MOBILITY MODELS IN VEHICULAR AD HOC NETWORKS: THE OVERTAKING IMPACT

D. Djenouri [*], W. Soualhi, E. Nekka
CERIST Center of Research, Ben-aknoun, BP 163, Algiers 16030, Algeria

ABSTRACT

Vehicular Ad hoc Networks (VANETs) are basically characterized by the high mobility of nodes, making the mobility model one of the most important parameters that should be carefully selected when evaluating any protocol. To get accurate results, the model should be as realistic as possible, and involve road-maps with all the constraints and facilities related to the vehicular movement. In this paper we first survey some mobility models that are relevant to VANETs, then we present our vehicular mobility simulator that allows to generate mobility trace files for both GloMoSim and ns2. Finally, we use our simulator together with GloMoSim to investigate the effect of some realistic parameters, namely the usage of a road map vs. movement in an open area, and especially the effect of the overtaking, one of the facilities enabled by many roads that has never been considered in literature.

Keywords

Vehicular Ad hoc Networks, Mobile Networks, Mobility Models, Mobility Simulation.

1. INTRODUCTION

One of the emerging applications of mobile Ad hoc Networking (MANET) is Vehicular Ad hoc Networking (VANET). In VANET, communications between nodes are direct, i.e. without relying on any dedicated infrastructure, contrary to earlier vehicular networks [1]. Although it is self-organized and easily to deploy, this infrastructureless network introduces many challenges that should be tackled before real implementations. For instance, to enable communications between nodes (vehicles) which are out of the power range of each other, some other intermediaries should act as routers to remedy the lack of dedicated routers. Thus, a distributed routing protocol needs to be employed. It is mandatory before passing to the real employment of a routing protocol (ass well as any other protocol or application) to evaluate it by simulation. The faithfulness of the simulation results is proportional to the realism of the parameters and the models used in the simulation. Particularly, it hugely depends on the accuracy of the mobility model to use. This latter can be defined as the pattern that establishes the nodes movement within a given area (termed the simulation area) during a simulation. The earlier mobility models used in MANET simulation assume the terrain to be without obstacles, and nodes to be able to move freely in the whole rectangular stimulation area. Random way-point (RWP) [2] is a typical example of such a kind of models, which is largely used in the literature, and available in many network simulators (such as ns2 and GloMoSim). This model defines the pause-time parameter, so that each node has phases of movement and others of pause. At the beginning, the node selects randomly and uniformly a destination towards it moves using a random and constant speed, which is also selected for each movement following a uniform distribution. Once it reaches this destination it stays there for the pause-time duration, then repeats the process. It has been illustrated that this model engenders, after a given simulation time, a spatial distribution of nodes concentrated around the center of the terrain, which is not uniform [3, 4].

Generally speaking, the assumption of an open terrain may be realistic for some applications of pedestrians, but it is inappropriate for VANET. More recent studies propose new models constrained to roads and obstacles, and thus are more suitable for VANET. Except the model of Gorgorin et al. [5], no model implements the possibility of overtaking, i.e. the possibility that a quick vehicle overtakes the in front vehicle(s) moving in a lower speed when overtaking is enabled and possible in a road segment. We think this should have a great effect on the network topology and should be considered. None of the previous work, including the one of Gorgorin et al. [5], investigated this issue. This latter just presents the proposed model and its implementation in a new simulator, but no simulation study has been driven. Our main contribution in this paper is the investigation of the overtaking impact upon the mobility model realism in VANET.

The rest of the paper is organized as follows: in the next section we present the related work, followed by a state of the art on the VANET mobility models. In Section 4, we describe our vehicular traffic simulator, followed in Section 5 by the simulation study performed using our simulator together with GloMoSim. Finally, we conclude the paper and sketch the open perspectives in the last section.

2. RELATED WORK

In [6] Camp et al. discuss a variety of mobility models used to evaluate ad hoc networks, and split them up into two categories: entity models and group models. The authors show by simulation how the choice of a mobility model can have a significant effect on the performance results. However, except city section [7], all the other models presented in this survey

---

*Corresponding author: *E-mail:* ddjenouri@mail.cerist.dz

ISSN: 1744-2397

are inappropriate for VANETs. In [3, 8], the authors classify the models according to the randomness of speeds and directions, and divide them into i) trace-based (deterministic) models, ii) constrained topology based models iii) and statistical (fully random) models. Basing on this classification, [8] provides a more recent survey of MANET's mobility models, presenting some ones that consider obstacles in the simulation area. Still, these models are suitable to simulate pedestrian movements, but not vehicles. In [9], the authors provide a mathematical and simulation investigations into some of the statistical (random) models, and combine them in the termed random trip. The most recent research efforts on mobility models focus on the vehicular ad hoc network application of MANET. In [10], the authors provide a general survey of VANET, and the existing challenges to overcome before the real deployment. In [1], the authors discuss the usefulness of VANET to ensure the vehicular traffic safety and facilities, as well as the advantages it provides compared to the other centralized technologies. Some specified applications have been proposed, such as the discovery of free parking places [11]. Regarding the mobility models, some new ones have been specifically proposed for VANET, such as [5, 12-15]. In this paper, we first present and discuss these new models, as well as those proposed for the general MANET that applies to VANET. We also present a new vehicular traffic simulator we implemented to generate movement trace files usable by some well-known network simulators, notably GloMoSim [16] and ns2 [17], then we use this tool along with GloMoSim to conduct a simulation comparative study. Our main contribution here is the investigation into the overtaking impact on the network performance.

3. VANET MOBILITY MODELS

In the following we review the models that were either specially devoted to VANET, or proposed in the general MANET context but usable in VANET. The criteria of applicability we consider here is the employment of road maps, and thus limiting the nodes movements into the roads, instead of moving them in a wide open area. As we will see, the considered parameters differ from a model to another. For instance, some models use traffic control mechanisms (stop signs and/or traffic lights) at intersections, and some others just assume continuous movement at these points. Some assume roads to be single-lane, but some others support multi-lanes roads. Some define the security distance, while others just ignore this parameter.

3.1 Freeway

Freeway is a generated-map-based model, defined in [18]. The simulation area, represented by a generated map, includes many freeways, each side of which is composed of many lanes. No urban roads, thus no intersections are considered in this model. At the beginning of the simulation, the nodes are randomly placed on the lanes, and move using history-based speeds, where the speed of each vehicle smoothly changes following a random acceleration. In addition to the realism related to the acceleration and the history based speed, the model defines a security distance that should be maintained between two subsequent vehicles in a lane. If the distance between two vehicles is less than this required minimal distance, the second one decelerates (its acceleration is forced to be negative) to enable the forward vehicle moving away. The change of lanes is not allowed in this model. The vehicle moves on the lane it is placed in until reaching the simulation area limit, then it is placed again randomly in another position and repeats the process. This scenario is definitely unrealistic.

3.2 Manhattan

This is also a generated-map-based model, introduced in [18] to simulate an urban environment.

Before starting a simulation, a map containing vertical and horizontal roads is generated. Each of these latter includes two lanes, allowing the movement in the two directions (north/south for the vertical roads and east/west for the horizontal ones). At the beginning of a simulation, vehicles are randomly put on the roads. Afterwards, they move continuously according to history-based speeds (exactly like Freeway). When reaching a crossroads, the vehicle randomly chooses a direction to follow. That is, continuing straight forward, turning left, or turning right. The probability of each decision are set by the authors respectively to 0.5, 0.25, 0.25. The security distance is also used in this model, and nodes follow the same strategy as in the freeway model to maintain this distance. But contrary to the previous model, a vehicle can change a lane at a crossroads. Nonetheless, there is no control mechanism at these points (crossroads), where nodes continue their movements without stopping, which is unrealistic.

3.3 City Section Mobility (CSM)

CSM [7] can be viewed as a hybrid model between RWP and Manhattan, as it introduces the principle of RWP, especially the pause-time and random selection destination, within a generated-map-based urban area. At each step of the vehicle's movement a random point is selected from the generated road map, towards which it moves following the shortest path. After reaching that destination, it remains there for a pause-time, then repeats the process. The speed of nodes are constrained by the security distance, along with the maximum speed limit of the road.

3.4 Rice University Model (RUM)

Thus far, we have presented models based on virtual generated maps. RUM [12] is very similar to CSM, but indeed it uses real maps obtained from the TIGER/Lines database [12]. For each road segment (a part of the road lying between two crossroads), the coordinates are extracted and converted using the Mercator projection [12]. The extracted points are then presented by a graph, where the crossroads are presented by vertices, and roads by weighted arcs. The weight of each arc is dynamically calculated in such a way to mimic the estimated time required for a vehicle to move over the corresponding segment, which is proportional to its maximum authorized speed, its distance, and the number of vehicles it currently contains. Therefore, the lower the weight, more the vehicles move freely in the segment. Note that the maximum authorized speed of a road segment depends on its type. Finally, we mention that like all the previous models, RUM defines no control mechanisms at crossroads.

3.5 Stop Sign Model (SSM)

Contrary to the previous models, SSM [14] integrates a traffic control mechanism. In every crossroads, a stop signal is put, which obliges vehicles to slow down and make a pause there. This model is based on real maps of the TIGER/Lines database, but all roads are assigned a single lane in each direction. A vehicle should never overtake its successor (like in all the models presented before), and should tune its speed to keep the security distance. If many vehicles arrive at an intersection at the same time, they make a queue, and each one waits for its successor to traverse the crossroads. This results in gathering of nodes, and hugely affects the network connectivity as well as the mobility (average speeds). According to the authors [14], the problem with this model is the unrealistic disposition of the stop signals, since it is impossible to find a region with stop signals at each intersection. Therefore, they proposed TSM [14], we describe hereafter.

3.6 Traffic Sign Model (TSM)

In this model, stop signals are replaced by traffic lights. A vehicle stops at a crossroads if it encounters a red stoplight, otherwise it continues its movement. When the first vehicle reaches the intersection, the light is randomly turned red with probability p (thus turned green with probability $1-p$). If it turns red, then it remains so for a random delay (pause-time) forcing the vehicle to stop, as well as the ones behind it. After the delay, it turns green then the nodes traverse the crossroads one after the other until the queue is empty. When the next vehicle arrives at the crossroads the process is repeated. TSM and SSM has been evaluated by simulation with ns2[17]. The results show that the two models are not significantly influenced by the speed of nodes (maximum speeds), due to the traffic control models that slow down the nodes and give more stability to the network [14]. When increasing the pause-time at the intersections, the authors remarked that the performances improved for both models, and that SSM gives better results than TSM when using the same pause-time. The authors argue this by the fact that in SSM nodes always stop at the intersections, unlike TSM. Nevertheless, in reality the pause-time for stop signals is shorter than the one of traffic lights, which makes TSM more stable indeed [14].

3.7 STRAW

STRAW [13] is also a model using real maps of TIGER/Line. Like the other models (except freeway), the roads include one lane in each direction, and is divided into segments. The model is basically composed of three modules: intra-segment mobility manager, inter-segment mobility manager, and finally the rout management and execution module. At the beginning of the simulation the nodes are placed randomly one behind the other, then move using the car following model [13] and try to accelerate until reaching the maximum speed of the segment. The first module manages this movement until reaching an intersection. The security distance is maintained, but the overtaking is not allowed. At crossroads the vehicles always slow down, even when they change a segment and turn without a full stop, which is realistic. The second module defines the traffic control mechanism including both stop signals and traffic lights, which are put on crossroads according to the class of the intersected roads. In addition to this usual control form, the module makes sure that the next segment to take contains enough available space before moving the vehicle towards it. If it is fully busy, the vehicle waits at the crossroads (at the end of the first segment). The last module selects the routes to be taken by each vehicle during the simulation. It implements two approaches: simple straw and straw OD. In the first one, the direction is randomly selected at each intersection, i.e. when reaching an intersection, the vehicle randomly decides whether to continue straight forward or to turn and change the road. On the other hand, in the second approach a destination is selected toward which the vehicle moves using the shortest path. The simulation study made by the authors [13] show that when using STRAW the reception ratio decreases from 43% up to 53% compared to movements in an open area. The results of this simulation also illustrate that the roads arrangement has an impact; scenarios with a high number of crossroads slow down the average speeds of nodes, which improves the reception ratio.

3.8 MOVE

MOVE [15] is a VANET's mobility model that uses as the compiler SUMO [19], which is a realistic vehicular traffics simulation model. SUMO is an open source application implemented with java, that integrates many realistic parameters such as realistic accelerations, the usage of real maps reflecting several types of roads (with multiple lanes), and traffic lights defining priorities between vehicles. Basically, MOVE is composed of two components; the road map editor and the vehicle movement editor. The former serves to manually and randomly generate a road map, either from TIGER/line files or Google earth files, whereas the latter allows to specify the properties of each vehicle, like the maximum speed, the acceleration, the probability of turning at crossroads, and the path to take. The information collected by the two editors are sent to the SUMO compiler, then a trace file in ns-2 or Qualnet format is generated. MOVE has been compared by simulation to RWP using AODV. The results show that MOVE causes a low reception rate.

3.9 Gorgorin et al. model

In addition to all the realistic parameters of the previous models, this one [5] implements an overtaking mechanism within multi-lane segments. A vehicle always tries to move on the most right lane (the lowest rang), except in case of overtaking during which it moves left, and at intersections in urban environments where it chooses the lane according to the next direction. A hierarchy of vehicle states is defined, respectively: free driving, approaching, following, and braking (in that order). When a vehicle is in another state than the free driving, it checks whether higher lanes allow it to pass to a higher state, and thus moves to the left lane to make an overtaking. Identically, a vehicle in a state different than braking checks whether the right lane allows it to at least stay in the same state and then moves right. Moreover, the model allows to specify the driver type, which affects many parameters of the vehicle (speed, acceleration, and others). Finally, note that the model includes both traffic lights and stop signals at intersections. One of these two different control mechanisms is put at each intersections according to the types of the intersecting segments. The most important parameter added in this model is the overtaking mechanism. However, no study investigating this issue has been done yet. Table 1 summarizes the features of all the models presented in this section.

Note that for space limitation"ss" stands for stop signs, and "tl" for traffic lights.

4. NEW MOBILITY SIMULATOR

We developed a multi-platform vehicular mobility simulator with the Microsoft Visual studio^{TM}.Net environment using C# programming language. It allows to simulate movements of vehicles within a selected road map, visualize the movements in realtime, and generate mobility trace files to be used by network simulators. The tool provides many visual facilities, and the mobility model implemented is parameterizable in such away to including many of the models presented in the previous section. Through the simulator visual interface, many parameters related to both the mobility model and the scenario to simulate are tunable. Hereafter, we illustrate the most important parameters.

4.1 Parameters related to the model

The mobility model we implemented can be considered as a mould of models, that can be shaped to simulate several models. This is by fixing the following parameters:

1) The use of either the random method or the shortest path method to generate the route to follow. This enables defining the inter-segment mobility, also known as macro mobility.

2) The possibility of overtaking: could be either enabled or disabled. Once enabled, a vehicle moving in a higher speed than another one ahead can overtake it whenever possible in multi-lane roads, as depicted in Fig. 1.

Table 1. VANET's mobility models features

Feature \ Model	Freeway	Manhattan	CSM	RUM	SSM	TSM	STRAW	MOVE	Gorgorin
Real maps	no	no	yes	yes	yes	yes	yes	yes	yes
# lanes/direction	many	one	one	one	one	one	one	many	many
Intersections	no	yes	yes	yes	yes	yes	yes	yes	yes
Changing lanes at intersections	no	yes	yes	yes	yes	yes	yes	yes	yes
Traffic control	no	no	no	no	ss	t1	both	t1 + priority	both
Overtaking	no	no	no	no	no	no	no	no	yes
Security distance	yes	yes	yes	no	yes	yes	yes	yes	yes
Pause-time	no	no	yes	no	yes	yes	yes	yes	yes

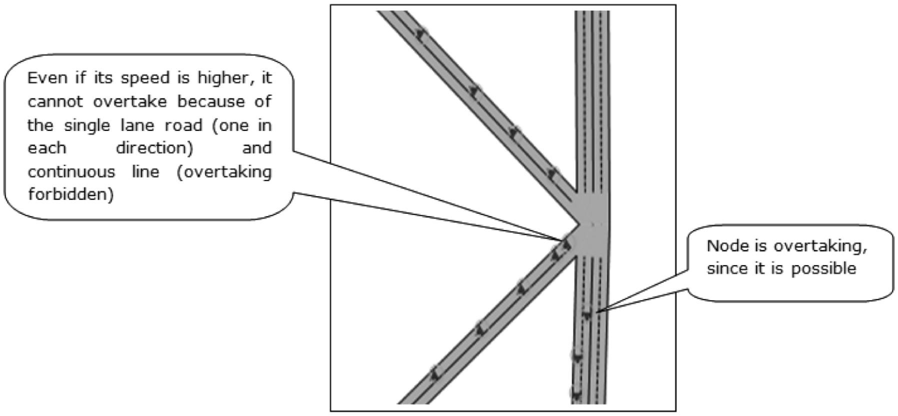


Figure 1. Overtaking Examples

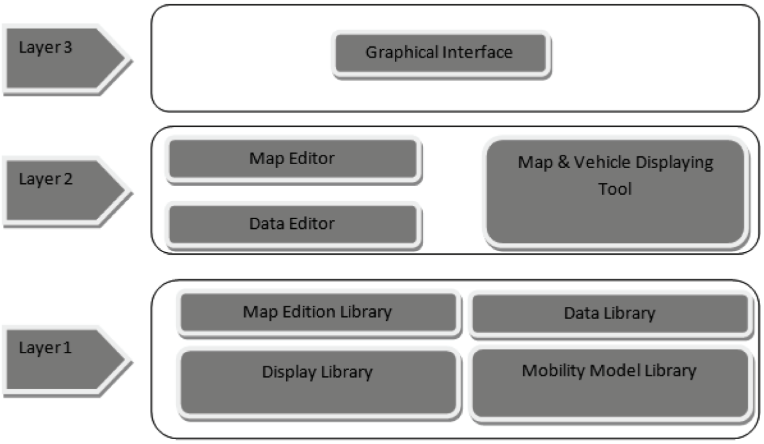


Figure 2. Application Frame Work

3) The use of traffic lights: For now, the implemented model includes only traffic lights, no stop signs. The employment of traffic light at intersections could be selected, and the user can define pause-time duration when encountering red light for each type of segment intersections. A type of intersection is defined by the combination of the segments forming it, e.g intersection of three segments of type 1, two segments of type 1 and one of type 2, three of type 2, and so on. The segment type is proportional to the number of lanes it includes in each direction (e.g. type1 segment includes one lane in each direction, type 2 two lanes, and so on). Our simulator supports the commonest existing types (up to three-lane segments), and is easily extendable to support four-lane and five-lane segments.

4) The pause-time after each movement.

This way, many models can be simulated. For instance, when activating shortest path and traffic lights without overtaking we can get a TSM-like model. The RUM model can be fixed by deactivating traffic lights and set pause-time to 0. In addition to the previous values, using a grill map allows to simulate Manhattan, and a map without intersections including only multi-lane segments to simulate freeway. But note that models that use stop signs, such as STRAW, are not supported thus far.

4.2 Parameters related to scenarios

The other parameters related to the setting of a scenario are:

1) The type of the map: The map to be used and its size could be either a real one picked from Tiger/Line database, or a generic one. For the latter it can be drawn using a drawing tool included in our simulator, allowing to stack a real map that can be obtained from google earth and to draw the road map to be used over it.

2) Number of nodes.

3) Simulation time.

4) Maximum speed authorized on each type of segments.

5) Discreetization time: the time unite that represents the interval between two events (node positions) in the simulator; it is trivial that the less the value the more precession and realism we get, but this would require more time and resources.

Once the parameters are fixed, the simulation can be launched. The user can choose to visualize the simulation to see the movements of nodes, along with their IDs, speeds, and accelerations in realtime. This is helpful for tuning the value of some parameters, but would require a great deal of time. Once all the parameters are fixed, the user can choose to directly generate the mobility trace file without visualization. The user also has the possibility to specify the format of the output (trace) file. For the time being, the simulator supports GloMoSim and ns2 formats.

4.3 Application Framework

Our application is composed of three layers, as depicted in Fig. 2. The graphical interface layer, or layer 3, is the interface through which the user interact with the application, thus we tried to make it as convenient and user-friendly as possible. It includes two modes: the drawing mode in which the user can draw a road map, possibly using real image (of google earth) on which he/she stacks the map to draw, then save the map in the TIGER/Line format. Our graphical interface also provides the potability to edit an existing map in TIGER/Lines to graphically modify it. The second mode is the simulation mode in which the user defines the parameters' values and launches the simulation. He can choose whether to display or not the movements, speeds, and vehicle numbers. In all cases he will get the trace file for the selected network simulator. The second layer is the gateway between the graphical interface and the libraries of layer 1. It is composed of: i) the map editor, for map drawing and editing, ii) data editor, through which the user can define the different parameters given in the previous section, as well as many others such as the map coordinates and dimensions, and displaying options, iii) the map & vehicle displayer is the last component that is responsible for displaying the map and the vehicles during the simulation, following the user specifications provided by the previous component. The last layer (layer one) is the kernel of our application that includes four libraries defining all the classes the other layers use. The first library is map edition library, containing the class (clTraceSeg), which is in charge of the addition/deletion of segments, as well as the loading/saving of maps in TIGER/Lines format. The second one is the data library that includes clCarte, the class containing all the data structures of our application, as well as methods facilitating the access and the modification of these structures. The third library is of the mobility models, composed of several classes implementing parameters related to them. The last library is the display library, which encapsulates classes managing the map and vehicles display, with the specified options (speeds, accelerations, IDs, and so on)

5. SIMULATION RESULTS AND DISCUSSIONS

5.1 Simulation Setup and Metrics

We have driven a simulation study using our mobility simulator together with GloMoSim. The former has been used to generate mobility trace files, used by the latter which is a well-known network simulator [16]. The aim is to show the impact of using a road map versus simulating movements in an open area, and more importantly to investigate how the overtaking influences the mobility and thus the simulation results. Therefore, we compared three models: i) random way point (RWP), which generates movements in open area, ii) TSM like model, which implements almost all the realistic vehicular movement constraints and facilities except the overtaking, and finally ii) a model much similar to the one of Gorgorin et al., enabling overtaking whenever possible. We call the last model VanetSim, as it looks like the most realistic compared to the other ones. We varied the number of nodes, the maximum speed, and the network traffic load, and measured the following metrics.

5.1.1 The relative mobility

This metric was defined by Larsson et al. [20]. It measures the average changes in distances between nodes, instead of the changes from a fixed terrestrial point as commonly used when expressing speeds. This definition more accurately reflects the network topological change, as it considers booth speeds and directions. It is given by the following formula:

$$Mob = \sum_{i=1}^{n} \frac{M_i}{n}$$

$$M_x = \sum_{t=0}^{T-\Delta t} \frac{\left| A_x(t) - A_x(t+\Delta t) \right|}{T}$$

$$A_x(t) = \sum_{i=1}^{n} \frac{dist(n_x, n_i)}{n-1}$$

where:

$dist(n_x, n_i)$: the distance between nodes n_x and n_i.

n: number of nodes.

$A_x(t)$: the average distance between node x and all the other nodes, at time t.

M_x: the average relative mobility of node x regarding all other nodes, during the simulation time T.

T: simulation time

Δt : time period used in computation.

After each Δt, $A_x(t)$ is calculated, i.e. it is calculated for: $t = 0$, $t = \Delta t$, $t = 2\Delta t, \ldots$, $t = T$. Measuring this metric allows to show the effect of the model on the relative mobility, thus the network topology. The next metrics will illustrate the effects of such a mobility vis-`a-vis the network performance.

5.1.2 Delivery ratio

It is the number of received packets divided by the number of sent packets, computed at the application layer level. It reflects the reliability which is a very important issue. High values of this parameter reflects a good reliability, i.e. low packet loss.

5.1.3 End-to-end delay

It is the average time separating the data packets sending from source nodes and their arriving at destination ones, at the application layer level. This metric is very important to study the quality of service, especially for realtime applications. These two performance metrics are the most relevant that could be influenced by the mobility. Energy consumption could also be affected, but we think it has no importance in VANETs, since nodes are usually supplied with relatively highly autonomous resources (vehicle batteries).

Table 2. Simulation fixed parameters

Parameter	Value
Simulation time	900s
Power-range	250m
Propagation model	Free-space
Simulation terrain	$800 \times 800\text{m}^2$
Routing protocol	AODV
MAC protocol	IEEE802.11
Discretization time	100ms

The fixed parameters of our scenarios are depicted in Table 2. We just point out that the terrain is without obstacles for RWP, and for the other models we generated different road segments of different type within this area, including intersections and traffic light at each intersection. Also, note that in addition to 900 simulation time, in the mobility simulation phase we used an overheating period of 1000s, during which nodes move from their initial positions without taking any measurements. In the next section, the simulation results will be presented.

5.2 Results

Fig. 3a shows the relative mobility vs. the maximum effective speed used in the simulation, expressed as the rate of the maximum allowed speed at each segment. We used 130*km/hr* for three-lane segments, 90*km/hr* for segments of two lanes, and 50*km/hr* for single-lane segments. Hence, the maximum speed a vehicle can reach in our scenarios is a ratio of these values, e.g. for measurements of 15% the maximum speeds are respectively 19.5*km/hr*, 13.5*km/hr*, and 7.5*km/hr*. The plots of Fig. 3a clearly illustrate the difference in term of relative mobility between a model enabling the overtaking (Vanet-Sim) and the one that does not allow this possibility (TSM-like). The values of both models increase with the maximum speed, and more importantly the difference between the two models also raises the speed, which is due to the overtaking. The same metric vs. the number of nodes is depicted in Fig. 3b. We remark here that the mobility goes down while the nodes number goes up. This can be argued by the growing in node density that causes higher load on the vehicular traffic and more queueing delays at intersections (trafficjam). The most important issue to notice is the huge difference between the two models. The two figures show clearly that the overtaking has a dramatic affect on the mobility. In the following we investigate the impact of this mobility on the performance metrics. We fix the maximum speed to 100% and vary the number of nodes as well as the number of CBR sources (traffic).

From Fig 4a and 4b, we remark that the reception ratio of VanetSim is the worst, and RWP is the best. Overall, this metric rises with nodes number and decreases with the load. The increase in nodes number increases the connectivity, which improves the delivery. On the other hand, the increase in load engenders high collisions, thus high packet loss. The worst results of VanetSim are basically due to the overtaking, while the best results of RWP are owing to the nonuniform distribution of nodes and their concentration at the center of the simulation area, causing high node density [4].

Like the delivery ratio, Fig. 5a and 5b confirm the effects of overtaking and road maps usage. Identically, RWP has the lowest (best) delay,a nd VanetSim the highest one. We also remark from Fig. 5b that the delay rises with the network load. This is due to the increase in collisions, which rises the number of packets to be retransmitted before arriving to their final destination.

5.3 Discussions

The comparison between RWP that operates in open space areas and the two other road-map models shows that using road-map movement, which is clearly more realistic in VANET than open space movement, has an impact and considerably reduces the performances, confirming thus the results reported in literature [13]. More importantly, when comparing the overtaking-enabling model (Vanet-Sim) with TSM-like model that does not permit the overtaking and indeed differs from the first one only with respect to this parameter, we clearly see a dramatic gap in relative mobility between the two models. The consequences of this difference caused by overtaking have been quantitatively assessed through two important parameters, say the end-to-end delay and the packet delivery ratio. The results illustrate a considerable decline in performances due to overtaking, which caused lower delivery ratio and higher delay. To summarize, our simulation results revealed the impact the overtaking can have on the simulation results, confirming that this realistic possibility should not be ignored as done in all the simulation studies reported in literature until now.

6. CONCLUSIONS

In this paper our contribution was three folds. First, we reviewed the current mobility models of vehicular ad hoc networks and provided an up-todate survey. We remarked through this study that the overtaking possibility of vehicles in multi-lanes roads was ignored, and in all the simulations the authors just assume that a node's speed is constrained to the speed of the following vehicle. We then presented in the second part the mobility simulator we developed, which is multi-plat-form and integrates a variety of realistic parameters of vehicular movements, enabling to simulate many models and generating trace files to be used by landmark network simulators (for now GloMoSim and ns2). This can provide a useful tool to the scientific community intersected in any field of VANET, or vehicular networks in general, involving a simulation study. Finally, we used our simulator together with GloMoSim to compare the employment of a road map vs. the movement in an open area, and especially to reveal the impact of the overtaking on the network performance, which represents our main contribution. The simulation study consists of comparing three models: RWP that is an open-area basic model, a TSM-like model, and finally a model that integrates almost all the realistic features, and differers from TSM merely by the fact that it enables the overtaking. The results illustrate that the employment of road-maps, as well as enabling the overtaking in multi-lane road segments whenever possible have huge affects on the performance results. Therefore, the overtaking possibility should carefully be considered when evaluating any protocol or applications in VANET, such that to obtain more faithful results. Some perspectives arise from this work such as: i) the integration of stop-signs to the simulator, along with the priority option, ii) the integration of one-way roads (for urban environment), and eightlanes (four in each direction) or even more to enable the simulation of some existing highways, and finally iii) extending our simulator to a parallel version and to other mobility trace file formats of other network simulators.

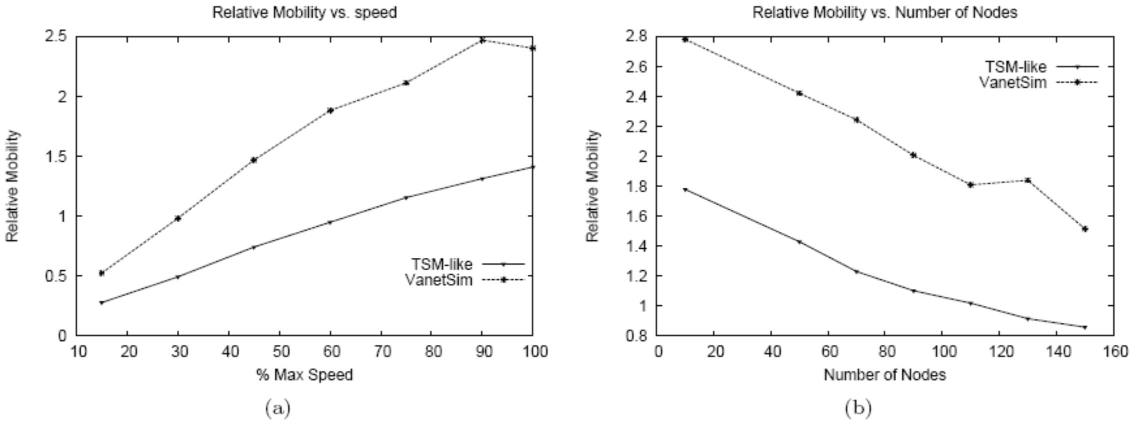


Figure 3. Relative Mobility

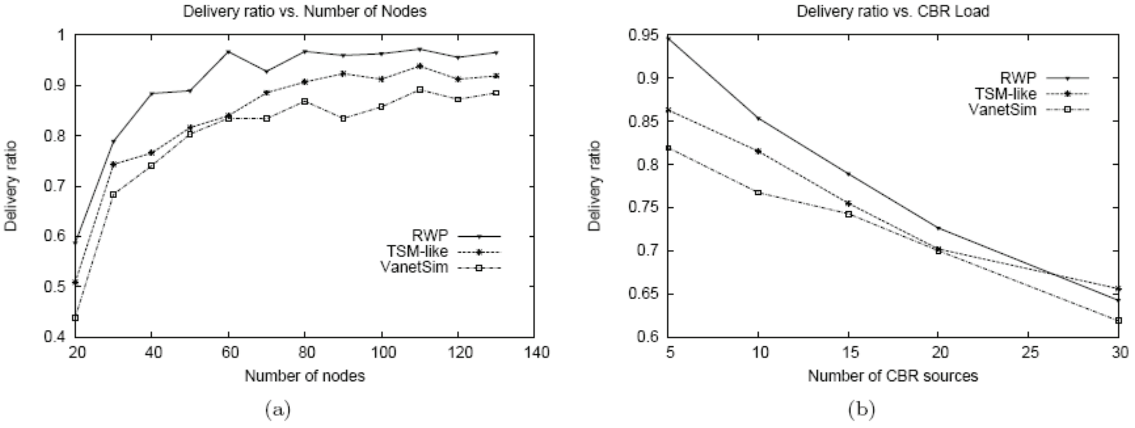
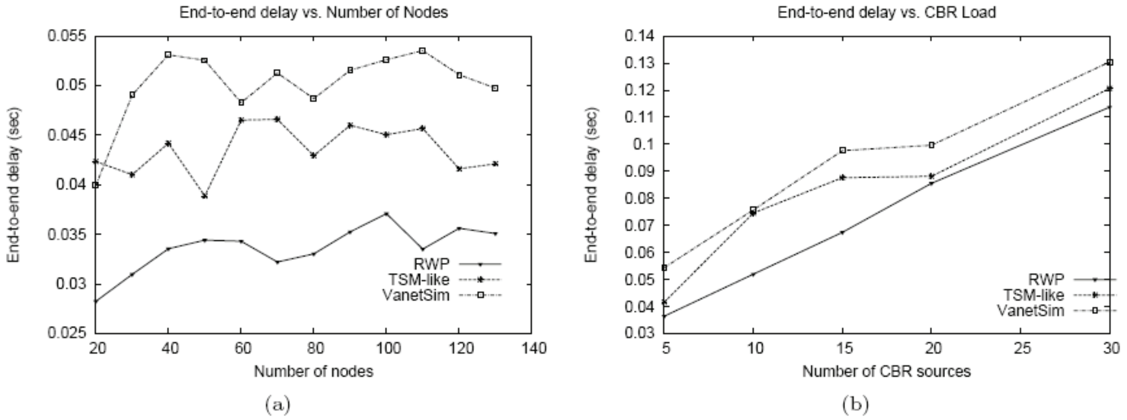


Figure 4. Delivery Ratio



(a) (b)

Figure 5. End-to-end Delay

REFERENCES

[1] I. Chisalita, N. Shahmehri, "Vehicular communication: A candidate technology for traffic safety," in Proc. IEEE International Conference on Systems, Man and Cybernetics, October 2004, pp. 3903-3908.

[2] B. David, A. David, "Dynamic source routing in adhoc wireless networks", in Mobile Computing, Kluwer Academic, 1996, Vol. 35, pp. 153-181.

[3] C. Bettstetter, "Smooth is better than sharp: a random mobility model for simulation of wireless networks", in Proc. 4th ACM international workshop on Modeling, analysis and simulation of wireless and mobile systems (MSWIM '01), ACM Press, 2001, pp. 19-27.

[4] C. Bettstetter, G. Resta, P. Santi, "The node distribution of the random way point mobility model for wireless ad hoc networks", IEEE Transactions on Mobile Computing, Vol. 2, No. 3, pp. 257-269, 2003.

[5] C. Gorgorin, V. Gradinescu, R. Diaconescu, V.Cristea, L. Ifode, "An integrated vehicular and network simulator for vehicular ad-hoc networks", in Proc. 20th European Simulation and Modelling Conference, May 2006.

[6] T. Camp, J. Boleng, V. Davies, "A survey of mobility models for ad hoc network research", Wireless Communications and Mobile Computing (WCMC): Special issue on Mobile Ad Hoc Networking: Research, Trends and Applications, Wiley and sons Publisher, Vol. 2, No. 5, 2002, pp. 483-502.

[7] V. Davies,"Evaluatingmobility modelswithin anadhoc network", Colorado School of Mines, Colorado, USA, Tech. Rep., 2000.

[8] Q. Zheng, X. Hong, S. Ray, "Recent advances in mobility modeling for mobile ad hoc network research", in Proc. 42nd ACM annual Southeast regional conference (ACM-SE). ACM Press, 2004, pp. 70-75.

[9] J. Y. L. Boudec, M. Vojnovic, "Perfect simulation and stationarity of a class of mobility models", in Proc. 24th IEEE Annual Joint Conference on Computer Communications and Networking INFOCOM'05, 2005, pp. 2743-2754.

[10] J. Luo, J. P. Hubaux, "A survey of inter-vehicle communication", School of computer and Communication Sciences, EPEL, Tech. Rep. IC/2004/24, 2004.

[11] M. Caliskan, D. Graupner, M. Mauve, "Decentralized discovery of free parking places", in Proc. 3rd International ACM Workshop on Vehicular ad hoc networks, VANET'06. ACM Press, 2006, pp. 30-39.

[12] D. B. Johnson, A. K. Saha, "Modeling mobility for vehicular ad-hoc networks", inProc. 1st ACM international workshop on Vehicular ad hoc networks, ACM Press, 2004, pp. 91-92.

[13] D. R. Choffnes, F. E. Bustamante, "An integrated mobility and traffic model for vehicular wireless networks", in Proc. 2nd ACM international workshop on Vehicular ad hoc networks, VANET'05, ACM Press, 2005, pp. 69-78.

[14] A. Mahajan, N. Potnis, K. Gopalan, A. I. A. Wang, "Urban mobility models for vanets", in Proc. 2nd IEEE International Workshop on Next Generation Wireless Networks, December 2006.

[15] F. K. Karnadi, Z. H. Mo, K. chan Lan, "Rapidgeneration of realistic mobility models for vanet", in Proc. IEEE Wireless Communications and Networking Conference, 2007, pp. 2506-2511.

[16] X. Zeng, R. Bagrodia, M. Gerla, "Glomosim: A library for the parallel simulation of large-scale wireless networks", in Proc. 12th Workshop on Parallel and distributed Simulation. PADS'98, Banff, Alberta, Canada, May 1998, pp. 154-161.

[17] "The Network Simulator: ns2", Available at http://www.isi.edu/nsnam/ns/, last accessed date: 30/10/2007

[18] F. Bai, N. Sadagopan, A. Helmy, "Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for ad hoc networks", in Proc. 22th IEEE Annual Joint Conference on Computer Communications and Networking INFOCOM'03, 2003, pp. 825-835.

[19] "SUMO Simulation of Urban MObility", Available at http://sumo.sourceforge.net/, last accessed date: 30/10/2007.

[20] P. Johansson, T. Larsson, N. Hedman, "Scenario-based performance analysis of routing protocols for mobile ad-hoc networks", in Proc. 5th Annual ACM/IEEE International

Conference on Mobile Computing and Networking (MOBICOM). Seattle, Washington, USA, 1999, pp. 15-19.

Biographies

Djamel Djenouri obtained the engineering degree in computer science, the master degree in computer science, and finally the PhD in computer science from the University of Science and Technology USTHB (Algiers) respectively in 2001, 2003, and 2007. During his PhD, he visited the technical university of Compienge, France, and the John Moors university of Liverpool, UK. He also participated in many international conferences and published some journal papers. He works mainly on ad hoc networking, especially on the following topics: security, power management, routing protocols, MAC protocols, vehicular networks, and sensor networks. He is currently a research assistant at the CERIST center of research in Algiers. He was an ACM student member, and won the ANDRU's best paper awards of the year 2008, supported by the Algerian ministry of higher eduction and scientific research.

Wassim Soualhi is a computer engineer, graduated from INI, Algiers, Algeria in June 2007, after working during his last year of graduation at CERIST on a research project dealing with vehicular mobility models in VANET, during which he participated in the development of the mobility simulator and the simulation analysis described in this paper. He was a member of INI for US, with which he participated in the Microsoft Imagine Cup competition and won the first national part of the contest in 2006, allowing him to reach the international competition that took place in Agra (India). Currently, he is working towards his master in Networking Technology at The National High School of Telecommunication (TelecomParistech) in France. He is also in a part-time placement at Alcatel-lucent France Corporation, where he is working on hi-tech solution integration, such as WiMAX, IMS, IPTV and MobileTV.

Nekka El Malik is a computer engineer, graduated from INI, Algiers, Algeria in June 2007, after working during his last year of graduation at CERIST on a research project dealing with vehicular mobility models in VANET. In this project he participated in the development of the mobility simulator and the simulation analysis described in this paper. In September 2007, he joined the university of Versailles in order to follow a vocational training on networks, entitled IRS. Currently, he carries out a six month work placement at H5 audits, a company specialized in network metrology and auditing in France.