Mobility Models for Vehicular Ad Hoc Networks: A Survey and Taxonomy

Jérôme Härri, Fethi Filali, and Christian Bonnet

***Abstract*—Vehicular Ad-hoc Networks (VANETs) have been recently attracting an increasing attention from both research and industry communities. One of the challenges posed by the study of VANETs is the definition of a vehicular mobility model providing an accurate and realistic vehicular mobility description at both macroscopic and microscopic levels. Another challenge is to be able to dynamically alter this vehicular mobility as a consequence of the vehicular communication protocols. Many mobility models have been developed by the community in order to solve these two issues. However, due to the large number of available models claiming to be adapted to vehicular traffic, and also due to their different and somehow incomparable features, understanding their true characteristics, their degree of realism with respect to vehicular mobility, and real capabilities is a hard task.**

In this survey, we first introduce a framework that proposes a guideline for the generation of vehicular mobility models. Then, we illustrate the different approaches chosen by the community for the development of vehicular mobility models and their interactions with network simulators. Finally, we propose an overview and taxonomy of a large range of mobility models available for vehicular ad hoc networks. The objective is to provide readers with a guideline to easily understand and objectively compare the different models, and eventually identify the one required for their needs.

***Index Terms*—Vehicular Mobility Models, Survey, Taxonomy, Classification, Architecture, Traffic Simulator, Vehicle Ad Hoc Networks, Inter-Vehicle Communications (IVC), Car-2-X Communication, Traffic Telematics.**

I. INTRODUCTION

VEHICULAR communication is seen as a key technology for improving road safety and comfort through Intelligent Transportation Systems (ITS). The growing interest toward the possible applications of wireless technologies to a vehicular environment has recently led consortia (US VII [1], EU C2C-CC [2]) and standardization bodies (IEEE [3]) to develop technologies and protocols for data transmission between vehicles and between vehicles and road infrastructures.

Manuscript received 25 February 2007; revised 9 August 2008. Part of this work has been done while J. Härri was supported by EURECOM. Another part has been supported by the German Ministry of Education and Research (BMB-F) for the "Network-on-Wheels" project under contract no. 01AK064F, and by the Ministry of Science, Research and the Arts of Baden-Württemberg (Az: Zu 33-827.377/19,20), the Klaus Tschira Stiftung, the INIT GmbH and the PTV AG for the research group on Traffic Telematics. EURECOM's research is partially supported by its industrial members: BMW Group Research & Technology - BMW Group Company, Cisco Systems, Monaco Telecom, Orange, SAP, SFR, Sharp, STEricsson, Swisscom, Thales, Symantec.

Jérôme Härri is with Institute of Telematics, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe, Germany, (e-mail: jerome.haerri@kit.edu).

Fethi Filali and Christian Bonnet are with EURECOM Département Communications Mobiles, 06904 Sophia-Antipolis, France, (e-mail: {filali,bonnet}@eurecom.fr).

Digital Object Identifier 10.1109/SURV.2009.090403.

Vehicular Ad-hoc Networks (VANETs) represent a rapidly emerging, particularly challenging class of Mobile Ad Hoc Networks (MANETs). VANETs are distributed, self-organizing communication networks built up from moving vehicles, and are thus characterized by very high speeds and limited degrees of freedom in nodes movement patterns. Such particular features often make standard networking protocols inefficient or unusable in VANETs. The current major objective of protocols developed for VANET is to benefit from the messages exchanged between cars to alter traffic, either for safety issues with advanced safety messages, or for traffic management in order to avoid traffic jams. In both cases, a strong interaction has been defined between the network protocol and vehicular mobility. In the standard approach, data traffic is altered by mobility. Yet in VANETs, mobility may also be influenced by data traffic. There is therefore a specific bidirectional relationship between mobility and wireless communication in VANETs. When considering the huge impact that the deployment of VANET technologies could have on the automotive market, the growing effort in the development of communication protocols and mobility models specific to vehicular networks is easily understandable.

Whereas it is crucial to test and evaluate protocol implementations in real testbed environments, logistic difficulties, economic issues and technology limitations make simulations the mean of choice in the validation of networking protocols for VANETs, and a widely adopted first step in development of real world technologies. A critical aspect in a simulation study of VANETs is the need for a mobility model reflecting the real behavior of vehicular traffic. Moreover, mobility models are required to be dynamically reconfigurable in order to reflect the effects of a particular communication protocol on vehicular traffic. The community therefore started to work on the development, or the revamping, of mobility models specific to vehicular motions. After a few years of exciting developments, a large variety of models are available, varying from the most trivial to the most realistic ones, or from freely available models to commercial-based traffic simulators. Unfortunately, this development has been uncoordinated, as each research group basically developed one model fulfilling its own specific needs. Beside the large variety of available models, it is also hard for someone starting in this area to easily understand each model's characteristics, compare its assets and drawbacks, and choose the best available solution for its needs.

In this original survey, we provide a detailed survey and comparison of mobility models available for vehicular ad hoc networks. We first introduce a framework that a mobility model should follow in order to realistically describe vehicular

1553-877X/09/$25.00 © 2009 IEEE

motion patterns. We then expose the different approaches followed by the community for the development of mobility models. We further illustrate the methodology for the collaboration between mobility models and network simulators such that a network protocol influences vehicular behaviors and conversely, mobility patterns alter the efficiency of a network protocol. Next, we detail the different criteria that we will use to classify the different models. These criteria have been defined as building blocks based on the framework and enhanced by the different categories of mobility models. Finally, we briefly describe each model, highlighting its assets and drawbacks, and provide an intuitive set of classification tables that allows readers to easily understand each model's characteristics. Our objective is to help readers to better compare the different solutions available in the literature, and find the one tailored to their needs. We eventually discuss the outcome of three years of developments, and suggest promising new research directions remaining in this field. We wrote this survey to offer to a large set of potential readers some understanding on the current status and the future orientations that could be taken in vehicular mobility modeling in the near future. It is therefore targeted to readers curious to discover this field of research, but also to experts in this area.

The rest of this paper is organized as follows. Section II introduces the framework describing the major required components for a mobility model, while Section III provides a description of the process of generating vehicular mobility models. We then cover the relationship between network and traffic simulators in Section IV, we detail the different criteria used to classify the various models in Section V, and finally propose a detailed survey and a taxonomy of mobility models available to the vehicular networking community in Section VI. Eventually, Section VII discusses the current state and future directions in the area of vehicular mobility modeling, and Section VIII concludes this survey.

II. A Framework for Realistic Vehicular Mobility Models

In literature, vehicular mobility models are usually classified as either macroscopic or microscopic [63]. The *macroscopic* description models gross quantities of interest, such as vehicular density or mean velocity, treating vehicular traffic according to fluid dynamics, while the *microscopic* description considers each vehicle as a distinct entity, modeling its behavior in a more precise, but computationally more expensive way.

Yet, a micro-macro approach may be seen more as a broad classification schema than a formal description of the models' functionalities in each class. A more precise way that we suggest for looking at mobility models consists in identifying functional blocks: motion constraints, traffic generator, time and external influences. On the one hand, *Motion constraints* describe the relative degree of freedom of each vehicle. Macroscopically, motion constraints are streets or buildings, but microscopically, constraints are modeled by neighboring cars, pedestrians, or by modelization's diversities either due to the type of car or to the driver's habits. On the other hand, the *traffic generator* defines different kinds of cars and deals with their interactions according to the environment under study. Macroscopically, it models traffic densities, speeds and flows, while microscopically it deals with properties like the inter-distance between cars, acceleration, braking, overtaking. Another important aspect of realistic motion modeling is *time*, which can be seen as the third functional block that describes different mobility configurations for a specific time of the day or day of the week. Finally, we also have to add a fourth fundamental block, the *External Influence*, modeling the impact of a communication protocol or any other source of information on the motion patterns.

According to the concept map in Fig. 1, mobility models intended to generate realistic vehicular motion patterns should include the following building blocks:

- *Accurate and realistic topological maps:* street topologies should manage different densities of intersections, contain multiple lanes, different categories of streets and their associated speed limitations.
- *Obstacles:* obstacles should be understood in a wide sense, as both constraints to cars mobility and hurdles to wireless communications.
- *Attraction/repulsion points:* initial and final destinations of road trips are not random. Most of the time, drivers are moving to similar final destinations, called attraction points (e.g. office), or from similar initial locations, called repulsion points (e.g. home), a feature that creates bottlenecks.
- *Vehicles characteristics:* each category of vehicle has its own characteristics, which has an impact on a set of traffic parameters. For example, macroscopically speaking, some urban streets and highways are forbidden to trucks depending on the time of the day. Microscopically speaking, acceleration, deceleration and speed capabilities of a car or a trucks are different.Accounting for these characteristics alters the traffic generator engine when modeling realistic vehicular motions.
- *Trip motion:* a trip is macroscopically seen as a set of source and destination points in the urban area. Different drivers may have diverse interests, which affect its trip selection.
- *Path motion:* a path is macroscopically seen as the set of road segments taken by a car on its trip between an initial and a destination point. As it may also be observed in real life, drivers do not randomly choose the next heading when reaching an intersection, as it is currently the case in most vehicular networking traffic simulations. Instead, they choose their paths according to a set of constraints such as speed limitations, time of the day, road congestion, distance, and even drivers personal habits.
- *Smooth deceleration and acceleration:* vehicles do not abruptly break and accelerate. Models for decelerations and accelerations should consequently be considered.
- *Human driving patterns:* drivers interact with their environments, not only with respect to static obstacles, but also to dynamic obstacles, such as neighboring cars and pedestrians. Accordingly, the mobility model should control the mutual interactions between vehicles, such as overtaking, traffic jam, preferred paths.
- *Intersection Management:* It corresponds to the process of controlling an intersection, and may either be modeled

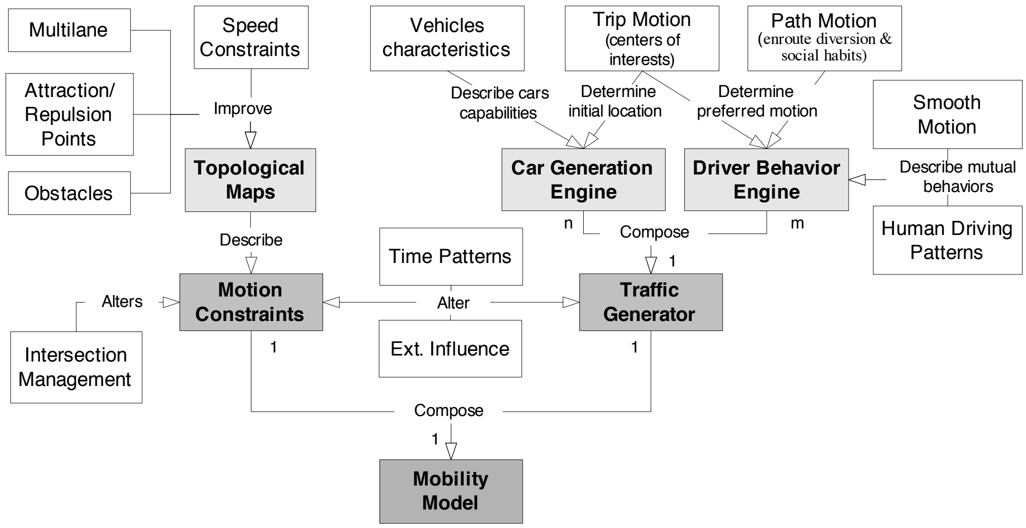


Fig. 1. Proposed concept map for the generation of realistic vehicular mobility models.

as a static obstacle (stop signs), a conditional obstacle (yield sign), or a time-dependent obstacle (traffic lights). It is a key part in this framework that however only influences the *Motion Constraint* block, as the *Traffic Generator* block cannot not see the difference between a stop sign or a high density traffic. Both are interpreted as a motion constraint.

- *Time patterns:* traffic density is not identical during the day. A heterogeneous traffic density is always observed at peak times, such as rush hours or during special events. This block influences the *Motion Constrains* and the *Traffic Generator* blocks, as it may alter the trip or path computation, and also the attraction/repulsion points.
- *External Influence:* some motion patterns cannot be proactively configured by vehicular mobility models as they are externally influenced. This category models the impact of accidents, temporary road works, or real-time knowledge of the traffic status on the motion constraints and the traffic generator blocks. Communication systems are the primary source of information about external influence.

All building blocks described here form our proposed framework that should be followed by designers of mobility models specific to vehicular motions. The more building blocks a vehicular mobility model includes, the more realistic it is. These building blocks will therefore be the base of our taxonomy of the different available approaches in vehicular mobility modeling that will be discussed in the next section.

III. Generating Mobility Models for Vehicular Networks

Although being a promising approach, the proposed guidelines illustrated in the previous section suffer from non negligible limitations. Indeed, parameters defining the different major building blocks such as *topological maps*, *car generation engine*, or *driver behavior engine* cannot be randomly chosen but must reflect realistic configurations. Therefore, due to the large complexity to obtain such kind of information, the research community took more simplistic assumptions and neglected several blocks. As we will show throughout this paper, most models available nowadays include a topological map, or at least a graph, as motion constraints. However, they do not include speed constraints or more generally attraction or repulsion points. The *car generation engine* block is also widely absent from all models, and the *driver behavior engine* is limited to smooth accelerations or decelerations.

Globally, the development of vehicular mobility models may be classified in four different classes: *Synthetic Models* wrapping all models based on mathematical models, *Survey-based Models* extracting mobility patterns from surveys, *Trace-based Models* generating mobility patterns from real mobility traces, and finally *Traffic Simulators-based Models*, where the vehicular mobility traces are extracted from a detailed traffic simulator. A proposed classification is illustrated in Fig. 2

In the rest of this section, we describe the challenges specific to each class, and provide examples of solutions proposed by the community. As some techniques to design mobility models are not specific to vehicular mobility, and as the objective is to show the different directions followed in various fields that could also be applied to vehicular models, some solutions described in this section have been designed for pedestrian mobility.

A. Synthetic Models

The first and most well known class includes synthetic models. Major studies have been undertaken in order to develop mathematical models reflecting a realistic physical effect. Fiore [4] wrote a complete survey of models falling into this category. We shortly summarize the classification he developed. For a more complete version, we refer the reader to [4]. According to Fiore's classification, Synthetic models may be separated into five classes: *stochastic models* wrapping all models containing purely random motions, *traffic stream models* looking at vehicular mobility as hydrodynamic phenomenon, *Car Following Models*, where the behavior of each driver is modeled according to vehicles ahead, *Queue*

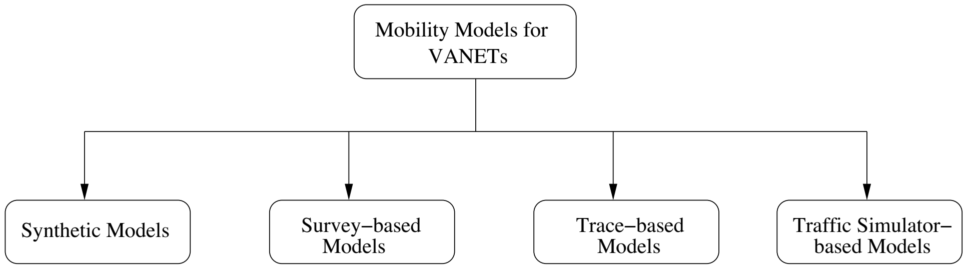


Fig. 2. Classification of Vehicular Mobility Models

Models which models roads as FIFO queues and cars as clients, and *Behavioral Models* where each movement is determined by behavioral rules such as social influences. Fig. 3 illustrates Fiore's classification.

Validating a mathematical model is an important step in order to guarantee its realism compared to real mobility. One solution is to gather mobility traces by large measurement campaigns and then compare the patterns with those developed by the synthetic model. In [5], authors proposed to tune some key characteristics of the RWP such as average speed and rest times using real life data. The Weighted Waypoint Model (WWM) [6] is a second attempt to tune the parameters of a synthetic model using real traces. The WWM adds the notion of preference to the Random Waypoint. The calibration of this "preference criterion" has been performed based on mobility traces obtained inside the USC campus.

A major limitation of most synthetic models comes from the complexity to model detailed human behaviors. Drivers are far from being machines and cannot be programmed to follow a specific behavior in all cases. They instead respond to stimuli and local perturbations that may have a global effect on traffic modeling. Accordingly, realistic mobility modeling should also consider behavioral theory. Although being related to human mobility, Musolesi *et al*. illustrated this approach in [10] by developing a synthetic mobility model based on social network theory, then validated it using real traces. They showed that the model was a good approximation of human movement patterns.

To conclude, the general approach behind synthetic mobility models is first to try to understand a particular movement, then to develop a mathematical model, and finally to try to reproduce it. Yet, some movements, or more specifically their mutual interactions, often make a mathematical model either too complex or impossible to establish at all. The following approaches therefore aim at recreating an approximation of the movements based on observations or surveys.

B. Survey-based Models

Surveys are an important source of macroscopic mobility information. The major large scale surveys are provided by the US department of Labor, which gathered extensive statistics of US workers' behaviors, spanning from the commuting time or lunch time, to traveling distance or preferred lunch types. By including such kind of statistics into a mobility model, one is able to develop a generic mobility model able to reproduce the pseudo-random or deterministic behavior observed in the real urban traffic.

The UDel Mobility Model [11] typically falls into this category. The mobility simulator is based on surveys from a number of research areas including time-use studies performed by the US Department of Labor and Statistics, time-use studies by the business research communities, or pedestrians and vehicle mobility studies by the urban planning and traffic engineering communities. Based on this data, the mobility simulator models arrival times at work, lunch time, breaks/errands, pedestrian and vehicular dynamics (e.g. realistic speed-distance relationship and passing dynamics), and workday time-use such as meeting size, frequency, and duration. Vehicle traffic is derived from vehicle traffic statistics collected by state and local governments such that it is able to model vehicle dynamics and diurnal street usage. We can also cite the Agenda-based [12] mobility model, which combines both social activities and geographic movements. The movement of each node is based on an individual agenda, which includes all kinds of activities in a specific day. Data from the US National Household Travel Survey has been used to obtain activity distributions, occupation distributions and dwell time distributions.

A complex and computationally demanding vehicular mobility model is proposed by the ETH [14] and generates public and private vehicular traffic over real regional road maps of Switzerland with a high level of realism within a period of 24 hours. The model is calibrated using data from census and other local or national mobility surveys or statistics. A similar approach has been followed at the Los Alamos Research Labs, USA, but using more precise statistics from various urban traffic management systems such as sensors at traffic lights and measured traffic flows. These last two examples are the illustration of the limitation of the survey-based approach, as survey or statistical data are only able to provide a coarse grain mobility as illustrated in the previous paragraph. If we need a more detailed and realistic mobility representation, then we still require a complex synthetic model and calibrate it using surveys or statistics.

C. Trace-based Models

Due to the complexity of modeling vehicular mobility, only few very complex synthetic models are able to come close to a realistic modeling of motion patterns. A different approach could also followed. Instead of developing complex models and then calibrating them using mobility traces or surveys, a crucial time could be saved by directly extracting generic mobility patterns from movement traces. Such approach recently became increasingly popular as mobility traces started

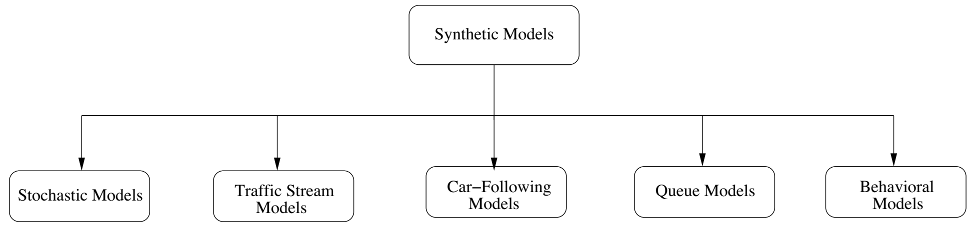


Fig. 3. Classification of Synthetic Mobility Models (Fiore [4])

to be gathered through the various measurement campaigns launched by projects such as CrawDaD [13], UMASSDieselNet [15], MIT Reality Mining [17], USC MobiLib [18], or Cabspotting [16]. The most difficult part in this approach is to extrapolate patterns not observed directly by traces. By using complex mathematical models, it is possible to predict mobility patterns not reported in the traces to some extent. The limitation is also often linked to the class of the measurement campaign. For instance, if motion traces have been gathered for bus systems, an extrapolated model cannot be applied to the traffic of personal vehicles.

Another limitation for the creation of trace-based vehicular mobility models is the limited availability of vehicular traces. Some research groups are currently implementing testbeds, but the outcome might not be available soon, if they are even made available to the public.

We list here the different existing solutions to extract mobility models from existing mobility trace. To the best of our knowledge, only three of them are related to vehicular mobility. The first one, developed in conjunction with the German Fleetnet and the Network on Wheels projects [7], [8], is based on traces measured by Daimler AG on a highway section and visualized by the HWGui [9]. The second one, created by the University of Massachusetts and named DieselNet [15], provides mobility traces of a bus system in the city of Amherst, MA USA. The Cabspotting project [16] is the third source of vehicular mobility traces. The project equipped all taxi vehicles in the San Francisco Bay Area and provides a live visualization of a complete taxi system. Unfortunately, all other approaches have only been proposed for model human mobility. As porting the same kind of techniques to model vehicular motion is still an open research field, we decided to cite them to provide a prospective reader with some insights on the possible directions to be followed for the generation of trace-based vehicular mobility models.

In [19], Tuduce *et al.* presented a mobility model based on real data from the campus wireless LAN at ETH Zurich. They used a simulation area divided into squares and derived the probability of transitions between adjacent squares from access points' data. In [20], Yoon *et al.* combined coarse-grained wireless traces, i.e., association data between WiFi users and access points, with a map of the area over which the traces were collected in order to generate a probabilistic mobility model representative of real movements. They derived a discrete time Markov Chain which does not only consider the current location, but also the previous locations and also the origin and the destination of the path. Unfortunately, the study does not consider correlations between nodes.

Kim *et al.* [21] described another measurement technique for extracting users' mobility characteristics from coarse-grained wireless traces. They derived locations of users over time in order to emphasize popular regions. Their major findings was that, unlike standard synthetic mobility models, the speed and the pause times follow a log-normal distribution. They also confirmed that the direction of movement closely reflect the direction of roads and walkways, and thus cannot be modeled by a uniform distribution. Similarly to [19], they ignored correlation between adjacent nodes.

Hou *et al.* [22] modeled users' mobility by a semi-Markov process with a Markov Renewal Process associated to access points' connection time instants. Unlike previous studies, this work is able to model how users' mobility is correlated in time at different time scales. The authors also performed a transient analysis of the semi-Markov process and extracted a timed location prediction algorithm, which is able to accurately predict users future locations. This work is moreover the first attempt to characterize the correlation between movements of individual users.

Chaintreau *et al.* [23] studied the inter-contact times between wireless devices carried by human beings using coarse-grained wireless traces and also experimental testbeds using iMotes. Their major breakthrough was that, unlike the widely accepted assumption that the inter-contact times follow an exponential distribution, a more realistic assumption should be that the distribution exhibits a heavy tail similar to a power law distribution. Another study from Srinivasan *et al.*([24]) analyzed the student contact patterns in a university campus using class time-tables and class attendance data. A major restricted assumption has been taken, which forces students to either be in classrooms or in some randomly chosen communication hubs. The authors showed that, in this configuration, most students experienced inter-contact times of the order of magnitude of few hours. However, unlike other studies (such as [23]), the inter-contact times do not follow a power law distribution. This is where the limitation from trace-base mobility modeling appears. Indeed, this study is specific to class attendance, and results remain also specific to the environment where the study has been made.

By using traces, various research teams have therefore been able to extract mobility models that would reflect more realistically the pedestrian motion patterns we experience in real life. A major result from trace-based mobility modeling, which is at odd with hypothesis used by synthetic models, is that the speed and pause time distributions follow a log-normal and not a uniform distribution, and the inter-contact time should be modeled by a power law and not by an exponential distribution. Synthetic models should accordingly be extended to show similar characteristics in order to be realistic.

Following a similar direction, if we can obtain an insight of the real distributions of speed, pause times, or inter-contact times in vehicular motion by using real traces, we could therefore accordingly configure the synthetic models.

D. Traffic Simulator-based Models

By refining synthetic models and going through an intense validation process based on real traces or behavior surveys, some companies or research teams gave birth to realistic traffic simulators. Developed for urban traffic engineering, fine grain simulators such as PARAMICS [25], CORSIM [26], VISSIM [27], TRANSIMS [28] or SUMO [35], are able to model urban microscopic traffic, energy consumption, or even pollution or noise level monitoring. However, these simulators cannot be used straightaway for network simulators, as no interface have been developed and traces are mutually incompatible[1]. In addition, these traffic simulators are commercial products and might require the purchase of a license. These issues were sufficient to justify the development of the novel off-the-shelf vehicular mobility models that we are going to describe in this paper.

By developing parser between traffic and network simulators input files, the end-user could however gain access to validated traffic patterns and would therefore be able to obtain a level of detail that is not reached by any actual vehicular mobility model. The major drawback of this approach is the configuration complexity of these traffic simulators, as the calibration usually requires to tweak a large set of parameters. But more important, the level of details required for vehicular network simulators may not be as demanding as that for traffic analysis, as global vehicular mobility patterns and not the exact vehicular behaviors are by far sufficient in most cases. Finally, the purchase of a commercial license for the use of a commercial traffic simulator may even be waived, as some university programs offering a free of charge use of some commercial traffic simulators (VISSIM for instance) may be found.

E. Validation

The discussion on the generation of mobility models would not have been complete without addressing the issue of their validation. In the community, a misunderstanding exists about the word *realistic*. In many approaches, modelers assume that their synthetic mobility model is realistic because it uses behavioral models that are close to reality. For example, a model considering a microscopic interaction between cars is usually assumed to be *more realistic* than the Random Waypoint Model. It can however not directly be assumed *realistic*, as the only method to assess the realism is by comparing the motion patterns with real topologies. This method is called *Validation*.

Validation may be performed in two ways. The first method is to compute the error between the traces generated by the synthetic model and real mobility traces. If the access to real traces is not possible, a two step approach called *delegated validation* is also possible, for which traces from a validated model are used instead of real ones.

All commercial traffic simulators (CORSIM, VISSIM, etc..) and also some free mobility models (SUMO, SHIFT) have been validated based on real traces. Consequently, any model presented in this paper that would be based on these solutions could therefore also be assumed to be *valid*.

Delegated validation has been notably followed by the authors of VanetMobiSim, as they could not gain access to real vehicular mobility traces. By comparing the motion patterns between CORSIM and VanetMobiSim in a similar environment, the authors prove that their simulator was able to reproduce the same patterns, and thus could be considered as valid. As the access to real vehicular traces is time and resource demanding, this two-step validation approach could be followed by other synthetic vehicular mobility models.

Generally, the validation step should be followed and successfully completed by any mobility model before being used to evaluate protocols for vehicular networks. However, as we will illustrate in this survey, only a few did it.

F. Discussion

In this section, we illustrated the different available approaches for the development of mobility models. If the motion patterns are not too complex, one solution is to develop a mathematical model and reproduce them at a high precision level. If not, another solution is to try to approximate the movements based on observed movement patterns. This approach has the limitation of the modeling of global mobility patterns instead of precise movements. The generated model is also not able to reproduce a pattern not observed by traces or surveys. It has yet the advantage to be able to represent a particular mobility that would be too complex to model by mathematical equations.

Yet, both approaches are not mutually exclusive, as the choice between one or the other highly depends on the application requirements. For example, if the application is a vehicular safety protocol, the mobility model must represent the real motion at a high level of precision, and thus must be generated by a synthetic model. In contrast, when testing a data dissemination protocol, the gross motion patterns are sufficient and a trace or survey-based model may therefore be envisioned. As we will show in the next section, the traffic simulator approach is also showing an increasing popularity as it allows to obtain a level of precision that cannot be reached by any synthetic model currently available.

IV. Mobility Models and Network Simulators: The Mute talking to the Deaf

In the previous section, we described the different methods to develop mobility models adapted to vehicular traffic. Yet, in order to be used by the networking community, these models need to be made available to network simulators. This simple compatibility issue is unfortunately a conundrum. The worlds of *Mobility Models* and *Network Simulators* may be compared to a mute talking to a deaf. They have never been created to

[1] VISSIM may be programmed to output a particular trace format, while SUMO has recently been extended to export traces usable by network simulators.

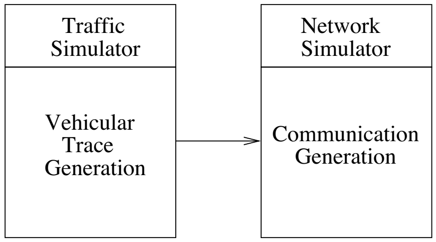


Fig. 4. Interaction between Network and Traffic Simulators: The Isolated Case

communicate, and even worse, they have been designed to be controlled separately, with almost no interaction whatsoever. When imagining the promising applications that could be obtained from vehicular networks, where communication could alter mobility, and where mobility would improve network capacity, this situation is a major setback to the development of vehicular networks.

In this section, we illustrate the need to create an interaction between a mobility model and a network simulator. We also describe the different approaches that can be followed depending on the applications' required level of interactions.

A. Isolated Mobility Models

Initially, mobility was seen by network simulators as random perturbations from optimum static configurations. Then, in order to provide a basic control to the use on the mobility patterns, network simulators became able to load mobility scenarios. However, as illustrated in Fig. 4, the different models must be generated prior to the simulation and must be parsed by the simulator according to a predefined trace format. Then, modifications of the mobility scenario are not possible, and no interaction therefore exists between these two worlds. Unfortunately, all historical models and most of the recent mobility models available to the research community fall into this category (see Section VI).

In spite of the limitation described in the previous paragraph, the networking community accommodated itself very well with isolated models, as this category has the advantage of allowing independent developments in mobility and network modeling. It is also only recently that the need for a more significant interaction appeared, as specific applications in vehicular communications, such as safety or traffic management, illustrated the requirement for a better compliance between both worlds.

B. Embedded Mobility Models

If network simulators cannot fully interact with mobility simulators, another solution is to replace both of them by simplistic off-the-shelf discrete even simulators. Accordingly, new simplistic network simulators were created, where the lack of elaborated protocol stacks was compensated by a native collaboration between the networking and the mobility worlds (see Fig. 5).

MoVes [32] is an embedded system that generates vehicular mobility traces and also contains a basic network simulator.

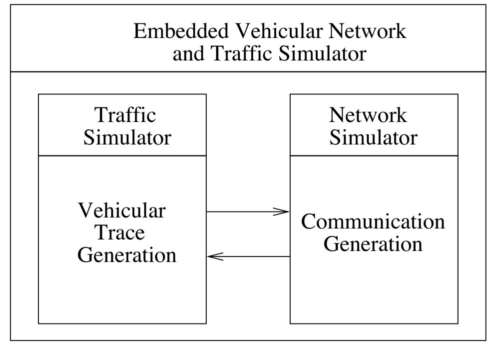


Fig. 5. Interaction between Network and Traffic Simulators: The Embedded Case

The major asset of this project is its ability to partition the geographical area into clusters, parallelizing and distributing the processing of each task. Although the mobility model reaches a sufficient level of detail, the project's drawback is the poor network simulator, which, at the time of writing this survey, only includes a basic physical and MAC layer architecture and totally lacks routing protocols. In [34], Gorgorin *et al.* also proposed an embedded vehicular and network simulator. The authors developed their own traffic and network simulator. Although being basic, the traffic model brings a sufficient level of details. However, the network simulator part is, by far, the major limitation, as it is only modeled by a simplistic discrete event simulator handling a basic radio propagation and CSMA/CA MAC protocol.

The main asset of the embedded approach is to have both models natively and efficiently interacting. However, their major limitation is actually the poor quality of the network simulator. Indeed, this approach has been so far only used to study basic network effects and could not pass the test of recent mobile ad hoc routing protocols requiring realistic and standardized physical and MAC layers. And this may also be criticized, as the actual direction in network simulations is a specific compliance with standard protocols and computational efficiency through parallel and distributed computing. Yet, this approach could be envisioned in the development of ns-3 for instance, as the new simulator and successor of ns-2 is currently at the architecture development phase. An extension to natively support a bi-directional interaction with a mobility model engine could therefore be proposed.

C. Federated Mobility Models

The last possible approach is to federate existing network simulators and mobility models through a set of interfaces (see Fig. 6). This solution has been taken, for example, by Prof. Fujimoto and his group in Georgia Tech [37]. They generated a simulation infrastructure composed of two independent commercial simulation packages running in a distributed fashion over multiple networked computers. They federated a validated traffic simulator, CORSIM, with a state-of-the-art network simulator, QualNet, using a distributed simulation software package called the Federated Simulations Development Kit (FDK) [38], which provides services to exchange data and synchronize the computation. In order to allow a direct interaction between the two simulators, a common message format has been defined between CORSIM and QualNet for vehicle status and position information. During the initialization, the road network topology is transmitted to QualNet. Once the distributed simulation begins, position updates are sent to QualNet and are mapped to mobile nodes in the wireless simulation. Accordingly, both simulators work in parallel and thus may dynamically interact on each other by altering for example mobility patterns based on network flows, and vice and versa. As mentioned by the authors, a major limitation comes however from the complex calibration of CORSIM and from its large number of configuration parameters that must be tweaked in order to fit to the modeled urban area. A similar solution has been taken by a team from UC Davis [39] with a simulation tool federating the network simulator Swans and a synthetic traffic model.

Another promising approach is called *TraNS* [41], and also aims at federating a traffic simulator SUMO and a network simulator ns-2. Using an interface called *Interpreter*[2], traces extracted from SUMO are transmitted to ns-2, and conversely, instructions from ns-2 are sent to SUMO for traffic tuning. The MSIE [42] project is an alternate approach using VISSIM instead of SUMO. This project is also more complete, as it proposes to interlink different simulators for traffic, network, and application simulations. The major actual limitations are first the communication latency between the different simulators, and second the expensive price of a commercial license for VISSIM. Unfortunately, the interlinking interface itself is also not freely available at this time. In [43], Schroth *et al.* chose to replace VISSIM by a complete tool called the *CARISMA* traffic simulator. Although not being as complete as VISSIM or SUMO, it helps to accurately evaluate the effects of car-to-car messaging systems in the presence of urban impediments by benefiting from the federated approach, and a ”real-time” trip (re-)configuration.

By interlinking independent and validated simulators, the federated approach is able to benefit from the best of both worlds, as state-of-the-art mobility models may be adapted to work with modern and efficient network simulators. However, it is computationally demanding as both simulators need to be run simultaneously, and the development of the interlinking interface may not be a straightforward task depending on the targeted network and traffic simulators. However, the networking and mobility modeling communities have a mutual interest in working together. At the time of promising benefits obtained from the various cross-layer approaches in network research, the ability to proactively or reactively act on mobility patterns in order to improve network efficiency and radio propagation, or even more promising, the ability to alter mobility patterns based on dynamic traffic events received by means of communication protocols, will probably be a central approach in future networking research projects.

[2]On the latest version of TraNS, the Interpreter has been replaced by the TraCI interface.

D. Application-centric vs. Network Centric Approach

With more interactions between a traffic and a network simulator, new applications become possible. But with new applications come also new requirements. For example, a possible traffic safety application such as a *collision avoidance system* takes control of a vehicle in order to avoid an imminent impact. For that matter, one of the application's requirements for the mobility model is first to be able to alter the car's trajectory, and second to control its influence on surrounding vehicle. Yet, the information triggering such brutal trajectory changes comes from the network simulator. Accordingly, a total interaction between a traffic and a network simulator is necessary. Another very important requirement is the latency and precision of the interaction between the two modules, as wrong or late information is not better than no information at all. For that matter, Killat *et al.* [44] studied the latency and synchronization requirements between these two modules. On the one hand, if the objective is a perfect synchronization between nodes' positions in the mobility model and in the network simulator, then a significant computational load is only used by the interface module and therefore alters the simulation performance. On the other hand, if the synchronization task is relaxed, the positions in the mobility model and the network simulator will be different, thus creating inconsistent decisions by both modules. Killat *et al.* showed that a trade-off should and could be found depending on the application needs.

These are actually just brief examples of a new paradigm called the *Application Centric* approach, where mobility is controlled and altered by an application. Opposite to this view is the traditional *Network Centric* approach, for which mobility is only required to test the resilience of a network protocol.

The isolated category is unfortunately unable to model application-centric approaches, such as traffic efficiently or traffic safety, as motion patterns cannot be altered according to dynamic traffic information. To a larger extent, the application-centric approach also needs to be carefully considered for the *embedded* category, as the level of realism of the traffic or network modules may not be sufficient to some applications. The *federated* category perfectly fits to the requirements of the application centric approach, first as a total mutual interaction is possible, and second as it regroups two independent solutions potentially allowing to choose the optimal traffic or network simulator as a function of the application requirements.

When Considering Application Centric approaches, the question of the controllability of the traffic simulator based on information received from a network simulator (embedded or federated) should also be mentioned. This controllability may be described by another paradigm in mobility modeling called *Agent Centric vs. Flow Centric*. In the “Flow Centric” approach, vehicles trajectories are macroscopically modeled as flows and thus no individual control on vehicles is possible. Two popular methods have been developed to generate the cars' trajectories: *Stochastic turns* and *O-D Matrix*. The former generates turning probabilities for each intersection, while the latter generates a set of trajectories between origin and destination points. We emphasize that the number of

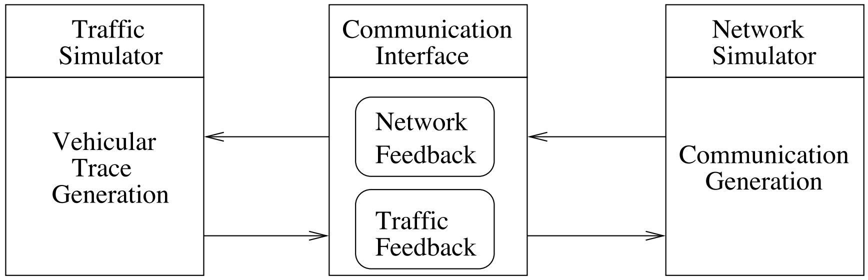


Fig. 6. Interaction between Network and Traffic Simulators: The Federated Case

trajectories is independent of and usually significantly less than the simulated cars. A basic configuration could be 10 O-D trajectories or flows for 1000 cars. In both methods, no individual behavior is possible, as all cars assigned to a trajectory must follow it from the beginning to the end, and this, often several times. In the contrary, the "Agent Centric" approach models the behavior of each driver microscopically, mesoscopically and macroscopically. The driver fully controls the trajectory it follows, as at least one trajectory is generated per driver that indicates the individually optimal path from a source to a destination.

When considering flow-centric models for an application-centric study, the challenge comes when the trajectory of a single car, or a subset of cars, must be altered. For the stochastic turn approach, the solution is to change the turning probabilities of all intersections accessing the road segment containing an accident. However, traffic management is not possible, as it is often necessary to go several intersections away from the accident to find a proper detour. For the O-D Matrix approach, *alternate routes* are pre-computed indicating what the flow of vehicles should do in case of accidents or traffic jams. Yet, the different alternate paths being pro-actively computed, it is either not possible or very complex to optimize them according to the traffic. In opposite, when considering the *Agent Centric* approach, traffic management only requires to change the weights on the specific road elements and to recompute the optimal path for a specific vehicle on-the-fly.

The challenge between the *Agent Centric* and *Flow Centric* approaches is actually the trade-off between *control* and *performance*. Indeed, the former is able to fully control individual behaviors at the cost of an increased computational cost that the latter does not require. But the latter suffers from a reduced control on the real vehicular behaviors. We believe yet that an increased control on the driver behavior will be necessary for future *Application Centric* solutions, such as safety or traffic management and thus will require agent-centric modeling.

E. Discussion

In the community, the three described approaches exist and are still currently developed in parallel. The success of the isolated case, and thus its survivability, comes from its simplicity and universality. Once a mobility model is developed for a network simulator, it could, for instance, be easily extended to support another one. The choice between the three different approaches therefore clearly depends on the application requirements. For example, if only a limited interaction between the mobility and network simulators is necessary (the Network Centric approach), the isolated case should be chosen. Instead, the federated case should have the favors of the community working in safety and traffic management applications. At this time, we recommend that the embedded case should only be chosen either for efficient simulation performances, or for application evaluations that would only require a coarse simulation granularity.

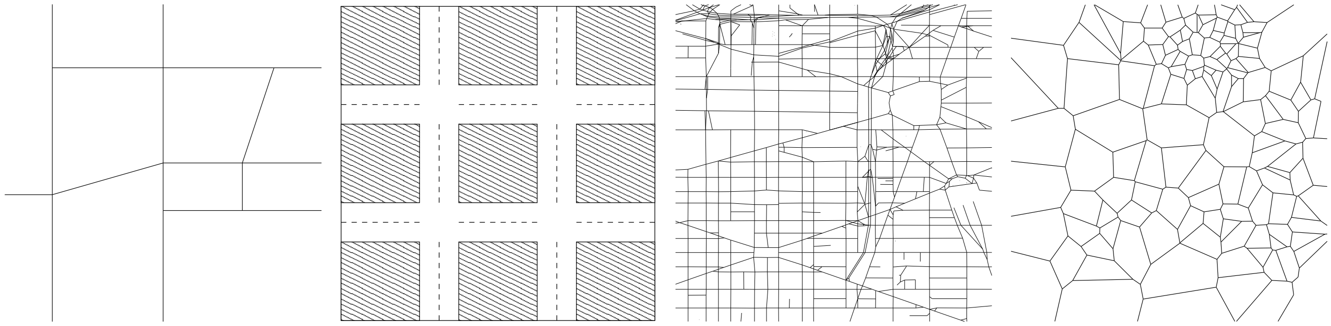
V. TAXONOMY: DESCRIPTION OF THE CRITERIA

Prior to providing a classification of the major mobility models for vehicular networks, we need to define the criteria based on which we will generate the taxonomy. In this section, we introduce a set of criteria extracted from the framework described in Section II and with which we will be able to better differentiate, classify, and evaluate the degree of realism of the different mobility models. The proposed criteria fall in four major categories: **Motion Constraints**, **Traffic Generator**, **Simulator Related** and **Recommendation**. The objective is to let the reader see and easily compare the different key features contained by each model and choose the best model fitted to its requirements. According to the framework, the more building blocks a vehicular mobility model includes, the more realistic it is. The evaluation section also provides a basic recommendation for each model as a function of their level of realism.

A. **Motion Constraints**

The Motion Constraints functional block spans the whole set of impediments that alters free-space mobility. When considering motion constraints, we do not only take into account the road topology, but also include all influences on vehicles movement patterns on the road topology such as the presence of intersection policies, speed limitations, single direction roads, multi-lane features, or even the effects of points of interest. We therefore define the following criteria:

- *Graph* – The mobility is restricted to a graph.
- *Source and Destination Points* – These positions may be either random, random restricted on a graph, or based on a set of attraction or repulsion points of interest.
- *Intersection Policy* – It describes the kind of intersection policy added to the Motion Constraints, such as stop signs or traffic lights, if any.
- *Multi-lane* – The topology includes a multiple number of lanes, potentially allowing lane changes or not.



(a) User-defined topology (b) Manhattan-grid map topology (c) TIGER map topology (d) Clustered Voronoi

Fig. 7. Road topologies examples

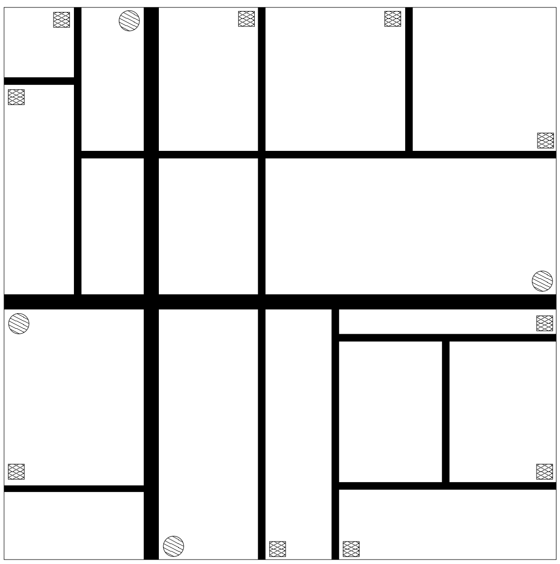


Fig. 8. Example of Attraction Points on a User-defined graph

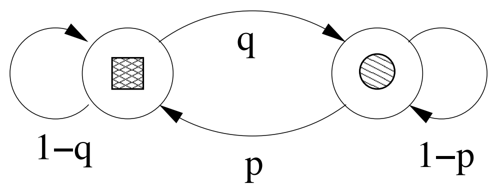


Fig. 9. Activity-based Sequence between the attraction points in Fig. 8

- *Speed Limitation* – The topology includes speed limitations, either homogeneous or per-road segment.

1) Graph: The selection of the road topology is a key factor for obtaining realistic results when simulating vehicular movements. Indeed, the length of streets, the frequency of intersections, or the density of buildings can greatly affect important mobility metrics such as the minimum, maximum and average speed of cars, or their density over the simulated map. We categorize the graphs by the following building blocks:

- *User-defined* – The road topology is specified by listing the graph's vertices and their interconnecting edges.
- *Random* – A random graph is generated, which are often *Manhattan-grid*, *Spider*, or *Voronoi*.
- *Maps* – The road topology is extracted from real maps obtained from different topological standards, such as *GDF*, *TIGER*, or *GIS Arcview*.

Examples of the possible topologies are shown in Fig. 7.

2) Source and Destination Points: Beside the popular, yet unrealistic, random Source and Destination (S-D) points, *Attraction or Repulsion* points are particular S-D points that have potentially attractive or repulsive features. Time alters the set of S-D points, as depending on the time of the day, some S-D pairs will be preferred among others. For instance, in a weekly morning, residential areas are repulsion points and office building are attraction points, as a large majority of vehicles are moving from the former and to the latter. A shopping center will seldom be chosen as source or destination point early morning, but could be more often chosen later during the day. We depict the use of attraction points on a user-defined graph in Fig. 8, where a round represents entry/exit points of high-speed roads (thick lines), and a square entry/exit points of normal-speed roads (thin lines) respectively.

3) Intersection Policy: An intersection policy adds a handling capability to the behavior of vehicles approaching a crossing. Intersections are seen by cars as static yet time varying obstacles, and thus fall into the *Motion Constraints* category. In most cases, only two different intersection scenarios are considered: a crossroad regulated by stop signs, or ruled by traffic lights. However, parameters such as the traffic light transition time, or the policy with respect to stop signs, constrain the motion of vehicles approaching an intersection. For example, when the intersection policy is time dependent, such as a traffic light, the *Traffic Generator* may acquire the current state of the traffic light in the car's direction of movement and may act accordingly. The effect of each intersection policy therefore significantly depends on the interaction between the Motion Constraints and the Traffic Generator blocks.

B. Traffic Generator

The Traffic Generator block is the second major category proposed by our framework and is in charge of generating vehicles and modeling their mobility by respecting all constraints introduced by the Motion Constraints block. In the proposed taxonomy, the Traffic Generator may be composed of the following building blocks:

- *Trip Generation* – A trip may be randomly generated between the source and destination points, or set according to a sequence of activities.
- *Path Computation* – It provides the algorithms used to generate the complete path between a source and a destination point on the trip created by the trip generator.
- *Human Mobility Patterns* – The car's internal motion and its interaction with other cars may be inspired from human motions described by mathematical models such as *Car Following*, *Behavioral Model*, or not.
- *Lane Changing* – Describes the kind of overtaking models implemented by the traffic generator, if any.
- *Velocity* – The simulated velocity may be uniform, smooth or road-dependent.
- *Intersection Management* – Describes the behavior with respect to the intersection policy contained in the Motion Constraints block, if any.

1) Trip Generation: The trip generation determines a trip's source and the destination points on the topological map. Generally, the trip generator randomly selects source and destination points on the map, without any correlation between previous and future destinations or without considering *Time Patterns*. It may also restrict the different source-destination points between the *Points of Interest* from the *Motion Constraints* class. Pushing the realism further, the generation of an *activity sequence* is another restriction in vehicles spatial and temporal distributions. A set of source and destination points are explicitly provided by the *Motion Constraint* category, and cars are forced to move among them in a predefined sequence. In particular, multiple sets of points of interest can be specified, along with the probability matrix of a vehicle moving from one set to another. Fig. 9 illustrates an activity sequence generated from a first order Markov chain between two categories of attractions points provided by the *Motion Constraints* class.

2) Path Generation: Once source and destination points have been selected for a trip, the path generator is in charge of first creating the precise itinerary followed by the car on the topological map, yet by respecting motion constraints such as single direction roads or restricted road access per category of cars or time of the day. Different algorithms may be employed for that matter (Dijkstra Shortest Path for instance). Depending on the level of realism provided by the path generator and the level of description from the motion constraints, the path may consider traffic congestion or speed limitations on each road segment in order to provide the fastest path between a source and a destination point. The path generator also considers *External Influences*, and is therefore in charge of finding an alternate route in case an unexpected event occurs on the original path, such as accidents, road blocks, or heavy traffic.

3) Human Mobility Patterns: The category containing human mobility patterns is probably with the intersection management, the category which has the most significant impact on the realism of vehicular mobility. Indeed, beside moving from one point to another respecting motion restrictions, it is obvious that vehicles are involved in a complex interaction between each others and controlled by human beings. This interaction is often referred to as micro-mobility, as it refers to the control of acceleration, deceleration levels, and reaction time in order to maintain a safe inter-distance between successive cars. In the following, we shortly describe the most widely used vehicular specific micro-mobility models aiming at extrapolating human mobility pattern. We refer to [4] for a larger coverage of the different microscopic mobility models.

a) Car Following Model (CFM): The class of *car following* models (CFM) is the major class of microscopic models implementing human mobility patterns. The CFM adapts a following car's mobility according to a set of rules in order to avoid any contact with the leading vehicle. A general schema is illustrated in Fig. 10. Brackstone in [46] classified Car Following Models in five classes: *GHR Models*, *Psycho-Physical Models*, *Linear Models*, *Cellular Automata*, *Fuzzy Logic Models*. A description of the differences between these models is out of scope of this paper. We refer the interested reader to [47]. We only list here the widely used models in traffic simulations, and which may be found in the models described in this survey.

- Krauss Model (KM) [48]
- Nagel and Schreckenberg Model (N-SCHR) [49]
- Wiedeman Psycho-Physical Model (Psycho) [50]
- General Motors Model (GM) [51]
- Gipps Model (GP) [52]
- Intelligent Driver Model (IDM) [53]

4) Lane Changing Models: Despite the large attention given to the driving tasks in general (such as car following models), much less attention has been directed to lane changing. Modeling lane changing behaviors is actually as complex as modeling human motion patterns. Indeed, a realistic modeling should actually include three parts: the **need** for lane changing, the **possibility** of lane changing, and the **trajectory** used for lane changing. Each part is important for a realistic lane changing attempt. And unlike car following models, it also needs to consider nearby cars and traffic flow information. Most of the models are based on a *Gap Acceptance* threshold [54] or a set of rules [55]. But recent approaches ([56], [57]) also consider forced merging, behavior aspects or game theory. Lane changing is not widely considered in open vehicular mobility models. In this survey, we mostly found

- Gibbs Model for Lane Changing (GP-LC) [54] and its variations such as [56]
- Wiedeman Psycho-Physical Model for Lange Changing (Psycho-LC) [58]
- MOBIL [57]

5) Intersection Management: As mentioned in Section II, the intersection management block does not directly impact the traffic generator, as each intersection management policy is simply seen as obstacles and the intersection management only acts on the first vehicle on each road, letting the car following model automatically adapt the behavior of the following cars.

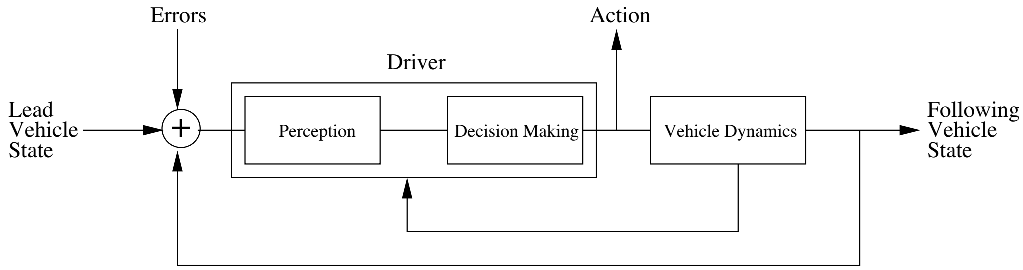


Fig. 10. General Schema for Car Following Models

However, we added an intersection management block in the traffic generator functional block in order to illustrate the behavior of the next direction policy followed by cars after the intersection. This feature could also be seen as part of the path computation, but for models that do not have a path computation, the different paths are selected at each intersection. Moreover, another feature that belongs to the traffic generator is the required speed adaptation as a function of the policy and the intersections status at the time a car is approaching. Summarizing, the traffic generator needs to know if an obstacle exists at the intersection or not, and which will be the next direction that the car will take in order to adapt the velocity. The strategies found in the described models may be classified in the following two criteria:

- Stochastic Turns: When a trip or a path is not provided, cars choose the next direction according to a probability density function and adapt their speed accordingly.
- Predefined Turns: When a path is provided (either flow-centric or agent-centric), the next direction is predefined and speeds already estimated.

C. Simulator Related Criteria

As illustrated in Section IV, the interaction with a network simulator or with the end-user by visualization means, is also a crucial capability for vehicular mobility models. We therefore provide here the following additional building blocks:

- *Visualization* – The model includes a visualization tool, or not.
- *Output* – This block describes the kind of network output generated by an isolated mobility model, or to which network simulator a federated model is interlinked, such as ns-2 or QualNet. This block does not make sense for the embedded models.
- *Platform* – Provides the programming language on which the simulator has been developed.
- *Class* – Indicates the level of interaction between the mobility model and a network simulator. The criteria are based on Section IV.

D. Recommendation

This last criterion is proposed in order to provide an evaluation on the appropriateness of each mobility model and help readers choose the most fitted one for their needs. Yet, choosing between one model or another significantly depends on application requiring vehicular communication. In order to provide a fair recommendation, we used the following three criteria:

- *Minimum Requirements for Realism* – Evaluated according to the study by Fiore *et al.* [31], we judge the minimum mobility requirements for *highway* or for *urban* scenarios, or for none of them depending on the features available in each model.
- *Radio Obstacles* – The model considers radio obstacles, either in the form of an obstacle topology for network simulator and a propagation computation interface for network simulators, or simply a radio propagation trace file. The availability of radio obstacles is absolutely required in urban scenario but less critical in highway scenarios. Radio obstacles are however only considered as *optional* due to the limited number of models fulfilling this requirement.
- *Validation* – It is checked when a model has been validated either using the direct or the delegated procedure. Even though a validated model would be highly preferable, we chose to add this requirement also as *optional* due to the small number of validated models.

In the next section, we will therefore propose a recommendation according to the requirements of these two groups (see Table I).

VI. The Taxonomy of Vehicular Mobility Models

After having introduced the criteria for the taxonomy, we simultaneously provide in this section a brief description of the major mobility models available to the vehicular networking community, and classify them in Table II, Table III and Table IV according to the building blocks introduced in Section V.

A. Remarks on Requirements for Realism

In this section, we intend to describe models depending on their level of realism. However, what is "realism", and what is "being realistic" in vehicular mobility modeling? This question is particularly complex to answer, as only few studies have been performed on this particular feature for urban environments. Helbing [63] illustrated that a car-to-car interaction was a minimum requirement for a realistically modeling of vehicular motion patterns. But this study was targeted at studying flows on highways and his conclusions are not sufficient for an urban environment. A more recent study

TABLE I
MINIMUM REQUIREMENTS FOR REALISTIC MOBILITY MODELS

Environment	Minimum Requirements			Optional	
	CFM	Intersection	Lane Changing	Radio Obstacles	Validation
Highway	x		x		x
Urban	x	x		x	x

performed by Fiore *et al.* *[31]* illustrated that, in urban environments, only models implementing simultaneously a car-to-car and a car-to-infrastructure interaction could pass standard traffic theory test for urban environments. Therefore, and to use the building blocks described in Section II, the minimal requirements for realistic modeling in highway environments are human driving patterns. In urban environment instead, models should also include an intersection management, consequently implementing an interaction between the motion constraints and the traffic generator functional blocks.

These suggestions are actually minimal requirements, as there is not at this time any study showing what are the sufficient requirements for mobility modeling. A *sufficient requirements* set for mobility modeling would mean that any extra requirement not contained in the sufficient set does not have a significant impact on the motion patterns. By definition, this set depends on an application.

When discussing the appropriate models to use for vehicular communication applications, we must also extend our analysis to the network simulators for the class of embedded vehicular mobility models. Indeed, with the isolated or federated class of models, the end-user has the choice to use the most appropriate network simulator. It is unfortunately not the case for embedded models that natively contain a basic network simulator that could not contain a sufficient level of realism to be appropriate for each application. When considering realism of network simulators, all underlying functionalities must be conform to a specification and well configured. The objective is that the probability of reception of each packet is as realistic as in real environments.

B. Traffic Simulators

The development of traffic simulators started decades ago with the modeling of critical highway sections of highways or urban intersections for traffic engineering. For example, most of the realistic traffic simulation tools, such as PARAMICS [25], CORSIM [26], VISSIM [27] or TRANSIMS [28] have been developed to analyze vehicular mobility at both microscopic and macroscopic levels with a very hight level of detail. With no exception, all these tools provide a precision that is not reached by any model described in this survey. Moreover, they have been all validated as providing realistic vehicular mobility. For a complete review and comparison of such traffic simulation tools, the interested reader can refer to [62].

Before moving further, we would like to answer a question which might appear obvious: if *Traffic Simulators* are able to provide realistic and validated vehicular motion patterns, why do we need alternate solutions?

In order to give an answer to this question, we have to understand what are the intended target of traffic simulators. They have been created for transportation, traffic, and civil engineers. With the exception of few teams that developed parsers (e.g. [60], [61]), or federated a traffic simulator with a network simulator (such as FDK [38]), these tools have not been designed for the generation of movement traces usable by networking simulators. For example, CORSIM does not output anything else than statistics. And copyrights forbid anyone from changing this significant limitation. Yet, solutions have been obtained for the particular case of CORSIM by creating a specific parser to extract vehicular mobility information. Yet, this solution is exceptional, as CORSIM needs to communicate with the visualizer using byte codes containing mobility information. If it had not been the case, that solution would not have been possible. A second significant limitation of traffic simulators is the complexity of their calibration. The level of realism involved in traffic and transportation planning is so high that potential users need to tweak a large set of parameters, each of them influencing vehicular traffic. This process is very time consuming, and for a large part, also appears useless to networking researchers, as network simulations do not require such levels of details for vehicular mobility. The objective of vehicular mobility models is to provide vehicular mobility patterns tailored to the specific needs of vehicular networking. The set of parameters are therefore usually significantly reduced, and the simulator's calibration faster. Last but not least, another significant impediment for the use of *Traffic Simulator* is the commercial license that needs to be acquired, which could be significantly onerous in some cases. For all these reasons, a large community started to work on the development of free vehicular mobility models tailored to network research, and which provide interactions with network simulators.

Yet, some open source traffic simulators have been created, validated, and made available to the community. The *Simulation of Urban MObility (SUMO)* [35] is an example of such open source, highly portable, road traffic simulation package designed to handle large road networks. For the Traffic Generator part, route assignments may be imported from various sources. The human driving patterns block in SUMO is a car following model, and the path motion block includes a stochastic traffic assignment modeled by a probabilistic route choice. For the motion constraints, SUMO contains parsers for various formats, ranging from TIGER, GIS Arcview, or even to VISSIM. SUMO is also able to output traces straightforwardly usable by network simulators. In order to ease the configuration of SUMO, we also mention *The Mobility Model Generator for Vehicular Networks (MOVE)* [36]. As a simple parser for SUMO, it yet enhances SUMO's complex configuration with a nice and efficient GUI, and inherits all its features.

The CARISMA traffic simulator [43] is a realistic simulator containing both a motion constraints block and traffic generator block. It includes a stop sign intersection management and imports real topological maps in ESRI standard [81]. It provides a real-time trip management, which is a very interesting feature for the evaluation of car-to-car messaging, and also implements the Krauss's car following model as human driving patterns. This model has also been federated with ns-2 for a realistic evaluation of vehicle-to-vehicle messaging systems. One major limitation of CARISMA comes from the ESRI shape files, which are not publicly available unless you buy ESRI products. Moreover, lane changing models and complex intersection managements are not considered at that time.

Finally, another important traffic simulator is the SHIFT Traffic Simulator [83]. It has been developed by PATH at the UC Berkley, and is now a well established traffic simulator that generates vehicles' trajectories according to validated CFM models. More specifically, SHIFT is a programming language with simulation semantics, which was used in *SmartAHS* as a mean for the specification, simulation and evaluation of the modeling and control of Automated Highway Systems (AHS). The major restriction of this simulator is its limitation to the modeling of segments of highways, the lack of topology modeling in the motion constraints, and the lack of trip and path motion blocks.

C. Network Simulators

Before moving forward, we would like to shortly introduce the various network simulators that are available to the vehicular networking community, as they play a key role in the classification introduced in this paper. Similarly to traffic simulators, we may also classify the simulators as commercial-based or free-of-charge. We emphasize that licenses are sometimes available free of charge to university programs.

1) Commercial-based Network Simulators: Probably the most widely used commercial-based network simulator is the Opnet Modeler [72], an efficient discrete event simulator containing hundreds of network protocols, including a complete wireless suite. Although the license may be onerous, Opnet offers a university program with a free license for the fully functional Opnet Modeler. The alter-ego of Opnet is Qualnet [71], which has the interesting feature of managing multi-processors parallel computing. Similar to Opnet, it contains a large library of wireless network protocols, including the MANET suite. Unfortunately, the fully functional Qualnet, including the parallel computing feature is not available free of charge. Finally, the OmnNet++ simulator is the last major licensed-based network simulator proposed to network research, which also proposes various network protocols libraries. Omnet++ is offered free of charge for academic and educational purposes, but a commercial use requires the purchase of a license.

2) Open Source Network Simulators: As an alternative to commercial-based network simulators, non-commercial ones are also available. Yet, the extent of the community behind a particular simulator significantly determines the various available protocols, as only a limited set of functionalities are provided by default. The most widely used network simulator is ns-2 [68], which did not earn its popularity from its simplicity or efficiency, but from its modularity and universality. Extended to wireless networks by the Monarch Project at Rice University, it became, de facto, the reference simulator for MANETs. It contains the major layers of the OSI stack, and do to its popularity, all major protocols for MANETs. The natural alternative to ns-2 is GloMoSim [70], which is the free version of QualNet [71] with reduced capabilities. It has the same functionalities and objectives as ns-2. The set of available protocols is however reduced due to the smaller GloMoSim community. Swans [73] is a scalable network simulator coded in Java. Although having suffered from a limited popularity at the beginning, it has now reached a sufficient large community and also contains the major protocols for MANETs. Finally, The Georgia Tech Network Simulator (GTNetS) is a full-featured network simulation environment that allows to study the behavior of moderate to large scale networks. GTNetS strict compliance with different protocol stack layers makes it an easier solution for the development of network protocols for MANETs.

Considering Vehicular Networks, ns-2 seems the most adapted simulator among the previously described solutions, as its physical Layer and MAC layer has recently been extended and validated for the modeling of vehicular networks [69], an outbreak filling a major gap in previous evaluations of VANET protocols.

D. Isolated Vehicular Models

In this section, we describe various vehicular mobility models that are able to generate traces for network simulators, yet without being able to interact with them. When confronted to the need for more realism compared to the legacy models, a schism appeared that depends on the direction followed by the community. Isolated vehicular mobility models may therefore be classified into the following evolutionary classes: *Legacy Models*, *Improved Motion Constraints*, *Improved Traffic Generator*, *Improved Motion Constraints and Traffic Generator*.

1) Legacy Mobility Models: In this section, we cover the legacy mobility models that have been the source of the initial evaluations of network protocols. Several models to generate nodes mobility patterns have been proposed, when mobility has first been taken into account in simulation of wireless networks. The Random Waypoint model, the Random Walk model, the Reference Point Group (or Platoon) model, the Node Following model, the Gauss-Markov model, just to cite the most known ones, all include the generation of random linear constant speed movements within a topology boundary. Further works added pause times, reflection on boundaries, acceleration and deceleration. The simplicity of use was the reason for the success to the Random Waypoint model in particular. However, the intrinsic nature of such mobility models produces unrealistic movement patterns compared to some real world behaviors. Nevertheless, random models are still widely used in the study of Mobile Ad-hoc Networks (MANETs).

As far as Vehicular Ad-hoc Networks (VANETs) are concerned, it soon became clear that using any of the aforementioned models would produce non significant results. Consequently, the research community started seeking more realistic models. The simple Freeway model and Manhattan (or Grid) models were the initial steps. Then, more complex projects were started involving the generation of mobility patterns based on real road maps or monitoring of real vehicular movements in cities. However, in most of these models, only graph-based motion constraints are considered. Although car-to-car interactions are a fundamental factor to take into account when dealing with vehicular mobility [63], little or no attention was paid to the traffic generator block and its interaction with the motion constraints block. A more complete and detailed survey of the legacy mobility models can be found in [64]–[66].

2) Improved Motion Constraints: The probable first improvement to the legacy models has been provided by the BonnMotion tool [76], which implements most of the random mobility models presented in [64], including the Manhattan model, but restricts nodes' motion constraints on a grid and provides traces for ns-2 and GloMoSim. When related to our proposed framework, we can easily see that the structure is definitely too simple to represent realistic motions, as it only models basic motion constraints without intersection management, and without a traffic generator.

The MONARCH project [78] proposed a tool that extracts road topologies from real road maps obtained from the TIGER [79] database. The capability of generating topologies from real maps is considered in the framework, but the lack of traffic generator makes it difficult to represent a complete mobility model. Indeed, this mobility model is simply a Random Waypoint Model restricted on a graph extracted from real topological cities. Although bringing some spatial correlations, it absolutely lacks time, car-to-car, and car-to-infrastructures correlations. Besides, the authors confirmed that their model showed similar patterns as the RWM.

The Obstacle Mobility Model [82] takes a different approach in the objective of obtaining a realistic urban network in the presence of building constellations as motion constraints. Instead of extracting data from TIGER files, the simulator uses random building corners and Voronoi tessellations to define movement paths between buildings. It also includes a radio propagation and attenuation model based on the constellation of obstacles. According to this model, movements are restricted to paths defined by the Voronoi graph. The Voronoi Model [91] is another illustration of how Voronoi graphs proposed by some simulators could be refined and improved to generate smoother roads as motion constraints. Unlike other mobility models including Voronoi tessellations, this *Voronoi Model* does not model roads as graph edges, but instead as Voronoi channels. A Voronoi channel is a spatial area obtained after multiple applications of a Voronoi Tessellation algorithm. It provides a global moving direction, while keeping some degree of liberty in the local direction patterns. Most of the contributions are related to the improvement of the motion constraints component as a promising random topology generator, while the traffic generator engine is a simple implementation of a Random Walk within each Voronoi channel. These two models' absolute lack of a human motion, trip and path motion blocks makes it unrealistic for vehicular mobility modeling.

3) Improved Traffic Generator: In parallel to, and almost independently from, the improvement of the Motion Constrains block, the community also improved the Traffic Generator block, extending its simple Random Walk from a random source to a random destination. Most of the following models use a simple graph-based motion constraints, which have been proved not to bring a significant change to vehicular motion compared to the RWM by the MONARCH study [78]. Yet, by modifying the general movement of vehicles, either by altering the trip or the path, the realism may be improved.

To that objective, the GEMM tool [77] might have been the first one to provide an extension to the BonnMotion model, as it improved the traffic generator by introducing the concept of human mobility dynamics, such as *Attraction Points (AP)*, *Activity*, or *Roles*. Attraction points reflect a destination interest to multiple people, such as grocery stores or restaurants. Activities are the process of moving to an attraction point and staying there, while roles characterize the mobility tendencies intrinsic to different classes of people. While the basic concept is interesting, its implementation by the traffic generator is limited to a simple enhanced RWM between APs. It however represents an initial attempt to improve the realism of mobility models by considering human mobility dynamics.

The CanuMobiSim tool [86] is another attempt to improve the realism of vehicular motion, although the objective was a general mobility by providing many different synthetic models. As for previous approaches, the motion constraints block is limited to a graph topology, possibly extracted from detailed Geographical Data Files (GDF), and the tool can generate mobility traces for *ns-2* and *GloMoSim*. However, unlike many other tools, the CanuMobiSim tool was the first one to implement a traffic stream model and a car following model (CFM) as an attempt to reproduce human mobility patterns by the traffic generator. Respectively, the *Fluid Traffic Model* adjusts the speed given vehicles local density, and the *Intelligent Driver Model (IDM)* adapts the velocity depending on movements between neighboring vehicles. Also unlike other tools, CanuMobiSim includes a complex trip and path generator. The trip generator is based on a probability matrix, which models a sequence of activities, while the path generator creates basic source-destination paths, either using the Dijkstra shortest path algorithm, local traffic density information, or similarly to GEMM, model trips between *Attraction Points*. This solution is actually the only fully implemented and available solution considering a heterogeneous class of users and destinations. In order to improve its modeling capability, CanuMobiSim has been extended (see [87]) by the same authors, and now includes a radio propagation module for ns-2 and QualNet.

4) Improved Motion Constraints and Traffic Generator: Although the two previous improving directions could bring more realism toward the modeling of vehicular motions, one major feature is globally missing in all previous proposed solutions: an interaction between the motion constraints and the traffic generator blocks. And this may only be reached by a joint improvement of the motion constraints and the traffic generator. For instance, the feature that will mostly impact vehicular motion patterns is the existence of traffic signs, such as stop signs or traffic signs, and their influence on the traffic generator. According to [31], this is the minimal requirements to make mobility models realistic in urban environments.

The Street Random Waypoint (STRAW) tool [84] is based on the freely available Scalable Wireless Ad Hoc Network Simulator (SWANS). STRAW contains a complex motion constraints block obtained by extracting urban topologies from the TIGER database, and by adding intersections. The traffic generator block, on the other hand, contains the Nagel-Schreckenberg car following model for the implementation of human motion patterns. STRAW is therefore one of the few mobility tools that implements a complex intersection management using traffic lights and traffic signs. Thanks to this, vehicles are showing a more realistic behavior when reaching intersections. The concept behind STRAW is very similar to the framework described in section II, as it contains accurate mobility constraints as well as a traffic generator engine. STRAW also includes several implementations of transport, routing and media access protocols, as they are not present in the original SWANS software. The main drawback of the tool is the smaller diffusion of the SWANS platform compared to ns-2. Indeed, a simulator used by a large community has a higher chance to contain the state-of-the-art protocols and communication concepts, and thus could allow an easier comparison between different approaches. Results obtained from it could therefore have a higher chance to be considered seriously.

The next evolution step offered the SSM/TSM model [89] and the Generic Mobility Simulation Framework (GMSF) [90]. Although both models appeared after STRAW or VanetMobiSim, they are introduced at this position in order to show the different models depending on the progressive evolution of their realism. The SSM/TSM represents actually three different mobility models, a *Stop Sign Model (SSM)*, a *Probabilistic Traffic Sign Model (PTSM)*, and a *Traffic Sign Model (TSM)*. The motion constraints part is dealt using a TIGER parser and also includes multi-lane features, while the traffic generator contains a very basic interaction between following cars in order to avoid collisions. The GMSF model provides very similar features, but with the characteristic to use topological maps and detailed mobility traces from Switzerland. Yet, at the time of this survey and for both simulators, the traffic generator is not able to benefit from the availability of multi-lanes and does not contain any overtaking capabilities. SSM, PTSM, TSM and GMSF include a road-dependent velocity distribution, and goes further than STRAW by including a path generator using speed limitations per road segment. Similarly to STRAW, SSM/TSM and GMSF have been specifically designed to model vehicles' motions at intersections, and the same observation provided by the authors of STRAW may be obtained by SSM/TSM or GMSF.

The VanetMobiSim [29] tool further enhanced the level of realism achieved so far, as it is able to model stop signs, traffic lights, safe inter-distance management, but also a behavior-based traffic generator including human mobility dynamics. The Motion Constraints category contains various road topology definitions too, ranging from realistic GDF [80] or TIGER [79], to user-defined or random topologies. The traffic generator lets the user define the trip generation between random source-destination points, or based on a sequence of activities. Moreover, the path used on the defined trip is also configurable between a distance shortest-path, a road-speed shortest path, and a density shortest path. It finally generates traces for various network simulators, which is a significant asset compared to STRAW. The interaction between the motion constraints and the traffic generator is also more complex than STRAW, SSM/TSM or GMSF, as it also includes overtaking capabilities. VanetMobiSim is, at the time of this survey, one of the most realistic and configurable synthetic model for the generation of vehicular motion patterns in the category of isolated models, as with the exception of radio obstacles and the external influences blocks, it contains all other building blocks from the framework in Section II. In successive studies by the authors of VanetMobiSim ([30], [59]), it has been shown that a basic intersection management, such as a simple stop sign, was able to produce a clustering effect at intersection, and a significant speed decay from the configured average speed. In urban environment, this effect is better known under the name *Traffic Jam*, and is hardly represented in most of the actual simulators. In a recent study [31], the authors also showed that the minimum requirements for a realistic modeling of vehicular motions was a realistic *car following model*, an *intersection management*, and a *multi-lane capability*. Finally, unlike all solutions previously described in this section, VanetMobiSim has also been validated by the delegated validation process against CORSIM, and is therefore able to reproduce realistic vehicular motions.

Finally, the UDel Models [11] are also a set of mobility and radio propagation models created for large-scale urban mesh and vehicular networks. The urban mobility part is significantly different from all previous approaches, as the detailed urban vehicular and pedestrian mobility is calibrated based on surveys. And unlike most of the models found in this survey, the generated simulator also includes a detailed urban propagation model, and an accurate map builder capable of parsing GIS dataset and adding realistic radio obstacles. UDel's simulator models vehicular mobility very realistically, thank to its strict compliance to the proposed framework. It however falls into the *isolated* category, as it only separately generates mobility and radio propagation traces usable by ns-2 or Qualnet.

Due to the lack of interaction between the *Motion Constraints* and the *Traffic Generator* classes, applications such as traffic safety or efficiency cannot be studied with *isolated mobility models*.

E. Embedded Vehicular Mobility Models

The second category of mobility models available to the community working in VANET and ITS contains the embedded mobility models. As described in Section IV, embedded models integrate mobility and networking capabilities in a unique model. As their development started later, this kind of models is usually more advanced than their counter part in the isolated case. Yet, the networking capabilities are also often limited.

TABLE II
MOTION CONSTRAINTS FEATURES IN MAJOR VEHICULAR MOBILITY MODELS

	Motion Constraints						
	Graph			Multi-lane	Speed Limitations	S-D Points	Intersection Policy
	User Defined	Random	Map				
Virtual Track [96]	yes	no	TIGER [79]	no	no	random	no
IMPORTANT [75]	no	Grid	no	no	no	random	Stop signs
BonnMotion [76]	no	Grid	no	no	no	random	Stop signs
RiceM [78]	no	no	TIGER	no	no	random	no
SHIFT [83]	yes	no	no	yes	yes	AP	no
STRAW [84]	no	no	TIGER	no	no	random	Stop signs, traffic lights
GrooveSim [85]	no	no	TIGER	no	yes	random	no
Obstacle [82]	no	Voronoi	no	no	no	random	no
Voronoi [91]	no	Voronoi	no	no	no	random	no
GEMM [77]	no	Grid	no	no	no	AP	no
CanuMobiSim [86]	yes	no	GDF [80]	no	no	random, AP	no
City [88]	no	Grid	no	no	no	random	Stop signs
SSM/TSM [89] GMSF [90]	no	Grid	TIGER	no	yes	random	Stop signs, traffic lights
UDel Models [11]	yes	no	ESRI [81]	yes	yes	random	Stop signs
VanetMobiSim [29]	yes	Voronoi	TIGER, GDF	yes	yes	random, AP	Stop signs, traffic lights
MoVES [32]	no	no	GPSTrack [94]	no	yes	random	stop signs, traffic lights
Gorgorin [34]	no	no	TIGER	yes	no	random	Stop signs, traffic lights
AutoMesh [40]	no	no	TIGER	yes	yes	random	Stop signs
NCTUns 5.0 [33]	yes	no	no	yes	yes	random	Stop signs, traffic lights
SUMO [35] MOVE [36] TraNS [41]	yes	grid, spider	TIGER	yes	yes	random, AP	Stop signs
MobiREAL [92]	yes	no	no	no	no	random	no
CARISMA [43]	no	no	ESRI [81]	yes	yes	random	Stop signs
VGrid [39]	yes	no	no	yes	yes	no	no
S-D: Source-Destination; *AP*: Attraction Point;							

Among the embedded models, The GrooveSim/GrooveNet[3] tool [85] was the first to appear in this area as a joint mobility and communication simulator. The Motion Constraints block contains topological maps from the TIGER database. As a self-sustained software, GrooveSim neither models vehicles-based human motion patterns, nor produces traces usable by any other network simulator than Groovesim. The interesting feature of this model is the non uniform distribution of speeds (although also found in SSM/TSM and VanetMobiSim). Indeed, motion constraints also include road-specific speed limitations with a speed-based path generator. Although it might look straightforward, this type of motion constraints is, at this time, considered by only a few simulators. The authors illustrated how vehicles were naturally choosing roads with the highest speed limitations on their journey. The main drawback of this tool is however the missing of a car-following model and the lack of mobility traces for network simulators.

Still with the objective to accurately model vehicles' specific motions, the City Model [88] has also been developed as an embedded system for testing routing protocols. Traces for network simulators are therefore not provided. The traffic generator includes a basic car following model based on the IDM and enhanced by a simplistic crossing management. Crossings are modeled like static obstacles by the Motion Constraints block. Vehicles change their direction according to a particular probability density function. This simulator mostly lacks modularity mostly due to its unique grid-based motion constraints, to its limitation to stochastic turns by the trip and path generators, and to the lack of human mobility patterns.

In [32], Bononi *et al.* created MoVes, a complex mobility generator on top of a scalable distributed simulation middleware Artis [93]. The interesting feature in MoVes with respect to the framework described earlier in this paper, is the availability of a driver characterization with human driving patterns, possibility providing different kind of mobility patterns depending on the drivers' category. The traffic generator in MoVes features the Psycho-Physical car following model. Motion constraints include GPS maps, thanks to the GPS TrackMaker program [94], and intersection managements. However, MoVes does not include any lane changing model

[3]GroveNet is the core project of modeling and visualizing vehicular mobility, while GrooveSim is the mobility model itself. In the rest of this paper, we will simply refer to GrooveSim for the joint mobility model and visualizer.

TABLE III
FEATURES INCLUDED BY THE TRAFFIC GENERATOR IN MAJOR VEHICULAR MOBILITY MODELS

	Traffic Generator					
	Trip Generation	Path Computation	Human Patterns	Lane Changing	Velocity	Intersection
VirtualTrack [96]	random S-D	RWP	no	no	uniform	no
IMPORTANT [75]	random S-D	RWM, RWalk	CFM	no	smooth	no
BonnMotion [76]	random S-D	RWM	no	no	uniform	no
RiceM [78]	random S-D	Dijkstra	no	no	uniform	no
SHIFT [83]	no	no	CFM	yes	smooth	no
STRAW [84]	random	RWalk, Dijkstra	CFM (N-SCH)	yes	smooth	stochastic, predefined turns
GrooveSim [85]	random	RWalk, Dijkstra	no	no	smooth, markov, density, speed-limit	stochastic, predefined
Obstacle [82]	random	Dijkstra	no	no	uniform	no
VoronoiM [91]	random	RWalk	no	no	uniform	no
GEMM [77]	random	RWM	no	no	uniform	no
CanuMobiSim [86]	random, activity	RWP, Dijkstra, density	IDM	no	uniform	no
City [88]	random	RWM	IDM	no	smooth	stochastic turns
SSM/TSM [89] GMSF [90]	random	Dijkstra	IDM	no	uniform	stochastic turn
Udel Models [11]	random	RWalk	CFM	yes (GP_LC) [56]	smooth	stochastic turns
VanetMobiSim [29]	random, user defined, activity	RWP, density, Dijkstra, speed	IDM, IDM_IM, IDM_LC	MOBIL	smooth, density	predefined
MoVES [32]	random	RWalk	CFM (Psycho)	no	uniform	stochastic
Gorgorin [34]	random	RWalk	CFM (Psycho)	CFM (Psycho-LC)	smooth	no turns
AutoMesh [40]	random	Dijikstra, activity, speed	IDM	no	uniform	stochastic, predefined
NCTUns 5.0 [33]	random, user defined	RWalk ,user defined	CFM	yes	smooth	stochastic, predefined
SUMO [35] MOVE [36] TraNS [41]	random S-D, activity	RWalk, Dijkstra	CFM (Krauss)	yes	smooth	stochastic turns
MobiREAL [92]	random	RWalk	CPE	no	uniform	no
CARISMA [43]	random S-D	Dijkstra, speed, density	CFM (Krauss)	no	smooth	predefined turns
VGrid [39]	no	no	IDM	yes	smooth	no

CFM: Car Following Model; *IDM*: Intelligent Driver Model *CPE*: Condition-Probability-Event;
IDM_IM: IDM with Intersection Management; *IDM_LC*: IDM with Lange Changes;

and no trip or path generation is supported at the time of the writing of this survey. The major asset of this tool is its parallel computing features. Its limitation is its weak networking capabilities.

Gorgorin *et al*. [34] also embedded a network and a mobility simulator. A major limitation in this model, at this time, is the relative simplicity of both simulators. Indeed, although motion constraints are imported from TIGER maps and the traffic generator includes a similar car following model as VISSIM, it does not consider any realistic path or trip generation. Moreover, similarly to MoVes, the network simulator also suffers from its simplistic architecture and from its poor diffusion compared to QualNet, OpNet or ns-2.

Vyyuru *et al*. proposed AutoMesh [40], a realistic simulation framework for VANETs. It is composed of a set of modules controlling all parts of a realistic simulation. It includes a *driving simulator module*, a *radio propagation module*, and *a network simulator module*. All modules are interlinked with feedback in order that any alteration made in one module influences the other modules. At the stage of the development of AutoMesh, the driving simulator module includes a random trip generator, a basic intersection management, and the IDM model. It has to be noted that the radio propagation module is very detailed, using 3D maps and digital elevation models in order to obtain a realistic radio propagation model in urban areas. This is a rare and valuable feature in the different solutions we describe in this paper.

The last tool falling into the embedded category is NCTUns [33], which is not specifically focused on vehicular mobility, but provides a full range of network stack simulation tools. The tool yet contains sufficient functionalities for vehicular motion modeling, such as human driving patterns, car following models, and intersection managements. It is also the only tool described in this survey which provides emulation capabilities and not only simulation functionalities, which allows a more precise evaluation of the behavior of network protocols. The networking part has also recently been extended to handle the IEEE WAVE 802.11p MAC for vehicular environment.

When considering the realism of the mobility functions found in these embedded models, a sufficient level of realism is provided for the modeling of vehicular motions. Yet, the network function might not be able to compete against ns-2 or Qualnet in term of networking capabilities. Yet, due to an improved computational approach, they are usually able to model a bigger network than ns-2 for instance. They could therefore be chosen for large scale simulations when QualNet is not available. Moreover, due to the coarse-grained network simulation part, traffic safety applications should not be evaluated using *embedded* mobility models.

TABLE IV
SIMULATOR ORIENTED AND EVALUATION FEATURES OF MAJOR VEHICULAR MOBILITY MODELS

	Simulator Related				Evaluation		
	Visualization Tool	Output	Platform	Class	Min. Rqrmt.	Radio Obstacles	Validation
Virtual Track [96]	no	ns-2, QualNet	C++	isolated	no	no	no
IMPORTANT [75]	no	ns-2	C++	isolated	no	no	no
BonnMotion [76]	yes	ns-2, QualNet	Java	isolated	no	no	no
RiceM [78]	no	ns-2, QualNet	C++	isolated	no	no	no
SHIFT [83]	yes	no	C++/SHIFT	isolated	highway	no	only highway
STRAW [84]	yes	Swans	JiST-Swans	isolated	urban, highway	no	no
GrooveSim [85]	yes	no	C++	isolated	no	no	no
Obstacle [82]	yes	ns-2, QualNet	C++	isolated	no	yes	no
VoronoiM [91]	no	ns-2	C++	isolated	no	no	no
GEMM [77]	no	ns-2	Java	isolated	no	no	no
CanuMobiSim [86]	yes	ns-2, QualNet,	Java	isolated	highway	yes	no
City [88]	yes	ns-2	C++	isolated	urban, highway	no	no
SSM/TSM [89] GMSF [90]	no	ns-2	C++	isolated	urban, highway	no	no
UDel Models [11]	yes	ns-2, QualNet	C++	isolated	urban, highway	yes	no
VanetMobiSim [29]	yes	NS-2, QualNet	Java	isolated	urban, highway	yes	yes
MoVES [32]	yes	X	C++	embedded	urban, highway	no	no
Gorgorin [34]	yes	X	C++	embedded	highway	no	only highway
AutoMesh [40]	yes	X	C++	embedded	urban, highway	yes	no
NCTUns 5.0 [33]	yes	X	C++	embedded	urban, highway	no	no
SUMO [35] MOVE [36] TraNS [41]	yes	ns-2, QualNet	C++	federated	urban, highway	no	yes
MobiREAL [92]	yes	GTNetS	C++	federated	urban	yes	yes
CARISMA [43]	yes	ns-2, QualNet	C++	federated	urban, highway	yes	no
VGrid [39]	yes	Swans	Java JiST	federated	urban, highway	no	no

isolated: Isolated class of mobility models; *embedded*: Embedded class of mobility models;
federated: Federated class of mobility models; *CFM*: Car Following Model

F. The Federated Mobility Models

After having described models falling into isolated and embedded categories, we now address the *Federated* category, which has the potential of providing very advanced features for vehicular motion modeling and networking capabilities, but also has the limitation of a complex configuration and possibly license restrictions depending on the traffic simulator. As mentioned in Section IV, instead of re-inventing the wheel, the community tried to federate already existing and validated network and mobility simulators, with the clear objective of taking advantage of the experience gathered in both fields. It is also worth to be noted that, as the solutions described here are based on commercial-based or open source traffic generators, the mobility patterns are the most realistic ones that may be obtained by any vehicular mobility model. These models indeed include almost all building blocks from the framework in Section II. Most interesting, they are also all considered as validated.

In the category of open-source models, TraNS [41] is a very promising approach using SUMO as its root functionality and federating the network simulator ns-2. Using an interface called *Interpreter*[4], traces extracted from SUMO are transmitted to ns-2, and conversely, instructions from ns-2 are sent to SUMO for traffic tuning. Interactions between the vehicular traffic and the network may be accordingly implemented. Thanks to SUMO, the vehicular mobility traces are among the most realistic traces currently available. One limitation that is under study at the time of this survey is SUMO's inability to alter the trajectories of cars based on an unpredicted traffic impediment (car accident for instance). This issue has been acknowledged by the authors as a limitation in TraNS's ability to study the effects of safety protocols on the evolution of urban traffic[5].

Another federated approach is the VGrid project [39], which federates SWANS as network simulator, the author's defined synthetic traffic model implementing the Nagel-Schreckenberg model, and a game theory approach similar to MOBIL for lane changes. This model is particularly designed to study the effects on traffic of accident alert messages. However, the traffic generator is limited to the modeling of straight roads (like highways) and lacks a correct modeling of urban

[4]This interface has recently been extended to TraCI.

[5]At the time of this survey, the authors of TraNS and SUMO were working on TraCI, a solution to solve this limitation.

areas including intersection management, and a realistic trip and path generator. It is therefore not able to model the non-uniform distribution of cars. Yet, it is able to show the evolution of speed with respect to traffic density thanks to the car following model. It is a nice attempt to model the effects of safety messages, or any other traffic protocol, on the evolution of vehicular traffic.

A similar approach to TraNS called MSIE [42] adopts the same structure, but uses VISSIM instead of SUMO for the vehicular mobility model part. It therefore falls into the commercial-based models, even though VISSIM also exists with a university license. Similarly to TraNS, MSIE provides an important level of details for the vehicular motion, as VISSIM is probably the most realistic commercial-based traffic generator at this time. It also provides functionalities to re-route vehicles based on dynamic traffic changes. Still in the commercial-based category, FDK [38] federates QualNet as network simulator and CORSIM as vehicular traffic generator. It provides similar performance as MSIE or TraNS (although using a more efficient network simulator).

A special attention should also be brought to a novel solution named *MobiREAL* [92] based on the Georgia Tech network simulator. Although focusing on the modeling of pedestrian mobility, its strict compliance with the proposed framework and its novel approach of cognitive modeling makes it very promising for a future extension to vehicular mobility. The most interesting features is that MobiREAL enables to change a node or a class of nodes' mobility behavior depending on a given application context. This particular application context is modeled by a *Condition Probability Event (CPE)*, a probabilistic rule-based mobility model describing the behavior of mobile nodes, which is often used in cognitive modeling of human behaviors. As most of the recent mobility models, MobiREAL is able to include geographical information. Moreover, it is also able to use this information to generate obstacles and more specifically, it is able to model radio interference and fading on the simulation field. MobiREAL's major drawback at this time is the limited diffusion of Georgia Tech Network Simulator (GTNets) and the lack a model for vehicular mobility.

VII. Discussion

The study of vehicular mobility started in the 60's for the analysis of vehicular traffic in order to design coherent traffic management systems. It has been recently revamped with the sudden interest for vehicular networks and the lack of vehicular mobility models natively available in network simulators. As it has been shown in this survey, the area of vehicular mobility modeling is vast and fruitful in term of outcomes for the community working in vehicular communication. And after three years of exciting developments, we propose in this section to take a step back to look at where we stand and to the new challenges ahead of us.

The objective of the community was first to reach a higher realism in the representation of vehicular movements, and second to be able to influence these movements with respect of the applications under study. From what has been shown in this survey, our opinion is that both objectives have been reached. Indeed, *VanetMobiSim* or *STRAW* in the category of isolated models, or MoVes and AutoMesh in the category of embedded models, are able to fulfill all requirements for a realistic modeling of vehicular motion patterns. And in the federated category, solutions like *TraNS* or *MSIE* will soon provide a full control on vehicular mobility in reaction to vehicular safety applications for instance. In [31], Fiore *et al.* illustrated the minimum requirements to realistically model vehicular motion. A question that is still open is what the sufficient requirements are. At some point, we guess that the level of precision in the modeling will stop impacting the evaluation of vehicular communication protocols.

One could argue that some applications might only need a coarse-grain intersection management, while some safety protocols could need to obtain fine-grain information in order to realistically model accidents, and their impacts on urban traffic. This question remains open and will depend on the requirements of the future safety applications that are still under development. We may only argue that the most promising models in the three different categories could all be extended to fulfill the new requirements. However, our feeling is that we might not need a significantly higher level of precision that we already have.

Another question that should be addressed relates to the configuration of each mobility model. More realism often means more parameters to tune. Yet, what are the right values, and how to control them? Do we even have a slight idea of the correct configuration parameters we should set?

A significant obstacle to really answer this question comes from the large deviation between the transient and the steady state phases, if such two phases even exist for vehicular motion. By studying traffic traces, we are able to extract traffic parameters that cannot be used to configure models, as such parameters represent the state of traffic at the steady-state phase, while configuration parameters usually control the state at the early transient phase. An example is the speed or the density of cars. If we configure a mobility model to have a speed s or a density d, we want these values to be at steady-state and not during the transient phase. Nowadays, parameters at the steady-state phase can only be extracted a posteriori, and are thus not easily controllable. The danger of not being able to control the real parameters is the inability to reproduce the results, or fairly compare different protocols.

More than setting parameters, evaluating protocols often requires the definition of different evaluation environments. The community also faces the almost impossible task of choosing among a large set of different vehicular environments, each of them having a significant impact on the protocols under study. One approach to solve this issue would be for the community to agree on a common set of parameters as well as benchmark values for the evaluation metrics. For instance, if we need to evaluate a protocol in urban area with heterogeneous cars and drivers, the benchmark would provide a set of parameters the model should follow. The community could therefore be able to obtain a fair comparison between different approaches.

In this survey and also in the issues raised in that section, we discussed the modeling of vehicular motion patterns. The straightforward impact being to see how mobility influences network and application protocols. And the answer is not yet

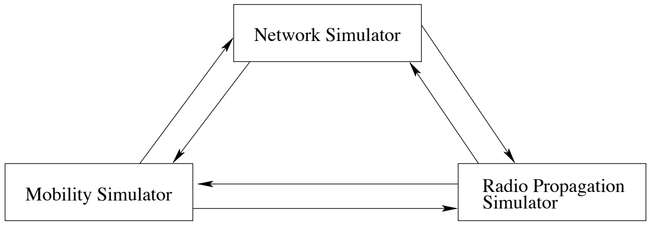


Fig. 11. Trium Vira in Realistic Simulation

clear. However, vehicular networks are also a wireless environment and it is expected from the radio link to have an impact as least as important as mobility. As a matter of fact, the quality of the radio channel between two vehicles significantly depends on the mutual mobility. A realistic radio propagation and fading model should therefore be implemented, which would also consider radio obstacles possibly blocking the signal. A radio module could do this task but its precise location should be discussed. At this time, a very simplified version of such module is almost exclusively contained in network simulators. The radio module needs to have access to mobility patterns and topographic information to model the radio channel. It should therefore be included in mobility models. Yet, the radio module also requires to have access to the data traffic model from each of the potential transmitters, which justifies its inclusion into network simulators. To our point of view, and similarly to the federated or embedded approaches, a *trium vira* should be created, where three modules control each part of a realistic simulation of vehicular networks, and interact with each others in order for each parameter to impact other modules (see Fig. 11). This might be a motivating future direction for the community working on vehicular mobility modeling, and as a matter of fact, most of the key players in that field have already envisioned this next step.

VIII. Conclusion

As a prospective technology, Vehicular Ad Hoc Networks (VANETs) have recently been attracting an increasing attention from both research and industry communities. One of the fastest growing domain of interest in VANETs is safety, where communications are exchanged in order to improve the driver's responsiveness and safety in case of road incidents. VANETs characteristics are a higher mobility and a limited degree of freedom in the mobility patterns. Such particular features make standard networking protocols inefficient or unusable in VANETs. Accordingly, one of the critical aspects when testing VANETs protocols is the use of mobility models that reflect as closely as possible the real behavior of vehicular traffic. In this survey, we first presented a framework which should be followed for the generation of realistic vehicular mobility patterns, then we covered the different approaches in vehicular mobility modeling, and proposed a classification of vehicular mobility models according to the techniques used for their creation. We finally described the most popular models available to the research community at this time, and provided their detailed taxonomy according to criteria based on natural building blocks required for realistic vehicular mobility modeling.

Unlike MANETs, the major objective of VANET protocols is a direct alteration of the traffic patterns for safety or trip optimizations. Accordingly, we also described the new trend to interlink traffic and network simulators in order to create a cross-layer collaboration between routing and mobility schemes. As far as the authors are concerned, this is the first article which clearly addresses this issue in perspective to other approaches, and provides an insight of the future research directions in joint traffic and network simulations.

The aim of this survey was to allow a comprehensive understanding of the various emerging developments of realistic vehicular traffic generators, the different methods, their justifications, and the interlinking with network simulators. We hope it will be a good guideline for people interested in understanding the unique relationship between traffic models and network protocols in vehicular networks. This article also provided a large coverage of the most popular mobility models for vehicular networks, and could thus be a starting point for people coming to this field or desiring to increase their knowledge in Vehicular Mobility Modeling.

Acknowledgments

The authors would like to thank the four anonymous reviewers for their valuable remarks and the time they invested to improve the quality of this survey.

References

[1] Vehicle Infrastructure Integration (VII), http://www.its.dot.gov/vii/.
[2] Car-2-Car Communication Consortium (C2C-CC),http://www.car-to-car.org/.
[3] Wireless Access for the Vehicular Environment (WAVE), http://grouper.ieee.org/groups/802/11/Reports/tgp_update.htm.
[4] M. Fiore, "Vehicular Mobility and Network Simulation", to be published in *Handbook on Vehicular Networks*, S. Olariu and M. C. Weigle Eds., Taylor and Francis, 2008.
[5] A. Rojas, P. Branch, and G. Armitage, "Experimental validation of the random waypoint mobility model through a real world mobility trace for large geographical areas", in *Proc. 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems*, Montreal, Canada, October 2005.
[6] W. Hsu, K. Merchant, H. Shu, C. Hsu, and A. Helmy, "Weighted Waypoint Mobility Model and its Impact on Ad Hoc Networks", in *Proc. ACM Mobile Computer Communications Review (MC2R)*, Jan 2005.
[7] The Fleetnet Project - Internet on the Road, http://www.et2.tu-harburg.de/fleetnet
[8] The Network on Wheels Project, http://www.network-on-wheels.de
[9] The HWGui Project - Visualizing, Evaluating and Transforming Movement Patterns of Vehicles on Highways, http://www.informatik.uni-mannheim.de/pi4.data/content/projects/hwgui
[10] M. Musolesi and C. Mascolo, "A Community Based Mobility Model for Ad Hoc Network Research" In *Proc. 2nd ACM/SIGMOBILE International Workshop on Multi-hop Ad Hoc Networks: from theory to reality (REALMAN'06)*, Florence, Italy, May 2006.
[11] UDel Models For Simulation of Urban Mobile Wireless Networks, http://udelmodels.eecis.udel.edu/
[12] Q. Zheng, X. Hong, and J. Liu, "An Agenda-based Mobility Model" in *Proc. 39th IEEE Annual Simulation Symposium (ANSS-39-2006)*, Huntsville AL , April 2006.
[13] Community Resource for Archiving Wireless Data At Dartmouth (Crawdad), http://crawdad.cs.dartmouth.edu
[14] Realistic Vehicular Traces http://lst.inf.ethz.ch/ad-hoc/car-traces
[15] UMass Diverse Outdoor Mobile Environment (DOME), http://prisms.cs.umass.edu/dome
[16] The Cabspotting Project, http://cabspotting.org
[17] MIT Media Lab: Reality Mining, http://reality.media.mit.edu
[18] Community-wide Library of Mobility and Wireless Networks Measurements, http://nile.usc.edu/MobiLib

[19] C. Tuduce and T. Gross, "A Mobility Model Based on WLAN Traces and its Validation", in *Proc. 25th Annual IEEE Conference on Computer Communications (INFOCOM'05)*, Miami, March 2005.

[20] J. Yoon, B. D. Noble, M. Liu, and M. Kim, "Building realistic mobility models from coarse-grained traces", in *Proc. ACM International Conference On Mobile Systems, Applications And Services (MobiSys'06)*, pp. 177-190, 2006.

[21] M. Kim, D. Kotz, and S. Kim, "Extracting a Mobility Model from Real User Traces", in *Proc. 26th Annual IEEE Conference on Computer Communications (INFOCOM'06)*, Barcelona, April 2006.

[22] J-K. Lee and J. Hou, "Modeling Steady-state and Transient Behaviors of User Mobility: Formulation, Analysis, and Application", in *Proc. 7th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC'06)*, Florence, May 2006.

[23] A. Chaintreau, P. Hui, C. Diot, R. Gass, and J. Crowcroft, "Impact of Human Mobility on the Design of Opportunistic Forwarding Algorithms" in *Proc. 26th Annual IEEE Conference on Computer Communications (INFOCOM'06)*, Barcelona, April 2006.

[24] V. Srinivasan, M. Motani, and W. T. Ooi, "Analysis and Implications of Student Contact Patterns Derived from Campus Schedules", in *Proc. The 12th ACM International Conference on Mobile Computing and Networking (MOBICOM'06)*, Los Angeles, 2006.

[25] Paramics: Microscopic Traffic Simulation, http://www.paramics-online.com/

[26] CORSIM: Microscopic Traffic Simulation Model, http://www-mctrans.ce.ufl.edu/featured/TSIS/Version5/corsim.htm

[27] ptv simulation - VISSIM, http://www.english.ptv.de/cgi-bin/traffic/traf_vissim.pl

[28] TRANSIMS, transims.tsasa.lanl.gov

[29] VanetMobiSim Project Home Page, http://vanet.eurecom.fr.

[30] J. Härri, F. Filali, and C. Bonnet, "On Meaningful Parameters for Routing in VANETs Urban Environments under Realistic Mobility Patterns", in *Proc. 1st IEEE Workshop on Automotive Networking and Applications (AutoNet'06, in conjunction with IEEE Globecom 2006)*, December 2006.

[31] M. Fiore, J. Härri, F. Filali, and C. Bonnet, "Understanding Vehicular Mobility in Network Simulation", in *Proc. 1st IEEE Workshop on Mobile Vehicular Networks (MoveNet'07, in conjunction with IEEE MASS'07)*, October 2007.

[32] L. Bononi, M. Di Felice, M. Bertini, and E. Croci, "Parallel and Distributed Simulation of Wireless Vehicular Ad Hoc Networks", in *Proc. ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, Torresmolinos, Spain, 2006.

[33] S.Y. Wang and C.L. Chou, "NCTUns Simulator for Wireless Vehicular Ad Hoc Network Research", to appear in *Ad Hoc Networks: New Research*, Nova Science Publishers, 2008. Project website: http://nsl10.csie.nctu.edu.tw.

[34] C. Gorgorin, V. Gradinescu, R. Diaconescu, V. Cristea, and L. Ifode, "An Integrated Vehicular and Network Simulator for Vehicular Ad-Hoc Networks", in *Proc. European Simulation and Modeling Conference (ESM)*, Bonn, Germany, May 2006.

[35] SUMO - Simulation of Urban Mobility, http://sumo.sourceforge.net

[36] F. Karnadi, Z. Mo, and K.-C. Lan, "Rapid Generation of Realistic Mobility Models for VANET", in *Proc. IEEE Wireless Communication and Networking Conference (WCNC'07)*, March 2007.

[37] H. Wu, J. Lee, M. Hunter, R. M. Fujimoto, R. L. Guensler, and J. Ko, "Simulated Vehicle-to-Vehicle Message Propagation Efficiency on Atlanta's I-75 Corridor," in *Transportation Research Board Conference Proceedings*, Washington D.C., 2005.

[38] The Federated Simulations Development Kit (FDK), http://www-static.cc.gatech.edu/computing/pads/fdk.html

[39] Vehicular Grid Computing (VGrid), http://wwwcsif.cs.ucdavis.edu/~VGrid

[40] R. Vuyyuru and K. Oguchi, "Vehicle-to-Vehicle Ad Hoc Communication Protocol Evaluation using Simulation Framework", in *Proc. 4th IEEE/IFIP Wireless On demand Networks and Services*, pp. 100-106, Austria 2007.

[41] Traffic and Network Simulation Environment http://wiki.epfl.ch/trans/

[42] Multiple Simulator Interlinking Environment for C2CC in VANETs, http://www.cn.uni-duesseldorf.de/projects/MSIE

[43] C. Schroth, F. Dötzer, T. Kosch, B. Ostermaier, and M. Strassberger, "Simulating the traffic effects of vehicle-to-vehicle messaging systems", in *Proc. 5th International Conference on ITS Telecommunications*, Brest, France, 2005.

[44] M. Killat, F. Schmidt-Eisenlohr, G. Göbel, T. Kosch, and H. Hartenstein, "On the Accuracy of Coupling a Mobility and a Communication Simulator for VANETs", in *Proc. 4th International Workshop on Intelligent Transportation (WIT)*, pp. 137-142, 2007.

[45] A. Wegener, M. Piorkowski, M. Raya, H. Hellbr"uck, S. Fischer, and J. P. Hubaux, "TraCI: An Interface for Coupling Road Traffic and Network Simulators", in *Proc. 11th Communications and Networking Simulation Symposium (CNS'08)*, Ottawa, Canada, 2008.

[46] M. Brackstone and M. McDonald, "Car-following: a historical review", Transportation Research Part F: Traffic Psychology and Behaviour, Volume 2, Number 4, pp. 181-196, 1999.

[47] S. Panwai and H. Dia, "Comparative Evaluation of Microscopic Car Following Behavior", in *IEEE Trans. Intell. Transport. Syst.*, Vol. 6 No. 3, pp. 314-325, September 2005.

[48] S. Krauss, P. Wagner, and C. Gawron, "Metastable States in a Microscopic Model of Traffic Flow" Physical Review E, volume 55, number 304, pages 55-97; May, 1997.

[49] K. Nagel and M. Schreckenberg, "A cellular automaton model for freeway traffic" Journal de Physic I, vol. 1992, no. 2, pp. 2221-2229, 1992.

[50] R. Wiedemann, "Simulation des Straenverkehrsflusses", in *Schriftenreihe Heft 8, Institute for Transportation Science*, University of Karlsruhe, Germany, 1974.

[51] R. E. Chandler, R. Herman, and E. W. Montroll, "Traffic Dynamics: Studies in Car Following", Operations Research, Vol. 6, No. 2, pp. 165-184, 1958.

[52] P. G. Gipps, "A behavioral car following model for computer simulation", Transportation Research Board 15, pp. 105–111, 1981.

[53] M. Trieber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations", Phys. Rev. E 62, Issue 2, August 2000.

[54] P. G. Gipps, "A Model for the Structure of Lane Changing Decisions", Transportation Research Board 20, pp. 107–120. 1986.

[55] D. Chowdhury, D.E. Wolf, and M. Schreckenberg, "Particle hopping models for two-lane traffic with two kinds of vehicles:Effects of lane-changing rules", Physica A, pp. 235-687, 1997.

[56] K. Ahmed, "Modeling Drivers' Acceleration and Lane Changing Behavior", Ph.D Thesis, MIT Press, USA, 1999.

[57] M. Treiber and D. Helbing, "Realistische Mikrosimulation von Strassenverkehr mit einem einfachen Modell", 16th Symposium Simulationstechnik ASIM 2002, Rostock, September 2002.

[58] R. Wiedemann, "Modeling of RTI-Elements on multi-lane roads", in *Advanced Telematics in Road Transport*, edited by the Commission of the European Community, vol. DG XIII, 1991.

[59] M. Fiore, J. Härri, F. Filali, and C. Bonnet, "Vehicular Mobility Simulation for VANETs", in *Proc. 40th IEEE Annual Simulation Symposium (ANSS-40 2007)*, March 2007.

[60] J. Yin, T. ElBatt, G.Yeung, B. Ryu, and S. Habermas, "Performance Evaluation of Safety Applications over DSRC Vehicular Ad Hoc Networks", in *Proc. 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET)*, October 2004.

[61] H. Wu, R. Fujimoto, R. Guensler, and M. Hunter, "MDDV: Mobility-Centric Data Dissemination Algorithm for Vehicular Networks", in *Proc. 1st ACM International Workshop on Vehicular Ad Hoc Networks (VANET)*, October 2004.

[62] S. Boxill and L. Yu, "An Evaluation of Traffic Simulation Models for Supporting ITS Development", Technical Report 167602-1, Texas Southern University, October 2000.

[63] D. Helbing, "Traffic and Related Self-driven Many-particles Systems", Rev. Modern Physics, Vol. 73, pp. 1067-1141, 2001.

[64] T. Camp, J. Boleng, and V. Davies, "A Survey of Mobility Models for Ad Noc Network Research", Wireless Communications and Mobile Computing, Vol. 2, Issue 5, pp. 483-502, September 2002.

[65] F. Bai and A. Helmy, "A Survey of Mobility Modeling and Analysis in Wireless Ad Hoc Networks", *Book chapter in Wireless Ad Hoc and Sensor Networks, Kluwer academic Publishers*, June 2004.

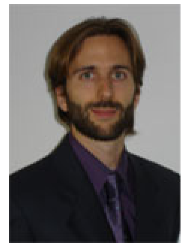
[66] J.-Y. Le Boudec and M. Vojnovic, "Perfect Simulation and Stationarity of a Class of Mobility Models", in *Proc. 25th Annual IEEE Conference on Computer Communications (INFOCOM'05)*, Miami, FL, March 2005.

[67] J. Härri, F. Filali, and C. Bonnet, "A Framework for Mobility Models Generation and its Application to Inter-Vehicular Networks", 3rd IEEE International Workshop on Mobility Management and Wireless Access (MobiWac'05), Maui, Hawaii, U.S.A., June 2005,

[68] The network simulator ns-2, http://www.isi.edu/nsnam/ns.

[69] Q. Chen, F. Schmidt-Eisenlohr, D. Jiang, M. Torrent-Moreno, L. Delgrossi, and H. Hartenstein, "Overhaul of IEEE 802.11 Modeling and Simulation in NS-2", http://dsn.tm.uni-karlsruhe.de/english/Overhaul_NS-2.php

[70] Global Mobile Information Systems Simulation Library, http://pcl.cs.ucla.edu/projects/glomosim.
[71] QualNet Developer http://www.scalable-networks.com
[72] The OPNET Modeler, http://www.opnet.com/products/modeler/home.html.
[73] Scalable Wireless Ad-hoc Network Simulator (SWANS), http://jist.ece.cornell.edu/
[74] The Georgia Tech Network Simulator (GTNetS), http://www.ece.gatech.edu/research/labs/MANIACS/GTNetS/
[75] F. Bai, N. Sadagopan, and A. Helmy, "The IMPORTANT Framework for Analyzing the Impact of Mobility on Performance of Routing for Ad Hoc Networks", *AdHoc Networks Journal - Elsevier Science*, Vol. 1, Issue 4, pp. 383-403, November 2003.
[76] BonnMotion, http://web.informatik.uni-bonn.de/IV/BonnMotion
[77] M.J. Feeley, N.C. Hutchinson, and S. Ray, "Realistic mobility for mobile ad hoc network simulation", in *Lecture Notes in Computer Science (LNCS)*, Vol. 3158, pp. 324-329, 2004.
[78] A. K. Saha and D. Johnson, "Modeling Mobility for Vehicular Ad Hoc Networks", *Poster Session, 1st ACM Workshop on Vehicular Ad Hoc Networks (VANET 2004)*, October 2004.
[79] U.S. Census Bureau - Topologically Integrated Geographic Encoding and Referencing (TIGER) system, http://www.census.gov/geo/www/tiger.
[80] Ertico, http://www.ertico.com/en/links/links/gdf_-_geographic_data_files.htm
[81] ESRI Shape Files, http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf
[82] The Obstacle Mobility Model moment.cs.ucsb.edu/mobility/
[83] The SHIFT Traffic Simulator, http://path.berkeley.edu
[84] D. Choffnes and F. Bustamante, STreet RAndom Waypoint (STRAW), http://www.aqualab.cs.northwestern.edu/projects/STRAW/index.php
[85] GrooveNet: Vehicular Network Virtualization Platform, http://www.andrew.cmu.edu/user/rahulm/Research/GrooveNet.
[86] CANU Project Home Page, http://canu.informatik.uni-stuttgart.de.
[87] Radio Propagation Modeling http://www.ipvs.uni-stuttgart.de/abteilungen/vs/abteilung/mitarbeiter/eigenes/stepania/start/downloads
[88] S. Jaap, M. Bechler, and L. Wolf, "Evaluation of Routing Protocols for Vehicular Ad Hoc Networks in City Traffic Scenarios", in *Proc. 5th International Conference on Intelligent Transportation Systems Telecommunications (ITST)*, June 2005.
[89] A. Mahajan, N. Potnis, K. Gopalan, and A. Wang, "Evaluation of Mobility Models for Vehicular Ad-hoc Network Simulations", *Technical Report N.051220*, Florida State University, 2005.
[90] P. Sommer, R. Baumann, F. Legendre, and B. Plattner, "Generic Mobility Simulation Framework", http://polar9.ethz.ch/gmsf/
[91] H. M. Zimmermann and I. Gruber, "A Voronoi-based Mobility Model for Urban Environments", in *Proc. European Wireless 2005 (EW'05)*, April 2005.
[92] MobiREAL – A Realistic Network Simulator, http://www.mobireal.net
[93] ARTIS: Advanced Runtime Infrastructure, http://pads.cs.unibo.it/dokuwiki/doku.php?id=pads:artis-doc
[94] GPS TrackMaker, http://www.gpstm.com
[95] A. Jardosh, E. Belding-Royer, K. C. Almeroth, and S. Suri, "Toward realistic mobility models for mobile ad hoc networks", in *Proc. 9th Annual International Conference on Mobile Computing and Networking (MobiCom 2003)*, September 2003.
[96] B. Zhou, K. Xu, and M. Gerla "Group and Swarm Mobility Models for Ad Hoc Network Scenarios Using Virtual Tracks", in *Proc. of IEEE MILCOM'04*, 2004.



Jérôme Härri Dr. Jérôme Härri is an assistant professor and leading investigator of the junior research group on Traffic Telematics (VTM) at the Karlsruhe Institute of Technology (KIT). Prior to joining the KIT, he was a research assistant and Ph.D student at EURECOM in Sophia-Antipolis, France working on mobility modelling and management for Mobile Wireless Ad Hoc Networks. He has also been a visiting scholar at Prof. Gerla's network research lab at the University of California in Los Angeles (UCLA) in 2006. His research interests include inter-vehicular communication, vehicular mobility modelling and management and delay critical or tolerant networks. He is currently involved into the activities of the Car-2-Car Communication Consortium (C2C-CC) in the working groups COM and SIM. He is a member of IEEE, IEEE Communications Society and Vehicular Society. He holds a diploma and a doctoral degree in Telecommunication from the Swiss Institute Technology (EPFL), Lausanne, Switzerland.

Fethi Filali Fethi Filali received his Computer Science Engineering and DEA degrees from the National College of Informatics (ENSI) in 1998 and 1999, respectively. At the end of 1999, he joined the Plante research team at INRIA (National research institute in informatics and control) in Sophia-Antipolis to prepare a Ph.D. in Computer Science which he has defended on November 2002. During 2003, he was an ATER (Attach Temporaire d'Enseignement et de Recherche) at the Universit of Nice Sophia-Antipolis (UNSA) and he joined on September 2003 the Mobile Communications department of EURECOM in Sophia-Antipolis as an Assistant Professor. He is/was involved in several French-funded (Dipcast, Constellation, Rhodos, Cosinus, Airnet, WiNEM) and IST FP6/7 (Widens, Newcom, Daidalos, E2R, Multinet, Unite, Chorist, iTETRIS, Newcom++) projects. In the context of some of these projects, he designed and developed an open, flexible and efficient architecture for the support of heterogeneous radio technologies. This architecture was integrated in EURECOM's wireless software-radio platform. His current research interests include WIMAX (802.16)-related communication mechanisms, QoS support in IEEE 802.11-based networks, sensor and actuator networks (SANETs), vehicle adhoc networks (VANETs), routing and TCP performance in wireless networks, etc.. He served as a technical reviewer of several international conferences and journals. Additionally, he is a member of IEEE and IEEE Communications Society. In April 2008, he was awarded the Habilitation Diriger des Recherches from the University of Nice. Sophia-Antipolis for his research on wireless networking.

Christian Bonnet Christian Bonnet joined EURECOM as an associate professor in 1992. Since 1998 he is at the head of the Mobile Communications Department of Eurecom. He has been involved in numerous research projects related to advance mobile networks in the field of QoS and IPv6: (IST) MULTINET, CHORIST, UNITE, DAIDALOS, Moby Dick, (RNRT) COSINUS, AIRNET, SAMU, PLATON, @IRS++.. Christian Bonnet was born in Paris on September 17, 1955. He received an engineering degree from Ecole Nationale des Mines de Nancy in 1978. He first started his career in the domain of programming languages. He participated to the definition of the programming language ADA and the implementation of the first European ADA compiler within the ALSYS Company. He joined GSI-TECSI in 1983 as a consultant. He worked on different projects related to radio communications systems, value added networks and real time systems. In 1987 he was appointed Director of the Real Time Department of GSI Tecsi and Responsible for the R&D European IST projects (DESCARTES, DRAGON, REX.). He co-authored more than 100 publications in international conferences and magazines.