ITSUMO: An Intelligent Transportation System for Urban Mobility

Bruno Castro da Silva, Ana L.C. Bazzan*, Gustavo K. Andriotti, Filipe Lopes, and Denise de Oliveira

Instituto de Informática, UFRGS, C.P. 15064, 91501-970, Porto Alegre, RS, Brazil

Abstract. It is well-known that big cities suffer from traffic congestion and all consequences that come with it. This is an especial problem in cities in developing countries where the public transportation system is not reliable and where the fleet of vehicles tend to be old thus increasing air pollution. There is no turnkey solution for this problem, but several improvements have been suggested in the field of urban and traffic management, provided an information system is built which can provide information to both the traffic experts and the user of the system. Such an information system has to incorporate features of an ITS and an ATIS. An underline assumption is that there is a simulation model to provide certain kinds of information in forecast. This paper discusses the model and implementation of such an information system which is based on a microscopic model of simulation and on cellular automata and is implemented using agent technologies and with a bottom-up philosophy in mind. We give here an overview of the project, the details of the modules (data, simulation, driver and information/visualization), as well as discuss an application of the simulation tool.

1 Introduction

The increasing urban mobility poses challenges to traffic engineers, urban planning experts, and researchers involved with optimization and information technology. One way – certainly not the only one, but one to be used in conjunction with others – to cope with this problem is to approach it from the information point of view. Our assumption is that urban mobility both produces information and increases the demand for it, a loop which has been the motivation for previous works [1, 2].

One way to deal with this information loop is to collect data, produce a forecast and broadcast it to the traffic system users. It is well-known that the *same* information when provided to *all* users can be harmful. This is certainly a point but will not be directly tackled in this paper, although the simulation infrastructure provided here can deal with such situations and in fact we have already been doing case studies [3]. In order to produce the information to be broadcasted, a fast simulation model is necessary. We use the Nagel–Schreckenberg approach [4] which is a microscopic model for traffic simulation originally based on cellular–automata (CA), as detailled in Section 2.

The motivation for the development of the simulation module is twofold. First the fact that although several similar tools exist, they are built normally with the objective

* Corresponding author

T. Böhme et al. (Eds.): IICS 2004, LNCS 3473, pp. 224–235, 2005.

to tackle isolated problems. Currently traffic engineers only have a collection of isolated tools to i) track and log flow and density of vehicles (induced loop detectors, cameras, etc.); ii) perform off-line simulations (e.g. to evaluate the impact of building a shopping mall); iii) adjust timing of traffic lights. None of these tools is really suitable for forecast. The new generation of traffic simulators are usually too slow for the faster-than-real-time simulations which are needed for traffic forecast, or are too expensive for budgets of cities in developing countries.

Second, both commercial and academic tools are black-box systems in the sense that generally one has no access to the source code. This is no big problem to users like traffic engineers working in control centers of municipal traffic authorities with a reasonable budget. But that poses a problem for i) anyone working in research; ii) anyone who wants minimal changes in output reports or, more important, to those who want to finetune the simulation model; iii) those aiming at integrating the simulator to other tools.

Therefore, driven by the incentives given by Brazilian funding agencies regarding free software, the project SISCOT (Integrated System for Simulation, Control and Optimization) was developed [2]. The aim of this project was to build a traffic simulator based on free software and on a CA model, to be used freely for academic and operational purposes.

We are now developing an extension of the SISCOT project, called ITSUMO – Intelligent Transportation System for Urban Mobility[1]. One of the aims of the project is to create an information system for urban mobility capable of integrating different functionalities, such as a simple traffic control (mostly based on control of traffic lights), traffic management and real-time information providing via internet and/or mobile phone. The idea is to distribute the system under free-software license.

The development of the project involves people with different backgrounds: physicists (microscopic traffic simulation model), computer scientists (modeling the driver and integration of it to the anticipatory traffic forecast tool), and traffic engineers (as users and evaluators of the approach). Once this infrastructure is ready, it can be used both for academic purposes (e.g. testing new approaches for synchronization of traffic lights, simulating drivers decision-making, etc.) or for operational purposes.

To address these objectives, a novel feature added to the basic simulation model is the possibility to easily define other modules such as drivers as intelligent agents, in contrast to the current models that are pure reactive and ignore the mental state (informational and motivational) from the drivers.

The kernel of the simulator is ready and was programmed using the object-oriented paradigm in order to facilitate the portability and increase the quality of documentation. The idea is that anyone can aggregate further code when necessary. Also a database and a visualization module are at final development stages, as well as the basic information module which is closely tied to the visualization, in case of information via internet.

To illustrate the benefits of our approach, consider this fact which has been taken from the daily newspaper "Folha de São Paulo" (online edition of April 21, 2005): "São Paulo (city) reaches traffic jam record: 178 Km. of jam were recorded at 7 p.m.". In Feb. of 2002, that amount was 130 kilometers. Since then, such level of traffic jam

[1] http://www.inf.ufrgs.br/˜mas/traffic

was reported many times due to the lack of timely information to users of the traffic system (public transportation, trucks and vehicle drivers).

The benefits for users is clear: once a forecast is provided or at least some information is given to them, they are able to better plan their trips. This has an impact on travel times, fuel consumption and acceptance of public transportation.

The remaining of this text is organized as follows: the next section details the microscopic simulation model. Section 3 presents the overview of the project and the main modules: data model, simulation kernel, infrastructure / module to model drivers decision-making, visualization and information to the users. Section 4 presents and discusses the use of ITSUMO project in a a traffic simulation of a region of the city of Porto Alegre, Brazil. The last section concludes and outlines the future research directions.

2 Microscopic Simulation: Nagel–Schreckenberg Model and Extensions

Basically, there are two approaches to model traffic movement, namely the macroscopic and the microscopic. The former is mainly concerned with the movement of platoons of vehicles, focusing on the aggregate level. At this level, the system as a whole behaves as if all traffic agents (drivers) make decisions in a deterministic and rational way. It considers only averaged vehicle densities but not individual traffic participants. On the other hand, in the microscopic model of simulation, one may go to the individual level. Each road can be described as detailed as desired, given the computational restrictions, thus permitting a more realistic modeling of drivers' behavior. In the microscopic approach both travel and/or route choices may be considered, which is a key issue in simulating traffic since those choices are becoming increasingly more complex. Also, individual traffic lights can be modelled according to several approaches, from classical off-line coordination to recently proposed ones (negotiation, communication-free, via game theory, reinforcement learning, swarm intelligence, etc).

2.1 Nagel–Schreckenberg Model

In this project, in order to achieve the necessary simplicity and performance, we use the Nagel–Schreckenberg model [4] which is a microscopic model for traffic simulation originally based on cellular–automata (CA).

In short, each road is divided in cells with a fixed length. This allows the representation of a road as an array where vehicles occupy discrete positions. Each vehicle travels with a speed based on the number of cells it currently may advance without hitting another vehicle. The vehicle behavior is expressed by rules that represent a special form of car–following behavior. This simple, yet valid microscopic traffic model, can be implemented in such an efficient way that is good enough for real-time simulation and control of traffic.

Given that every vehicle has a nonnegative integer speed (v, limited to $vmax$), the following four rules are verified simultaneously for all vehicles in a CA way:

1. Movement: each vehicle advances v cells at each time step;
2. Acceleration: each vehicle's speed is increased by one unit, up to $vmax$ (the maximum speed) or the gap (the number of empty cells in front of a vehicle);

3. Interaction: if the vehicle ahead is too close, v is decreased by one unit;
4. Randomization: vehicles decelerate with probability p in order to simulate the non-deterministic dynamics of vehicle movement, which occurs mainly because of the different driving behaviors.

These CA rules imply that certain conditions must be met. First, the simulation must be discrete both in time and space. Second, all network cells must be updated simultaneously. This condition implies that the vehicles' movement must occur in two steps: a) all drivers must take their decisions of movement and b) all vehicles' status must be updated by taking into account the driver decision and the restrictions and limitations of the vehicle and the scenario being simulated.

Although these rules might seem too simplistic, investigations reported in the literature showed that the cellular automaton model is capable of reproducing macroscopic traffic flow features including realistic lane changing behavior (for an overview see [5]). Later on, this basic model was extended to deal with enhanced scenarios [6–8]. In the urban traffic scenario, there was the need to add more elements such as traffic lights and more complex intersections.

The first concept of the Nagel–Schreckenberg model was a very simple CA that only intend to mimic a single-lane circular circuit on a highway. Improvements have been made on that basic model to make it possible to simulate other scenarios. There are two major extensions to the model to make it suitable for more complex simulations:

– Multi-lane extension: several different extensions had been made to achieve this characteristic, such as [5] and [9]. These improvements allow to perform realistic simulations with more than one lane, since vehicles can change lanes. Therefore it is possible for the vehicles to perform a take-over maneuver and avoid slowing down by changing to a faster lane.
– Density inversion: This characteristic observed in real traffic is important to make simulations more realistic. Density inversion is a phenomenon that occurs when the flow of vehicles reaches a certain threshold. Then the lane on the left side gets a higher density than the lane on the right side. This empirical phenomenon is reproduced to some extensions of the original model (see [9] and [10]).

2.2 The Traffic Network Representation

As for the network representation, each road is described as a composition of nodes and edges representing intersections and roads, respectively. Each edge is composed by a set of connected sections. A section represents a portion of the road that is delimited by two nodes. Each section is formed by a set of lanesets, which represent groups of lanes with the same orientation. The lowest-level structure in the network representation is the lane itself. Lanes are subdivided into cells of fixed length, which can be either empty or occupied by a vehicle. The cell size is chosen so to represent the average space a vehicle occupies in a traffic jam, but this value can be adjusted. It is important to do this adjustment so that each time step in the simulation corresponds to the desired time frame, that is, the simulator must be adjusted in order to use discrete values that faithfully represent real-life measurements.

As said before, a node is an important component of a road. It represents the connection between two sections and is an abstraction of a real-life crossing. A node may contain four different objects: sources, sinks, sensors and traffic lights[2]. Sources and sinks are objects used to adjust the simulation parameters to known values of traffic flow (e.g. collected by loop induced detectors). This way, the traffic manager or the user responsable for the simulation may configure a node in a way that vehicles are inserted into the network according to fixed rules. Currently, nodes equipped with a Source object can generate vehicles according to a constant flow, a constant probability, a variable probability or a variable flow. The Sink object is just the opposite of the Source. It is responsible for removing vehicles of the simulation. Note that dead-end streets must obligatorily be linked to a Sink object. Currently the simulator can automatically handle this kind of requirement preventing the user from facing the tedious task of adapting the real-life traffic network to the simulator standards.

Moreover, the simulator supports two other objects: Sensors and Traffic Lights. The purpose of a Sensor object is to collect all sorts of information about the scenario being simulated, such as the lane occupation rate, the average vehicle speed in a street, in/out flow of vehicles in a specific laneset, etc. Traffic Lights control a set of signal plans which reflects the movement constraints of the real traffic network.

3 Description of the System

3.1 Overview of the Project

The ITSUMO project uses data from several sources, which can be both off-line information (maintained by different providers), and on-line information (e.g. traffic flow). The network topology data, for instance, is provided by traffic authorities. Currently, given that there is a lack of on-line data, which is not collected in the city of Porto Alegre so far, we work basically with off-line data. However, the system is designed to use real-time information.

Most of the information is stored in a database, described in the next section. There is also a module for definition of drivers decision-making, which is optional. If the user does not define particular classes of drivers, then the simulation is performed using the standard driver model described in Section 2. In any case, the simulator kernel retrieves the necessary information from the database, performs the computation of the vehicles' movement, updates the network state and stores this in flat files. If special drivers are defined, then the simulator also loads such models in a way described in Section 3.4.

A visualization module retrieves data originated from the microscopic simulation and exhibits a graphical representation of the traffic simulation. The ITSUMO system is thus composed by four distinct modules: the data module, the simulation kernel, the driver definition module, and the visualization module. These modules are further detailed next.

3.2 Data Module

This module manages the traffic information data that serves as the basis to the microscopic simulation. This information is stored in a relational database, which means that

[2] Sources and Sinks are inspired in the proposals discussed in [7, 8, 11].

the data is represented in the form of tables. Queries to these tables are made by means of the SQL language.

The data module uses the PostgreSQL relational database system[3]. Since performance is a key issue in real-time simulation, a database with simple interface and quick response is required. PostgresSQL is a free implementation of SQL/92 standard with a fair time response to the database queries. Another important issue is the possibility to work with spatial data in order to make use of geographically referenced data. This capability is provided by PostGIS, an extention of PostgreSQL to support spatial data and geographic objects. Note that the topological data existing in the database is not yet geographically referenced, although we plan to use these capabilities in the future. Actually, spatial positions are implemented by means of a series of attributes on each node, such as its spatial cartesian coordinates. Besides that, the database makes use of several foreign-key constrains in order to guarantee the consistence of the spatial relationship between objects.

The database structure reflects the way the simulation occurs. Thus, the data module is in charge of providing the necessary means to specify several topology details and network constrains. The database entities and the attributes they model are described below:

- **General Settings:** topology name, traffic system orientation (right-handed, left-handed), cell size, real-life correspondence of each iteration, frequency of sensor measurements, starting hour of the simulation and *default* deceleration probability
- **Network:** network name and its settings
- **Node:** cartesian coordinates of the node and the associated network
- **Street:** street name and the associated network
- **Section:** section name, whether it is preferential, delimiting nodes and the associated street
- **Laneset:** length, associated section
- **Lane:** maximum speed, width, associated laneset
- **Laneset Probability:** set of probabilities of moving a vehicle to a specific laneset upon passing through a node and the associated lane where these probabilities are valid
- **Turning Probability:** set of allowed vehicle movements according to its current lane
- **Traffic Light:** set of signal plans, associated with a node
- **Signal Plan:** set of lane-to-laneset allowed movements in a specific order, cycle time and offset
- **Sink:** probability of removal of a vehicle and the associated node
- **Source:** source behavior (insertion of vehicles according to fixed values per iteration, insertion of vehicles according to constant/variable probabilities, etc), laneset where the vehicles should be inserted and the node which contains this Source.

The above entities also describe the relationship between the several traffic network components. For instance, each lane is associated with exactly one laneset, and each

[3] Actually all technologies used in this project are free software, such as PHP, Apache Web Server, Java, JVM and Debian servers.

section is composed by exactly two nodes. The data module was designed in a way to ease the task of specifying the traffic network topology.

3.3 Simulation Module

This module was developed using C++/PostgreSQL platform and is responsible for simulating the traffic flow in a given traffic network. The simulation kernel was implemented over the ideas of a previous version of the simulator [2]. The simulator implemented in [2] imposed serious restrictions to the feature-adding process, making it hard to add new functionalities and delaying the process of implementing the necessary extensions.

The current version of the simulator solves an issue inherent to the basic microscopic model described in Section 2, namely the difficulty to represent urban scenarios. Thus, the simulation tool we developed supports different kinds of elements, such as lanes, lanesets, streets, sections, vehicles, sources and sinks (of vehicles), sensors and detectors, traffic lights, etc. Some special functions are still under development, which will be briefly explained in Section 5.

The simulation occurs in discrete steps and is implemented as a series of updates in the drivers' decisions of movement, followed by simultaneous updates in the vehicles' positions in the network and finally updates in the status of objects such as nodes, sources, sinks and traffic lights. Each update in a node or traffic light may modify its current behavior depending on its internal state and on the current traffic flow. Sources and sinks are also activated whenever a vehicle passes over a crossing. This is the moment in the simulation when vehicles might be inserted or removed from the network. Note that special drivers (such as floating cars) are not removed from the simulation. Also, they do follow specific routes, in opposition to standard drivers which move through the network in a non-deterministic way. The last action to be performed in each iteration is to fire the sensors and detectors, updating their internal status and eventually displaying partial simulation results.

The simulation results can be formatted according to the user needs. The most usual formats are the "cell map" and the "laneset occupation map". The former is a representation of a set of cells associated with each lane in the network. This set of cells indicates which portions of the lane are occupied by which vehicle, providing the most detailed output possible. On the other hand, the "laneset occupation map" is a high-level output which specifies the rate of occupation (density) for each laneset in the network. This output is generally more useful since it hides the individual status of the vehicles and focus on the overall stochastic behavior of the simulation.

3.4 Driver Model

Modeling a driver can be approached in different ways, depending on the purpose of the simulation. In most cases, the objective is to simulate the collective or macroscopic behavior. However, this behavior emerges out of individual ones. Simple algorithms, like the Nagel–Schreckenberg model, can be used to describe vehicles movement without loosing significant simulation fidelity with reality.

A driver model can be splitted into two different components: decision about movement and planning. These two parts together can guide a vehicle in the network. The movement decision is responsible for a vehicle's short-time movement. On the other hand, the planning decision is related to a more sophisticated decision-making, e.g. which direction to turn and what to do in case a jam exists.

The movement decision-making is in charge of local optimization, i. e., to decide whether it is better to take-over the headway vehicle, to change lane and to avoid collisions with other vehicles. More complex than the former, the planning decision-making component is in charge of deciding which path to follow from one point to other in the traffic network. To make such decisions it is necessary to observe the traffic forecast (if available), the path restrictions, etc. Moreover, it is necessary to implement some kind of learning skills in the driver module so that it improves future decisions. Most of the optimization performed intends to reduce the amount of time to travel from an origin to a destination. The planning component can use several heuristics and methods to perform those tasks.

3.5 Visualization Module

This is the module that allows the graphical visualization of the simulation results. The visualization can be either in a macroscopic or microscopic level. At a macroscopic level, the visualization considers only data which reflect the overall behavior of the network, abstracting details and providing an useful tool to capture the big picture of what is happening in a specific scenario. It is useful to provide this kind of information via the internet, as it has been the case in [12].

On the other hand, the microscopic level visualization provides an interface through which one can see, both in 2D and 3D manner, individual vehicles movement. In order to obtain a more realistic and detailed visualization, this module is being developed using the OpenGL graphic library, enabling features such as walk-through navigation and detail-focused interfaces. An example of the visualization is depicted in Figure 1.

The visualization of data at microscopic level is the kernel of the information system for the traffic engineer and/or the urban planning experts. At this level, the whole system can be used to perform simulations in order to check what happens if a shopping mall is built in a specific place, or a new arterial is added to the network (as it was the case recently in Porto Alegre), or traffic lights are synchronized in a given way. In the next section we discuss a scenario from Porto Alegre in which the ITSUMO system was used: a simulation of an arterial with different rates of vehicles insertion in given nodes of the network.

4 Experiment: A Scenario from Porto Alegre

The ITSUMO system is capable of simulating traffic flow considering several simulation parameters. These parameters, such as the probability of inserting new vehicles in a given street and the deceleration probability reflect the overall behavior of the network being simulated as well as the drivers' behavior.

The probability of inserting vehicles in the simulation reflects the inflow of vehicles in a given location. The deceleration probability, on the other hand, represents the

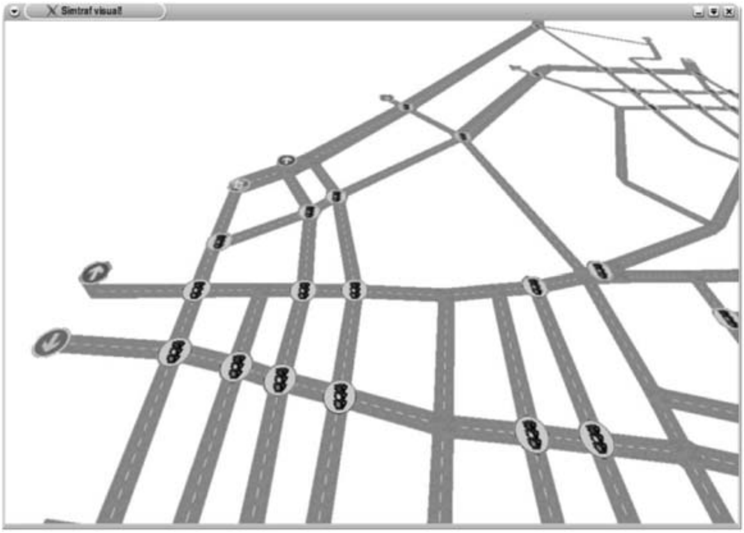

Fig. 1. Example of the visualization module interface

tendency of drivers to slow down occasionally, which is a known cause for quick and sporadic traffic jams.

The ITSUMO system was applied in a scenario which models an important part of an arterial street located in Porto Alegre. This street, called Independencia, is composed by several sections and traffic lights. There are six points of insertion of vehicles (sources): five of them are located in the transversals and insert vehicles with a constant rate of 0.1 per iteration. The insertion probability of the other source, located in the beginning of the Independencia street, will be one of the varying parameters of the simulation. Note that since we measure the vehicle density only in the Independencia street itself, we chose to keep the insertion probability in transversal streets as small as possible in order to minimize their influence on the main traffic.

The scenario also contains four traffic lights, as shown in figure 2, each one operating with two phases. The first phase gives green time to the flow in the main street (75% of the cycle time); the other phase gives green time to the flow in streets that cross the Independencia avenue (25% of the cycle). The traffic lights are not synchronized in any way. All vehicles leave the network in the last node of Independencia, depicted as the rightmost node in Figure 2.

The goal of this test scenario is to show that this type of simulation can be done with the ITSUMO system, and how the results can be used in order to re-plan and optimize the already existing traffic infrastructure.

In order to depict the effect of two important simulation parameters, namely the deceleration probability and the inflow of vehicles in the Independencia street, we present a table containing the average vehicle density in the street according to the variation of these parameters.

Table 1 shows how the traffic flow varies according to the parameters. Note that an increase in the probability of deceleration leads to a proportional increase in the street

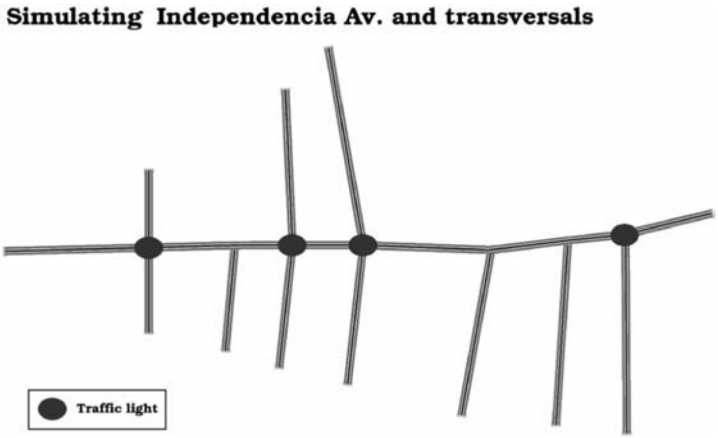

Fig. 2. 3D visualization of the Independencia avenue

Table 1. Density on Main Street after 20000 time steps

	Deceleration probability			
Main Source Probability	0.1	0.3	0.5	0.75
0.1	0.10	0.10	0.11	0.12
0.25	0.13	0.14	0.15	0.21
0.5	0.19	0.21	0.25	0.46
0.75	0.27	0.29	0.36	0.51
1.0	0.34	0.39	0.45	0.52

density, since the average speed decreases and the effect of a deceleration is cumulative in the street. In a similar manner, an increase in the probability of insertion of vehicles leads to a proportional increase in the density, since there are more vehicles using a limited street area. Moreover, the density values depicted in Table 1 depend on the relation between the street length and the duration of the traffic light phase. This relation is important due to the fact that longer traffic light phases tend to create cycles of density with great amplitudes. This effect, which causes the density of vehicles in the street to vary between zero and something close to the maximum possible value (1), can be seen in Figure 3.

In order to optimize network parameters such as the traffic light cycle time and the phase time, one can estimate parameters like sources probabilities according to data generated by induced loop detectors, cameras, etc. These data is thus used to calibrate the simulation parameters and allows the usage of the ITSUMO system to control and optimize the existing traffic infrastructure.

5 Conclusion and Outlook

The ITSUMO system is capable of dealing with several aspects of the simulation of a traffic scenario, such as the driver behavior, traffic lights coordination, traffic jam

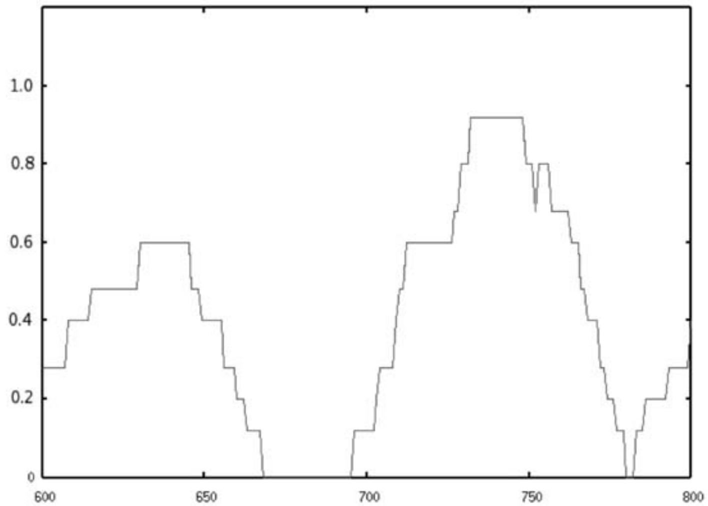

Fig. 3. Variation of density over time

prediction, among others. The driver module makes the simulation more realistic since it allows the modeling and use of several and arbitrary drivers behavior simultaneously.

Currently, the ITSUMO system provides an useful tool to simulate and predict traffic conditions, as well as to support urban planning tasks. The system itself is independent of driver model and supports the adjustment of several parameters that can be used to reflect the characteristics of the traffic network. The use of the ITSUMO simulation module facilitates the implementation of concrete actions that can lead to a better traffic flow control, reducing the average commuting time thus promoting a better life quality in cities with complex urbanization.

We plan to extend ITSUMO to consider other kinds of information such as those related to weather forecast. Our future goal is to focus on the development of ITSUMO as an information system for urban mobility, making it capable of integrating functionalities that will vary from simple traffic control to information providing via internet and/or mobile phone.

Acknowledgments

Filipe Lopes was supported by CNPq (project SincMobil). CNPq also partially supports Ana L. C. Bazzan. Denise de Oliveira's master scholarship was funded by CAPES.

References

1. Bazzan, A.L.C., Wahle, J., Klügl, F.: Agents in traffic modelling – from reactive to social behaviour. In: Advances in Artificial Intelligence. Number 1701 in LNAI, Berlin/Heidelberg, Springer (1999) Also appeared as Proc. of the U.K. Special Interest Group on Multi-Agent Systems (UKMAS), Bristol, UK.

2. Andriotti, G.K., Bazzan, A.L.C.: An object-oriented microscopic traffic simulator. In: Proceedings of the XXVIII Latin-American Conference on Informatics (CLEI 2002), Montevideo, Uruguay (2002)
3. Klügl, F., Bazzan, A.L.C.: Route decision behaviour in a commuting scenario. Journal of Artificial Societies and Social Simulation **7** (2004) http://jasss.soc.surrey.ac.uk/7/1/1.html.
4. Nagel, K., Schreckenberg, M.: A cellular automaton model for freeway traffic. J. Phys. I France **2** (1992) 2221
5. Nagel, K., Wolf, D.E., Wagner, P., Simon, P.: Two-lane traffic rules for cellular automata: A systematic approach. Phys. Rev. E **58** (1998) 1425
6. Esser, J., Schreckenberg, M.: Microscopic simulation of urban traffic based on cellular automata. International Journal of Modern Physics C **8** (1997) 1025
7. Esser, J., Neubert, L., Wahle, J., Schreckenberg, M.: Microscopic online simulation of urban traffic. In Ceder, A., ed.: Proc. of the 14th ITS, Pergamon (1999) 535–554
8. Rickert, M., Nagel, K., Schreckenberg, M., Latour, A.: Two lane traffic simulation on cellular automata. Physica A **231** (1996) 534
9. Wagner, T., Garvey, A., Lesser, V.R.: Complex goal criteria and its application in design-to-criteria scheduling. In: Fourteenth National Conference on Artificial Intelligence. (1997)
10. Knospe, W., Santen, L., Schadschneider, A., Schreckenberg, M.: A realistic two-lane traffic model for highway traffic. J. Phys. A (2003) 3369–3388
11. Nagel, K., Esser, J., Rickert, M.: Large–scale traffic simulations for transport planning. Ann. Rev. Of Comp. Phys. (2000) 151–202
12. Hafstein, S.F., Chrobok, R., Pottmeier, A., Schreckenberg, M., Mazur, F.C.: A high-resolution cellular automata traffic simulation model with application in a freeway traffic information system. Comp-aided Civil Eng **19** (2004) 338–338