Temporal Evolution of Vehicular Network Simulators: Challenges and Perspectives

Mauricio J. Silva[1], Genilson I. Silva[2], Fernando A. Teixeira[3] and Ricardo A. Oliveira[1]
[1]*Universidade Federal de Ouro Preto, Ouro Preto, MG, Brazil*
[2]*Instituto Federal Sudeste Minas Gerais, Barbacena, MG, Brazil*
[3]*Universidade Federal de Sao Joao Del Rei, Sao Joao Del Rei, MG, Brazil*

Keywords: Vehicular Network Simulators, Traffic Simulators, Mobility Models.

Abstract: New proposals of applications and protocols for vehicular networks appear every day. Its crucial to evaluate, test and validate these proposals on a large scale before deploying them in the real world. Simulation is by far the preferred method by the community when conducting the evaluation. In this paper we survey the main simulators for vehicular networks and show how they evolved over time. Thus, we provide information that leads to an understanding of how, and how long does it take for the scientific community to absorb a new simulator proposal. Additionally, valuable insights are presented to help researchers make better choices when selecting the appropriate simulator to evaluate new proposals.

1 INTRODUCTION

Communication-capable vehicles allowed the emergence of collaborative Intelligent Transport Systems (ITS). The ITS comprises services developed with the aim to improve the road safety (e.g., warning about road conditions), reduce traffic jams (e. g., intelligent intersections) and make the travel more enjoyable for drivers (e.g., informing about gas stations nearby, roadside restaurants, among other things) (Fischer, 2015).

The design and evaluation of algorithms and protocols for ITS have been a constant topic of research (Tornell et al., 2015). Given the unique characteristics of vehicular networks, recent works have pointed towards the emergence of new traffic models and the impact analysis of these models on the behavior of proposed protocols and algorithms (Arellano and Mahgoub, 2013; Zemouri et al., 2012). These models must be fed by data collected in current networks so that it is possible to make a reasonably accurate estimate of the performance of new applications. Such an assessment is a challenge, and can usually be done using three different methods, which are Mathematical Analysis, Field Operational Tests and Simulations (Guan et al., 2014; Nimje and Dorle, 2013). Each of these methods has its advantages and disadvantages (summarized on Table 1), and the method to be used should be chosen cautiously as it directly influences the results (Harri et al., 2009; Eckhoff and Sommer, 2015).

Mathematical analysis allows an analytical study of the problem, and can provide valuable information, allowing a better understanding of the designed system. Statistical distributions are used to generate the models that are necessary for the simulation. However, this method tends to simplify certain simulation parameters such as the mobility models. Such simplifications can lead to inaccurate results.

Field Operational Tests allow a better evaluation of applications and protocols for vehicular networks. In this type of analysis, the devices are exposed to real environments, which can lead to unpredicted situations. The disadvantages related to this type of test usually involve high costs in terms of time and money as well as the difficulty to perform large-scale tests.

Simulations make it possible to assess the new proposals on a large scale and at low cost. However, similarly to what happens in the mathematical analysis, complex models need to be simplified so that they can be simulated. Again, this simplification must be done cautiously, as they may make the results inaccurate.

Researchers prefer to evaluate their proposals through the simulation method, which demands simulators that produce results increasingly closer to reality (Harri et al., 2009). Researchers used to believe

Silva, M., Silva, G., Teixeira, F. and Oliveira, R.
Temporal Evolution of Vehicular Network Simulators: Challenges and Perspectives.
DOI: 10.5220/0006696900510060
In *Proceedings of the 20th International Conference on Enterprise Information Systems (ICEIS 2018)*, pages 51-60
ISBN: 978-989-758-298-1

Table 1: A comparison of assessment methods for ITS applications.

Assessment method	Category	Advantages	Drawbacks
Mathematical Analysis		Valuable insights on overall behavior; Lower and upper bounds of problem.	Undergo simplifications that compromise the results
Simulation	Offline	Reproducible; Faster; Acceptable degree of realism	Changes in one domain simulator do not affect the other; nodes mobility are immutable; scalability by cloning
	Embedded	Reproducible; Reliable degree of realism	Communication does not change mobility and vice-versa.
	Online	Communication changes might affect mobility and vice-versa; Results more realistic; Scalable	Unreproducible; Hard to set and parameterize;
FOTs		Real scenario, generating real data;	Not scalable; Few prototypes or vehicles; High costs (time and money)

that Vehicular Networks were a specific application of Mobile Networks, and for that reason, random models have been applied to simulate the vehicles' mobility. It was not long ago that they realized that vehicular networks had their own characteristics, so specific mobility models should be proposed to represent them. The vehicular mobility models evolved from random models, where the mobility of all nodes was generated in a single file that was used as input to a network simulator (called offline), to the behavior-based models, where network and traffic simulators interact to represent the behavior of the driver (called online).

Unfortunately, the more realistic a simulator is, the harder it is to use it. Because of this, one of the biggest difficulties in simulating a vehicular network is to ensure that the mobility patterns of a real environment are reproduced. According to (Sommer and Dressler, 2008) and (Marfia et al., 2007), existing simulators that use models that are able to generate scenarios closer to the real ones, are often complex to use. As a result, many of the new proposed protocols and algorithms are evaluated using customized simulators, which introduce bias and compromise the reproducibility of these algorithms by other members of the scientific community (Mota et al., 2014).

This article provides an overview of the vehicular network simulators evolution, answering the following questions: how did the evolution of vehicular network simulators happened from the beginning until now? How long does the scientific community need to absorb a new approach for a vehicular network simulator? Do we have a pragmatic solution for vehicular network simulators? In addition to answering these questions, we will provide valuable insights to help researchers make better choices when selecting the appropriate simulator to evaluate new proposals.

The rest of this article is organized as follows: Section 2 presents the necessary requirements to obtain better results in specific-domain simulators. In Section 3 we survey the most used simulators to assess algorithms and protocols for vehicular networks, describing their features and drawbacks. Section 4 discusses the temporal evolution and the future challenges related to vehicular simulations. Finally, we conclude the paper in Section 5.

2 OVERVIEW OF DOMAIN-SPECIFIC SIMULATORS

Vehicular networks have drawn attention from the academia, industry and government. The goal is not only to develop applications that contribute to a safer transport system, but also to propose new applications based on high quality of service. Such applications must be tested, evaluated and validated in a controlled environment before being deployed in the real world.

Simulation is the preferred method by the academic community to evaluate new proposals for vehicle networks, because it enables scalable evaluations, with low cost and an acceptable degree of realism. Scalability and low cost are well defined requirements in a simulation. However, vehicular networks have their own mobility requirements, and for a simulation to have an acceptable degree of realism it is necessary that the mobility models comply with such requirements. To resolve this issue, three approaches can be used, which (Zemouri et al., 2012) called offline, embedded and online (see Figure 1). We will use the same classification because we believe that in addition to being simple, it encompasses unambiguously all existing proposals.

In the offline approach 1(a), trace files are extracted from the real movements of vehicles, which are either obtained from navigation systems, or generated by a vehicular mobility simulator that has real mobility capable models. This approach allows the nodes in the network simulator to move in accordance with the data read from the trace files. It is called offline because the trace file is not modified after it has been generated. This implies that there is no interaction between the mobility and network simulators. Some simulators that are in this category are BonnMotion (Aschenbruck et al., 2010), MOVE (Karnadi et al., 2007) and VanetMobiSim (Härri et al., 2006).

In the embedded approach 1(b) the traffic and network simulators are natively coupled to form a single simulator. In fact, most of the platforms of this category were either designed for traffic simulations or for network simulations, not both. Some modules that would allow such platforms to be used in vehicular networks were later implemented, for instance, VISSIM that has MOVE (Karnadi et al., 2007), the NCTUns (Wang and Chou, 2009) and VCOM (Killat et al., 2007) as embedded modules.

The online approach 1(c) was introduced to address the limitations of the previous two approaches. It provides a bidirectional communication between network simulators (developed by computer scientists) and traffic simulators (developed by traffic engineers). Thus, vehicular mobility is handled by experts in traffic, and communication is handled by experts in computing. In the online approach the network simulator controls the traffic simulator by sending commands that modify the behavior of the nodes. The traffic simulator responds to the network simulator with the position of the affected node. This approach is considered, thus far, the best solution for simulations in vehicular networks, for it allows a high degree of realism.

Vehicular networks simulators face many challenges. Some of them are related to the communication, more specifically, physical layer, link layer, and, in some cases, transport layer. The other challenges are related to the mobility, which is one of the main factors that affect the assessment of protocols and applications for vehicular networks. Communication challenges are dealt with by the network simulator, which must implement all models needed to represent a real vehicle-to-vehicle communication. Mobility challenges are dealt with by the mobility simulator, which is responsible for all models needed to represent a real vehicle mobility, including change lane models, driver behavior models, traffic sign models, etc. In the following sections, we present the main simulators used by the scientific community in each specific-domain (network and mobility), and also show how these simulators were combined to achieve better results.

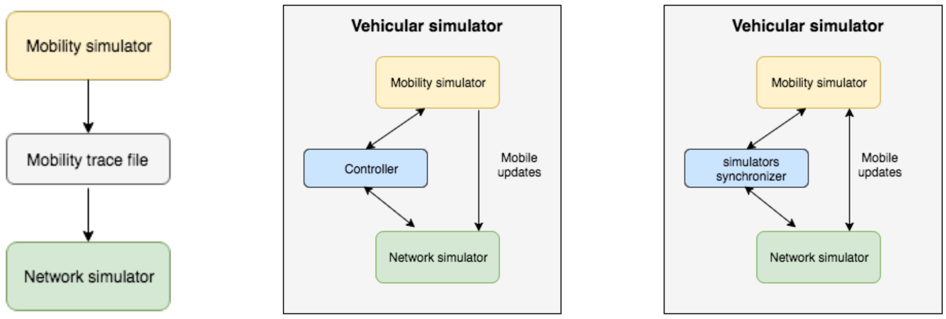
2.1 Network Simulators

Simulations based on discrete events have become the main method used by simulators. In this type of simulator, simulation behavior is not based on continuous equations, but rather in discrete events distributed in time. The most used simulators for vehicle networks are OMNeT ++, ns-2/ns-3, Jist/Swans, GloMoSim and GTNetS. All of them are based on discrete events and will be discussed in greater detail below.

OMNeT++ (Varga, 2010) is a discrete event simulator that allows communication modeling in networks, parallel systems and distributed systems. Available since 1997 under the GPL, it has become one of the most used simulators by the scientific community. The OMNeT++ was designed from the beginning to support large-scale simulations. For this reason, it was built completely modular, enabling better reuse of code and facilitating the implementation of new libraries that extend its functionality. As an example, we can mention the INET, which is an open-source framework that has the models to simulate mobile, wired and wireless networks. The OMNeT++ modular feature, allows each model to be implemented separately and then combined to form a protocol stack similar to the real one. INET has models of the physical layer (PPP, Ethernet and 802.11), various communication protocols (IPV4, IPV6, TCP, SCTP, UDP) and various application models.

The ns-2 (ns2, 2017) simulator is one of the most popular and it is widely used by the academic community. Its first version was launched in 1996, and derived from its ns-1 predecessor. It includes detailed models of a great number of TCP variations and many applications (such as HTTP traffic). It also supports wired and wireless networks modeling. In the wireless networks field, there are models for routing algorithms such as AODV and DSR, as well as models for the MAC protocol of the 802.11b protocol specification.

Scalable Wireless Ad hoc Network Simulator (SWANS) (Barr et al., 2005) is a wireless network simulator that can be used for sensor networks. It was built on top of the *Java in Simulation Time* (JIST) simulation platform. The JIST is general purpose *engine* simulation based on discrete events, which was developed in the JAVA language. This simulator is fo-



(a) *Offline category* (b) *Embedded category* (c) *Online category*

Figure 1: Categories of Vehicular network simulators.

cused on high performance and efficiency, and it simulates networks with four times higher performances than the ns-2 networks, with the same system requirements and level of detail (Kargl and Schoch, 2007). Jist/Swans was developed to meet the needs related to simulations of wireless networks and sensor networks. One of the main advantages of Jist/Swans is that it allows simulation of networks that require large-scale tests.

GloMoSim is a modular library for parallel simulation of wireless networks. This library was developed to be extended and combined and it has an API defined to each layer of the communication protocol stack. The GloMoSim has implemented: MAC layer of 802.11b protocol in detail; routing algorithms to wireless networks (AODV, DSR, and some others); transport layer protocol TCP; some application in the application layer level. Moreover, new protocols can be developed to extend GloMoSim capabilities. Executing a parallel model in GloMoSim is often transparent to the user. An interface can be used to set up the simulation parameters and also to designate a mapping strategy for running a parallel simulation.

GTNetS is a network simulator that had its development focused on parallelism. For this reason, it has shown a good performance as well as good scalability, even considering networks with millions of elements. The GTNetS was implemented in C++ using the object-oriented paradigm, which allows an easy extension of existing protocols. The GTNetS implements the following models: on the physical layer, it has implemented the 802.11, for wireless simulations, and the 802.3, for wired simulations. On the network layer, it has implemented the routing protocols AODV and DSR for wireless and the BGP and IEGRP for wired simulations. On the transport layer, it has implemented the TCP and UDP protocols. Finally, the application layer includes implementations for FTP, P2P, and client-server applications. The community has not provided support for GTNetS since 2008.

Although all networks simulators mentioned above have mobility models for mobile network simulations, they should not be used to simulate a vehicular network. That is because the network simulators implement random models, and as we have seen before, these models do not represent real mobility. A good alternative is to delegate the vehicular mobility to a traffic simulator.

2.2 Traffic Simulators

When vehicular networks emerged, researchers believed that it was a specific application of mobile networks. It means that at the beginning, vehicular network protocols were evaluated using random models. Such models worked perfectly for mobile networks, because generally it was about networks that simulated human behavior in an open field, such as a university campus or a conference. But it soon became clear that the random models did not represent the vehicular mobility, which produced undesirable results when evaluating new protocols.

Since then the study of vehicular mobility has become an open topic of constant research. This researches resulted in an evolution from random models, where the direction, speed and origin and destination points were chosen completely randomly, to models that extract information from actual maps and aim to generate vehicular mobility that are closer and closer to reality.

Thus far, several proposals of models and tools that simulate vehicular mobility have emerged. Some of them are based on mathematical models that simulate the streets as well as the driver's behavior. Others use maps to extract all kinds of information possible, for instance, the limits of the streets, the number of lanes, the direction of the tracks, the speed limits of each vehicle category,etc. Some of the main mobility simulators will be mentioned below.

VISSIM, proposed by (Fellendorf, 1994) in 1994, is a stochastic simulator of microscopic vehicular mobility, meaning that it simulates the behavior of each car individually. The quality of a traffic simulator is highly dependent on the quality of the traffic flow model. For this reason, the cars in queue and the lane change models are both part of VISSIM kernel. The model of cars in queue describes the behavior of a vehicle with respect to the vehicle that is in front of it, and may include overtaking models when there is a lane change model. Instead of using a deterministic cars in queue model, VISSIM uses a model based on a psychological study created in 1974 (Fritzsche and Ag, 1994). This approach uses random models to calculate the parameters that determine the potential acceleration of each driver.

VISSIM has standardized and well-defined interfaces that allow C-based programs to be implemented and integrated to it. This allowed the first Bidirectionally coupled simulators to emerge (Lochert et al., 2005) (Gorgorin et al., 2006). Bidirectionally coupled simulators are formed by the combination of a traffic simulator with a network simulator.

Another traffic simulator widely used by the academic community is the SUMO (Krajzewicz et al., 2012). SUMO is the acronym for *Simulation of Urban Mobility*, and it is a platform for microscopic traffic simulation, intermodal and multimodal, of continuous space and discrete events. The development of SUMO started in 2001, but it was only in 2002 that it was released under the GPL (Krajzewicz et al., 2002).

SUMO is not only a traffic simulator, but rather a set of applications that help perform and prepare traffic simulations. To allow greater flexibility, various configuration file formats are supported. These files can be imported from other tools or generated by SUMO itself. As previously mentioned, simulations representing the real world need high quality mobility models. For this reason, the SUMO has tools to generate the network topology, the vehicles and the traffic demand.

In SUMO, the real-world networks are represented as graphs, where nodes are the intersections and streets are represented by edges. The intersections consist of their own position, plus information about their shape and right-of-way rules. The edges are one-way connections between two nodes, and they have geometry information, the permitted classes of vehicles and the maximum speed allowed. Two tools can be used to generate the network topology, which are the *"netconverter"* and *"netgenerate"*. The *"netconverter"* allows the topology to be imported from other tools, such as VISSIM, OpenStreetMaps, etc. The *"netgenerate"* allows the generation of three different types of networks, which are manhatam grid, circular spider network and random network, as shown in Figure 2.

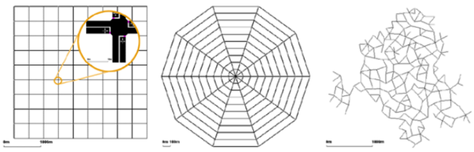


Figure 2: Examples of networks generated by *"netgenerate"*, from left to right, manhatam grid, circular spider network and random network (figure extracted from (Krajzewicz et al., 2012)).

SUMO is a purely microscopic traffic simulator. Because it is multimodal, it allows not only car traffic modeling, but also the modeling of public transport systems, rail systems and any other system that may influence or participate in the simulation.

The VanetMobiSim (Härri et al., 2006) is an extension of CanuMobiSim (Tian et al., 2002), a general purpose user mobility simulator. Coded in Java, both are platform independent and produce traces that can be used by different network simulators such as ns-2 (ns2, 2017), QualNet (qua, 2017) and GloMoSim (Zeng et al., 1998). CanuMobiSim provides an easily extensible architecture for mobility. However, the fact that it is designed for multiple purposes causes the level of detail in specific scenarios to be reduced. The VanetMobiSim therefore is a dedicated extension for vehicle networks.

As we have seen, a critical aspect for vehicle networks is the need for a simulation to reflect, as closely as possible, the actual behavior of vehicular traffic. When dealing with vehicular mobility models, we can separate the scenarios in a macro and a micro views.

In the macro view, both the network topology and its structure (number of lanes and direction) must be taken into account. Other factors that are relevant are the characteristics of the traffic (speed limits and vehicle restrictions by class), the presence of traffic restrictions (traffic signs and traffic lights) and finally the effects caused by points of interest (path between home and job).

Micro mobility refers to the behavior of each driver individually when interacting with other drivers or the road infrastructure. Examples of parameters that need to be informed to the models of micro mobility are: Travel speed in different traffic conditions, deceleration and overtaking criteria, driver behavior in the presence of intersections and traffic lights and general attitudes of the driver (which are usually related to age, sex, maturity, etc.). To model micro mobility parameters the VanetMobSim implemented two models that are familiar to researchers in the field, which are the FTM (Seskar et al., 1992) and the

IDM (Treiber et al., 2000).

STRAW (Choffnes and Bustamante, 2005) (acronym of STreet RAndom Waypoint) was developed in 2005 and is publicly available[1] for download. It was implemented as an extension of SWANS (Scalable Wireless Ad Hoc Network Simulator) (Barr, 2004), a Java-based, publicly available, and scalable wireless network simulator. The STRAW extract topology information, like road names, location, and shapes of roads, from a TIGER data set to create the topology of the network. According to (Rothery, 1992), the the car-following model is used to control the nodes movement and intersection management. As a drawback, the STRAW does not support lane changing and not consider a vehicle's current lane when it attempts to make a turn.

MOVE (Karnadi et al., 2007) is implemented in Java and runs atop of SUMO. MOVE consists of two main components: the Map Editor and the Vehicle Movement Editor. To build the road topology, the Map Editor allows three possibilities, which are: (i) the user can create the map manually, (ii) The map can be generate automatically and; (iii) the map can be imported from existing real maps, such as TIGER database. The Vehicle Movement Editor allows the user to specify the trip and the route that vehicles should take. Then, the data is fed into SUMO to generate a mobility trace, which can be used by network simulators such as ns-2 and QualNet to simulate a realistic vehicle movement. One of the main strengths of MOVE is the fact that it was implemented to allow users to rapidly generate realistic mobility models for vehicular networks.

In the previous sections, we presented the main simulators for a specific-domains which are relevant to vehicular networks. In the next section we will help the reader understand how the vehicular network researchers combined traffic simulators with network simulators to get a reliable analysis of protocols and applications on vehicular environments.

3 VEHICULAR NETWORK SIMULATORS

It is clear to vehicular networks researchers that neither traffic simulators nor network simulators meet all the requirements for a simulation. An alternative to solve this problem would be to record the mobility generated by a traffic simulator in a trace file and then use this file on the network simulator to update the position of the nodes. However, this approach does not allow mobility to be influenced by the network simulator, and therefore, does not reflect the topology changes. The current state of the art is the bidirectional coupling between traffic and network simulators. To make it possible, both simulators run the same simulation and exchange information on the status of each node. Although this approach enables high degree of realism, it requires the exchanging of a lot of messages between the simulators, which results in high computational cost when it is used in large scenarios. Characteristics and restrictions of bidirectionally coupled simulators are discussed below.

The *MSIECV*, also called *VISSIM/NS-2*, (Lochert et al., 2005) is the first simulator to propose the bidirectionally coupling between traffic and network simulators. The MSIECV architecture combines the ns-2 network simulator, VISSIM traffic simulator, and the Matlab/Simulink applications simulator. A simulation controller was implemented to manage the interaction between all simulators. A sync class was implemented to ensure that traffic and network simulators are synchronized in time during their execution.

GrooveNet (Mangharam et al., 2006), for Linux, was developed in 2006, and it was implemented in C++ and Qt. All types of vehicles communications are supported, that is, vehicle-to-vehicle, and vehicle-to-infrastructure through DSRC and 802.11. For vehicular mobility, the GooveNet includes the models for car-following, traffic light, lane changing and simulated GPS. Despite the authors say that the models were validated, they did not provide any information about its implementation. The main characteristic highlighted by the authors, is the ability of GrooveNet to make hybrid simulations, including real and simulated vehicles.

VanetSim (Tomandl et al., 2014) had its first version developed in 2008, but it was only in 2014 that its stable version was made available. Developed in Java, it is open source (GNU GPLv3) and can be downloaded in http://www.vanet-simulator.org/. The VanetSim was specifically designed to analyze attacks on privacy and security. To ensure a approximated real-world simulation providing realistic results, the VanetSim implements the state-of-the-art micro-mobility model (Krauß, 1998) and allows the importing of the network topology from OpenStreetMap[2].

(Wu et al., 2005) proposed a simulator that combines CORSIM to control the mobility of vehicles and the QualNet to model the communication between them. These two simulators were combined using a distributed simulation software pack-

[1] http://http://aqualab.cs.northwestern.edu/projects/111-c3-car-to-car-cooperation-for-vehicular-ad-hoc-networks

[2] http://www.openstreetmap.org

age called the Federal Simulations Development Kit (FDK) (McLean et al., 2001). The FDK implements services defined in the Interface Specification of High-level Architecture (Russo et al., 1995). Also, a Communication Layer was developed to define interactions between CORSIM and QualNet.

VCOM (Killat et al., 2007) is a hybrid library which combines the micro traffic simulator VISSIM with a discrete event based inter-vehicle communication simulator. The VCOM takes advantage from mathematical modeling to reduce the number of events generated by the communication. According to the authors, this approach can overcome ns-2 by a considerable speed-up. Also, an application module that contains all application logic provides a well-defined interface to simplify the implementation and evaluation of new applications.

NCTUns (Wang et al., 2007) and MoVES (Bononi et al., 2008) are embedded simulators developed respectively in 2007 and 2008, which implement their own network and traffic models. The difference between them is that MoVES was developed to be a parallel and distributed simulator. Another simulator that was proposed in the same year as NCTUns is the AutoMesh (Vuyyuru and Oguchi, 2007). The AutoMesh implements a custom mobility simulator, but uses ns-2 as network simulator.

The *ExNS3* (Arbabi and Weigle, 2010) is extended from the ns-3 and implements custom traffic models to be applied in vehicular network simulations. Unlike ExNS3, the *TraNS* (Piórkowski et al., 2008) uses ns-2 for network simulation and SUMO for traffic simulation. A TraCI interface is used to allow data exchange between the network and traffic simulators.

Veins is a simulation framework that provides coupling of the OMNET++ network simulator with the SUMO traffic simulator. It was initially proposed in 2008 by (Sommer and Dressler, 2008), after the authors discuss the development of simulators for vehicular networks. But it was only in (Sommer et al., 2011), in 2011, that it gained greater visibility in the academic community. The coupling between the OMNeT++ and SUMO happens through dedicated communication modules that have been implemented for both. During the simulation, these communication modules exchange information over TCP.

The iTETRIS (Rondinone et al., 2013) is a simulation platform developed in 2013, which is freely available to members of the iTETRIS community. It integrates and extends the SUMO and ns-3, which are two open source platforms widely used for traffic and network simulations. The iTETRIS is an open source platform, and its architecture is completely modular, which facilitates for the community to expand it in the future. It was designed to be aligned with international standards, more specifically, to be compatible with the ETSI architecture for intelligent transport systems, and it allows simulations to use either the 802.11p (Teixeira et al., 2014) or the ETSI ITS G5 standards.

Unlike the other bidirectionally coupled simulators, the VSimRTI (Schnemann, 2011) is a runtime simulation infrastructure that enables integration between any pair of simulators. Its goal is to make it as easy as possible for the user to prepare and implement a simulation. To achieve this, it uses a high-level architecture (HLA) simulation and modeling standard defined by the IEEE (893, 2000). For immediate use, a set of simulators is already coupled to VSimRTI, such as the traffic simulators VISSIM and SUMO, the network simulators JIST/SWANS and OMNeT ++ and the application simulator VSimRTI_APP as well as various data analysis tools.

The OVNIS (Pign et al., 2010), proposed in 2010, is one of the bidirectionally coupled simulator that uses SUMO as traffic simulator and ns-3 as network simulator. Another simulator that uses SUMO and ns-3 is the HINTS (Zemouri et al., 2012), proposed in 2012. HINTS differs from previously mentioned simulators by using a hybrid approach to generate vehicular mobility. It manages to bring together the best of both worlds, that is, the flexibility of the online approach and low computational cost of the offline approach. The authors mention that the new approach advances the state of the art in terms of performance by using resources more efficiently, thereby reducing the simulation time and the computational cost.

Although the bidirectionally coupled simulators allow simulations to be performed with a high degree of realism, some issues that are not treated by them need to be considered. As an example, we can mention the traffic demand, in which unrealistic traffic can be generated if random origin and destination points are chosen. Or, the poor quality of the maps, where the absence or incompleteness of information can influence network topology. Another factor that should be taken into consideration is the presence of different elements in the network. In the case of vehicular networks, future efforts should be applied to the insertion of elements in the simulation such as *Unmanned Aerial Vehicles* (UAVs), people walking and autonomous cars.

4 DISCUSSIONS

To understand the current state and future challenges in vehicular network simulators it is necessary to

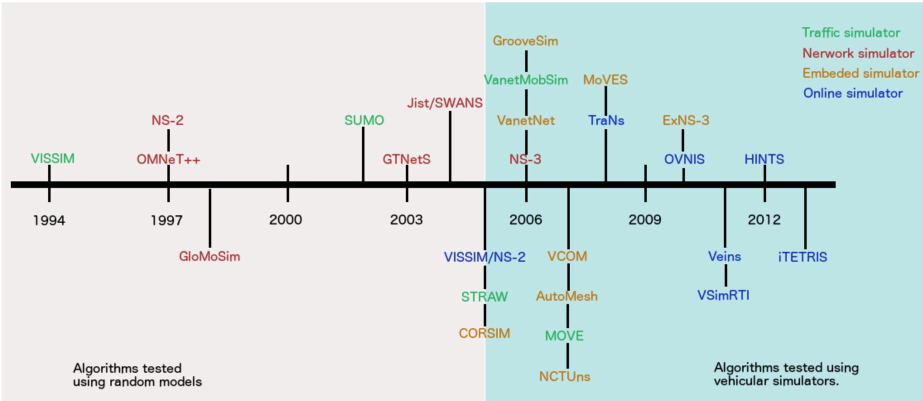


Figure 3: Evolution timeline of vehicular network simulators.

know how the evolution happened from the beginning. As we can see on Figure 3, until 2005 the applications and protocols proposed for vehicular networks were evaluated using random models. That is because all existing simulators until then had others purposes, more specifically, the traffic simulators were developed and used exclusively by traffic engineers, and network simulators were proposed to be applied in other types of networks, typically, sensor networks.

This scenario changes when, in 2005, (Lochert et al., 2005) proposed the MSIECV, which combines VISSIM (a micro-traffic simulator) with ns-2 (a network simulator) to generate scenarios that are closer to real ones. It is important to highlight that the technique proposed by (Lochert et al., 2005) comprises the current state-of-the-art. Although MSIECV was the first vehicular network simulator to combine traffic and network simulators (online simulators), it was only with Veins that this approach gained prominence in the scientific community. The Veins simulator was published three years latter by (Sommer and Dressler, 2008) and was consolidated in (Sommer et al., 2011).

We believe that one of the reasons for the lack of attention of the scientific community with the MSIECV, is the fact that VISSIM is a commercial tool that is not freely available. On the other hand, the Veins uses two freely and well-established tools that have a bigger support of the community, which are OMNeT++ and SUMO.

The Figure 3 also shows that most of the existing vehicular network simulators were proposed over a period of 3 years after the first one was proposed, in 2005. During this period, the computer scientists used to believe that implementing mathematical models to represent vehicular mobility was the better solution (embedded simulators). In 2008, (Sommer et al., 2011) turned the bidirectionally coupling between traffic and network simulators more popular, thus gaining attention of the researchers. It is important to notice, that the scientific community took five year to absorb the concept of online simulators, and three year more to start to use them.

As far as we know, after 2013 to date, there has been no significant effort to propose new vehicular network simulators. We believe that this is due to the fact that the scientific community changed focus and concentrated their efforts on implementing new models and improving the existing ones. Some examples of these models are: 802.11p (Eckhoff et al., 2012), DSRC/WAVE (Eckhoff and Sommer, 2012), Obstacle Shadowing (Sommer et al., 2014) and Antenna Patterns (Eckhoff et al., 2016)

5 CONCLUSION

New proposals of applications and protocols for vehicular networks appear every day. Simulation is the preferred method by the community when conducting the evaluation. In this context, this paper presented a temporal evolution of vehicular applications assessment from random models until the emergence and subsequent development of vehicular network simulators. We discussed the problems related to random models used to assess vehicular applications and how the bidirectionally coupling of network and traffic simulators can solve these problems. Additionally, we showed that the scientific community took more than five years to consolidate and use a new simulator paradigm.

ACKNOWLEDGEMENT

We acknowledge support from the Brazilian research agency (CNPq), the Research Foundation of the State of Minas Gerais (FAPEMIG), the CAPES Foundation and the Federal University of Ouro Preto (UFOP).

REFERENCES

(2000). IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules. *IEEE Std. 1516-2000*, pages i–22.

(2017). Ns-2. http://www.isi.edu/nsnam/ns/. Accessed: 2017-09-29.

(2017). Qualnet. http://web.scalable-networks.com/content/qualnet. Accessed: 2017-09-30.

Arbabi, H. and Weigle, M. C. (2010). Highway mobility and vehicular ad-hoc networks in ns-3. In *Proceedings of the Winter Simulation Conference*.

Arellano, W. and Mahgoub, I. (2013). Trafficmodeler extensions: A case for rapid vanet simulation using, omne, sumo, and veins. In *10th High Capacity Optical Networks and Emerging/Enabling Technologies*.

Aschenbruck, N., Ernst, R., Gerhards-Padilla, E., and Schwamborn, M. (2010). Bonnmotion: A mobility scenario generation and analysis tool. In *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*.

Barr, R. (2004). *An Efficient, Unifying Approach to Simulation Using Virtual Machines*. PhD thesis.

Barr, R., Haas, Z. J., and Van Renesse, R. (2005). Scalable wireless ad hoc network simulation. *Handbook on Theoretical and Algorithmic Aspects of Sensor, Ad hoc Wireless, and Peer-to-Peer Networks*, pages 297–311.

Bononi, L., Felice, M. D., DAngelo, G., Bracuto, M., and Donatiello, L. (2008). Moves: A framework for parallel and distributed simulation of wireless vehicular ad hoc networks. *Computer Networks*, 52:155–179.

Choffnes, D. R. and Bustamante, F. E. (2005). An integrated mobility and traffic model for vehicular wireless networks. In *Proceedings of the 2Nd ACM International Workshop on Vehicular Ad Hoc Networks*.

Eckhoff, D., Brummer, A., and Sommer, C. (2016). On the Impact of Antenna Patterns on VANET Simulation. In *8th IEEE Vehicular Networking Conference*.

Eckhoff, D. and Sommer, C. (2012). A Multi-Channel IEEE 1609.4 and 802.11p EDCA Model for the Veins Framework. In *5th ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems*.

Eckhoff, D. and Sommer, C. (2015). Simulative performance evaluation of vehicular networks. *Vehicular Communications and Networks: Architectures, Protocols, Operation and Deployment*, pages 255–274.

Eckhoff, D., Sommer, C., and Dressler, F. (2012). On the Necessity of Accurate IEEE 802.11p Models for IVC Protocol Simulation. In *75th IEEE Vehicular Technology Conference*, pages 1–5.

Fellendorf, M. (1994). Vissim: A microscopic simulation tool to evaluate actuated signal control including bus priority. In *64th Institute of Transportation Engineers Annual Meeting*.

Fischer, H.-J. (2015). *Standardization and Harmonization Activities Towards a Global C-ITS*, pages 23–36. Springer International Publishing.

Fritzsche, H.-T. and Ag, D.-b. (1994). Amodel for traffic simulation. *Traffic Engineering and Control*, 35:317–321.

Gorgorin, C., Gradinescu, V., Diaconescu, R., Cristea, V., and Ifode, L. (2006). An integrated vehicular and network simulator for vehicular ad-hoc networks. In *Proceedings of the European Simulation and Modelling Conference*.

Guan, S., Grande, R. E. D., and Boukerche, A. (2014). Real-time 3d visualization for distributed simulations of vanets. In *Proceedings of the IEEE/ACM 18th International Symposium on Distributed Simulation and Real Time Applications*.

Harri, J., Filali, F., and Bonnet, C. (2009). Mobility models for vehicular ad hoc networks: a survey and taxonomy. *IEEE Communications Surveys Tutorials*, 11(4):19–41.

Härri, J., Filali, F., Bonnet, C., and Fiore, M. (2006). Vanetmobisim: Generating realistic mobility patterns for vanets. In *Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks*.

Kargl, F. and Schoch, E. (2007). Simulation of manets: A qualitative comparison between jist/swans and ns-2. In *Proceedings of the 1st International Workshop on System Evaluation for Mobile Platforms*.

Karnadi, F. K., Mo, Z. H., and c. Lan, K. (2007). Rapid generation of realistic mobility models for vanet. In *2007 IEEE Wireless Communications and Networking Conference*.

Killat, M., Schmidt-Eisenlohr, F., Hartenstein, H., Rössel, C., Vortisch, P., Assenmacher, S., and Busch, F. (2007). Enabling efficient and accurate large-scale simulations of vanets for vehicular traffic management. In *Proceedings of the 4th ACM International Workshop on Vehicular Ad Hoc Networks*.

Krajzewicz, D., Erdmann, J., Behrisch, M., and Bieker, L. (2012). Recent development and applications of SUMO - Simulation of Urban MObility. *International Journal On Advances in Systems and Measurements*, 5:128–138.

Krajzewicz, D., Hertkorn, G., Rössel, C., and Wagner, P. (2002). Sumo (simulation of urban mobility)-an open-source traffic simulation. In *Proceedings of the 4th Middle East Symposium on Simulation and Modelling*.

Krauß, S. (1998). *Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics*. PhD thesis.

Lochert, C., Barthels, A., Cervantes, A., Mauve, M., and Caliskan, M. (2005). Multiple simulator interlinking environment for ivc. In *Proceedings of the 2Nd ACM International Workshop on Vehicular Ad Hoc Networks*.

Mangharam, R., Weller, D., Rajkumar, R., Mudalige, P., and Bai, F. (2006). Groovenet: A hybrid simulator for vehicle-to-vehicle networks. In *3rd Annual Interna-*

tional Conference on Mobile and Ubiquitous Systems - Workshops.

Marfia, G., Pau, G., De Sena, E., Giordano, E., and Gerla, M. (2007). Evaluating vehicle network strategies for downtown portland: Opportunistic infrastructure and the importance of realistic mobility models. In *Proceedings of the 1st International MobiSys Workshop on Mobile Opportunistic Networking*.

McLean, T., Fujimoto, R., and Fitzgibbons, B. (2001). Next generation real-time rti software. In *5th IEEE International Workshop on Distributed Simulation and Real-Time Applications.*

Mota, V. F., Cunha, F. D., Macedo, D. F., Nogueira, J. M., and Loureiro, A. A. (2014). Protocols, mobility models and tools in opportunistic networks: A survey. *Computer Communications*, 48:5–19.

Nimje, T. G. and Dorle, S. (2013). A survey on various mobility models to improve realistic simulation and accuracy of ivc protocols. In *International Conference on Emerging Trends in Computing, Communication and Nanotechnology (ICE-CCN)*.

Pign, Y., Danoy, G., and Bouvry, P. (2010). A platform for realistic online vehicular network management. In *IEEE Globecom Workshops*.

Piórkowski, M., Raya, M., Lugo, A. L., Papadimitratos, P., Grossglauser, M., and Hubaux, J.-P. (2008). Trans: Realistic joint traffic and network simulator for vanets. *SIGMOBILE Mobile Computer Communication Review*, 12:31–33.

Rondinone, M., Maneros, J., Krajzewicz, D., Bauza, R., Cataldi, P., Hrizi, F., Gozalvez, J., Kumar, V., Rckl, M., Lin, L., Lazaro, O., Leguay, J., Hrri, J., Vaz, S., Lopez, Y., Sepulcre, M., Wetterwald, M., Blokpoel, R., and Cartolano, F. (2013). itetris: A modular simulation platform for the large scale evaluation of cooperative {ITS} applications. *Simulation Modelling Practice and Theory*, 34:99–125.

Rothery, R. W. (1992). Car following models. *Traffic flow theory*.

Russo, K. L., Shuette, L. C., Smith, J. E., and McGuire, M. E. (1995). Effectiveness of various new bandwidth reduction techniques in modsaf. In *Proceedings of the 13th Workshop on Standards for the Interoperability of Distributed Simulations*.

Schnemann, B. (2011). {V2X} simulation runtime infrastructure vsimrti: An assessment tool to design smart traffic management systems. *Computer Networks*, 55:3189–3198.

Seskar, I., Maric, S. V., Holtzman, J., and Wasserman, J. (1992). Rate of location area updates in cellular systems. In *IEEE 42nd Vehicular Technology Conference*.

Sommer, C. and Dressler, F. (2008). Progressing toward realistic mobility models in vanet simulations. *IEEE Communications Magazine*, 46(11):132–137.

Sommer, C., Eckhoff, D., and Dressler, F. (2014). IVC in Cities: Signal Attenuation by Buildings and How Parked Cars Can Improve the Situation. *IEEE Transactions on Mobile Computing*, 13(8):1733–1745.

Sommer, C., German, R., and Dressler, F. (2011). Bidirectionally coupled network and road traffic simulation for improved ivc analysis. *IEEE Transactions on Mobile Computing*, 10:3–15.

Teixeira, F. A., e Silva, V. F., Leoni, J. L., Macedo, D. F., and Nogueira, J. M. S. (2014). Vehicular networks using the IEEE 802.11p standard: An experimental analysis. *Vehicular Communications*, 1:91–96.

Tian, J., Hahner, J., Becker, C., Stepanov, I., and Rothermel, K. (2002). Graph-based mobility model for mobile ad hoc network simulation. In *Simulation Symposium, 2002. Proceedings. 35th Annual*.

Tomandl, A., Herrmann, D., Fuchs, K. P., Federrath, H., and Scheuer, F. (2014). Vanetsim: An open source simulator for security and privacy concepts in vanets. In *12th International Conference on High Performance Computing Simulation (HPCS)*.

Tornell, S. M., Calafate, C. T., Cano, J. C., and Manzoni, P. (2015). Dtn protocols for vehicular networks: An application oriented overview. *IEEE Communications Surveys Tutorials*, 17:868–887.

Treiber, M., Hennecke, A., and Helbing, D. (2000). Congested traffic states in empirical observations and microscopic simulations. *Physical Review E*, 62:1805–1824.

Varga, A. (2010). *OMNeT++*, pages 35–59. Springer Berlin Heidelberg.

Vuyyuru, R. and Oguchi, K. (2007). Vehicle-to- vehicle ad hoc communication protocol evaluation using realistic simulation framework. In *2007 Fourth Annual Conference on Wireless on Demand Network Systems and Services*.

Wang, S. and Chou, C. (2009). {NCTUns} tool for wireless vehicular communication network researches. *Simulation Modelling Practice and Theory*, 17(7):1211–1226.

Wang, S. Y., Chou, C. L., Chiu, Y. H., Tzeng, Y. S., Hsu, M. S., Cheng, Y. W., Liu, W. L., and Ho, T. W. (2007). Nctuns 4.0: An integrated simulation platform for vehicular traffic, communication, and network researches. In *IEEE 66th Vehicular Technology Conference*.

Wu, H., Lee, J., Hunter, M., Fujimoto, R., Guensler, R., and Ko, J. (2005). Efficiency of simulated vehicle-to-vehicle message propagation in atlanta, georgia, i-75 corridor. *Transportation Research Record: Journal of the Transportation Research Board*, (1910):82–89.

Zemouri, S., Mehar, S., and Senouci, S. M. (2012). Hints: A novel approach for realistic simulations of vehicular communications. In *Proceedings of the 4th Global Information Infrastructure and Networking Symposium.*

Zeng, X., Bagrodia, R., and Gerla, M. (1998). Glomosim: a library for parallel simulation of large-scale wireless networks. In *Proceedings 12th Workshop on Parallel and Distributed Simulation.*