Large-Scale Simulation of V2V Environments

Hugo Conceição, Luís Damas,
Michel Ferreira
DCC/FCUP & LIACC
University of Porto
R. Campo Alegre, 1021/1055
4169-007 Porto, Portugal
{hc,luis,michel}@dcc.fc.up.pt

João Barros
DCC/FCUP & IT
University of Porto
R. Campo Alegre, 1021/1055
4169-007 Porto, Portugal
barros@dcc.fc.up.pt

ABSTRACT

Providing vehicles with enhanced ability to communicate and exchange real-time data with neighboring vehicles opens up a variety of complex challenges that can only be met by combining different research fronts such as wireless communications, information processing, self-organization protocols and collaborative optimization. The difficulty in performing real tests in this area forces the use of computer simulation. In this paper we introduce an efficient simulation framework for large scale vehicle-to-vehicle (V2V) networks in urban environments. Our main contribution is a sophisticated traffic simulator, which is oriented towards simulating car-to-car communications, and relies on a global positioning server in order to convey location information for micro-simulated vehicles. To illustrate the various studies made possible by our simulation system, we provide a preliminary characterization of how the wireless transmission range in an urban-like environment affects the freshness and inter-vehicle propagation characteristics of real-time mobility information.

General Terms

Design, Performance, Experimentation

Keywords

Large-scale simulation, traffic telematics, wireless networks

1. INTRODUCTION

The advent of wireless ad-hoc vehicle-to-vehicle (V2V) networks, i.e. groups of vehicles equipped with the ability to communicate over the ether and to self-organize into a collaborative mesh (see e.g. [1]), opens a myriad of possibilities towards sharing and exploiting highly dynamic geographic information. One of the most interesting V2V technologies is the possibility for vehicles to exchange information about traffic conditions, collected on the fly while traveling the roads. Based on this data, vehicles are able to calculate optimal paths using fresh estimates about road congestion.

Understanding the implications of propagating dynamic traffic information among vehicles on the overall congestion behavior of the entire road network is one of our goals, because it is essential to the success of V2V technologies. The added difficulty in performing real tests for these environments forces the use of computer simulation, which constitutes a significant challenge per se due to the complexity of traffic/human behavior, network topology and node interactions. This is shown by the high number of traffic models developed over sixty years of research in this area [2]. Enabling the interaction between a layer of car-to-car communication simulation and a layer of microscopic traffic simulation in order to obtain a realistic simulation of a V2V environment is also one of our main goals. The propagation of road congestion conditions can cause vehicles to dinamically change their route, which should be reflected on the traffic simulation engine.

Motivated by this need, our main contributions are as follows:

- *Large-Scale Simulation of V2V Environments:* we present a real-time open-source simulation framework for moving vehicles in an urban environment that includes multiple driving states, inter-vehicle communication and sophisticated visualization, providing to the community a comprehensive mean for the realistic study of V2V environments.
- *Cooperative Navigation Protocol:* We implement and test a very low complexity communication protocol that leverages the exchange of fresh traffic information between vehicles for optimized routing and navigation;
- *Analysis of the Impact of the Communication Radius:* We provide preliminary results obtained with our framework, demonstrating the importance of the communication range and other key aspects on the propagation of information, as well as the relevant possibilities opened by our software.

The rest of the paper is organized as follows. Section 2 provides a brief overview of the state of the art in terms of traffic simulation. Our general system setup is described in Section 3, whereas Section 4 outlines the details of our simulation framework and traffic data exchange protocol. Finally, the integration results are presented in Section 5 and the paper concludes with Section 6.

SAC'08 March 16-20, 2008, Fortaleza, Ceará, Brazil

2. RELATED WORK

Traffic simulation has been an object of intense study for the past sixty years. There exists many traffic simulators but just recently started to appear software directed to the study of V2V environments, with the capacity to simulate both the car traffic and the mobile network. From what exists in terms of V2V environment simulators, we choose two as examples and term of comparison:

1. TraNS (Traffic and Network Simulation Environment) [3] is a simulation environment that integrates both traffic and network simulators and aims to provide a tool to build realistic simulations of Vehicular Ad-hoc NETworks.

2. NCTUns [4] network simulator and emulator is prepared to simulate various protocols in specific car traffic situations. It can also simulate fixed nodes and perform both network and traffic simulation at the same time. NCTUns is very extensible and so adaptable to many environments.

In order to study the implications of the propagation of dynamic information between vehicles (e.g.: about the traffic conditions), it is essential that the traffic and the network simulators are able to interact, so that vehicles can react to the information (e.g.: change route). TraNS uses SUMO [5] as traffic simulator and NS-2 [6] as network simulator. It provides a way to integrate these two packages by generating a file from the traffic simulation that can be interpreted by the network simulator, and so both simulations can not occur at the same time. Due to this design principle, it becomes difficult to simulate the reaction of vehicles to information exchange.

V2V technologies are of greater application in urban areas. Unlike NCTUns, the software presented in this paper supports standard map file formats, making it possible to use maps provided by governmental institutions. This provides a way to do a more realistic simulation and also a mean to both empirical and scientifically compare the traffic generated by the simulator with the real traffic of the city. Our software is directed to simulate in an urban environment, supporting large-scale simulation of thousands of cars and roads, and specific situations like, for instance, cars parking for a while. Although presently and unlike NCTUns and TraNS, the network simulation is only done on the application layer of the OSI model, it can be used to perform valuable tests like the one presented in Section 5.

3. SYSTEM SETUP

In this section, we present our system setup. We describe our modeling assumptions regarding vehicles, mobility sensors and maps. Also, we discuss the communication aspects that are relevant in an environment of vehicle-to-vehicle wireless transmission.

3.1 Traffic Modeling Assumptions

Our simulation prototype for cooperative vehicular navigation, named DIVERT (Development of Inter-VEhicular Reliable Telematics), is currently setup over a vectorial representation of the road network of the city of Porto, the second largest in Portugal. This road network covers an area of 62 square kilometers, with 1941 streets summing up to 965 kilometers of extension. DIVERT is not dependent on this map and can be setup with maps of other cities or regions, which follow the Open GeoSpatial Consortium specifications.

In DIVERT we model two types of vehicles: vehicles which circulate and communicate, called *sensors*; and vehicles which just circulate. Within each type of vehicle, DIVERT further divides in normal and large-sized vehicles, associating appropriate mobility patterns to each. These mobility patterns are also individually influenced by random initialization of attributes such as acceleration, braking, aggressiveness and risk tolerance. Sensors add an attribute of wireless transmission range. An interesting attribute regarding the interaction of the two simulation layers is the "*patience threshold*" of each vehicle regarding congested roads, affecting the volatileness of its route plan.

DIVERT uses the following two vectorial maps of the road network:

1. A simple two-layer map of the road central axles, representing, through polylines, the geometry of intersection free road segments, and their topological connectivity. A copy of this map is present in every sensor, and is used for positioning of GPS readings and for the collaborative propagation of mobility conditions on road segments.

2. A multi-layer map describing in detail the road network, including information of road segment lanes, lane-level connectivity, intersection visibility, location of traffic lights, traffic lights inter-relationships, speed limits on segments, and parking.

A raster layer from satellite images further improves the visualization of traffic simulation in our prototype, which is currently based on 2D data. Work is undergoing for a 3D model, which will not only allow more realistic accelerations parameterized by steepness, but also enable an improved modeling of wireless transmission ranges, which accounts for fading, reflection and shadowing effects based on a 3D buildings layer.

Regarding vehicle routes, DIVERT currently uses an hybrid model of pre-defined routes and randomly generated routes. For randomly generated routes, our system arbitrarily selects an origin and a destination and calculates the route based on a shortest-path algorithm, either parameterized by distance or by time. Shortest-path based on time uses not only the speed limits of segments, but mainly the dynamic calibration of average mobility derived from previous simulation results. Pre-defined routes have an associated frequency and have been carefully chosen to approximate the simulation to our perception of traffic distribution in Porto city.

Traffic simulation is parameterized by the number of vehicles and the percentage of these vehicles which are sensors. Simulation is initialized by randomly placing each vehicle in an arbitrary point of its route. Vehicles which arrive at their destination are either removed or parked for a parametrized time. New vehicles also show up during the simulation, either from entry points in the map, or from parking lanes of segments, as DIVERT tries to maintain the targeted number of vehicles for the simulation.

It should be noted that the simultaneous micro-simulation of thousands of vehicles, with the degree of sophistication present in DIVERT, poses major challenges in term of optimization of algorithms and efficiency of data-structures. Although it is out of the scope of this paper to exhaustively describe the techniques implemented in our traffic simulator, we will later address the most relevant aspects concerning efficiency.

3.2 Communication Aspects

In order to capture the inter-vehicle communication aspects it is necessary to define the level of abstraction with respect to the physical communication channel and the protocol architecture. At the current preliminary stage, we opted for a very simple broadcasting model, in which vehicles are able to communicate with each other if their distance is below a certain threshold, which is determined by the transmission radius. The resulting random geometric graph is widely accepted as a simple yet reasonable first-order approximation of the connectivity pattern attained by a mobile ad-hoc network [7]. A more elaborate approach would be to consider path loss, multi-path and shadowing effects, however this would incur in a high penalty in terms of simulation complexity. Another alternative would be to consider collisions and packet losses over the wireless medium. We are currently considering the possibility of integrating these aspects in our simulation in order to obtain a richer connectivity profile.

4. SIMULATING A V2V ENVIRONMENT

In this section we describe how a traffic simulation layer, which acts as a global positioning server, and a simulation layer of wireless inter-vehicle transmission, are put together to provide a realistic simulation of large-scale V2V systems in action.

4.1 The Traffic Simulator

The traffic simulator is the main piece of software of the DIVERT framework. It is an open-source C++ implementation, currently listing 50000 lines of code, including the graphical interface and visualization component. At url: `http://myddas.dcc.fc.up.pt/divert` we provide a short video of the V2V simulation over Porto city, which allows the reader to preview and evaluate the degree of sophistication of the simulator described in this paper.

Section 3 has already described the main modeling assumptions implemented in the simulator. A key point in the simulator implementation is related to efficiency. As we mentioned earlier, the simultaneous and real-time micro-simulation of thousands of vehicles, in a single computer, is only possible using highly optimized algorithms and data structures. The two main techniques implemented in DIVERT are explained next.

The first technique is inspired by what happens with human drivers and their distinct focusing effort in different driving environments. For instance, driving straight ahead in an intersection-free road segment requires much less focusing effort than driving in an intersection, which can require analysing incoming vehicles, respecting traffic lights, or perform lane changing. Our traffic simulator uses geographic information to distinguish between vehicles traversing intersection-free segments and vehicles traversing intersections. In the first case, simulation is very simple and computationally efficient, essentially implementing a Car Following model[8], where distance to the preceding vehicle is the main variable being monitored. In the latter case, simulation is much more computationally expensive, with a large number of variables being accounted for and a much larger degree of interacting vehicles. As expected, at a given instant, most vehicles are simulated in intersection-free segments, which ensures overall efficiency.

A second technique, also uses geographic information to partition the road network layer in order to explore multi-threading and parallelism in multi-core processors. Given the partitioning in regions of the city map, each region is asynchronously handled by a simulation thread. A special synchronization region is created to connect asynchronous regions, so that vehicles can only move from an asynchronous region to another through the interconnecting region, which ensures synchronization (through the use of locks) guaranteeing that no two asynchronous regions can be concurrently modifying it. Although we have been running our simulations in a single computer, the simulator is prepared to explore parallelism using multiple computers.

The simulator is also able to emulate a GPS receiver in each of the sensor vehicles. The interface with the communication simulation layer is done through GPS-like sentences, where the simulator generates for each sensor its position, in terms of latitude and longitude, and its velocity vector, together with a global timestamp which provides the time-synchronization among the inter-vehicle data exchange. Using these GPS-like sentences for the interface allows the simulation of realistic scenarios where there is a positioning error associated with each GPS reading. It is a task of the communication simulation layer to implement a map-match algorithm to obtain the actual road segment, as done by GPS navigation systems installed in vehicles.

4.2 Communication Simulation

This layer simulates the actions performed by each sensor in terms of the communication with other sensors. We now describe these actions.

As mentioned above, the traffic simulator acts as a positioning server. It provides the following data structure to each sensor:

GPS INFORMATION

$TS \leftarrow$ timestamp

$(X,Y) \leftarrow$ position point

$(V_x,V_y) \leftarrow$ velocity vector

Each sensor saves the following data structure:

DATA STRUCTURE D

$S \leftarrow$ array with information about each road segment

$S_i \leftarrow$ ith element of S

$t \leftarrow$ information freshness

$av \leftarrow$ average speed

$(t_i, av_i) \leftarrow S_i$

After matching the GPS point to a road segment (we consider a maximum error of 15 meters in our simulation), the current sensor updates its D data structure, refreshing its information about the road segment average speed with the velocity value reported by the GPS sentence. The updating of the average speed and freshness values about a given segment is done incrementally through aggregation of all the GPS readings collected by the sensor while traversing a road segment.

Besides being updated "*on-the-fly*" by the GPS sentences received by a given sensor, the D data structures are also updated through a basic communication protocol that enable the moving vehicles to exchange mobility information about the road network. For now, we are assuming that the lower layers of the protocol stack deal with all issues pertaining packet errors and losses, interference, collisions and congestion, such that each node can broadcast its information reliably to its neighbors.

Each vehicle j that abides by the protocol performs the following steps:

1. Wait for Packet Transmission by neighboring node h;
2. Upon reception of a new packet D_{nodeh}, update table D_{nodej} according to algorithm 1;

Algorithm 1 update table values, re-order, purge

```
$NS \leftarrow$ number of map segments
$t_x \leftarrow$ information freshness of segment x
$av_x \leftarrow$ average speed on segment x
for $i$ to $NS$ do
  if $t_{inodej} > t_{inodeh}$ then
    $t_{inodej} \leftarrow t_{inodeh}$
    $av_{inodej} \leftarrow av_{inodeh}$
  end if
end for
```

3. Wait for pre-determine retention time;
4. Broadcast Packet D_{nodej} to neighboring nodes.

Note that a certain amount of retention time is necessary to avoid the so called broadcast storms in the network. Ideally, the number of transmissions within a limited time interval should depend on the density of vehicles in a certain area of the road networks, as well as on the requirements with respect to the freshness of the traffic data for effective navigation. Introducing these parameters in the protocol is part of our ongoing work.

5. INTEGRATION RESULTS

In this section we present results for the integration of the traffic simulation layer with the communication simulation layer. Our evaluation is aimed at providing insight on the relation between the wireless transmission range and the quality of the mobility information, in terms of *freshness* and *map-coverage*. The analysis and graphical representation of the results has been done with Spatial-Yap, a logic-based geographic information system [9].

We have run our simulations with 5000 vehicles, of which 500 are sensors, over the map of Porto city. Simulation time was 20 minutes for each simulation (1200 interfacing steps, one per second). We evaluate mobility progagation using wireless transmission ranges of 30, 60, 120 and 180 meters. Compared with the DSRC maximum communication range for safety applications, 50 to 300 meters [10], our ranges are conservative, due to the urban environment where the simulations are ran and the fact that our model does not account for fading, reflection and shadowing effects caused by buildings.

Table 1 presents a comparison of relevant statistics for the four simulations with these varying ranges. The first row lists the total number of communications performed during the 20 minutes of the simulations. The second and third rows list, respectively, the average and maximum number of sensors communicating at each instant, t, of the 1200 seconds of the simulation, in the entire city area. The fourth row lists the average size of the packets transmitted between sensors. Sensors only transmit information about segments whose freshness is greater than $(t-300)$, where t is the current simulation step (second). This is the condition pruning the size of the transmitted packets. The fifth and sixth rows list, respectively, the average and maximum number of refreshed segments per communication. A segment is refreshed in a given sensor if its freshness value is lower than the value being transmitted. Finally, seventh and eighth rows list, respectively, the average and maximum number of vehicles with which one vehicle communicates per instant. These values allow evaluating the necessary bandwidth for the communications to be completed in the one second period assumed in the simulation.

From Table 1, one of the most interesting values is the average number of refreshed segments per communication, ranging from 11 (60m) to 16 (30m). The reason for the higher value with a shorter communication range is explained by the smaller number of communications, which tend to be more informative when they happen. This small value of refreshed segments leads to the conclusion that it is probably possible to further prune the packets transmitted based of higher values of freshness, without major impacts on mobility information. This is exactly the kind of hint and immediate verification that the framework described in this paper enables.

In order to implement mobility-based navigation, vehicles should have, at a given instant, a good knowledge of traffic conditions, in terms of coverage, as well as in terms of freshness. In the evaluation of mobility propagation there are three dimensions that should be analyzed: segment coverage, freshness and number of sensors, forming a 3D data cube. In the analysis that follows, we have fixed freshness to a value lower than 5 minutes ($300t$) and fixed the number of sensors to 250, representing 50% of the total number of sensors. Our goal is to analyze the third dimension, segment coverage, seeing which segments are known to at least 50% of the sensors, with a freshness of at least $900t$, at the end of the $1200t$ of the simulation.

Figures 1(a), 1(b), 1(c), and 1(d) show, for the four different transmission ranges, the map segments about which 50% of the sensors have information fresher than 5 minutes at the end of the simulation.

In Fig. 1(a), the few highlighted segments are also the ones with greater traffic density. Because of this, when sensors enter this segments, they may take long to exit and so the information about other segments becomes obsolete, by our parameters.

In situations of great traffic congestion, a lot of data exchanges may occur in a segment, but what happens is that after a short period of time, every sensor vehicle has the same information and what they receive from others is not useful. The very low transmission range we used in the first simulation resulted in communications almost only with sensors in the same segment, and few with sensors in other segments. This seems to be the key factor for the insuccess of mobility propagation in this simulation.

When two sensors in different segments communicate, they always exchange new data. An optimal situation for this to happen is an intersection where sensors with distinct origins

Statistic	*30m*	*60m*	*120m*	*180m*
Number Communications	204798	303391	335859	343375
Average number of communicating vehicles per instant	189	276	383	458
Maximum number of communicating vehicles per instant	258	330	443	505
Average size of transmitted packets in bytes (pruned by freshness)	13344	20359	33973	47089
Average number of refreshed segments per communication	16	11	13	14
Maximum number of refreshed segments per communication	723	1554	2962	1900
Average number of vehicles with which one vehicle communicates per instant	0.35	0.51	0.70	0.84
Maximum number of vehicles with which one vehicle communicates per instant	3	3	6	11

Table 1: Simulation statistics with varying transmission ranges

and routes, meet and exchange data.

As we increase the transmission range in our tests, communication between sensors in different segments occurs more frequently, and as we can see in the respective figures, the results in terms of map coverage increase substantially.

From these results we conclude that there exists a critical value for the transmission radius after which the dissemination of traffic information is sufficient for a large number of vehicles to be able to compute a comprehensive and accurate congestion map. This observation is strikingly related to the physical phenomenon of percolation, which is well known to govern the connectivity of large classes of wireless networks [11]. Once the transmission radius is above the critical threshold, the graph representing wireless connectivity has a giant component on which traffic information flows very fast and over long distances. One may think the communication could be limited to intersections, but in situations of low traffic or if we scale up to a map of a country with long highways, the communication outside of the intersections is crucial.

6. CONCLUSIONS

We presented a simulation framework for a large-scale urban vehicular network in which a subset of nodes are capable of exchanging traffic data in real-time. Perhaps the most striking features of our prototype are its ability to micro-simulate a very large number of vehicles on a real road graph and the inclusion of communication aspects that allow the evaluation of vital parameters for collaborative navigation, specifically connectivity (as determined by the transmission radius) and freshness of data. Our results show that these parameters play a vital role in the view of the road network possessed by each vehicle in each time instant, thus determining its ability to find the best route given the congestion state of city.

As part of our ongoing work, we are refining the urban mesh (by adding three-dimensional data), the mobility patterns (by changing the routes according to dynamic traffic data) and the communication aspects (by including more sophisticated wireless propagation models). We believe that the presented methodology and tools can provide the means for a deeper understanding of cooperative navigation systems and their impact on the behavior of urban traffic.

7. ACKNOWLEDGMENTS

We would like to thank the Porto City Council for providing us with the maps of the road network. This work has been partially supported by the Portuguese Foundation for Science and Tecnology (FCT) under projects MYDDAS (POSC/EIA/59154/2004), JEDI (PTDC/EIA/66924/2006), STAMPA (PTDC/EIA/67738/2006), and by funds granted to LIACC and IT through the Programa de Financiamento Plurianual and POSC.

8. REFERENCES

[1] CAR 2 CAR Communication Consortium Manifesto. Version 1.1. http://www.car-2-car.org/, 2007.

[2] S. Hoogendoorn and P. Bovy. State-of-the-art of vehicular traffic flow modelling. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 215(4):283–303, 2001.

[3] Trans (traffic and network simulation environment). http://trans.epfl.ch/mediawiki/index.php/Main_Page.

[4] S. Y. W. et al. The design and implementation of the NCTUns 1.0 network simulator. *Computer Networks*, 42(2):175–197, June 2003.

[5] D. Krajzewicz, M. Hartinger, G. Hertkorn, P. Mieth, J. Ringel, C. Rssel, and P. Wagner. The "simulation of urban mobility" package: An open source traffic simulation. In *2003 European Simulation and Modelling Conference*, 2003.

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

[7] F. Xue and P. Kumar. The Number of Neighbors Needed for Connectivity of Wireless Networks. *Wireless Networks*, 10(2):169–181, 2004.

[8] G. Newell. Nonlinear Effects in the Dynamics of Car Following. *Operations Research*, 9(2):209–229, 1961.

[9] D. Vaz, M. Ferreira, and R. Lopes. Spatial-yap: A logic-based geographic information system. In V. Dahl and I. Niemelä, editors, *ICLP*, volume 4670 of *Lecture Notes in Computer Science*, pages 195–208. Springer, 2007.

[10] "Vehicle safety communications project-final report". Technical Report HS 810 591, USDOT, http://www-nrd.nhtsa.dot.gov/departments/nrd-12/pubsrev.html, April 2006.

[11] L. Booth, J. Bruck, M. Franceschetti, and R. Meester. Covering algorithms, continuum percolation and the geometry of wireless networks. *Ann. Appl. Probab*, 13(2):722–741, 2003.

a) transmission radius = 30 meters

b) transmission radius = 60 meters

c) transmission radius = 120 meters

d) transmission radius = 180 meters

Figure 1: Map coverage