A Parallelizable Probabilistic Cellular Automata Model for Highway Traffic Simulation: Periodic and Open Boundaries Conditions

Marcelo Zamith, Regina Célia P Leal - Toledo, Mauricio Kischinhevsky, Esteban Clua, Diego Brandão, Anselmo Montenegro and Edgar B. Lima

Federal Fluminense University, Passo da Pátria Street, 156, Niterói, Brazil

Abstract. A Cellular Automata (CA) model for traffic roads based on Nagel and Schreckenberg's model is presented. The probabilistic model, investigated in this work, models the individual behavior through an extension of the base model in which a continuous probability function is applied to define an expectancy of a follower with respect to its leading vehicle. This anticipatory feature leads to a counter flow velocity tunning. Moreover, two issues regarding the proposed model are discussed: the use of open boundary conditions (OBC) as well as its parallelizability. Thus, simulations are developed and discussed herein and compared to real data fundamental diagrams.

1 Introduction

The traffic flow increasingly affects the quality of life in modern societies. Traffic jams and their companion pollution generation, as well as mental stress effects, are some of the reasons to seek a better understanding of traffic flows. These help understand why the subject has been receiving so much attention for the last decades. Several attempts were made to mathematically model traffic flows. These models are organized as either macroscopic or microscopic. Among them, a specific microscopic model named Cellular Automata has been employed with good results, due to the fact that it mimics the whole spectrum of vehicles' movement. Some features about traffic control's strategies are discussed in [1], an on-line service using a CA model has been made available [2], so as to describe the freeway traffic in North Rhine-Westphalia. Although CA models have been applied for managing, describing and understanding the traffic characteristics, their application in this context is recent and a perfect model is yet to be developed. One of the main advantages of CA models is that they are easily implemented, lead to moderate computational cost and keep the basic features of the phenomena ([3], [4],[5]). In [6] open boundary conditions policies and inflow/outflow strategies are discussed.

Among the objectives of this work one has the improvement of the flow-density relation, leading to real data resembling transition to jammed flow. Moreover, due to the explicit nature of the algorithm, which allows the algorithm to be decoupled, i.e., another objective arises. This refers to the splitting of the computational procedure in two steps: the first is responsible for defining the new velocity and the second updates all vehicles' positions, as described below.

A continuous probability function properly defines the driver's average behavior. Moreover, during the determination of velocities' step, the counter flow velocity tunning gives rise to a recursive procedure that provides the new velocities' definition while preventing the occupation of a single cell by more than one vehicle. This is performed in such a way that velocities are adjusted to appropriate values even in the event of a velocity reduction by a leading vehicle. The inflow and outflow of vehicles to/from the highway is discussed, and a preliminary evaluation of the computation cost of parallel of proposed model.

This work is organized as follows: in Section 2, related work on cellular automata models applied to traffic highways is discussed; in Section 3 some concepts of traffic-flow theory are reviewed; Section 4 describes the proposed algorithm; Section 5 presents some test cases. Concluding remarks about the proposed method are discussed in Section 6.

2 Related Work

The class of Cellular Automata models applied for traffic problems is often organized in two sets, so-called deterministic and probabilistic models. Among them, Rule-184 [7] is one of the most known deterministic models. Fukui and Ishibashi [8] modified this model in order to introduce more velocity variations, supporting a variation of up to 5 cells per unit time simulation. The uncertainty introduced by probabilistic models tries to properly represent the behavior of drivers, so as to improve the flow-density relation. Nagel and Schreckenberg [9] pioneered in the study of this class of models. Their proposed algorithm, NaSch, has been largely employed, has become the basis for many improvements ever since. The algorithm consists of simple rules: all vehicles try to speed up to the maximum velocity allowed by the flow of vehicles or the road's speed limit. The uncertainty introduced in the algorithm tries to mimic the behavior of a driver in that this shall keep the vehicle's velocity or, without any apparent reason, simply reduce it. A subsequent adjustment of each vehicle's speed considers its distance to the one immediately ahead.

Slow-to-start models form a subset of the set of probabilistic models. Their main purpose is to represent the driver's eagerness for restoring his velocity when his vehicle is stopped, i.e., the inertia effect of the vehicles. Although slow-to-start models can represent the meta stability phase, drivers' conservative behavior leads to jammed flow with lower density than empirical data indicate.

Among the slow-to-start, one has the following: models discussed in [10] and [11] consider the space ahead after some simulation time, in order to restore the car's velocity. In [13] and [14] the flow adjustment method defines that a driver considers the space between vehicles ahead and adds the free space between them. The concept adopted in such method is that every driver considers that its leading vehicle (that is, the one immediately ahead) will move in the same velocity as in the previous time frame. Thus, a model with an anticipatory feature.

The anticipatory policy represents a level of expectancy that a driver has in relation to its leading vehicle. It is given by the distance between two vehicles plus the potential distance that the leader has for moving. The level of expectancy is quantified by a randomness related with the velocity of the leading vehicle. The other rule of anticipation delineates the situation when a (driver's) vehicle has a high level of expectancy that its leader is going to keep its velocity at next time frame and this does not happen.

3 Traffic-Flow Theory

The behavior of traffic is analyzed here on the basis of a set of variables: density describes the number of vehicles per unit length of a highway at some time (see Eq. 1) where n is the number of vehicles as well as L represents a part of the highway.

$$\rho = \frac{n}{L} \tag{1}$$

The average velocity, i.e., the averaged sum of velocities, is established as (Eq. 2)

$$\overline{v} = \frac{\sum_{i=1}^{n} v_i}{n} \tag{2}$$

Flow is defined as the number of vehicles that pass by a specific point of the highway per unit time, as in Eq. 3

$$J = \rho\overline{v} \tag{3}$$

By using Eqs. 2 and 3, J can be re-written as in Eq. 4.

$$J = \frac{\sum_{i=1}^{n} v_i}{L} \tag{4}$$

The equations, as described previously, show how to compute the variables at a particular time. However, when one tries to simulate realistic scenarios, many time steps should be used in order for the complexity of the phenomena to emerge. Thus, the equations shall be re-written for greater adequacy, as in Eq. 5, where m is the number of vehicles that pass at a highway section and T is a time period.

$$J = \frac{m}{T} \tag{5}$$

Moreover, the average velocity is re-written, so as to consider the vehicles that pass at a highway section, as in Eq.6.

$$\overline{v} = \frac{\sum_{i=1}^{m} v_i}{m} \tag{6}$$

In order to obtain the average density at a highway section, Eqs. 5 and 6 are replaced by Eq. 7, that is

$$\overline{\rho} = \frac{m^2}{T\sum_{i=1}^{m} v_i} \tag{7}$$

The equations above represent the behavior of traffic flow on a highway by means of vehicle counts over long periods of time. Typically, the analysis of traffic flow is performed by constructing the corresponding fundamental diagram. This depicts how flow and density relate. The theoretical model is illustrated in Fig. 1 and real data is presented in Fig. 2.

Fig. 1. Theoretical fundamental diagram

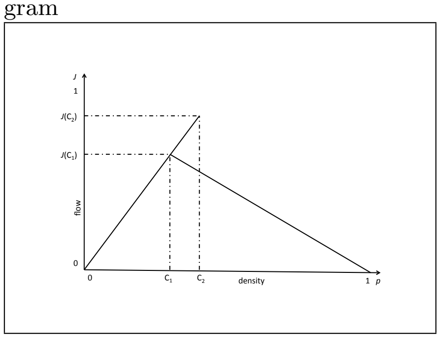
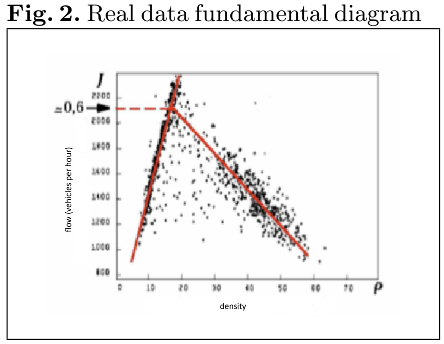


Fig. 2. Real data fundamental diagram



The fundamental diagram exhibits three well defined phases, namely: the first one represents the free-flow and corresponds to the region of low to medium density and weak interaction between vehicles. In this phase, the vehicles can move almost at the highway's speed limit, and the flow increases linearly with increasing density, density ranging in $0 \leq \rho \leq c_1$. The second phase presents medium and high density, the flow shall behave either free or jammed, i.e., the phase $c_1 < \rho < c_2$ shows a flow that is not defined only by density, but by the interaction between the vehicles instead. This phase is named the metastable phase. The last phase is $\rho > c_2$, and represents the jammed flow, where an increase in density forces a decrease in the flow.

Therefore, usable models have to represent both qualitatively (mandatory) as well as quantitatively (desirable) the fundamental diagram, making it possible for a coherent analysis of all variables involved in the process to be carried on.

4 The Proposed Model

In this section, NaSch's model is briefly reviewed, and its algorithmic description, Algorithm 1, provides a framework for the presentation of the new probabilistic model, which is based on NaSch's model and extends the anticipatory concept proposed by [14].

For NaSch's model, variables space and time are both discrete, so $t \in \mathbf{N}$, $x_i^t \in \mathbf{Z}$; The highway is considered to have periodic boundary condition, i.e., the position X is the same as position $X + L$ where L is the length of the circuit. Likewise, the ith vehicle is the same as the $(i + N)$th vehicle, where N is the number of vehicles. Each cell represents a length space of 7.5 meters and the time evolution scale is measured in seconds. Besides, the variable (v_i^t) denotes the velocity of the ith vehicle in time instant t, in cells per time, while x_i^t denotes the spatial position. The distance between two vehicles in a time instant is represented by d_i^t, and the maximum speed allowed is given by v_{max}. The probabilistic character of the model is represented by p and p_m. The first is chosen from the uniform distribution and the latter is an initial parameter. They represent the probability for a vehicle to preserve its speed at the next time instant $t + 1$.

Algorithm 1 NaSch's Algorithm

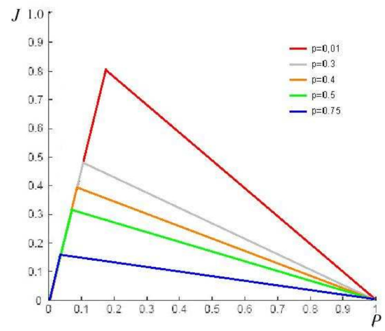
for from the last to the first vehicle **do**
    $v_i^t \leftarrow min[v_i^{t-1} + 1, v_{max}]$
    $d_i^t = x_i^{t-1} - x_{i-1}^{t-1} - 1$
    $p \leftarrow$ randomize number $\in [0, 1]$
    if $v_i^t > d_i^t$ **then**
        $v_i^t \leftarrow d_i^t$
    end if
    if $p \leq p_m$ and $v_i^{t-1} > 0$ **then**
        $v_i^t \leftarrow v_i^{t-1} - 1$
    end if
    $x_i^t \leftarrow x_i^{t-1} + v_i^t$
end for

Even though it does not generate the meta stability phase, NaSch model presents satisfactory results. The Fig 3 shows the effects of different values of p_m. These define in which density the free flow behavior switches into the jammed one.

Algorithms 2 and 3 describe the probabilistic model proposed. The main characteristic of the model is that it is explicit in time. Moreover, the counter flow velocity tunning ensures the correct velocity definition for each vehicle at each simulation instant, so that the occupation of a cell by two differnt vehicles is forbidden.

The model's algorithm is splitted in two stages: the first one is responsible for the correct velocity definition, the second updates position of each vehicle based on the velocity defined in the previous stage. Lines $2 - 14$ of algorithm 2 describe the first stage while the second is represented by line 17 of the same algorithm. Since velocity definition and position updating are independent tasks,

Fig. 3. Fundamental diagram of NaSch's model



they are not constrained by the update direction, that is, free access to the data structure is allowed.

Algorithm 2 The Proposed Algorithm - main algorithm

for all vehicles **do**
  $v_i^t \leftarrow min(v_i^{t-1} + 1, v_{max})$
  $p \leftarrow$ randomize number $\in [0, 1]$
  if $p \leq p_m$ and $v_i^{t-1} > 0$ **then**
    $v_i^t \leftarrow v_i^{t-1} - 1$
  end if
  $\alpha_i^t \leftarrow normal(\mu, \sigma)$
  $d_i^t = x_i^{t-1} - x_{i-1}^{t-1} - 1$
  $d_{is}^t \leftarrow d_i^t + [v_{i+1}^{t-1} \times (1 - \alpha_{i+1}^{t-1})]$
  if $v_i^t > d_{is}^t$ **then**
    $v_i^t \leftarrow d_{is}^t$
  end if
  if $[v_i^{t-1} \times (1 - \alpha_i^t)] > d_{is}^t$ **then**
    call Solver Vehicles' Cluster (i)
  end if
end for
for all vehicles **do**
  $x_i^t \leftarrow x_i^{t-1} + v_i^t$
end for

The explicit procedure of reverse adjustment of velocity is indicated by algorithm 3, which is part of algorithm 2 . This procedure is evoked when any vehicle reduces its velocity unexpectedly, making the algorithm redefined the velocities of all vehicles affected by this fact.

The policy employed for the anticipatory procedure adopts two different distances: the distance between the vehicle analyzed i and its leader (d_i^t); the ef-

Algorithm 3 The Proposed Algorithm - Solver Vehicles' Cluster algorithm

1: **while** $v_i^t > d_i^t$ **do**
2: $\quad\alpha_i^t \leftarrow normal(\mu, \sigma)$
3: $\quad d_i^t = x_i^{t-1} - x_{i-1}^{t-1} - 1$
4: $\quad d_{is}^t \leftarrow d_i^t + [v_{i+1}^t \times (1 - \alpha_{i+1}^t)]$
5: $\quad$**if** $v_i^t > d_{is}^t$ **then**
6: $\qquad v_i^t \leftarrow d_{is}^t$
7: $\quad$**end if**
8: $\quad$**if** $[v_i^t \times (1 - \alpha_i^t)] > d_{is}^t$ **then**
9: $\qquad$call Solver Vehicle's Cluster (i)
10: $\quad$**end if**
11: $\quad i \leftarrow i - 1$
12: **end while**

fective distance, which considers the distance between two vehicles increased by an expectation term (d_{is}^t). This term is an expectation level of how much the next vehicle will move, analyzing its velocity in the previous time instant. It is described by lines 9 and 4 of algorithms 2 and 3, respectively.

The procedure described in 3 is applied when the condition $\alpha_i v_i^{t-1} > d_{is}$ holds. This is used to restrict the number of vehicles that need to have their velocities updated. Then, the highway is divided in regions composed by clusters of vehicles that will be affected by sudden changes in other vehicles' velocities.

A cluster is identified by collecting members along the direction opposing the flow. One begins with a vehicle that does not update its velocity until the vehicle that is not affected by the decrease of the velocities of the vehicles belonging to the cluster. Additionally, the signalization process may be propagated as any vehicle in the cluster may move with a velocity which differs from the one it was expected to. This will result in the creation of subclusters inside the main cluster. This is computationally treated by using a recursive strategy. In a cluster where one vehicle does not move as expected, the iterative procedure is applied within all cluster and its subclusters.

The open boundary condition permits the use of different values for incrementing and decrementing vehicles at the highway. The velocity definition of the vehicle that is being inserted is another parameter that should be considered, i.e., the inflow boundary condition. Three approaches for defining the vehicles' velocity are adopted in this work: the first, the vehicles are added to the highway with velocity equal to $1 cell/s$; the second approach, the vehicles' velocity are equal to maximum velocity; in the last one, the vehicles' velocity is set according to the average velocity of outflowing vehicles. The Figs. 4 and 5 show the fundamental diagram of open boundary condition with inflowing velocities (V_i) equal to 1 and V_{max}, respectively. The fundamental diagram for the approach that employs velocity average of the outflowing vehicles is illustrated by Fig. 6.

The Fig. 11 shows the space time diagram. Thus, the flow is around to 35% of highway, some platoons of vehicles of formed and dissolved when the vehicles

Fig. 4. Flow-density diagram / $p = 0.30$ / $N(\mu = 0.5, \sigma = 0.1)$ / $V_i = V_{max}$

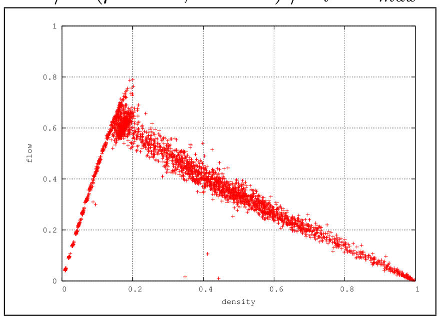
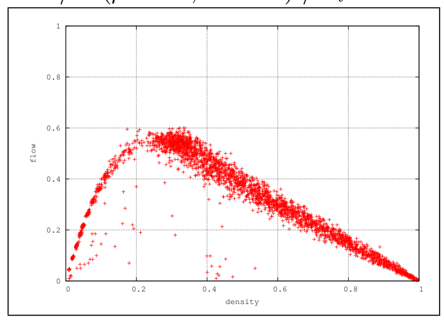


Fig. 5. Flow-density diagram / $p = 0.30$ / $N(\mu = 0.5, \sigma = 0.1)$ / $V_i = 1$



leave the highway and are replaced by inflowing others. This is a more realistic representation of reality than the one provided by periodic boundary conditions.

4.1 The Rejection Technique - Monte Carlo

In this work a recently proposed policy for anticipation is employed. A continuous probability function is used, representing a flexible and realistic model, instead of the fixed probabilities that describe all vehicles on the highway to behave equally as average.

The rejection technique employed herein shall be described by algorithm 4:

Algorithm 4 Rejection Rule (μ, σ)

```
1: repeat
2:     x ← randomize
3:     y ← randomize
4:     p(x) ← 1/(σ√(2π)) e^(−(x−μ)²/(2σ²))
5: until y > p_x
6: return ← x
```

4.2 Preliminary Evaluation of the Computational Cost of Parallelization

The scalability of a problem is a crucial issue when one discusses techniques which are to be applied to problems with many variables. The model presented, and its future developments, shall be applicable to tens of thousands of vehicles on a road network. The time explicit method allows the cells to be divided in subsets which can be updated in parallel.

Fig. 6. Fundamental diagrams - open boundary condition

Fig. 7. $p = 0.10$ - Normal

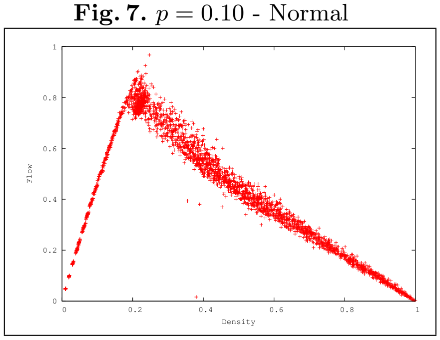


Fig. 8. $p = 0.10$ - Uniform

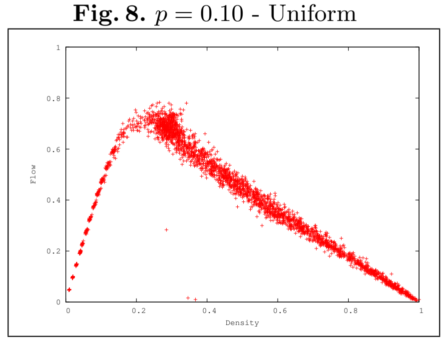


Fig. 9. $p = 0.40$ - Normal

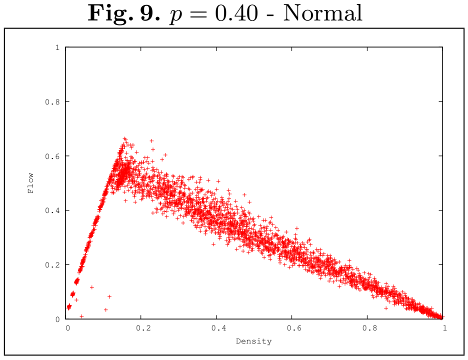
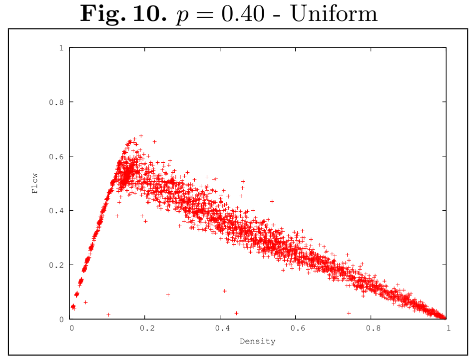


Fig. 10. $p = 0.40$ - Uniform

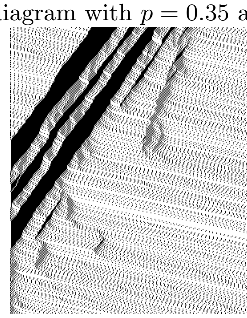


The proposed parallel model is composed by a domain partition, where each cluster node has a subset of cells of the problem. Moreover, each node must change information about boundaries (leftmost and rightmost) with its neighboring nodes. In order to guarantee the correct change information, two buffers are created: one for the leftmost cells and the other for rightmost cells, subset inflow and outflow respectively. The size of the buffers is based on the maximum speed for all cells, i.e., the maximum velocity of the highway. Their size is defined *a priori* for the simulation, and is a static parameter for the whole process.

Since data for velocity, behavior expectancy and vehicles' position are required during a simulation, the data structure should store information about these attributes. They shall require, e.g., 16 bytes per cell. Consider that each buffer has length 80 bytes, where (16 bytes per cell $\times 5$ maximum velocity $\times 2$ boundaries condition). Thus, the message from a node to its neighborhood is sent for each simulation instant.

The parallel approach of the proposed model must obey two constraints, which are applied only for the counter flow velocity tunning procedure. Before re-adjusting the velocity, all speeds must have been already defined. Thus, when

Fig. 11. Space time diagram with $p = 0.35$ and normal distribution



this procedure is invoked for one vehicle, it can remain in wait state until all its trailing vehicles within the cluster have their velocity set. The effectiveness of the parallelization is ensured by synchronization through a barrier. Furthermore, the update stage is completely parallelizable due to the fact that each vehicle has its position updated independently of the other.

As a preliminary evaluation of the computational cost of parallelization of the proposed model, some data which correspond to a sample architecture are chosen and the subsequent computational costs under very simple assumptions for a parallel implementation are obtained. These give rise to estimates which suggest a promising performance for the parallel version of the computational model.

The amount of message passing in the network depends on granularity of cells per node. Considering the cluster is composed by processors Intel Core 2 Quad 2.4 GHz CPU with gigabit architecture network and running MPI.

According to [15], the latency for MPI short message (MTU equal 1500 bytes) is 22 μs and the bandwidth up to 1.9 Gb/s. There are four messages communicated for each time step simulation, i.e., message passing cost per time simulation is $320 bytes$. Additionally, the cost of latency is 88 μ/s. Hence the cost per message passing is approximately 22μs plus $3.92136 \times 10^{-14} \mu/s$, i.e., the message passing cost is constant for all time steps.

Table 1 presents the elapsed time with processing of cells in the sequential approach. Next, considering that each node processes a subset of the domain and the information about latency, bandwidth as well as elapsed time with the processing cells, the effect of granularity (cells per node) is estimated. Upon inspection of table 1, the parallel approach apparently becomes a good strategy above 1000 cells per node, i.e., the elapsed time with message passing would represent less than 1% of the time.

5 Test and results

The tests translate qualitative results, since this work is a first approach of the model. This uses a set of parameters in order to define what one intends to

Table 1. *Elapsed time for processing cells and changing messages*

cells	elapsed time (μs)	cells processed / time	messages / time
1000	26802.938	0.996716778	0.003283222
5000	132526.938	0.999335984	0.000664016
10000	264681.938	0.999667525	0.000332475

simulate: simulation time, number of cells, number of vehicles and probability of maintaining the velocity are required to define any simulation. An important feature carried out by the model is the fact that the simulations are independent of the initial state, i.e., hundreds of time steps are performed before the "memory" of the any initial state stops influencing the dynamics of the process. One refers to this as the convergence of the simulation to satisfactory result.

All tests in this work were performed on an intel Core 2 Quad 2.4 GHz CPU with 4 GB of RAM with CENTos 64bits operation system. Each instance of test was executed with the configuration: periodic boundary condition, 300 cells, 10.000 simulation time steps and the first 1000 time transient time steps. The simulations were carried out using highway densities ranging from 0.01 to 0.99. The maximum speed value allowed is 5 cells per unit time.

In Figs. 12 different values for p_m (the uncertainly of a vehicle keeps its velocity in the next time instant) are considered; normal and uniform continuous probability functions with the same values of p_m parameter are used. The left column figures presents results with normal functions with parameters $\mu = 0.5$ and $\sigma = 0.1$, while the right column shows results with uniform functions. Both are built in accordance to the Monte Carlo method.

Despite uniform probability function shows a good flow-density relation, the values of p_m parameter do not give arise to the metastable phase as clearly as can be seen in the simulations for the normal probability function. Among the exhibited results for the p_m values, $p_m = 0.30$ with normal probability function closely resembles real data (see Fig. 15).

In Fig. 17 and 18 velocity-density and velocity-flow diagrams, respectively, are illustrated. For the former, a density increase leads to a speed decrease. The latter shows the relation between velocity and flow. For this, an increase in flow occurs almost with constant average velocity until the flow jams; next, the decrease in flow occurs for decreasing average speed.

5.1 Complexity and convergence of the simulation

The method is composed by two parts: velocity definition and position update. Thus, the complexity should be analyzed under the sequential paradigm.

In the algorithm the velocity definition and position update are applied to all vehicles. Both of them have complexity $O(n)$, where n is the number of vehicles, i.e., for each time simulation the algorithm is invoked for each vehicle for defining velocity followed by other n iterations in order to update their position.

Fig. 12. Fundamental diagrams - periodic condition

Fig. 13. $p = 0.10$ - Normal

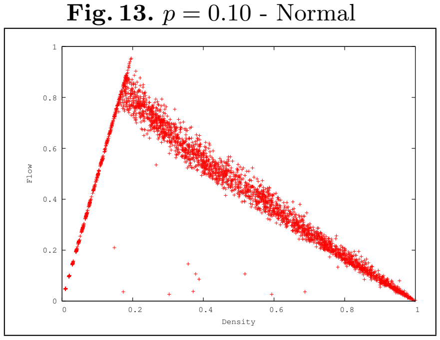


Fig. 14. $p = 0.10$ - Uniform

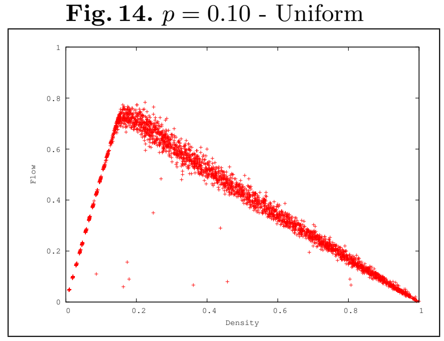


Fig. 15. $p = 0.30$ - Normal

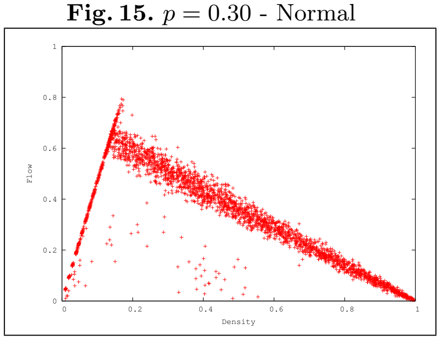
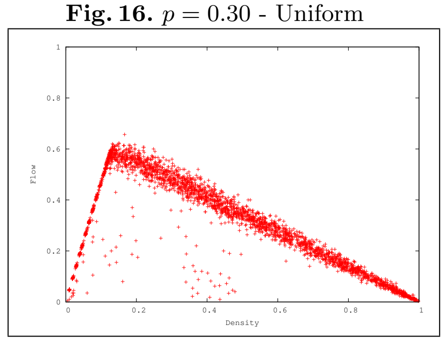


Fig. 16. $p = 0.30$ - Uniform



Besides, the counter flow velocity tunning should be analyzed separately from the ordinary velocity definition procedure (lines 1-16 of algorithm 2). When this special procedure is enabled, its complexity is $O(n^2)$ in the worst case, since the worst case represents invoking it for each vehicle and the first of all vehicles.

During the simulation, there is no significant impact in enabling the counter flow velocity tunning procedure. Fig. 19 illustrates the average number of re-adjustments per iteration within different values for p_m, using normal probability function. The worst case happens with $p_m = 0.10$, i.e., only 10% of drivers will keep their velocity in the next time instant. In this case, the average number of re-adjustments represents less than 1% of iteration. Therefore, the impact of re-adjustments does not represent a significant amount of the whole computation.

6 Conclusion

Since each driver's behavior does not fit an uniform distribution, the continuous probability function based on normal distribution provides an appropriate modeling for driver's behavior. Furthermore, the anticipatory feature of the drivers helps provide a model which appears to more closely represent the real world.

Fig. 17. Velocity-density diagram / $p = 0.30$ / $N(\mu = 0.5, \sigma = 0.1)$

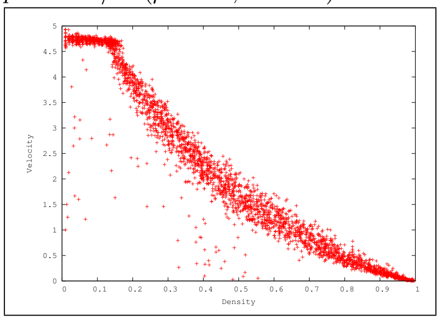


Fig. 18. Velocity-density diagram / $p = 0.30$ / $N(\mu = 0.5, \sigma = 0.1)$

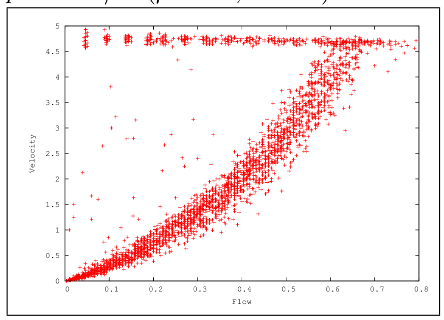
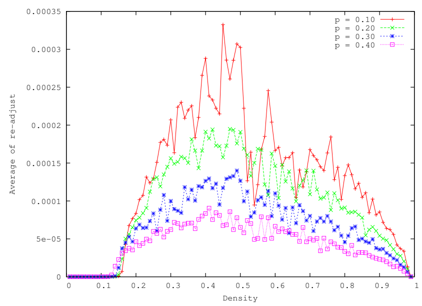


Fig. 19. Average of re-adjust



The model proposed in this work requires few variables to set up one simulation. Despite being an explicit method, it is stable. After the transient phase, the algorithm always reaches the proper state, which means the simulations converge. The parallelizability is discussed and a quantitative estimate seems promising for future development of a simulator for real traffic networks with tens of thousands of vehicles.

Results illustrate many realistic features and qualitative, as well as quantitative match with real data. Some improvements are under development, including multilane roads, traffic signs, and other aspects.

References

1. Wahle, J. and Neubert, L. and Esser, J. and Schreckenberg, M. A cellular automaton traffic flow model for online simulation of traffic Parallel Computing. **27** (2001) 719-735
2. Hafstein, S. F. and Chrobok, R. and Pottmeier, A. and Schreckenberg, M. A High-Resolution Cellular Automata Traffic Simulation Model with Application in a Free-

way Traffic Information System Computer-Aided Civil and Infrastructure Engineering. **19** (2004) 338-350
3. Nagatani, T.: Traffic states and fundamental diagram in cellular automaton model of vehicular traffic controlled by signals. Physica A: Statistical Mechanics and its Applications. **388** (2008) 1673–1681
4. Tian, R.: The mathematical solution of a cellular automaton model which simulates traffic flow with a slow-to-start effect. Discrete Applied Mathematics. **13** (2009) 2904–2917
5. Spyropoulou, I.: Modelling a signal controlled traffic stream using cellular automata. Transportation Research Part C: Emerging Technologies. **15** (2007) 175–190
6. Jiang, R. and Wu, Q.: Open boundaries in a cellular automata model for synchronized flow: effects of nonmonotonicity. Phys Rev E Stat Nonlin Soft Matter Phys. **68** (2003) 1539-3755
7. Wolfram, S.: Theory and applications of cellular automata Wolfram, S. (1986)
8. Fukui, M., Ishibashi , Y.: Traffic Flow in 1D Cellular Automaton Model Including Cars Moving with High Speed. Journal of the Physical Society of Japan. **65** (1996) 1868–1870
9. Nagel, K., Schreckenberg, M: A cellular automaton model for freeway traffic. Journal de Physique I. **2** 2221–2229
10. Fukui, M.,Ishibashi, Y.: Metastable states in cellular automata for traffic flow. Journal Euro Physics J. B. **5** (1996) 793–800
11. Takayasu, M. , Takayasu, H.: 1/f Noise in a Traffic Model. Factral 1. **5** (1993) 860–866
12. Benjamin, S., Johnson, N., Hui, P.: Cellular Automata Models of Traffic flow Along a Highway Containing a Junction. Journal of Physics A. **5** (1996) 3119–3127
13. Emmerich, H., Rank, E.: An Improved Cellular Automaton Model for Traffic Flow Simulation Journal of Physics A. **234** (1997) 676–686
14. Larraga, E. M., Rio, A. J., and Schadschneider, A.: New kind of phase separation in a CA traffic model with anticipation Journal of Physics A: Mathematical and General. **37** (2004) 3769–3781
15. Justin, H. and Wu-chun, F.: Research Note: Analyzing MPI performance over 10-Gigabit ethernet J. Parallel Distrib. Comput.. **65** (2005) 1253–126