A Cellular Automata Framework for Studying Expandable Traffic Flow Models

Ourania Hatzi
Stephanos Thomas
Vassilis Dalakas
Mara Nikolaidou
Dimosthenis Anagnostopoulos

Harokopio University of Athens
70 El. Venizelou Str, 17671
Athens, GREECE.

Keywords: Traffic Flow Modelling, Cellular Automata, Micro-simulation, MATLAB

Abstract

The introduction of methods from statistical physics for traffic flow modelling, such as cellular automata models, enabled the faster than real time simulation of large traffic networks. A key issue faced by such efforts is the expandability of the network model. This paper discusses the development of a cellular automata framework for micro-simulation of vehicle traffic flow in road networks and highways, which was implemented on MATLAB. Attention has been paid to the modularity of the proposed model, in order to provide not only expandability, but also parameterization and independency from simulation rules. Our goal is to accomodate the study of traffic critical properties by including several parameters in the model and facilitate the construction of hierarhical modular traffic models and the experimentation with alternative simulation rulesets. The first step of this integrated methodology is to validate the proposed framework, by presenting numerical results and comparisons.

1. INTRODUCTION

A wide range of different models for traffic flow has been developed by scientists from mathematics, physics and engineering to address different traffic problems. A prominent model among these is the cellular automata model. A cellular automaton (CA) is a discrete model consisting of an infinite, regular grid of cells, each in one of a finite number of states [1]. The basic model for traffic simulation is a single-lane model which is discrete in space and time [2]. The driving behaviour is specified by only a few simple rules without losing the essential phenomena in traffic dynamics. Complex traffic models are constructed by composing simple single-lane models, to represent, for example, multi-lane traffic and different types of vehicles. Cellular automaton model for traffic flow belongs to microscopic simulation methods. Micro-simulation is the dynamic and stochastic modelling of individual vehicle movements, within a system of transportation facilities. The motion of each vehicle of a traffic stream is considered depending on the motion of the preceding vehicle and the individual driving behaviour can be taken into account [3].

As changes in vehicle technology occur, i.e. new types of automatic or semi-automatic vehicles are being developed, and driver behaviour alters over time, the simulation rules for traffic flow of the past may have to be adjusted. The accomodation of such features would facilitate the exploration of traffic flow on a localized basis, where classical single-lane models might not be accurate enough. Furthermore, most traffic management projects require simulation study before realization, in order to be cost-effective. Therefore, great interest lies in the ability of simulation in contemporary urban complex environments [4-6]. So far, there is a variety of tools employing micro-simulation, such as TRANSIMS [7], which facilitate the parametric definition of road network models. However, model expandability and parametric composition is not provided, while simulation rules are embedded within the model and cannot be modified. In this paper a framework based on CA is proposed that enables the defintion and study of hierarhical, modular, parameterizable models, and facilitates the modeller's intervention not only to road network models but to simulation rules as well. The development of a software tool based on the proposed framework in MATLAB environment is also discussed.

The main obstacle encountered in our effort so far was the validation of the results due to the lack of adequate references and empirical results for complex models. This is not the case for simple, single lane models, where validation is not an issue, as plenty of works can be found in the literature [8-11]. Thus, it was decided to implement such models in the proposed system to accomodate validation. This first step was considered successful since simulation results approximated empirical results, which established the reliability of the tool, as discussed in the following sections. Although the proposed tool has the capability to accomodate complex models, validation is not yet performed, due to the aforementioned lack of empirical results. We are currently working towards this direction, gathering empirical results for an autonomous inner city network, while the exploitation of alternative simulation rules, depicting localized driving behaviour in the city of Athens, is the imminent goal of this effort. However, the provision of the parametrical environment presented in the following, which is able to embody the above features, is essential for the efficient exploitation of alternative traffic model scenarios.

The rest of the paper is organised as follows: Section 2 reviews the Cellular Automaton Model, Section 3 presents our framework and implementation efforts, as well as validation results, and Section 4 discusses future work and goals.

2. THE CELLULAR AUTOMATA MODEL

The proposed approach was based on the cellular automaton model for freeway traffic proposed by Kai Nagel and Michael Schreckenberg [1]. The model is defined by a vector which is comprised of sites, also known as *cells*. The size of each cell is usually set to 7.5 m. Each cell may either be empty, or occupied by a vehicle. In the original model, all vehicles are of the same size and have integer velocities between zero and u_{max}, which is a parameter of the model. An ordinary, realistic value for u_{max} is 5. Update of the vehicle position involves three rules, which, at each time step, determine the velocity of the vehicle. These rules are the following:

- *Acceleration*: if the velocity of the vehicle has not reached the maximum velocity u_{max}, it is increased by 1.
- *Slowing down*: if the gap between a vehicle and its preceding vehicle, counted in cells, is less than the velocity of the vehicle, the velocity is descreased to gap-1.
- *Randomization*: With a probability pdec, the velocity of a vehicle is decreased by 1.

The above rules are executed and velocity is calculated for each vehicle present in the system. Eventually, the position of each vehicle is updated according to this velocity. In order for the results to correspond to real-life measurements, each time step is considered to be 1 sec.

The proposal of the model applied two diferent types of boundary conditions to it: closed system and open system. Closed system referred to traffic in a circle, where no vehicles enter or leave the system, therefore the number of the vehicles is constant. On the other hand, an open system is more realistic for modelling urban traffic, since it assumes that there are entry and exit points where vehicles enter and leave the system, respectively. The second type of boundary conditions will be used in the proposed model as well.

In order to demonstrate the structure of a road and the effect of the above rules to the velocity of the vehicles, the following example is presented (Fig. 1).

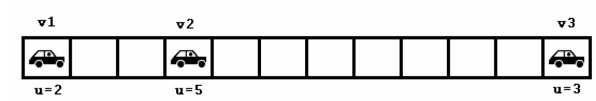

Figure 1. An example of a cellular automata model for traffic flow on a single-lane road.

The road shown above is comprised of twelve cells, three of which are occupied by vehicles with velocities as shown. The pdec for this example is set to 0.33, therefore, an average of 1 out of 3 vehicles will be forced to reduce its velocity. Applying the rules will result in the first vehicle (v1) increasing its velocity by 1 but then decreasing it again because of the small gap between the first and the second vehicle (v2). The second vehicle has already reached u_{max} therefore it will not further increase its velocity. In case randomization requires this vehicle to decelerare, its velocity will be reduced to 4. Finally, the third vehicle (v3) has a velocity that allows it to leave the system. The updates in the vehicle positions will take place, simulation time will advance and the next simulation step will begin by applying again the same rules.

The most important measures in the model are traffic flow and traffic density, which, on a single lane, are computed from the local measurement data. The traffic flow is defined as the number of vehicles per time unit and is calculated as follows:

$$q_j(x,t) = \frac{\Delta N_j(x,t)}{\Delta t}$$

The mean traffic density is defined as the number of vehicles per length unit and can be calculated approximately from the traffic flow and the mean velocity :

$$\rho_j(x,t) = \frac{\Delta N_j(x,t)}{\Delta x_j} \approx \frac{\Delta N_j(x,t)}{\Delta t \cdot v_j(x,t)} = \frac{q_j(x,t)}{v_j(x,t)}$$

The approach $\Delta xj \approx \Delta t \cdot vj\ (x, t)$ is valid only if all detected vehicles are moving with approximately the same velocity during the time interval Δt.

3. FRAMEWORK AND IMPLEMENTATION

The proposed framework is an integrated environment aiming at enhancing the expandability and parameterization of road network models and the manipulation of alternative simulation rules. Its architecture is modular, in order for the different parts to be independent from each other and easily modifiable. Simulation result exploitation and comparison features are embedded within the framework (Fig. 2).

The modularity of the approach and the proposed representation of road network models facilitate scalability when studying large road networks. The simulation rules remain the same, while the road networks can be arbitrarily large. Furthermore, new parts can be progressively added, at any time, to the existing road networks.

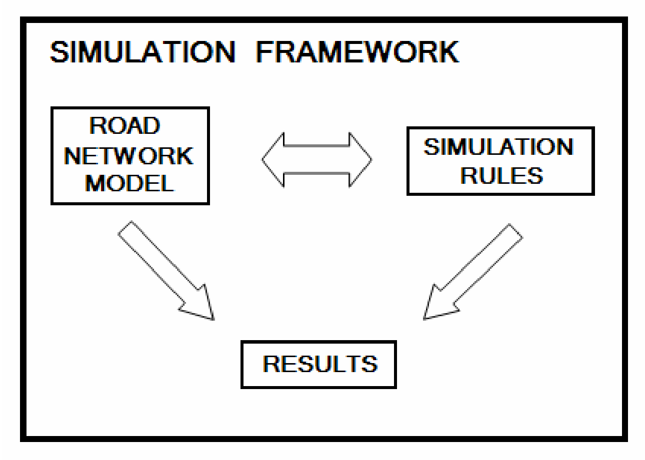

Figure 2. Proposed framework architecture.

The framework facilitates:

- Parametric definition of road network models.
- Easy modification and expansion of existing road network models.
- Adaptive ruleset that can easily be altered.
- Parametric simulation.
- Visualization of simulation results in charts and comparison with empirical data.

3.1. Conceptual model and network representation

The proposed methodology that will be presented in this section results in the construction of an integrated modular system for modelling of arbitrary road networks, implemented in MATLAB.

The conceptual model of the road network, as designed in the proposed system, consists of roads and junctions. Each road is divided into lanes and parts, and each part into cells. Junctions may be simple, in which case regular traffic laws indicate priority of vehicles, or may have traffic lights, in which case their indications are taken into account. Moreover, the system has entry and exit points, where vehicles enter and leave the system and statistics are gathered. As far as properties of the road parts are concerned, each part has a predefined capacity, which indicates the number of cells, and a deceleration factor, which indicates how often the vehicles slow down in this part of the road in a probabilistic way.

Each vehicle is represented as a separate entity which has its own parameters, such as velocity. These parameters are used in combination with the simulation rules and other safety parameters of the system, in order to produce the updates to the position of the vehicle. Vehicles are inserted in the system at entry points, according to a predefined arrival rate given by the user in advance as a parameter of each entry point. While they are in the system, they can occupy exactly one cell at any given time, as they are all considered to be of the same size. Finally, they leave the system at exit points, affecting the appropriate statistic measures.

When it comes to implementing the above model in MATLAB, a combination of arrays and other variables is used. The first step is to create a road network. This can be achieved through answering simple questions about its structure. To demonstrate this, the following simple road network is presented as an example (Fig. 3).

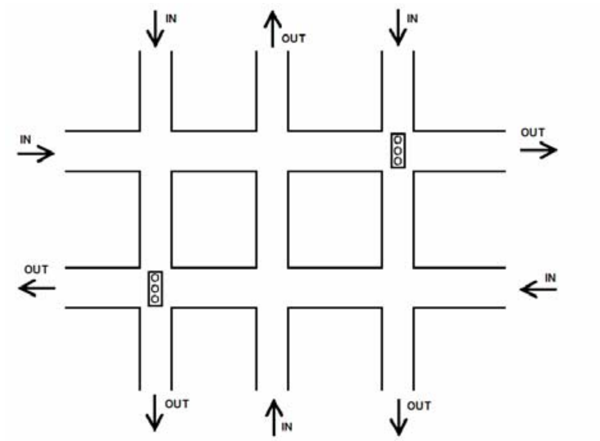

Figure 3. An example road with entry and exit points marked.

The network consists of five single-lane roads and junctions between them, two of which have traffic lights. The direction of the roads as well as the entry and exit points of the network are also marked.

Dividing the network into roads, parts and junctions results in the following schema (Fig. 4).

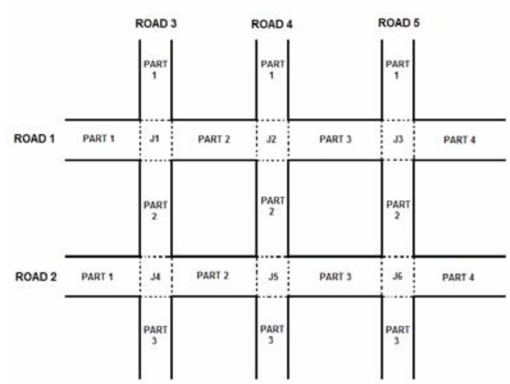

Figure 4. The representation of the road network in terms of cellular automata models.

A partial example demostrating the insertion of information for the first part of Road 1 to MATLAB follows (Fig. 5):

```
How many roads are there in the network?  5

How many parts do you have in road 1?  4

Is part/lane 1 an input point for the
network? (no=0 / yes=1):  1

Enter interarrival time (lambda):  1.5

Enter part/lane 1 capacity?  50

Which part/lane is on the right of
part/lane 1 (0 for no part/lane):  0

Which part/lane is on the left of part/lane
1 (0 for no part/lane):  0

What is the type of the intersection at the
end of this part/lane?
(roundabout=0, traffic light=1, dead end=2,
other lane=3, network end=4):  3

What is the id of the intersection at the
beginning of this part/lane?  0

What is the id of the intersection at the
end of this part/lane?  1
```

Figure 5. Example of road data input.

The information provided to the system includes the interarrival time for this part, since it is an entry point for the system, the capacity of the road part, its direction, its neighbouring lanes, if any, and the junctions at the beginning and end of the road part. The same procedure has to be followed once for every part in the system. Then, it can be saved and modified at a later time, used to run a traffic simulation with the simulation rules embodied in the system, or used with a different set of simulation rules.

It should be noted that the road network model is saved as a simple text file in order to be easily expanded and modified. For example, if the modeller wishes to add a crosswalk in part 3 of road 2, this part can be splitted into two parts, and the new information can substitute the old data in the text file, without any further changes. Likewise, new roads or road parts can be added to the network by appending new data to the text file.

Traffic light information is provided in a similar way. In the following example, information is given about the traffic light on Road 1 (Fig. 6).

```
Give the direction for the traffic light in
road 1 part 1 : 'E'

Give the category of this traffic light
(TLmain=0, TLturn=1, TLprio=2): 0

Give the time interval (sec) where this
traffic light is green: 40

Give the time interval (sec) where this
traffic light is orange: 5

Give the time interval (sec) where this
traffic light is red: 35

Give the point in time where this traffic
light turns green for the first time: 0
```

Figure 6. Example of traffic light data input.

3.2. Simulation

Each step of the simulation consists of three sub-steps:

- Importing the vehicles
- Updating vehicle positions
- Advancing the simulation time

The above steps comprise different modules of the system. Therefore, each one is independent from the other, thus enabling the modeller to experiment with different sets of simulation rules.

The implemented model in the proposed system is the original one by Nagel and Schreckenberg. The same simulation rules and conventions for cell length and maximum speed are followed. At each time step, the rules that determine the velocity of each vehicle are acceleration, slowing down and randomization, and in the end the position of the vehicle is updated according to this velocity. Randomization probability, referred to as pdec, is introduced in the system as a parameter of each road part, therefore each part may have a different probability.

The above rules are encoded in the system and executed consecutively. As long as a vehicle stays in the same part of the road, updating its position is a trivial task. However, attention must be paid to take the appropriate actions when a vehicle approaches the end of the road part it is moving in, and its velocity enables it to leave this road part. Priority in junctions and traffic light indications must be taken into account. Moreover, there is the possibility that the vehicle has reached a network exit point, therefore statistics must be updated.

3.3. Validation and results

In order to validate the proposed methodology and corresponding tool, a series of steps has to be performed: First of all, simple traffic models have to be simulated using the rules of the original CA model by Nagel and Schreckenberg. Moreover, numerical results have to be presented and compared to empirical results. A simple network that consists of one road with two parts is the example which was simulated in order to obtain the results demostrated. Simulation was performed for two different values of the probability factor pdec, which serves randomization purposes and shows how often vehicle velocity is decreased (Fig. 7). The measure of interest in this example is the number of vehicles that went through the network in each simulation step.

Taking into account the results from the simulation of simple models and their proximity to empirical results, the reliability of the proposed tool was established. Thus, further experimentation with more complex traffic models can be exploited.

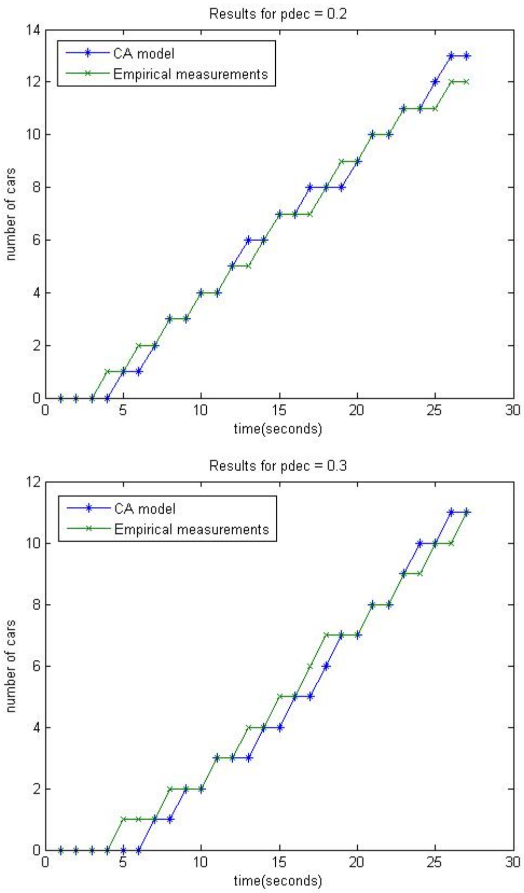

Figure 7. Comparison of simulation and empirical results.

4. CONCLUSION AND FUTURE WORK

The proposed framework accomodates the definition and experimentation of hierarhical modular models and simulation rules in an independent fashion. We are currently working on a case study of an autonomous inner city network in downtown Athens, for which empirical results are being gathered. This will allow the validation of complex traffic networks, while the exploitation of alternative simulation rules, depicting localized driving behaviour in the city of Athens, is also explored. Existing CA rules are enriched in order to ensure accuracy.

The next steps are summarized as following:

- The ability to incorporate other types of metrics in the system, such as environmental ones.
- Combination of vehicle and pedestrian traffic in a single simulation tool. This requires pairing simulation rules from both fields [12]. A useful example of the application of such simulations would be the determination of the exact location of crosswalks in urban areas, in order to optimize safety.

Acknowledgments

This research was supported in part by Pythagoras program (MIS 89198) co-funded by the Greek Government (25%) and the European Union (75%).

References

[1] von Neumann, J. 1966, *The Theory of self reproducing automata*. Edited by A. Burks, Univ. of Illinois Press, Urbana

[2] Nagel K and Schreckenberg M. 1992, “A cellular automaton model for freeway traffic”. J. Physique I 2 2221

[3] Knospe W., Santen L., Schadschneider A. and Schreckenberg M. 2001, “Single-vehicle data of highway traffic:microscopic description of traffic phases”. Phys. Rev. E at press

[4] 58th Traffic and Safety Conference, May 9-11,2007, Columbia, Missouri. *Available online via http://muconf.missouri.edu/traffic/index.html [accessed April 19, 2007]*.

[5] Ontario Traffic Conference, Ontario. *Available online via http://www.otc.org/otcweb2 [accessed April 19, 2007]*.

[6] European Transport Safety Council. *Avalable online via http://www.etsc.be/home.php [accessed April 19,2007]*.

[7] TRANSIMS, *The TRansportation ANalysis and SIMulation System project*. Los Alamos National Laboratory.

[8] Simon P. M. and Nagel K. 1998, “Simplified cellular automaton model for city traffic”. Phys. Rev. E 58, 1286 – 1295

[9] Fouladvand M. E., Shaebani M. R., Sadjadi Z., 2004, “Intelligent controlling simulation of traffic flow in a small city network”. Journal of the Physical Society of Japan, vol. 73, no11, pp. 3209-3214.

[10] Ping-Hua H., Ling-Jiang K., Mu-Ren, L. 2002 “A study of a main-road cellular automata traffic flow model”. Chinese Physics Beijing Journal, VOL 11; PART 7, pages 678-683.

[11] Barlovic R., Esser J., Froese K., Knospe W., Neubert L., Schreckenberg M., Wahle J., 2001, “A cellular automaton traffic flow model for online simulation of traffic”. Parallel Computing, Volume 27 , Issue 5, Special issue on cellular automata: from modeling to applications, Pages: 719 – 735.

[12] Blue, V.J., Embrechts, M.J., Adler, J.L. 1997, “Cellular automata modeling of pedestrian movements”. IEEE International Conference on Systems, Man, and Cybernetics 1997, 12-15 Oct 1997 Page(s):2320 – 2323 vol.3