Simulation Tools and Techniques for Vehicular Communications and Applications

Christoph Sommer, Jérôme Härri, Fatma Hrizi, Björn Schünemann, and Falko Dressler

Abstract In the domain of Inter-Vehicle Communication (IVC), even though first field operational tests are already going on, performance evaluation is still dominated by simulation experiments. Yet, they require a very specific methodology as well as adapted tools and models not straightforwardly found in other domains. In this chapter, we first describe the required methodology in terms of scalability and applicability to select the right models and their interactions. In particular, we classify each class of models as in increasing level of granularity, and discuss in details the trade-off between scalability and applicability typical to IVC simulations. We then introduce some of the most widely used and openly available simulation frameworks applicable to the domain of IVC, and emphasize their capabilities related to the required methodology. In particular, we present the IVC simulation toolkits Veins, iTETRIS, and VSimRTI, three prominent simulation platforms openly available for IVC simulations. To provide guidelines for efficient and scalable simulations of IVC applications, we discuss the appropriate selection of models and their level of granularity as function of the IVC application requirements, and provide an overview of their corresponding support in each of toolkit.

Christoph Sommer, Falko Dressler
CCS Group, Univ. Paderborn, Paderborn, Germany
e-mail: {sommer,dressler}@ccs-labs.org

Jérôme Härri, Fatma Hrizi
Mobile Communications EURECOM, Sophia Antipolis, France
e-mail: {haerri,hrizi}@eurecom.fr

Björn Schünemann
OKS / Daimler Center for Automotive IT Innovations, Technische Universität Berlin, Germany
e-mail: bjoern.schuenemann@dcaiti.com

1 Introduction

Vehicular networks, Intelligent Transportation Systems (ITS) supported by wireless communication, have become one of the key research areas in the wireless networking community. There have been major achievements in this field including new concepts for efficient IVC [25, 52, 19]. Aside from the obvious use of cellular networks, first standards for short range radio communication have been defined in this field based on Dedicated Short Range Communication (DSRC), most prominently the IEEE 802.11p protocol [20]. Furthermore, standardization bodies are working on higher layer protocols such as those based on the IEEE 1609.4 Wireless Access in Vehicular Environments (WAVE) [21] or ETSI ITS-G5 [14] standards for IVC. First concepts such as single-hop broadcasts using Cooperative Awareness Messages (CAMs) / Basic Safety Messagess (BSMs) or ETSI ITS G5 geocast ideas provide the floor for more mature applications and protocols [51]. The relevant industry players are now ready for deploying short range broadcasting systems based on DSRC/WAVE in addition to cellular networks. Yet, many open question and challenges remain that have to be worked on. Examples include protocol design on all layers and especially scalability of proposed concepts.

In general, such performance assessment can be based on analytical models, simulation, or experimentation. In the very complex vehicular environment, analytical models are frequently based on too many unrealistic assumptions. Experimentation is extremely important and field operational tests are currently being conducted in Germany, France, Austria, and new ones are just starting in the United States. Yet, those tests are limited in space and time, i.e., only a certain number of applications and protocol variants can be tested. Smaller scale experiments primarily help validating available models. Thus, simulation is and will be the primary tool for performance assessment [43, 12].

In the last few years, substantial progress has been achieved in the field of simulation tools and techniques helping to assess the performance of IVC protocols. Thus (and in contrast to early stages of IVC research) there are now a good number of tools and models ready for use and openly available for the research community. Still, choosing exactly the right set of techniques, tools, and models is very complex. *Consolidation efforts* and recent research on *integrated IVC simulation frameworks* led to the development of three promising candidates: Veins [46], iTETRIS [38], and VSimRTI [42].

In this survey chapter, we study what specific aspects of IVC have an influence on simulation experiments. Essentially, we discuss all the models that are specific to performance evaluation of ITS, i.e., wireless networking aspects and mobility modeling. We introduce three of the best known and openly available simulation frameworks, Veins, iTETRIS, and VSimRTI, to provide an overview of the various aspects that need to be considered for producing realistic results, both for researchers already active in the field and those interested in moving into vehicular networking research.

Our main motivation for this chapter is to make researchers working on IVC and ITS applications as well as those moving into the field aware that the mentioned consolidation efforts successfully converged and the most widely used simulation frameworks Veins, iTETRIS, and VSimRTI meanwhile offer all the components and models needed for performance evaluation of vehicular networking protocols and applications – and that matching realistic setups very closely.

Our contributions can be summarized as follows:

- We give a broad overview of necessary simulation models that are specific to the IVC domain and discuss the level of granularity and applicability required by IVC applications. In particular, we provide guidelines for the selection of the right models at the right level of granularity to optimize the tradeoff between simulation scalability, granularity and applicability.
- We discuss the current state of simulation technology that is available for applying these models for investigating IVC approaches and present three fully featured toolkits that are openly available: Veins, iTETRIS, and VSimRTI.

The remainder of this chapter is organized as follows. We first briefly study typical IVC applications and their requirements in terms of simulation granularity and scale in Section 2. We then discuss models that are specific to IVC simulations, in particular those used for the wireless communication and networking part (Section 3) and those to simulate vehicles' mobility (Section 4). In the main part of the chapter, we introduce network and mobility simulators (Section 5) that build the basis for all the leading integrated simulation frameworks. We outline the capabilities of the most widely used frameworks in Section 6. We finally discuss the outcome of this chapter in Section 7.

2 IVC Applications

The most distinctive aspect of the design of IVC systems is the role of the application, leading to a strong bi-polarity in the communication design: IVC applications dictate the requirements, whereas the access technology and the shared mobile wireless medium formulate the capabilities. The challenge is therefore to find a trade-off between such a Yin and Yang. Accordingly, the design of a communication solution *fitting all* relates to an engineering myth, and dedicated communication solutions must be developed for mostly each IVC application. This impacts not only the protocol design at different layers, but also the simulation toolkits employed for their evaluations.

In order to bring order to such a chaotic situation, IVC applications have been classified [25, 19] in three classes as function of their scale, infrastructure support, and communication sensitivity [52, 19]:

- *Safe Mobility* contains safety-of-life communications to notify drivers of dangerous situations [24]. Such applications are localized, and are very sensitive to transmission impairments, viz. delay and loss. They must also work without the help of any form of infrastructure.

- *Smart Mobility* gathers and disseminates information related to traffic state or navigation guidelines in a decentralized way [26]. As the precision of traffic information decays with distance and time, the scale is small to medium, and communications are delay bounded.
- *Connected Mobility* encompasses all applications providing Internet access, content exchange, or commercial advertisements [2]. Applications in this class are fundamentally large scale, and focus mostly on throughput maximization.

Such classification also impacts the models and toolkits required to evaluate the performance (or conformance) of IVC applications. Given the class of the IVC application, the selection of the models is a constant trade-off between performance and precision. Thus, given the constrained simulation capability, the modeling of only those behaviors that are effectively impacting the IVC application should be sought.

2.1 Required Criteria

In orde to help select the right model with the right level of abstraction required for a particular IVC application, we provide in this section a list of criteria guiding the IVC's requirements.

- scale - represents how large the scenario should be both spatial as well as number of actors (vehicles). It can ranges from small scale when an area spanning a couple of vehicles is required to large scale when a city-wide network is required.
- applicability - represents the accuracy of the models used by simulations. It can range from a high level of accuracy when models need to be as close as possible to real conditions, to low when abstractions can be tolerated.
- infrastructure - represents the support of communication infrastructures, which can either be inexistent, homogeneous (metro-scale or micro-scale) or heterogeneous (metro-scale or micro-scale)
- COM pattern - represents the communication pattern for IVC. It can range from pure direct patterns (WiFi-direct, IEEE DSRC, ETSI ITS-G5, LTE-Direct,...) to relay patterns (4G/5G cellular networks, WiFi, etc..).
- NET protocol type - represents the support of network protocols, such as IPv6 or non-IP (e.g. ETSI Geonet, WAVE WSMP, or application defined).

IVC Application	**scale**	**applicability**	**infrastructure**	**COM pattern**	**NET protocol**
Safe Mobility	small	high	no	direct	none
Smart Mobility	medium	medium	yes	direct & relay	IPv6 & non-IP
Connected Mobility	large	low	yes	direct & relay	IPv6

Table 1 Overview of different granularity/applicability requirements for IVC applications

Safe Mobility application simulation is critically reliant on accurate modeling of transmission impairments. Thus, these must be precisely reproduced. Accordingly, the communication models should come as close as possible to the real condition found by the application, and the network models must be as realistic as the ones found in chipsets and computers. The scale relates to the geographic area but also to the number of actors. Constrained by finite computational resources, the level of realism used by the models required by traffic safety applications limits their evaluation to a few actors in a small area.

Smart Mobility requires a large amount of actors to gather trustworthy traffic states. Here, the precision of the traffic states is more important than the communication or networking models employed to provide it. Therefore, the level of precision of the models must frequently be reduced to cope with the scale of the network.

Connected Mobility are typically characterized by their large scale. This forces a further abstraction in the level of precision of the models employed to evaluate them.

Accordingly, the performance evaluation of IVC applications requires models to be carefully selected for an evaluation scenario. Models may be categorized into different layers (PHY and Channel, MAC, Network, Mobility, etc.). Literature and the vehicular community developed different levels of abstractions in each layers, each of them adapted to different requirements from the IVC applications. A trustworthy evaluation scenario is therefore an appropriate selection of models and their level of abstraction, all of which need to be carefully considered – and described [23].

In the next sections, we will describe various levels of accuracy found in the different models employed in a typical simulation-based evaluation scenario. We will provide guidelines for their efficient selections as a function of the required level of applicability to IVC applications. In particular, for each modeling class, we will provide a classification table containing *safe*, *smart*, *cnt* entries (i.e. corresponding respectively to safe, smart and connected mobility classes) and indicating which models are required and at which level of granularity and applicability they should be used.

3 Wireless Communications & Networking

Simulation tools used to assess the performance of IVC protocols and applications must support a variety of simulation models that is very specific to this application scenario [12]. In particular, signal propagation models need to incorporate the environmental conditions of communicating vehicles, models for channel access need to be able to treat the CSMA behavior, congestion control or even multi-channel operations. Also, information dissemination schemes for IVC, such as beaconing [13] or geocasting [31], or even facilities-layer protocols such as BSM or CAM might also be required for IVC applications. In the following, we discuss both domains and give examples for models that are frequently used in this area.

3.1 Signal Propagation and fading

In application-layer centered simulations, the most commonly considered physical layer effects are throughput, delay, and Bit Error Rate (BER). As the first two are straightforward to model in any discrete event simulation, in the following, we will focus on considerations of BER calculations only.

We can distinguish three types of fading models that need to be considered depending on the granularity of the simulated applications:

- *Distance* based models simply help understanding free space radio transmissions to a certain extend. But even in freeway scenarios, shadowing by other vehicles and fast fading need to be investigated.
- *Shadowing* can be modeled in a very abstract way using stochastic models, or using very accurate shapes of the obstacles, either using geometry-based approaches of fine-grained ray tracing.
- *Fast fading* is typically caused by multi-path propagation and is frequently either ignored or modeled using stochastic models.

In their simplest case, *Distance based* models the success or failure of any transmission by p_{err} in a fully-deterministic fashion, by comparing the distance d of sender and receiver with a fixed threshold distance d_{max}, as follows.

$$p_{\text{err}} = \begin{cases} 1 & \text{if } d > d_{\text{max}}, \\ 0 & \text{else.} \end{cases} \tag{1}$$

This approach is widely known under the name of *unit disk* graph connectivity model, however, is only appropriate for simulations that model large scale data flows rather than individual transmissions [48]. Fading models may include a higher granularity by considering the Signal to Interference plus Noise Ratio (SINR) of signals to arrive at a BER of transmissions, making the decision whether a packet can be received probabilistic. The SINR is calculated at each potential receiver by weighting the received power level of any interfering transmission (and the noise floor) against the received power level of the signal to be decoded. For this calculation of the receiving power P_r, any such model will need to rely on a set of radio propagation models to predict losses between transmitting and receiving station, where P_t is the transmit power, G_t and G_r are the transmit (and receive) antenna gains, and L_x are terms capturing loss effects during transmission [1, 34]. The terms L_d,L_s, L_f represent the loss effect from distance, shadowing and fast fading respectively.

$$P_r[\text{dBm}] = P_t[\text{dBm}] + G_t[\text{dB}] + G_r[\text{dB}] - L_d[\text{dB}] - L_s[\text{dB}] - L_f[\text{dB}] \tag{2}$$

The simplest of such propagation models are empirical adaptations of a *free space* model, taking into account only the path loss L_d, i.e., loss in transmission power over distance, that is caused by the radiation pattern of ideal omni-directional antennae, depending on the wavelength λ and an empirically determined path loss coefficient α [37]. One step up are *two ray interference* models [40], which deterministically derive path loss caused by constructive/destructive self-interference effects of the signal component that is reflected on the ground, taking into account the height over ground of transmitter h_t and receiver antenna h_r together with an empirically determined relative permittivity of the ground ε_r [44].

Although providing an increasing level of realism toward large-scale fading impacts, they do not include the critical fading effects found in a vehicular environment, i.e. shadowing or fast fading. To consider them, a shadowing model L_s, and a fast fading model L_f must also be considered in the overall received power P_r.

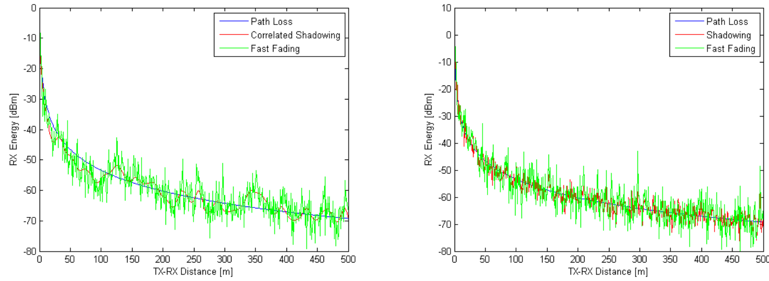
Shadowing models help determining additional factors that contribute to path loss such as obstacles, which shield a receiver from all or part of the radiated power. If all transmission attempts can be assumed to be uncorrelated in time (on the order of seconds) and space (on the order of tens of meters), such obstacles can be modeled using purely stochastic models, e.g., a log-normal shadowing model [10]. However, if these assumptions do not hold, i.e., if multiple transmission attempts are made at roughly the same point in time or space, correlated (space or time) models are required, as well as deterministic classification on how positions of buildings [45] and location of vehicles [6] impact line-of-sight and Fresnel zones, and as such the path loss. Also, shadowing models can be made arbitrarily complex, such as building precise fading database from measurements campaigns or by relying on *ray tracing* techniques [47], yet at the cost of extreme computational overhead or reduced spatial resolution. Simulations requiring this class of models also require a very specific simulation framework that are seldom found in available tools.

Additional small scale, or fast fading effects, can be added as further loss terms, to model Rayleigh or Rician fading. This typically would depend on the environment and mobility of vehicles. For example, a simple Nakagami-m model can stochastically capture narrow-band multi-path fading effects, considering the received power level and an empirically determined value of m, which can be used to adapt the model for differently pronounced line-of-sight components [40]. For further details related to vehicular channel and propagation models for IVC, we refer the reader to Chapter X12.

As depicted in Fig. 1 narrow-band fading, shadowing, and path loss models can then be cumulatively used to derive the SINR and, as described, the packet reception probability. The choice of the class of fading model as well as their level of granularity significantly depend on the IVC application requirements, as summarized in Table 2.

3.2 Channel Access

Channel Access protocols have to answer the basic question of how to efficiently share the radio channel resources between multiple transmitters. A strong particularity of IVC comes from the strict *distributed* requirement for channel access, opposed to personal communications where a centralized approach is mostly found (WLAN, 4G).



(a) Correlated Shadowing (b) Uncorrelated Shadowing

Fig. 1 Illustration of the cumulative impact of the three types of fading

mobility	**granularity**	**applicability**	**safe**	**smart**	**cnt**	**application**
distance	unit disk	low			✓	analytical/asymptotic
	free space	medium			✓	long range radio tx
	self-interference	high			✓	short range radio tx
shadow	ignored	low			✓	no buildings or vehicles
	stochastic	medium		✓	✓	signals uncorrelated in time and space
	stochastic correlations	high	✓			signal correlated in time and/or space
	geometry-based	high	✓			location-dependent fading
	ray tracing	very high	✓			specific real-world modeling
fast	ignored	medium		✓	✓	isolated protocol aspects
	stochastic	high	✓			system behavior

Table 2 Overview of different granularity/applicability propagation models and possible applications

Conceptually speaking, distributed channel access mechanisms for IVC have to provide the following three functions:

- *Multiple Access Control* - Provide fair access to the vehicular channel. It determines when a channel is busy or idle, models when a transmission has failed, and properly reacts.
- *Congestion Control* - Restrict the spatial or temporal usage of the vehicular channel to avoid degrading channel conditions.
- *Multi-Channel Control* - Distribute communications across multiple available IVC channels for efficient spectrum usage.

If packet reception probability is modeled independent of channel conditions (idle/busy) or if the probability of failed transmissions may be abstracted by a stochastic process, the simplest access control solution is to abstract packet reception by a packet reception probability modeled either by a deterministic function (e.g. *unit disk* model) or a stochastic model. The latter can take into account various communication parameters, such as transmit range, number of neighbors in range, transmit rate, or packet size, in order to model an decreasing reception probability with increasing load on the channel. It may also be noted that such approach is not strictly linked to a particular technology, as the stochastic reception model may be tuned to the communication specificities of LTE or a ITS G5 technologies. Simulations using such an *idealistic* model can certainly yield highly instructive results and can be performed extremely fast [27, 5]. It is in particular preferred for smart mobility applications.

The performance of safe mobility applications is critically reliant on the reception of a particular packet or even a signal, which requires fine grained multiple access models [49]. In such condition, estimating if and when a station might detect the channel to be busy is already non-trivial [8]. Accordingly, multiple access modeling may again have different levels of abstraction, ranging from a simple *CSMA/CA* listen-before-talk protocol to full fledged IEEE 802.11-2012 or even higher layer control protocols such as those in ITS G5. This granularity relates to a conformance level to a standard describing IVC protocols. A low conformance level will not reflect realistically how an IVC protocol would truly behave once deployed, whereas an IVC protocol with high level of conformance to a standard will provide a high level of confidence in measured effects. The selection of the appropriate conformance level for a multiple access model is a trade-off between confidence and performance, as modeling a high level of conformance requires also more computational resources.

Providing fair multi-access control does not specifically mean that the need of all transmitters may be supported. A higher congestion level impacts the delay, throughput and reliability of IVC protocols. In a vehicular context, distributed multi-channel mechanisms cannot react to the transmissions they cannot sense (e.g., hidden nodes). Congestion control mechanisms are therefore critical to maintain a particular Quality-of-Service (QoS), which can take the shape of a reduced transmit power, rate, or packet size in order to control the load on the channel. As depicted on Fig. 3.2, ETSI channel access regulations restrict the usage of the ITS-G5A channels for safety-related communications under a congestion control mechanism. If the objective of the IVC protocol or application is conformance, then such congestion control mechanism should be included and as close as possible to the standards, such as ETSI Decentralized Congestion Control (DCC). Congestion control mechanisms are also not specific to a particular access technology, as long as the transmit parameters may be controlled and altered on a per-packet basis. For further details related to DCC mechanisms for IVC, we refer the reader to Chapter X4.

Directives from EU ETSI and the US FCC allocate multiple channels to IVC applications. So far, most IVC focused on one channel, but IVC traffic should benefit from multiple channels as function of their types, either by traffic off-loading on alternative channels, or for IVC service management. Various standards for multi-channel systems exist, such as those based on the IEEE 1609.4 WAVE [21] or ETSI ITS-G5 [14] standards for IVC. With more than one channel to choose from (and potentially less transceivers than channels), the MAC layer has to pick which channel to listen to and which to pick for transmitting [9, 28]. Accordingly, packets cannot be received if a transmitter is away on a different channel, which generates delay and unreliability.

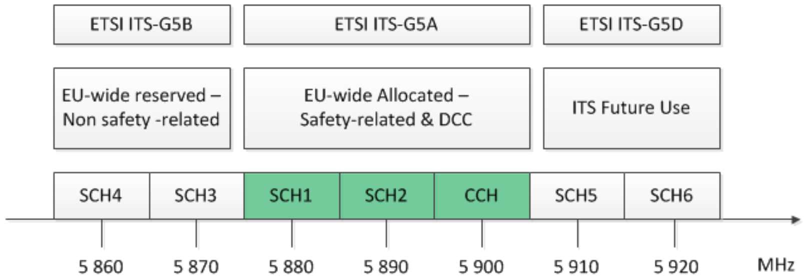


Fig. 2 ETSI channel allocation in EU

mobility	granularity	applicability	safe	smart	cnt	application
MAC	unit disk	low			✓	asymptotic performance
	stochastic	medium		✓	✓	analytical/asymptotical
	CSMA-CA	medium		✓	✓	packet level tests
	ITS-G5/ DSRC	high	✓	✓	✓	validation and conformance
congestion control	ignored	low			✓	large-scale test
	generic model	medium		✓	✓	channel load impact
	ITS DCC	high	✓	✓	✓	validation and conformance
multi-channel	ignored	medium	✓	✓	✓	isolated examination
	off-loading	high	✓			multi-channel congestion
	services	high		✓	✓	system behavior

Table 3 Overview of different granularity/applicability channel access models and possible applications

3.3 Networking

Network protocols have the responsibility of disseminating IVC information through vehicular networks or to a back-end or Internet service. Another objective of a networking layer is to provide a network-wide view of particular IVC parameters. The objective of this section is not to exhaustively describe vehicular networking protocols, but illustrate the link between the class of IVC applications and the required IVC networking stack.

Following Fig. 3, the vehicular networking stack is situated between IVC applications/facilities and IVC access technologies. Irrespective of the various IVC standards, network protocols may be classified in three types:

- *Null Net* - A network layer is inexistent. This is typically the case in the ISO CALM standard, where IVC packets contain all required information to be transmitted to IVC access technologies.
- *non-IP Net* - A non IP network layer is available. This is found in the WAVE Short Message Protocol (WSMP) stack and the ETSI Geo-networking stack, although contrary to ETSI Geo-networking, WSMP are mainly headers for IVC service management, and do not provide multi-hop networking support.
- *IPv6 Net* - An IPv6 stack is available, including the required IPv6 mobility management functions.

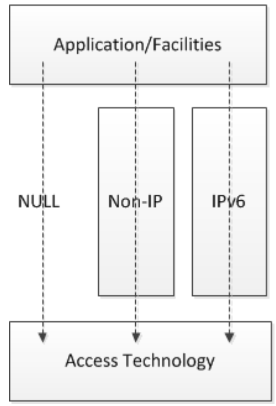


Fig. 3 Conceptual overview of the different NET layer stack for IVC applications

Safe mobility applications are typically restricted to single-hop, potentially including multi-hop piggybacking. Accordingly, a networking stack is most likely not required. For instance, both CAM and BSM are application/facilities layer messages and contain sufficient addressing information to support safety-related IVC. ETSI provides network support for multi-hop dissemination of emergency Decentralized Emergency Notification Message (DENM). Although a networking layer might not be required for most of the safe mobility applications, a Facilities layer stack would be required for high conformity with ETSI standards.

Smart mobility applications require data dissemination either horizontally (directly between vehicles) or vertically (to back-end or Internet services). In that case, the selection of the networking stack depends on the IVC protocols. As the Internet Engineering Task Force (IETF) does not currently provide IPv6 multi-hop support for ITS applications over a vehicular network, a non-IP stack would be required if smart mobility data needs to be disseminated between vehicles. If smart mobility traffic is mostly vertical, an IPv6 stack is required as vehicles need to interconnect to Internet.

As connected mobility applications are mostly based on keeping vehicles connected to the back-end or Internet, an IPv6 stack is critical to evaluate the performance of this class of IVC applications. Considering the potential high scale of vehicular traffic on the IPv6 back-end, efficient IPv6 mobility and traffic management might also be required.

It may also be noted that the selection of the networking stack also depends on the selected IVC as described in Table 4. For instance, if the selected IVC access technology is 4G/5G, then an IPv6 networking support is required. With the advent of the Internet-of-Things (IoT) paradidm, it is expected that future new networking layers will become available, including IoT-specific addressing and routing.

mobility	granularity	applicability	safe	smart	cnt	application
NULL		low	✓	✓	✓	cooperative awareness
Non-IP	1-hop broadcast	low	✓	✓	✓	neighborhood monitor
	geo-routing	medium	✓	✓	✓	isolated IVC protocols
	DTN	high		✓	✓	traffic offloading
IPv6	isolated	low			✓	static, or IP protocol test
	mobility mgnt	high		✓	✓	impact of mobility

Table 4 Overview of different granularity/applicability network models and possible applications

4 Mobility Modeling

When simulating IVC applications, the crucial role of vehicular mobility models has been illustrated by multiple studies [12, 15].

4.1 Overview and Constraints

As illustrated in Fig. 4, vehicular mobility models consist of three major blocks: *Motion Constraints*, *Mobility Demand*, and *Traffic Demands*, mutually influencing each other. Mobility and traffic demands provide a description of the desired mobility, whereas motion constraints limit them. The level of accuracy of a particular model depends on the granularity in each of these two blocks.

Considering motion constraints, we can distinguish three major levels – An increasing the level of accuracy in the motion constraint block shows a significant impact on mobility patterns, but the impact of traffic demand should also be carefully considered:

- *Random Topology* – Random graph representations, such as star, honeycombs, or Voronoi tessellations. They can be rapidly generated, but only provide an abstract topology representation.
- *Real Topological Maps* – Real street and cadastral data provide a higher accuracy to the graph representation, and aim at reflecting motion constraints found in urban areas.
- *Maps with added features* – Speed, access restrictions, turn limitations, and Points-of-Interest (POI) are added to the graph representation for an increased accuracy in motion constraints. Traffic lights, variable message signs, and similar technologies act as dynamic influence on road topology.
- *Maps with precise geometry* – Streets are no longer represented by a graph, but as a precise geometrical shape, including lane, sidewalk and any street-level geomety. This enables the precise positioning of vehicles and pedestrians in the network, in particular their interactions.

Mobility demands may also be classified as a function of their scope, granularity level, and characteristics. For many years, *random models* have been the preferred and easiest solutions to model mobility for communication networks. Their simplicity and stochastic properties are counter-balancing their limited precision to model vehicular mobility.

Depending on the scale of the evaluation, *flow models* such as microscopic models are needed, exactly representing the interaction of a vehicle with other cars, traffic regulations, street topologies, etc. For large-scale evaluation, mesoscopic models are usually a good compromise between accuracy and simulation performance. Finally, when pure physical laws cannot represent vehicular motions, *behavioral models* allow to describe patterns based on stimuli and responses, with the objective to represent the human behavior rather than flow or traffic equations [11].

Traffic Demands represent large scale mobility patterns of vehicular mobility. A *Random turns* approach operates a random selection of a new direction at each intersection. For an increased granularity, *Origin-Destination (O-D) matrices* are use to represent initial and destination areas, as well as optimal paths between them. For an even finer granularity, area-wide or city-wide *Population Models* may be obtained in order to adjust the O-D Matrices, as well as the preferences in paths and mobility options (bus, car, etc..).

A concept map for vehicular mobility models, where the three main blocks are enriched with added blocks and features, each of them providing an increased accuracy, but at an increased cost and complexity is shown in Fig. 4. The figure outlines the many aspects influencing motion constraints, mobility and traffic demand, all defining the vehicular mobility model. The relationship between the various levels of granularity in the mobility models and the IVC applications are illustrated in Table 5.

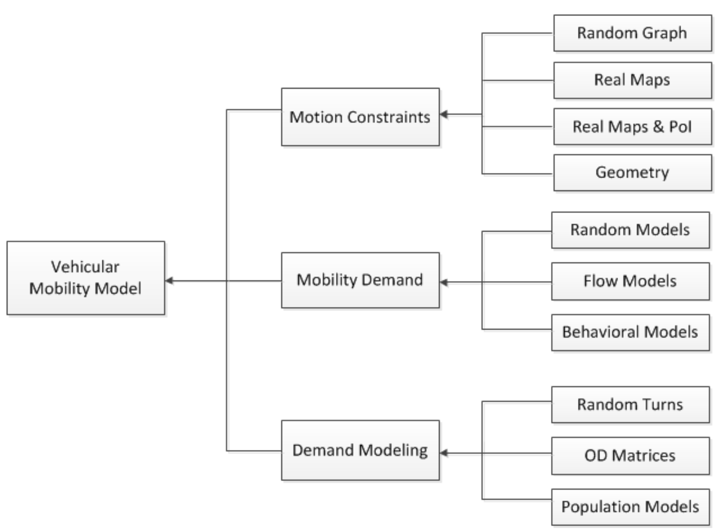


Fig. 4 Concept map for vehicular mobility models, where the two main blocks are enriched with added blocks and features, each of them providing an increased accuracy, but at an increased cost and complexity

mobility	granularity	applicability	safe	smart	cnt	application
Motion	random	low			✓	analytical evaluations
Constraints	maps	low			✓	impact of mobility on IVC
	maps and PoI	medium		✓		dynamic navigation
	street geometry	high	✓			safety and autonomous drive
Mobility	random	low			✓	analytical evaluations
Demands	flow	medium		✓	✓	traffic jam modeling
	behavioral	high	✓			accident modeling
Traffic	random	low			✓	analytical evaluations
Demands	OD Matrix	medium		✓	✓	social impact on smart mob.
	Population	high		✓		multi-modal IVC app.

Table 5 Overview of different granularity/applicability of mobility models and possible applications

4.2 Towards More Realistic Mobility Modeling

One of the challenges [15] is to develop a traffic demand providing an accurate vehicular mobility description at both macroscopic and microscopic levels. For instance, it could be shown that a flow model was an absolute minimum to reproduce the realistic microscopic Vehicle-to-Vehicle (V2V) interactions, typically observed when evaluating traffic safety applications. Traffic models are also required when large scale motion patterns are necessary, for instance when evaluating traffic efficiency applications.

Flow and traffic models are based on physical equations developed to provide accident-free and optimal traffic spreading. Given a specific situation, cars will always react the same way. In real live, drivers are not perfect and react to stimuli with different reactions as function of the context. Such patterns are represented by behavioral models, where reactions are not following any strict rule, but adapted to given contexts. These models are for instance the preferred choice to model accidents.

Finally, trace/survey models are experiencing an increasing interest from the community in building large-scale calibrated urban mobility models. Although providing a level of accuracy very close to the calibration data, the limitations are their complexity and the spatial and temporal constraints from the calibration data.

The choice of a mobility model and its level of accuracy depend on the application and protocol requirements, and should carefully be selected for accurate and scalable mobility modeling. Although all of the common network simulators have, by now, integrated support for node mobility, their mobility models' level of sophistication varies widely. It has long been established that the quality of results obtained from mobile ad hoc network simulations is heavily influenced by the quality of the employed mobility model [7]. Furthermore, the impact of mobility models on IVC simulation results, as well as the inadequacy of simple random mobility models, are well documented in the literature [39, 43].

For a detailed comprehensive discussion and current state-of-art of vehicular mobility models, we further refer the reader to Chapter X11.

5 Network and Mobility Simulation Tools

In the following, we introduce and briefly describe simulators frequently used for performance evaluation of IVC applications and protocols [23]. For a comprehensive discussion of most of the tools and methodology mentioned below, we refer the reader to the very complete survey by [17].

5.1 The ns-3 Network Simulator

The network simulator 3 (ns-3) [16] is an open discrete-event simulation environment that was designed to be the successor of the popular simulator ns-2. Aiming to be more scalable and more open for extension, it significantly differs from ns-2 with its novel structural and modular implementation.

The core architecture is object-oriented, it has been developed in C++, contrary to ns-2 (which is written in OTcl and C++), ns-3 optionally uses python scripts for simulations. Some C++ based models have been ported from ns-2 to ns-3. Being open also to commercial use, its base architecture has been designed to support network virtualization and real testbed integration.

The object aggregation model is the main feature of ns-3. Multiple objects could be linked together at run-time such as, nodes, applications and protocol stacks. This mechanism has been proposed as solution to the "weak base class" problem in C++ where the base class should be modified each time the programmer needs to reuse it with different configuration. Moreover, the aggregation model handle the access between the different aggregated objects and eases the automatic memory control.

Although less wide then ns-2, ns-3 support an increasing amount of models, from IEEE 802.11, 3GPP LTE, to IPv6 and selected MANET routing protocols. ns-3 is a packet-level simulator, but if the precise simulation of wireless transmissions at a signal level is required, ns-3 has been extended by mittag et al. [33] with a physical layer implementation, integrating OFDM symbol processing, space-time channel modeling as well as a precise reception model. This however requires significant computational resources, which restricts this module to be used on specific small-scale wireless communication scenarios.

ns-3 has also been extended by the iTETRIS [38] and the subsequent COLOMBO[1] projects with the support of a ETSI ITS compliant IVC protocol stack as depicted in Fig. 5.1. Major communication-related facilities, such as CAM, DENM and Service Announcement Message (SAM) have been integrated, as well as a non-IP

[1] http://www.colombo-fp7.eu/

ETSI Geo-networking stack supporting various geographic routing protocols (geo-unicast, geo-broadcast, geo-anycast) and Delay Tolerant Networks (DTN) functions. On the lower layers, heterogeneous access technologies are available, including ETSI ITS-G5. The management side of iTETRIS supports *multi-channel operations*, as well as a *multi-technology selections*. Whereas the former one is capable of offloading traffic between ITS G5 channels, the latter is capable of off-loading to multiple the access technology anf the network stack as function of the IVC application requirements. An ETSI compliant DCC is also available, first with a *DCC channel load monitor* function at the management layer, as well as a *DCC flow control* function in the data plane.

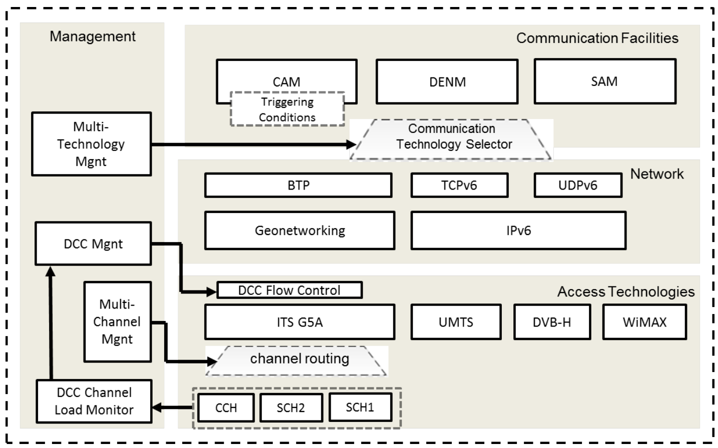


Fig. 5 ETSI ITS extensions on ns3

5.2 The OMNeT++ Simulation Environment

OMNeT++, now at version 4, is an Open Source simulation environment that is distributed free for non-commercial use [50]. A separate version of the same simulation environment which is licensed for commercial use is sold by *Simulcraft, Inc.* under the *OMNEST* brand. OMNeT++ comprises an IDE, an execution environment, and a simulation kernel. The IDE is based on Eclipse, enhanced with facilities to graphically assemble and configure simulations, as an alternative to editing the plain text files. The execution environment exists in two flavors. The command line based environment targets unattended batch runs on dedicated machines. The graphical environment better supports interactive interactions with components of a running simulation, allowing to directly monitor or alter internal states.

OMNeT++ enforces a strict separation of behavioral and descriptive code. All behavioral code (i.e., code specifying how simple modules handle and send messages, as well as how channels handle messages) is written as C++ code linking to the OMNeT++ kernel. All descriptive code (i.e., code declaring the structure of modules/channels and messages) is stored in plain-text *Message Definition* (msg) and *Network Description* (ned) files, respectively. All run-time configuration of modules is achieved by an *Initialization File* (ini). With all behavioral code being contained in a C++ program, OMNeT++ components can easily interface with third-party libraries and can be debugged using off-the-shelf utilities; thus it lends itself equally well to rapid prototyping and developing production quality applications.

Building on the discrete event simulation kernel are several module libraries modeling various protocol stacks. One popular example is the *MiXiM* [29] module library, which is focusing on accurate channel modeling and signal processing. Signals at a certain location are modeled as three-dimensional entities whose power level varies over both time and frequency. Calculating how such signals propagate in a simulation, as well as how they interfere with each other, is handled by MiXiM itself with no further effort from the model developer required. Thus, MiXiM lends itself very well to IVC simulation, where accurate models of common upper-layer Internet protocols matter less than precise simulation of wireless transmissions. If this is desired, another module library, the *INET Framework*, focuses on accurate representations of IPv4 and IPv6 as well as Internet transport layer protocols and applications, while overlay networks and peer to peer networking is the focus of the *Oversim* module library extension. Cellular networks are handled by yet other module libraries, like *SimuLTE* for simulating LTE and LTE advanced. Among the IVC specific models that are configured for use in OMNET++ are path loss models such as Two Ray Interference, shadowing by buildings and vehicles, IEEE 802.11p DSRC, and IEEE 1609.4 WAVE as well as auxiliary models such as driver behavior and emission computation. Models for the ETSI ITS G5 protocol stack, most importantly the DCC have been implemented. OMNeT++ acts as a framework for executing simulations that are assembled from these module libraries.

5.3 *JiST/SWANS*

The scalable wireless network simulator SWANS is built atop the JiST platform [3]. The JiST design is used to achieve high simulation throughput, save memory, and run standard Java network applications over simulated networks. JiST/SWANS programming code is open source and released under the Cornell Research Foundation license.

Particularly for vehicular ad hoc networks, a number of extensions and improvements for JiST/SWANS were developed at Ulm University [41]. For example, the DUCKS tool, uses configuration files to define a complete setup of a simulation study. Moreover, the DUCKS tool supports an extensible model for storing result data. A configuration file is used to set up how results are saved. Thus, simulation results can be stored, for instance, in a MySQL database.

5.4 The SUMO Simulation Environment

Today, a huge number of simulation environments exist which implement traffic microsimulation models. In the interest of comparability of research results, however, it is evidently more beneficial to use readily available Free and Open Source Software simulation environments. In this section, we briefly introduce the most popular of those [23], the SUMO [30] microscopic road traffic simulation environment. This simulator is in widespread use in the research community, which makes it easy to compare results from different simulations.

SUMO is Free and Open Source Software licensed under the *GNU General Public License* (version 2 or later), is highly portable, and allows high-performance simulations of multi-modal traffic in city-scale networks. Simulations in SUMO can be run both with and without the OpenGL-based GUI, which allows for direct interaction with a running simulation. In order to afford accurate simulations of a large number of vehicles, SUMO was designed to incorporate an adaptation of the aforementioned microscopic vehicle mobility model described by Kraußand, more recently, more complex mobility models such as the IDM model. The parameterization of vehicles can be freely chosen with each vehicle following a statically assigned route, a dynamically generated route, or driving according to a configured timetable. Traffic flows can be assigned manually, computed based on demand data, or generated completely at random. Each road in SUMO can consist of multiple lanes, each of which can be restricted to be usable only by certain vehicle classes. Individual lanes can have any shape and can be interconnected with junctions, with inter-junction traffic being regulated by simple right-of-way rules, by fixed-program traffic lights, or by demand-actuated traffic lights. For individual traffic, vehicle trips are generated either from Origin/Destination (OD) matrices or following random turns at intersections and a wide variety of mobility related metrics are collected.

The *TraCI* interface allows the interconnection of external control functions to SUMO. TraCI provides flexible open APIs to retrieve metrics (e.g., for combining them with network metrics into ITS-specific metrics) and to control most of the SUMO parameters, from the mobility model, road network, to traffic light control. Accordingly, the TraCI interface allows to add new functionalities to SUMO or connect it to other tools. For instance, it is possible to alter the microscopic model or even implement a completely new one, or interconnect an external traffic light control module to SUMO.

5.5 VISSIM

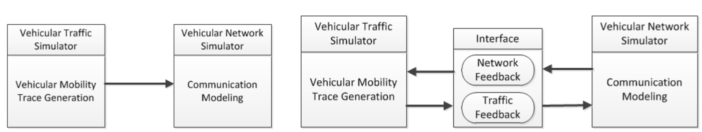
The commercial simulation software VISSIM [32] is a microscopic traffic simulator based on a psycho-physical driver behavior model for vehicles and a social force model for pedestrians. Simulated traffic scenarios can consist of cars, trucks, buses, two-wheelers (bicycles, motorcycles), and public transport (bus, tram, underground). Furthermore, VISSIM includes a dedicated pedestrian model. For individual traffic, vehicle trips are generated by OD matrices and for public transport, schedules are defined. The routing through the traffic network is done by predefined routes.

VISSIM provides interfaces to add new functionality or connect other tools. A DLL-interface allows to implement new driver-behavior models. A COM-interface can be used to have read and write access to simulation data during runtime of a simulation. As a result, a coupling with other simulation tools is possible allowing an interaction during the runtime of the simulation.

6 Integrated IVC Simulation Toolkits

A key difference between the evaluation of wireless communication application and IVC application, is that in the former, mobility is considered simply as a perturbation, whereas in the latter mobility is the application and must be able to be dynamically altered. Moreover, the various specific expertise in the required models for IVC applications, im particular when considering compliance to standards, makes it difficult to integrate all required models in a single simulator. Considering that each community (mobility, application, communication, etc..) traditionally relies on their specific simulators, an efficient strategy is to federate the required simulators in an IVC simulation framework.

In particular, when communication and mobility must interact, the traditional approach is to extract synthetic traces from traffic simulators in the shape of mobility files that may later be integrated into the network simulator using a dedicated parser (see Fig. 6(a)). This approach remains a favorite choice when mobility does not need to be influenced by vehicular communication or networking. Considering IVC applications, mobility shall be influenced, either to avoid accident or to reduce congestion. A bi-directional interaction is created between two (or more) simulators to be able to exchange mobility data or to influence the control of one or the other (see Fig. 6(b). This approach is currently the favorite choice for evaluation of IVC applications.



(a) Isolated Case (b) Federated Case

Fig. 6 Interactions between network and traffic simulators

There has been substantial effort leading to the development of quite a number of integrated IVC simulation frameworks. In the last years, consolidation efforts helped these efforts to converge. In the following, we introduce three promising candidates: Veins [46], iTETRIS [38], and VSimRTI [42].

6.1 Veins

Veins (Vehicles in Network Simulation) [46] is a simulation framework that is built on the aforementioned OMNeT++ simulation environment. It employs the OMNeT++ simulation kernel for discrete event simulation, that is, all simulation control and data collection is performed by OMNeT++. Veins instantiates SUMO to model vehicle movement and it provides a modular framework for the simulation of custom applications. In order to abstract away from discrete event simulation of wireless channels, e.g., managing event routing between nodes and modeling signal processing, the aforementioned MiXiM model suite is used, whereas dedicated model libraries are used for simulating, e.g., Internet protocols or cellular network communication.

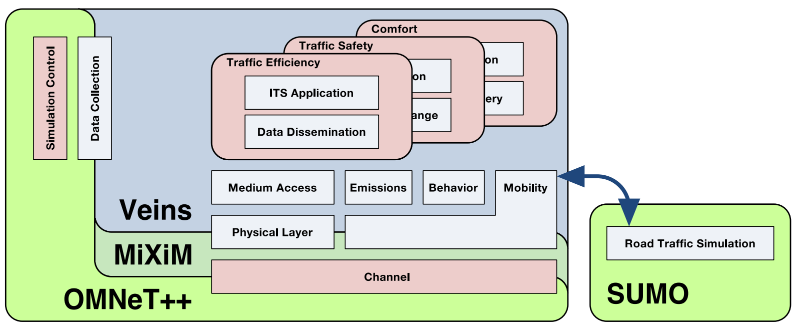


Fig. 7 The Veins simulation framework.

As illustrated in Fig. 7, Veins builds on this basis to provide a suite of models that can then, in turn, serve as a modular framework for simulating applications. Based on the suite of IVC models available in OMNET++, custom and application-specific data generation and dissemination protocols can be implemented, e.g., for traffic safety, traffic efficiency, and infotainment applications.

Such application simulations, and all used modules of Veins, are compiled and linked into an executable that can be run as a GUI application or as a command line batch simulation. Vehicle movement is simulated by a separate instance of the aforementioned SUMO road traffic simulator, which is instantiated, then controlled by the running simulation. In order to improve efficiency, Veins makes use of *object subscriptions* integrated [46] in SUMO, which allow it to request push notifications and updates from a running simulation, e.g., when vehicles are created or their state changes. The combination of precise channel and access models, behavior, and mobility feedback enables capturing a wide range of aspects necessary for, e.g., investigating intersection collision avoidance approaches [24].

More information, full source code, a beginner's tutorial, and related publications are available from the Veins website.[2]

[2] http://veins.car2x.org/

6.2 *iTETRIS*

The architecture of iTETRIS [38], illustrated in Fig. 8, is centered around the iTETRIS Control System (iCS) interface. Although being simulator agnostic, it federates the traffic simulator SUMO, the network simulator ns-3, and one or multiple instances of an ITS application simulator.

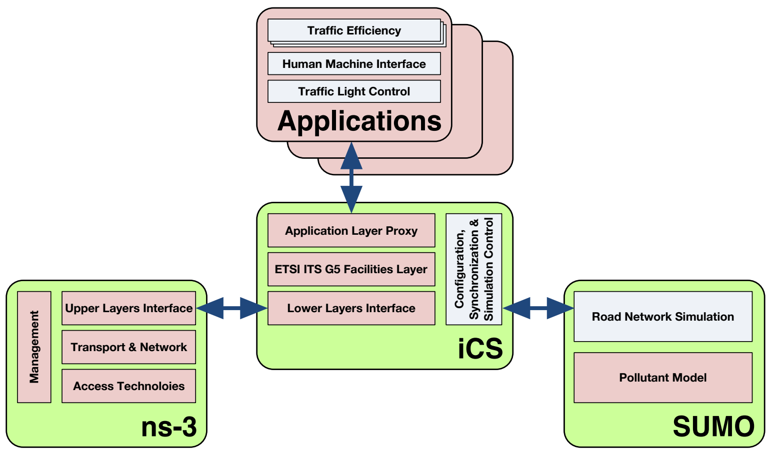
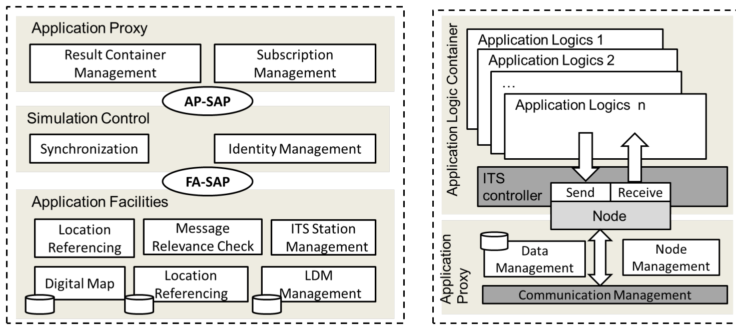


Fig. 8 The iTETRIS simulation platform.

The role of the iCS is yet more important, as it also integrates the IVC application aspects of the ETSI ITS Facilities (see Fig.9(a)), such as the *ITS station management* or *local dynamic maps*. It also contains a *result container* used to store and exchange generic data between multiple instances of the IVC application module. The most important module of the iCS is yet the *subscription* module, which implement a set of open APIs controlling the exchange of information between ns-3, SUMO and the IVC applications. This particularity allows the simulation of the dynamic interactions between mobility and networking on IVC applications.

A very unique aspect of iTETRIS is the IVC application simulator (see Fig.9(b)). It allows the evaluation of ITS applications with minimal efforts by embedding the application logics outside of the main other simulators. The architecture of the IVC simulator is separated in two layers. The first layer handles all connection primitives between the IVC application logics, the iCS, SUMO and ns3. A *Payload Storage* block is also available to keep IVC data local to the IVC application simulator. This is typically used to increase scalability, as the packet-level granularity of ns-3 makes that it does not need real data in the payload of the simulated packets. The second layer is a container for the IVC application logics. This layer has been extended by Bellavista *et al.* [4] with an innovative flexible higher-layer *Node* architecture that can support basic *send* and *receive* primitives very similarly to ns-3, as well as an *ITS controller* playing the role of an interface and a coordinator of IVC applications logics. This architecture allows an easy integration and interaction between the IVC application logics, the application simulator, and the other iTETRIS modules, both from modeling and simulation runtime perspective.



(a) iCS (b) IVC application ([4])

Fig. 9 Illustration of the architecture of the iTETRIS iCS and IVC applications

Finally, as IVC application logics running on multiple instances of the IVC application simulator need to exchange information, a specific open API allows for their interactions over the iCS. This is typically critical for parallel development of complex IVC applications, where for instance one IVC applications is in charge of monitoring traffic, while another one provides personalized navigation services.

More details related to the iCS and the IVC application simulator may be found in[18], and we refer interested readers to the iTETRIS community website[3] for more details on the iTETRIS platform.

6.3 VSimRTI

VSimRTI (Vehicle-to-X (V2X) Simulation Runtime Infrastructure) [42] goes one step further in decoupling individual components. It is a generalized framework coupling different simulators, each for a particular domain, following an ambassador concept inspired by some fundamental concepts of the IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) [22]. All management tasks, such as synchronization, interaction, and life-cycle management are handled completely by VSimRTI (see lower part of Fig. 10). Several optimization techniques, such as optimistic synchronization, target high performance simulations [35].

By implementing the generic VSimRTI interfaces (see upper part of Fig. 10), an easy integration and exchange of simulators is possible. Consequently, the deployment of simulators is enabled for each particular domain – allowing a realistic

[3] http://www.ict-itetris.eu/

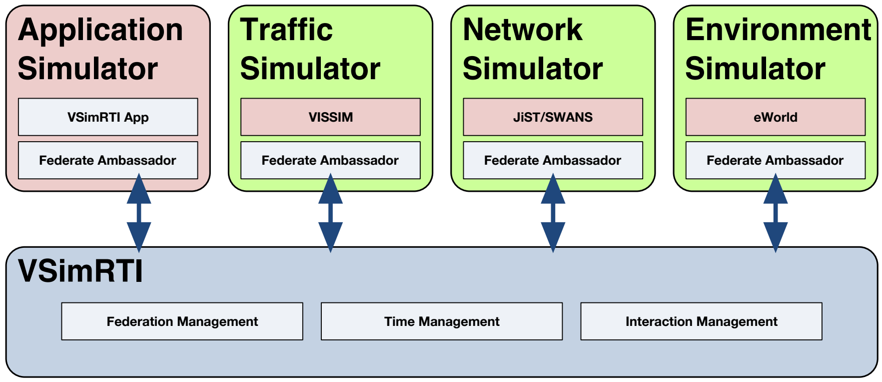


Fig. 10 VSimRTI sample simulator coupling.

presentation of vehicular traffic, emissions, wireless communication (cellular and ad-hoc), user behavior, and the modeling of mobility applications [42].

For immediate use, a set of simulators is already coupled with VSimRTI: the traffic simulators VISSIM and SUMO; the communication simulators ns-3, OMNeT++, JiST/SWANS, and a cellular communication simulator; a Java-based application simulator; and several visualization and analysis tools. Furthermore, VSimRTI is currently being extended to allow the simulation of electric mobility scenarios.

The application simulator of VSimRTI is optimized for the simulation of V2X applications. Applications run in a sandbox which offers vehicle-like interfaces, e.g., for requesting sensor data or interacting with communication modules. Data provided by traffic, communication network, and further connected simulators is transformed into a format used by components of real vehicles. To run an application, its logic is implemented in Java. Additionally, the VSimRTI application simulator supports various settings to specify the characteristics of an application and to configure its behavior, e.g., the CAM sending rate or the conditions for broadcasting a DENM can be defined [36].

Further information can be found on the VSimRTI website[4].

7 Discussion

We provided in this chapter a description of the specific simulation modeling requirements from the three different classes of IVC applications: *safe*, *smart* and *connected* mobility. Considering propagation, channel access, networking and mobility, we classified the various available levels of granularity available and classified them as function of their applicability to each class of IVC applications. One challenge is to identify the right trade-off between scalability, granularity and appli-

[4] http://www.dcaiti.tu-berlin.de/research/simulation/

cability. To this objective, we provided insight of the models critically required or not necessary to conduct simulations of IVC according to the right trade-off.

Another challenge is to use simulators that contain the required models and support the necessary level of granularity for each specific class of IVC applications. We summarized the current state of the art in simulation models and techniques and presented available integrated IVC simulation toolkits. The primary lesson learned is that the quality and comparability of simulation studies in the ITS world clearly improves over time, given that most of the described simulators include all major models at various levels of granularity. This and the tight integration of simulation components to create a unified bi-directionally coupled simulation of aspects has become a cornerstone of modern simulative performance studies. The consolidation efforts led to the emergence of three integrated and well accepted toolkits: Veins, iTETRIS, and VSimRTI. From our investigations, there is no clear winner among the three, all cover the aforementioned required aspects; the choice mainly depends on the protocols or applications under study.

In general, the credibility of IVC simulation studies, and most importantly, the reproducibility can substantially be increased by using just one of these toolkits. The main reason is that validated and openly available models that are unique in IVC can be used. This includes specific propagation, channel access, networking and realistic mobility models, which may be tightly and dynamically coupled and integrated with IVC applications.

Acknowledgement

EURECOM acknowledges the support of its industrial members: SFR, Orange, ST Microelectronics, BMW Group, SAP, Monaco Telecom, Symantec, IABG.

References

[1] Ahmed S, Karmakar GC, Kamruzzaman J (2010) An Environment-Aware Mobility Model for Wireless Ad Hoc Network. Elsevier Computer Networks 54(9):1470–1489

[2] Baiocchi A, Cuomo F (2013) Infotainment services based on push-mode dissemination in an integrated VANET and 3G architecture. Communications and Networks, Journal of 15(2):179–190, DOI 10.1109/JCN.2013.000031

[3] Barr R, Haas ZJ, van Renesse R (2005) JiST: an efficient approach to simulation using virtual machines. Software: Practice and Experience 35(6):539–576, DOI 10.1002/spe.647

[4] Bellavista P, Caselli F, Foschini L (2014) Implementing and evaluating v2x protocols over itetris: Traffic estimation in the colombo project. In: Proceedings of the Fourth ACM International Symposium on Development and Anal-

ysis of Intelligent Vehicular Networks and Applications, DIVANet '14, pp 25–32

[5] Bieker L, Krajzewicz D, Rckl M, Cappelle H (2010) Derivation of a fast, approximating 802.11p simulation model. In: 10th International Conference on Intelligent Transport Systems Telecommunications(ITST 2010), Kyoto, Japan

[6] Boban M, Vinhosa T, Barros J, Ferreira M, Tonguz OK (2011) Impact of Vehicles as Obstacles in vehicular Networks. IEEE Journal on Selected Areas in Communications (JSAC) 29(1):15–28

[7] Camp T, Boleng J, Davies V (2002) A Survey of Mobility Models for Ad Hoc Network Research. Wireless Communications and Mobile Computing, Special Issue on Mobile Ad Hoc Networking: Research, Trends and Applications 2(5):483–502

[8] Chen Q, Jiang D, Taliwal V, Delgrossi L (2006) IEEE 802.11 based vehicular communication simulation design for NS-2. In: 3rd ACM International Workshop on Vehicular Ad Hoc Networks (VANET 2006), ACM, Los Angeles, CA, pp 50–56, DOI 10.1145/1161064.1161073

[9] Cheng W, Cheng X, Znati T, Lu X, Lu Z (2009) The Complexity of Channel Scheduling in Multi-Radio Multi-Channel Wireless Networks. In: 28th IEEE Conference on Computer Communications (INFOCOM 2009), Rio de Janeiro, Brazil, pp 1512 –1520, DOI 10.1109/INFCOM.2009.5062068

[10] Chrysanthou C, Bertoni HL (1990) Variability of sector averaged signals for UHF propagation in cities. IEEE Transactions on Vehicular Technology 39(4):352–358, DOI 10.1109/25.61356

[11] Dressler F, Sommer C (2010) On the Impact of Human Driver Behavior on Intelligent Transportation Systems. In: 71st IEEE Vehicular Technology Conference (VTC2010-Spring), IEEE, Taipei, Taiwan, pp 1–5, DOI 10.1109/VETECS.2010.5493964

[12] Dressler F, Sommer C, Eckhoff D, Tonguz OK (2011) Towards Realistic Simulation of Inter-Vehicle Communication: Models, Techniques and Pitfalls. IEEE Vehicular Technology Magazine 6(3):43–51, DOI 10.1109/MVT.2011.941898

[13] van Eenennaam M, Wolterink W, Karagiannis G, Heijenk G (2009) Exploring the solution space of beaconing in VANETs. In: 1st IEEE Vehicular Networking Conference (VNC 2009), IEEE, Tokyo, Japan, DOI 10.1109/VNC.2009.5416370

[14] European Telecommunications Standards Institute (2009) Intelligent Transport Systems (ITS); European profile standard for the physical and medium access control layer of Intelligent Transport Systems operating in the 5 GHz frequency band. ES 202 663 V1.1.0, ETSI

[15] Fiore M, Härri J (2008) The networking shape of vehicular mobility. In: 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (Mobihoc 2008), ACM, Hong Kong, China, pp 261–272

[16] Font J, Iñigo P, Domínguez M, Sevillano JL, Amaya C (2010) Architecture, design and source code comparison of ns-2 and ns-3 network simulators. In: 2010 Spring Simulation Multiconference (SpringSim 2010), SCS, Orlando, FL

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

[18] Härri J, Cataldi P, Krajzewicz D, Blokpoel RJ, Lopez Y, Leguay J (2011) Modeling and simulating its applications with itetris. In: MSWiM'11, 14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems

[19] Hartenstein H, Laberteaux KP (2008) A Tutorial Survey on Vehicular Ad Hoc Networks. IEEE Communications Magazine 46(6):164–171

[20] Institute of Electrical and Electronics Engineers (2010) Wireless Access in Vehicular Environments. Std 802.11p-2010, IEEE

[21] Institute of Electrical and Electronics Engineers (2011) IEEE Trial-Use Standard for Wireless Access in Vehicular Environments (WAVE) - Multi-channel Operation. Std 1609.4, IEEE

[22] Institute of Electrical and Electronics Engineers and IEEE-SA Standards Board (2000) IEEE standard for modeling and simulation (M&S) High Level Architecture (HLA): Framework and Rules. IEEE Standard 1516. IEEE standard, Institute of Electrical and Electronics Engineers

[23] Joerer S, Dressler F, Sommer C (2012) Comparing Apples and Oranges? Trends in IVC Simulations. In: 9th ACM International Workshop on Vehicular Internetworking (VANET 2012), ACM, Low Wood Bay, Lake District, UK, pp 27–32, DOI 10.1145/2307888.2307895

[24] Joerer S, Segata M, Bloessl B, Lo Cigno R, Sommer C, Dressler F (2014) A Vehicular Networking Perspective on Estimating Vehicle Collision Probability at Intersections. IEEE Transactions on Vehicular Technology 63(4):1802–1812, DOI 10.1109/TVT.2013.2287343

[25] Karagiannis G, Altintas O, Ekici E, Heijenk G, Jarupan B, Lin K, Weil T (2011) Vehicular Networking: A Survey and Tutorial on Requirements, Architectures, Challenges, Standards and Solutions. IEEE Communications Surveys and Tutorials 13(4):584–616, DOI 10.1109/SURV.2011.061411.00019

[26] Katsaros K, Kernchen R, Dianati M, Rieck D, Zinoviou C (2011) Application of Vehicular Communications for Improving the Efficiency of Traffic in Urban Areas. Wireless Communications and Mobile Computing 11(12):1657–1667, DOI 10.1002/wcm.1233

[27] Killat M, Hartenstein H (2009) An empirical model for probability of packet reception in vehicular ad hoc networks. EURASIP Journal on Wireless Communications and Networking 2009(1):721,301

[28] Klingler F, Dressler F, Cao J, Sommer C (2013) Use Both Lanes: Multi-Channel Beaconing for Message Dissemination in Vehicular Networks. In: 10th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2013), IEEE, Banff, Canada, pp 162–169

[29] Köpke A, Swigulski M, Wessel K, Willkomm D, Haneveld PTK, Parker TEV, Visser OW, Lichte HS, Valentin S (2008) Simulating Wireless and Mobile Networks in OMNeT++ – The MiXiM Vision. In: 1st ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Net-

works and Systems (SIMUTools 2008): 1st ACM/ICST International Workshop on OMNeT++ (OMNeT++ 2008), ACM, Marseille, France
[30] Krajzewicz D, Hertkorn G, Rössel C, Wagner P (2002) SUMO (Simulation of Urban MObility); An open-source traffic simulation. In: 4th Middle East Symposium on Simulation and Modelling (MESM 2002), Sharjah, United Arab Emirates, pp 183–187
[31] Lee KC, Lee U, Gerla M (2009) TO-GO: TOpology-assist Geo-Opportunistic Routing in Urban Vehicular Grids. In: 6th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2009), IEEE, Snowbird, UT, pp 11–18
[32] Lownes NE, Machemehl RB (2006) VISSIM: a multi-parameter sensitivity analysis. In: 38th Winter Simulation Conference (WSC '06), IEEE, Monterey, CA, pp 1406–1413, DOI 10.1109/WSC.2006.323241
[33] Mittag J, Papanastasiou S, Hartenstein H, Strom E (2011) Enabling accurate cross-layer phy/mac/net simulation studies of vehicular communication networks. Proceedings of the IEEE 99(7):1311–1326
[34] Nagel R, Eichler S (2008) Efficient and Realistic Mobility and Channel Modeling for VANET Scenarios Using OMNeT++ and INET-Framework. In: 1st ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2008), ICST, Marseille, France, pp 1–8
[35] Naumann N, Schünemann B, Radusch I, Meinel C (2009) Improving v2x simulation performance with optimistic synchronization. In: Services Computing Conference, 2009. APSCC 2009. IEEE Asia-Pacific, pp 52–57, DOI 10.1109/APSCC.2009.5394142
[36] Protzmann R, Schünemann B, Radusch I (2014) A sensitive metric for the assessment of vehicular communication applications. In: Advanced Information Networking and Applications (AINA), 2014 IEEE 28th International Conference on, pp 697–703, DOI 10.1109/AINA.2014.86
[37] Rappaport TS (2009) Wireless Communications: Principles and Practice, 2nd edn. Prentice Hall
[38] Rondinone M, Maneros J, Krajzewicz D, Bauza R, Cataldi P, Hrizi F, Gozalvez J, Kumar V, Röckl M, Lin L, Lazaro O, Leguay J, Härri J, Vaz S, Lopez Y, Sepulcre M, Wetterwald M, Blokpoel R, 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, DOI 10.1016/j.simpat.2013.01.007
[39] Saha AK, Johnson DB (2004) Modeling mobility for vehicular ad-hoc networks. In: 1st ACM Workshop on Vehicular Ad Hoc Networks (VANET 2004), Philadelphia, PA, pp 91–92
[40] Saunders SR, Aragón-Zavala A (2007) Antennas and Propagation for Wireless Communication Systems, 2nd edn. Wiley
[41] Schoch E, Feiri M, Kargl F, Weber M (2008) Simulation of ad hoc networks: ns-2 compared to JiST/SWANS. In: 1st ACM/ICST International Conference

on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2008), ICST, Marseille, France
[42] Schünemann B (2011) V2X simulation runtime infrastructure VSimRTI: An assessment tool to design smart traffic management systems. Elsevier Computer Networks 55(14):3189–3198, DOI 10.1016/j.comnet.2011.05.005
[43] Sommer C, Dressler F (2008) Progressing Toward Realistic Mobility Models in VANET Simulations. IEEE Communications Magazine 46(11):132–137, DOI 10.1109/MCOM.2008.4689256
[44] Sommer C, Dressler F (2011) Using the Right Two-Ray Model? A Measurement based Evaluation of PHY Models in VANETs. In: 17th ACM International Conference on Mobile Computing and Networking (MobiCom 2011), Poster Session, ACM, Las Vegas, NV
[45] Sommer C, Eckhoff D, German R, Dressler F (2011) A Computationally Inexpensive Empirical Model of IEEE 802.11p Radio Shadowing in Urban Environments. In: 8th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (WONS 2011), IEEE, Bardonecchia, Italy, pp 84–90, DOI 10.1109/WONS.2011.5720204
[46] Sommer C, German R, Dressler F (2011) Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. IEEE Transactions on Mobile Computing 10(1):3–15, DOI 10.1109/TMC.2010.133
[47] Stepanov I, Rothermel K (2008) On the Impact of a More Realistic Physical Layer on MANET Simulations Results. Elsevier Ad Hoc Networks 6(1):61–78
[48] Torrent-Moreno M, Schmidt-Eisenlohr F, Füßler H, Hartenstein H (2006) Effects of a realistic channel model on packet forwarding in vehicular ad hoc networks. In: IEEE Wireless Communications and Networking Conference (WCNC 2006), IEEE, Las Vegas, NV, pp 385–391, DOI 10.1109/WCNC.2006.1683495
[49] Torrent-Moreno M, Mittag J, Santi P, Hartenstein H (2009) Vehicle-to-Vehicle Communication: Fair Transmit Power Control for Safety-Critical Information. IEEE Transactions on Vehicular Technology 58(7):3684–3703, DOI 10.1109/TVT.2009.2024873
[50] Varga A, Hornig R (2008) An overview of the OMNeT++ simulation environment. In: 1st ACM/ICST International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMUTools 2008), ACM, Marseille, France
[51] Werner M, Lupoaie R, Subramanian S, Jose J (2012) MAC Layer Performance of ITS G5 - Optimized DCC and Advanced Transmitter Coordination. In: 4th ETSI TC ITS Workshop, Doha, Qatar
[52] Willke TL, Tientrakool P, Maxemchuk NF (2009) A Survey of Inter-Vehicle Communication Protocols and Their Applications. IEEE Communications Surveys and Tutorials 11(2):3–20