message every $10s$. Every run simulates the aforementioned scenario during $2000s$. To measure the error introduced by the use of our cache and the differences between several cell sizes, we have determined the total number of received beacons, and the total number of detected collisions per simulation.

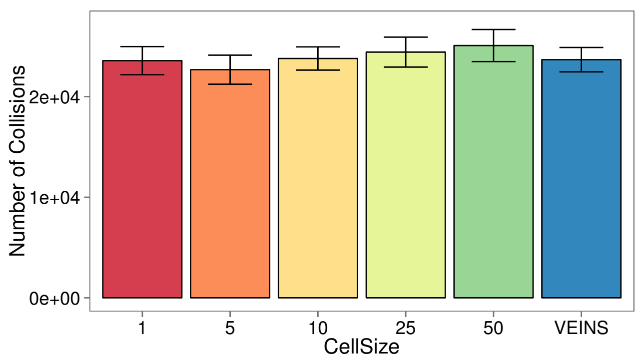
Figures 5a and 5b show the impact of our optimization in the simulation time. In the case of Washington map, a cache based on a grid cell size of just $1m$ allows saving 25% of the simulation time, while in the case of Milan map, it saves only 13%. Both figures reveal a clear relationship between the simulation time and the size of the grid cell. When comparing both maps, we appreciate that, in the case of the original VEINS implementation, simulations using the Washington map require approximately ten times more time than those simulations using the Milan map. This occurs because the definition of the Washington obstacles has more than 7000 buildings while the Milan map has less than 600.



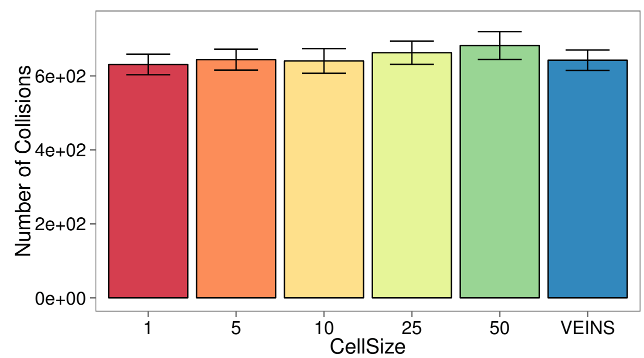
(a) Washington average number of received beacons.



(b) Milan average number of received beacons.

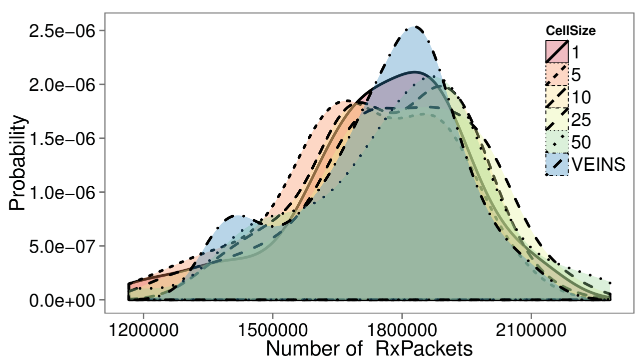


(c) Washington average number of collisions.

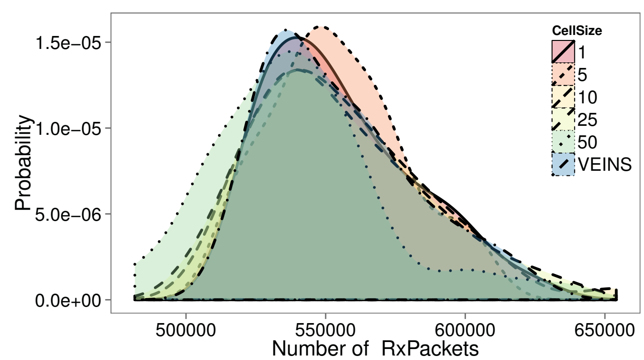


(d) Milan average number of collisions.

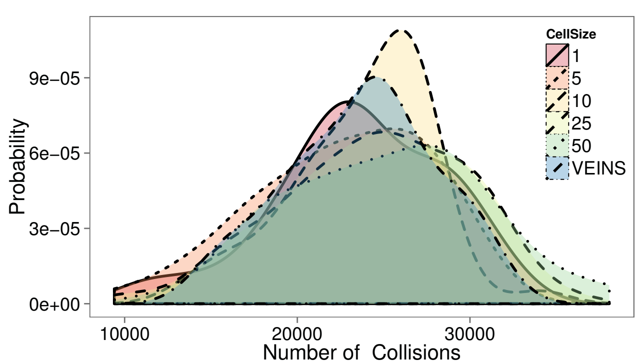
Fig. 6: Average measurements in the One-Hop scenario.



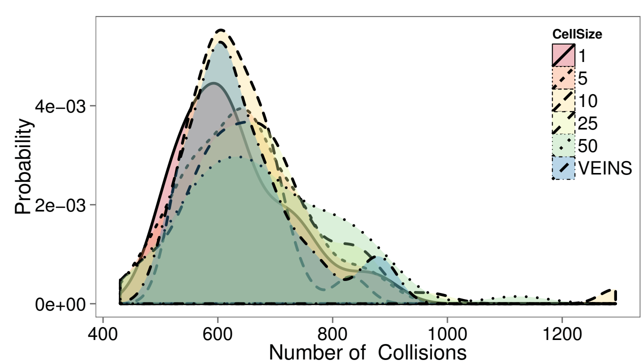
(a) Washington average number of received beacons..



(b) Milan average number of received beacons.



(c) Washington average number of collisions.



(d) Milan average number of collisions.

Fig. 7: Probability Density Function (PDF) of different measurements in the One-Hop scenario.

As we stated in section III, the amount of obstacles clearly increases the number of operations needed to compute each transmission. We can conclude that: the higher the number of obstacle in the simulation, the greater is the performance gain obtained by our proposed optimization.

Figures 6a and 6a show the average total number of received packets per simulation. We must highlight the high similitude between the results obtained by all the configurations when using either the Washington or the Milan map. If we compare the results obtained when using the Washington map against the ones obtained using the Milan map, we can easily appreciate that in the case of Washington, the number of received messages is higher than when using the Milan map. This is because, the Washington road layout has more open areas and wider avenues, which allows nodes to communicate without obstacles.

Figures 6c and 6d, represent the average number of detected collisions in each map. In the case of collisions, there is more diversity in the results. However, the larger confidence intervals obtained (comparing to figures 6a and 6b) indicates that this diversity may be due to random effects.

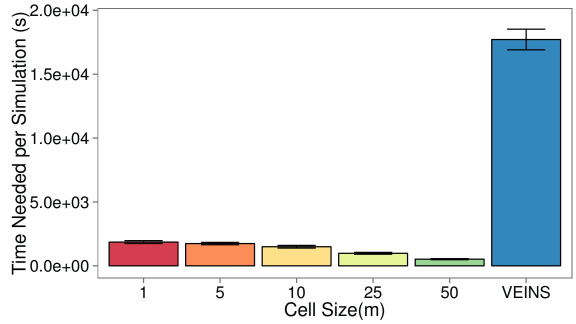
Figure 7 shows the PDFs of the previously exposed variables: total number of received packets and total number of detected collisions. Those figures confirm that the distribution obtained for both variables when using our proposed optimization is similar to the one obtained by the VEINS implementation. However, we can appreciate that when adopting larger grid cell sizes, the differences between the distribution obtained using cache and that obtained from the VEINS implementation become more evident.

In this simple one-hop scenario the results obtained using the original VEINS implementation do not significantly differ from the results obtained using our proposed optimization, even in the case of a 50 m grid cell size, which saves up to 95% and 75% of the simulation time when using Washington and Milan maps, respectively.

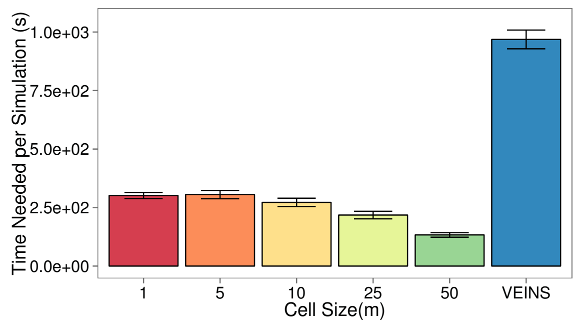
C. Impact of Caching in Multi-Hop Communication

To evaluate the impact of our proposed optimization in multi-hop communication, we have configured our vehicles to rebroadcast those beacons received for the first time. This broadcast method is commonly known as *flooding* [19]. Similarly to the previous configuration, each vehicle generates a new message every $10s$. Configuring this scenario to simulate $2000s$, as in the previous scenario, implied that simulations running the original VEINS implementation would require more than 25 days to complete. Due to this problem, we configured this scenario to simulate only $50s$. This simulated duration may not be enough to evaluate the performance of certain protocols, but is enough to evaluate the impact of our proposed optimization.

Figures 8a and 8b show the impact of our optimization on the simulation time. In this scenario, our proposed optimization when adopting a grid cell size of $1m$ is able to reduce the required time to run a simulation to 15% and 25% for the Washington and the Milan maps, respectively. We can



(a) Washington



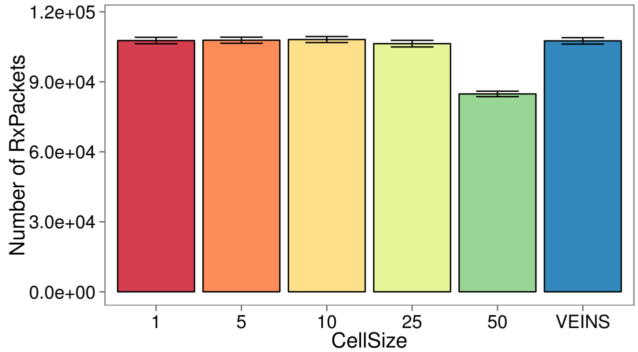
(b) Milan

Fig. 8: Average time needed per simulation for the multi-hop scenario.

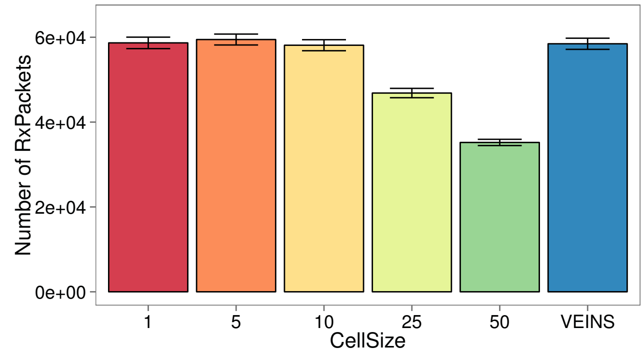
observe the same trend we observed previously in figure 5: the larger the grid cell size, the smaller the time required to run a simulation. Comparing this figure to figure 5 we appreciate that the impact of our proposed optimization is much more significant. Notice that, in this scenario, nodes exchange a higher number of messages and, therefore, the calculation of attenuation due to obstacles is more frequent. Moreover, in a flooding scheme, every new message triggers other nodes to forward a lot of messages in a short period of time, *i.e.* there are a lot of transmissions during a short period where nodes remain mostly static, clearly improving the chances of a cache hit.

Figure 9 shows the results obtained in this scenario for different configurations. It shows that, given the complexity of the scenario, simulations with large grid cell size significantly differ from the original VEINS implementation. Figures 9b and 9d show that, in Milan, the results obtained using a $25m$ grid cell size cache considerably differ from the ones obtained with the original VEINS implementation, while, in Washington (figures 9a and 9c), this divergence appears only when using a $50m$ grid cell size cache. Notice that the difference between both maps is due to the layout of both road networks, while the long streets and wide avenues of Washington increase the number of nodes in LOS, which are not affected by errors in the obstacle attenuation calculation. On the contrary the short and narrow streets of Milan, increases the error probability when using our proposed optimization.

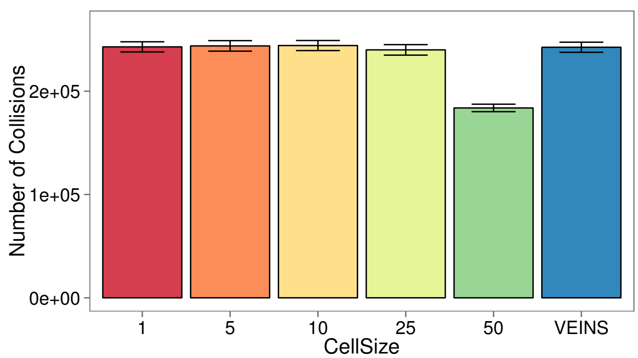
Figure 10 confirms the diversity in the results distribution when using large grid cell sizes, as described previously, while



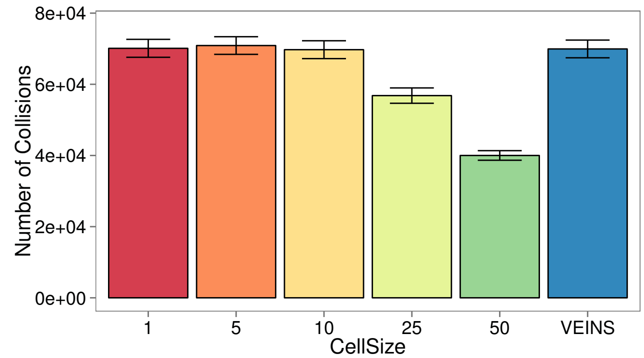
(a) Washington average number of received beacons.



(b) Milan average number of received beacons.

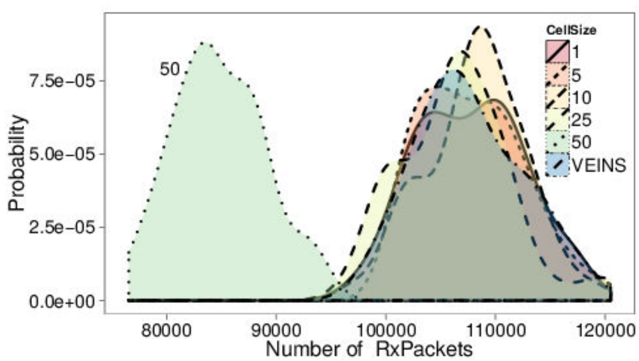


(c) Washington average number of collisions.

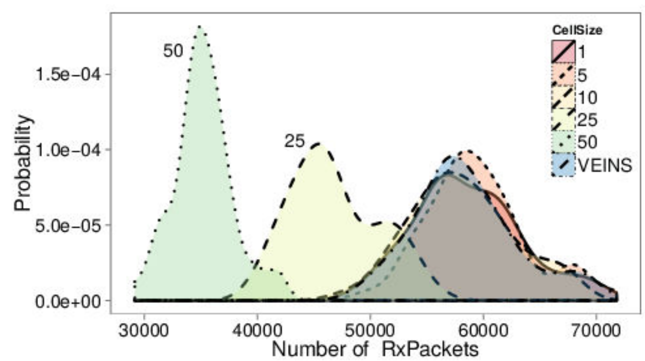


(d) Milan average number of collisions.

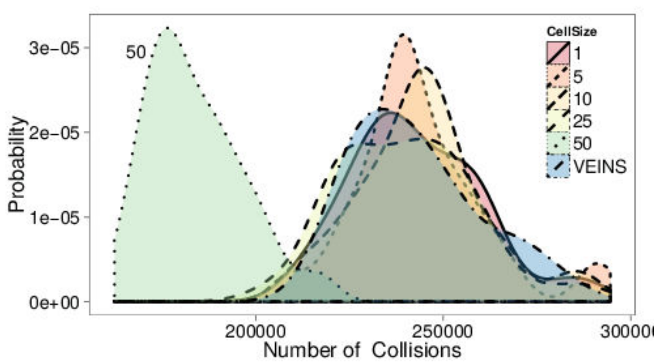
Fig. 9: Average measurements in the Multi-Hop scenario.



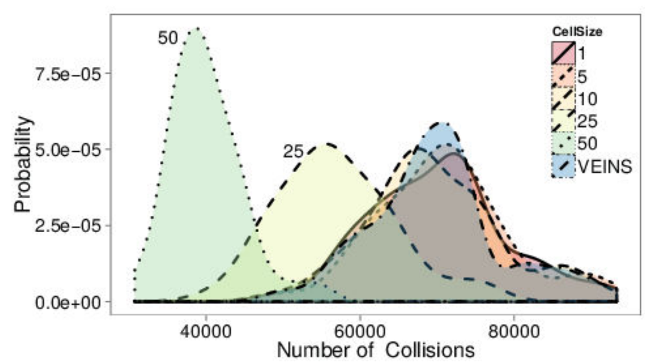
(a) Received beacons for in Washington.



(b) Received beacons in Milan.



(c) Number of collisions in Washington.



(d) Number of collisions in Milan.

Fig. 10: Probability Density Function (PDF) of different measurements in the Multi-Hop scenario.

at the same time it also confirms the high similarity between the results obtained using the smallest grid cell sizes and those obtained using the original VEINS implementation.

Overall, this experiments points out that the impact of our proposed optimization in time of performance and the accuracy the obtained results depends not only on the obstacle map, but also on the simulated protocol. In the conclusions we briefly discuss these issue and propose how to use our optimization properly, without compromising our final results.

VI. Conclusions and Discussion

In this paper we presented a simple but powerful optimization to the VEINS[20] obstacle shadowing model. By using a symmetric cache grid we have significantly reduced the time needed to simulate large networks, imposing a minimal overhead to simulation memory requirements. Our results demonstrated that the final outcomes, when using our optimized model, do not significantly differ from the outcomes obtained using the original model implemented in VEINS.

As seen in the previous sections, choosing a large grid cell sizes helps at accelerating simulations but may lead to biased results. When using our proposed optimization, the researcher must be conscious of this fact and act consequently. Thus, before publishing results that include simulations using our optimization, we recommend to cross validate the results running a small set of shorter simulations and then comparing the results obtained against the original VEINS implementation. This allows the researcher to confirm that the effect of caching did not bias the simulation results. We also recommend being conservative when choosing the grid cell size, and use always the smallest cell size that allows us running simulations in a reasonable time.

The areas of the simulated map where the received power is prone to experience errors due to the effects of the caching are restricted to those areas near the corners of obstacles. Therefore, the impact of caches with a grid cell size much smaller (at least one order of magnitude) than the smallest obstacle size is expected to be negligible.

We believe that there is room for more improvements in the obstacle shadowing model. For example, using a dynamic cache in which cells near the corners of the obstacles are subdivided into smaller cells to increase accuracy, while cells near the walls of the obstacles have large sizes, thereby speeding up simulations without affecting the results accuracy.

Acknowledgments

This work was partially supported by the *Ministerio de Economía y Competitividad*, Spain, under Grants TIN2011-27543-C03-01 and BES-2012-052673.

References

[1] H. Hartenstein and K. Laberteaux, "A tutorial survey on vehicular ad hoc networks," *IEEE Communications Magazine*, no. June, pp. 164–171, 2008.

[2] F. Rango, F. Veltri, P. Fazio, and S. Marano, "Two-level Trajectory-Based Routing Protocol for Vehicular Ad Hoc Networks in Freeway and Manhattan Environments," *Journal of Networks*, vol. 4, no. 9, 2009.

[3] S. Joerer, C. Sommer, and F. Dressler, "Toward Reproducibility and Comparability of IVC Simulation Studies: A Literature Survey," *IEEE Communications Magazine*, vol. 50, no. 10, pp. 82–88, Oct. 2012.

[4] P. Fazio, F. De Rango, C. Sottile, and A. F. Santamaria, "Routing Optimization in Vehicular Networks: A New Approach Based on Multi-objective Metrics and Minimum Spanning Tree," *International Journal of Distributed Sensor Networks*, vol. 2013, pp. 1–13, 2013.

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

[6] C. Sommer and D. Eckhoff, "A computationally inexpensive empirical model of IEEE 802.11 p radio shadowing in urban environments," *International Conference on Wireless On-Demand Network Systems and Services*, vol. 79, no. September, pp. 84–90, 2011.

[7] P. Fazio, F. D. Rango, and C. Sottile, "An On Demand Interference Aware Routing Protocol for VANETS," *Journal of Networks*, vol. 7, no. 11, 2012.

[8] M. Baguena, C. T. Calafate, J.-C. Cano, and P. Manzoni, "Towards realistic vehicular network simulation models," in *2012 IFIP Wireless Days*. Dublin, Ireland: IEEE, Nov. 2012, pp. 1–3.

[9] F. J. Martinez, C.-K. Toh, J.-C. Cano, C. T. Calafate, and P. Manzoni, "Realistic Radio Propagation Models (RPMs) for VANET Simulations," *2009 IEEE Wireless Communications and Networking Conference*, 2009.

[10] T. Rappaport, *Wireless Communications-principles and practice 2002, edition second*, 2nd ed. Prentice Hall PTR, Jan. 2002.

[11] A. Kuntz, F. Schmidt-Eisenlohr, O. Graute, H. Hartenstein, and M. Zitterbart, "Introducing probabilistic radio propagation models in OMNeT++ mobility framework and cross validation check with NS-2," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, 2008.

[12] M. Nakagami, "The m-distribution – A general formula of intensity distribution of rapid fading," in *Statistical Methods in Radio Wave Propagation*, W. C. Hoffmann, Ed., Elmsford, NY, 1960.

[13] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, "Overhaul of ieee 802.11 modeling and simulation in ns-2," in *Proceedings of the 10th ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*, ser. MSWiM '07, Array, Ed., vol. 2. New York, NY, USA: ACM, 2007, pp. 159–168.

[14] M. Catedra, J. Perez, F. S. de Adana, and O. Gutierrez, "Efficient ray-tracing techniques for three-dimensional analyses of propagation in mobile communications: application to picocell and microcell scenarios," *IEEE Antennas and Propagation Magazine*, vol. 40, 1998.

[15] E. Giordano, R. Frank, G. Pau, and M. Gerla, "CORNER: A Radio Propagation Model for VANETs in Urban Scenarios," *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1280–1294, 2011.

[16] "Omnet++," http://www.omnetpp.org/, April 2014.

[17] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO - Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128–138, December 2012.

[18] M. Báguena, S. M. Tornell, A. Torres, C. T. Calafate, J.-c. Cano, P. Manzoni, A. Torres, and C. D. V. S. N, "A Tool Offering Steady-State Simulations for VANETs," *Recent Advances in Communications and Networking Technology*, vol. 2, pp. 102–112, 2013.

[19] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking - MobiCom '99*, pp. 151–162, 1999.

[20] C. Sommer and F. Dressler, "Progressing toward realistic mobility models in VANET simulations," *Communications Magazine, IEEE*, vol. 46, no. 11, pp. 132–137, Nov. 2008.