

# Decentralized Method for Traffic Monitoring

Guillaume Sartoretti<sup>1,2</sup>, Jean-Luc Falcone<sup>1</sup>,  
Bastien Chopard<sup>1</sup>, and Martin J. Gander<sup>2</sup>

<sup>1</sup> Computer Science Department

<sup>2</sup> Department of Mathematics,  
University of Geneva, Switzerland

**Abstract.** We propose a decentralized method for traffic monitoring, fully distributed over the vehicles. An algorithm is provided, specifying which information should be tracked to reconstruct an instantaneous map of traffic flow. We test the accuracy of our method in a simple cellular automata traffic simulation model, for which the traffic condition can be controlled and analyzed theoretically.

## 1 Introduction

Nowadays in big cities, the need for an overview of the traffic situation is critical in order to avoid queues, accidents, and establish an efficient routing strategy. Current solutions often involves a centralized node which collects information from the drivers within the traffic, and broadcast it to let everyone make reasonable decisions. While this strategy works well, it raises the question of privacy: the central node knows everything about everyone in the network, and this centralization of the information may hit back if the knowledge is in the wrong hands.

In this paper, we define a fully decentralized communication method which allows drivers to gather information and share it anonymously with the neighboring drivers throughout their journey, letting everyone have a good view of the status of the traffic. Due to this type of information distribution, the disadvantages of centralization and the privacy violation can be avoided. However, the problem is not trivial because the system must converge quickly to an accurate and stable representation of the current traffic conditions, and it must be able to detect rapid changes due to accidents or rush hours.

To test the capability of our method to build an accurate traffic map, we use a simple traffic model in which the cars accumulate information as they move, through mutual exchanges. Our goal is to produce well defined traffic conditions that can then be compared with the estimate made by each car. The paper is organized as follows. We first propose a theoretical description of the car motion, from which the traffic flow can be predicted with the knowledge of only a few quantities. Then we define how the cars communicate and how they combine the information to progressively build a global traffic map. Finally we provide some examples of simulation, showing how accurately the traffic can be estimated. More detail about this study can be found in [1].

## 2 Traffic Model and Theoretical Framework

In order to produce simple traffic conditions suitable to test our communication method, we consider a Cellular Automata (CA) model. This approach has been widely used in traffic simulations, with excellent results (see for instance [2–6]).

### 2.1 Basic Model

Here we consider several road segments, interconnected through simple junctions. A road segment is a 1D vector of length  $L$ , where each component (cell) represents a possible location for a car. Let  $n_i(t)$  be the value of the cell  $i$  at time  $t$ . We define  $n_i(t) = 1$  if there is a car at location  $i$ , at time  $t$ . Otherwise, we define  $n_i(t) = 0$

Here we use Wolfram’s rule 184 [7] to describe the motion of the cars on a road:

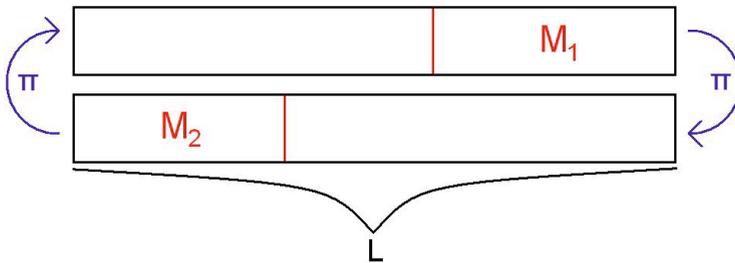
$$n_i(t + 1) = n_{i-1}(t)(1 - n_i(t)) + n_i(t)n_{i+1}(t) \quad \text{for } 1 < i < L \quad (1)$$

This means that a car moves to the next cell if and only if this next cell is free; otherwise it stays still. Roads are considered as one-way lanes. The two special locations are the beginning and the end of a road. These cells obey other rules that implement the chosen boundary conditions, for instance the junction that interconnects two segments.

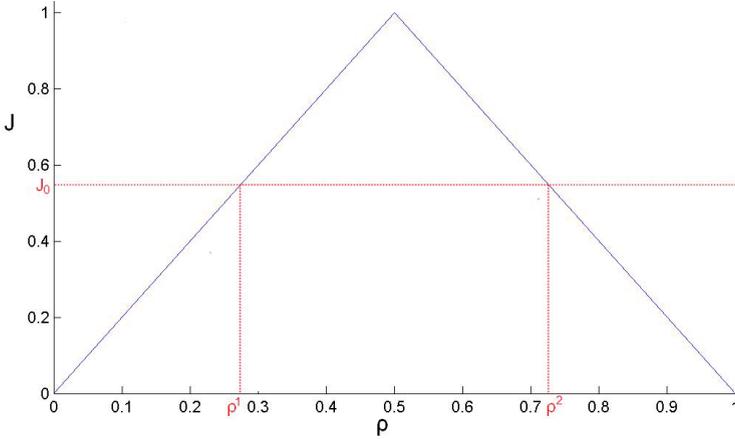
**Circuits.** A simple example of interconnection between two roads is the circuit shown in Fig. 1. It is composed of two side by side roads of length  $L_1 = L_2 = L$ . A car reaching the end of one of the roads may cross to the other road with probability  $\pi$  (as if there was a stop sign, or a traffic light), and given that the beginning of the other road is free.

Accordingly, the rule for cells 1 and  $L$  can be expressed as

$$\begin{aligned} n_1(t + 1) &= m_L(t)(1 - n_1(t)) \cdot Q(t) + n_1(t)n_2(t) \\ n_L(t + 1) &= n_{L-1}(t)(1 - n_L(t)) + n_L(t)(1 - m_1(t)) \cdot P(t) \end{aligned}$$



**Fig. 1.** Diagram of a two-road circuit, with an initial number  $M_1$  and  $M_2$  of car on each road



**Fig. 2.** Fundamental diagram of traffic, showing the flow  $J$  according to the density  $\rho$

where  $n_i$  and  $m_i$  denote the occupation numbers of the first and second roads, respectively. The quantities  $P(t)$  and  $Q(t)$  are Boolean random variables which are 1 with probability  $\pi$ . The equations for  $m_1$  and  $m_L$  are obtained by symmetry.

**Occupation Density of a Cell.** On a road segment, the car density  $\rho_i(t)$  on cell  $i$  is defined as  $\langle n_i(t) \rangle$ , the average of  $n_i$  (over time or space). We can write an evolution equation for  $\rho_i(t)$  by taking the average of eq. (1):

$$\rho_i(t + 1) = \rho_{i-1}(t) - \langle n_{i-1}(t) n_i(t) \rangle + \langle n_i(t) n_{i+1}(t) \rangle. \tag{2}$$

Note that  $\langle n_i(t) n_{i+1}(t) \rangle \neq \rho_i(t) \rho_{i+1}(t)$  in general, because  $n_i(t)$  and  $n_{i+1}(t)$  are highly correlated.

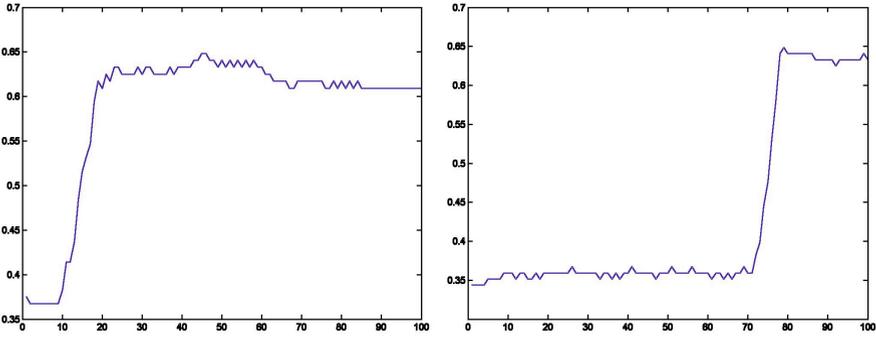
**Fundamental Diagram of Traffic.** The traffic flow  $J(t)$  in a road segment is defined as  $J(t) = \rho(t) v(t)$ , with  $v$  the average speed of the cars on that section. In a steady state, the flow obeys the *fundamental diagram of traffic* shown in Fig. 2 (see for instance [2] for a detailed calculation).

This diagram shows that for a given flow  $J_0$ , two different pairs of values  $(\rho^1, v^1)$  and  $(\rho^2, v^2)$  are possible:  $J_0 = \rho^1 v^1 = \rho^2 v^2$ , with the constraints  $\rho^1 < \frac{1}{2} < \rho^2$  and  $\rho^1 + \rho^2 = 1$ .

This means that a road with flow  $J_0$  may be separated into two zones, a low-density zone (a free moving zone where the density will be  $\rho^1$ ), and a high-density zone (a queue of cars, where the density will be  $\rho^2$ ).

## 2.2 Theoretical Description

**Queue Length.** Fig. 3 illustrates the typical density  $\rho_i$  at each cell  $\{1, \dots, L\}$  of one section of the two-road circuit shown in Fig. 1. This figure shows that



**Fig. 3.** Density plots for a road of length  $L = 10$ , averaged over 128 successive time iterations at different time of evolution. As we can see, the length of the queue can vary over time.

the queue length varies over time. In this section, we propose an analytical description of this behavior.

In a steady state, we can rewrite eq. (1) as

$$\rho_i = \rho_{i-1} - \langle n_{i-1}n_i \rangle + \langle n_i n_{i+1} \rangle, \tag{3}$$

In a steady state, the traffic flow  $J$  is constant. Therefore the density has two possible values  $p$  and  $q = 1 - p$ , as discussed above. Let us assume boundary conditions such that  $\rho_1 = p$  and  $\rho_L = q = 1 - p$ , with  $p < \frac{1}{2}$ . Then, eq. (2) is obeyed if

$$\rho_i = \begin{cases} p & \text{if } i \leq k \\ q & \text{if } i > k \end{cases} \quad \forall k \in \{1, 2, \dots, L\}. \tag{4}$$

This condition means that there is a change in the car density at a cell  $k$  along the road. This is interpreted as the presence of a queue that starts at position  $k$ .

We can check that eq. (4) obeys eq. (2) by computing the value of  $\langle n_{i-1}n_i \rangle$  and  $\langle n_i n_{i+1} \rangle$ . Let us consider the region where  $i \leq k$ , the low-density region. Since  $\rho < \frac{1}{2}$ , the cases where  $(n_i, n_{i+1}) = (1, 1)$  is impossible in the long run, as there is always enough space to let a free cell between two following cars. In such a case, we can only have  $n_i n_{i+1} = 0$ , so that

$$\langle n_i n_{i+1} \rangle = 0 \quad \text{if } \rho_i = p < \frac{1}{2} \tag{5}$$

In the high-density region,  $i > k$ , where  $\rho_i = q = 1 - p > \frac{1}{2}$ , we cannot have  $(n_i, n_{i+1}) = (0, 0)$  in the long run. The number of cars is too high to allow such configuration, and only the  $(1, 0)$ ,  $(0, 1)$  and  $(1, 1)$  configurations can be observed. Let us denote  $C = L - k$  the number of cells in the region of density  $q$ . We define  $c_1 = qC$  the number of cars in this region and  $c_0 = C - c_1$  the number of free cells. Since next to any free cell, there will be a car, there are  $c_0$

times the configuration (1, 0) and  $c_0$  times the (0, 1) The total number of pairs in this region being  $C$ , we can write the probability of having the (0, 1) or (1, 0) configurations as  $2c_0/C = 2(C - c_1)/C = 2 - 2c_1/C = 2 - 2q$ .

Since there is no (0, 0) configuration, the probability of the configuration (1, 1) is  $1 - (2 - 2q) = 2q - 1$ . Thus

$$\langle n_i n_{i+1} \rangle = 2q - 1 \quad \text{if } q > \frac{1}{2}. \tag{6}$$

Finally, for  $i = k$  we can assume that

$$\langle n_k n_{k+1} \rangle = 0 \tag{7}$$

This means that at the interface between the two regions, the configuration (1, 1) is impossible: if this configuration were possible, we should move the break point from  $k$  to  $k - 1$ , as  $k$  would already be in the high-density region. In other word, the cell  $k$  at which  $\langle n_k n_{k+1} \rangle$  switches to a non-zero value is defined as the start of the queue. Using eqs. 4, (5), (6), (7), it is easy to verify that eq. (3) is satisfied.

A consequence of this analysis is that the queue can a priori have any length because  $k$  can be any integer between 1 and  $L$ . However, there is a constraint related to the total number of cars. Let  $N$  be the number of car in the upper road of the circuit shown in Fig. 1. Then, we must have  $N = kp + (L - k)q$ . Thus,  $k = (Lq - N)/(q - p)$  and the fluctuation of  $k$  are related to the fluctuations of  $N$ . In the two-road circuit,  $N$  is constrained by the maximum number of car  $M_1 + M_2$  and the length  $L$  of the two road segments:  $\max(0, M_1 + M_2 - L) \leq N \leq \min(L, M_1 + M_2)$ . As the queue length is a quantity which can potentially vary a lot in our circuit, it is not well suited for testing the capability of our mapping method to estimate the traffic condition. It is better to consider a quantity which is easier to control and which is more stable. A better test is to check whether the cars can accurately estimate the value of  $\pi$ , the probability that a car can cross the last cell of the road.

**Traffic Flow in the Two-Road Circuit.** In our circuit, there is a relation between the densities  $p$  and  $q = 1 - p$ , and the value of  $\pi$ , the probability of not being stopped. In a steady state, the flow  $J$  must be constant all across the system. At the end of each road,  $J$  is given by the probability  $q$  to have a car times the probability that this car is allowed to continue. This assumes that the first cell of the next segment is always free, which is the case as no two cars can successively enter the low density region from the high density region (there must be a free cell between moving cars). Moreover, from the fundamental diagram, the flow in the low density region is equal to  $p$ . Thus

$$J = q\pi = p = 1 - q, \tag{8}$$

from which we conclude that  $\pi = p/q$  and  $p = \frac{\pi}{1+\pi} = 1 - q$ .

### 3 Decentralized Flow Estimation

We showed in the previous section that for a given flow at steady state, a road is composed of two regions, one with low density  $p$  and the other with high density  $q = 1 - p$ . We have also shown that the flow could be reconstructed when  $p$  and  $q$  are known. Thus, we could estimate the flow of each road segment, if we can estimate the values of  $p$  and  $q$ . In this section, we present a way to estimate those quantities by a decentralized exchange of information between neighboring cars.

#### 3.1 Global Map

We call *global map*, the real traffic map that each car will try to estimate. According to our discussion, we can restrict this map to a few cells in the network, those which are enough to predict the traffic flow in each road segment. Here, the map is composed of the measured densities at each road end,  $\hat{\rho}_2$  and  $\hat{\rho}_{L-1}$ . Those densities are computed as the average occupation of the cell during the last 128 iterations:

$$\hat{\rho}_i(t) = \frac{1}{128} \sum_{k=t-127}^t n_i(k) \quad (9)$$

The use of a “sliding time window” (of size 128 in our case) to compute the density of a cell, allows us to forget the remote past and to get a dynamic view at traffic evolutions. The size of 128 was chosen for practical implementation reasons, and provides good results as shown in the next section. If the window size is too small, fluctuations will dominate time-stable structures. If it is too large, temporary (but relevant) structures will become undetectable.

#### 3.2 Local Map

Every car in the model maintains an estimated map of traffic density, termed *local map*. This map is updated at each iteration by direct observations and communications with other cars. Cars can exchange information with neighboring cars. Here, during a single iteration, each car can communicate with at most with 5 other cars, the leading and trailing ones, and the three cars in the opposite lane.

For a street map with  $N_{\text{road}}$  road segments, the local map  $m_v$  of a car  $v \in \{1, 2, \dots, M_1 + M_2\}$  at iteration  $t$  is a matrix with 128 rows and  $2N_{\text{road}}$  columns. The rows correspond to the 128 previous time steps and the columns contain the tracked locations (both ends of each road segment). The elements of the map defined as

$$m_v(i, j) = \begin{cases} 1 & \text{if a car was recorded by } v \text{ at location } j \text{ at time } t - i + 1 \\ 0 & \text{else.} \end{cases} \quad (10)$$

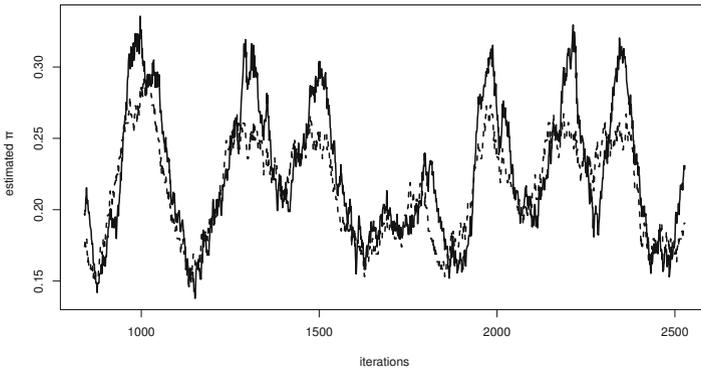
An estimate of the density of each location can be computed by averaging the corresponding column.

**Local Map Update.** All cars start with a local map filled with zeroes. At each iteration, they update the map according to the following steps: **(1)** The local map matrix rows are shifted down, eliminating the oldest row. The first row is filled with zeroes. **(2)** Each car adds 1 to the element of the first row corresponding to its current location, if it is a tracked location ( $i = 2$  or  $i = L-1$ ). **(3)** Each car sends its map to each neighboring car and receives maps from them, merging the information into its own local map with an element-wise logical **OR**.

It is easy to see that the logical **OR** of the first 128 lines of the local maps of all the cars in our circuit will produce the global map. Indeed, as all cars only put their own position at each step of the simulation in their local maps, the logical **OR** of all the local maps will give the occupation of each of the cars in the last 128 iterations. The question is whether our car to car communications will be fast enough to provide an accurate **OR** of all the local maps, and thus the global map.

## 4 Simulation Results

We present here some preliminary results demonstrating the capability of our decentralized communication method to build a reliable traffic map. Using the circuit defined in Sec. 2.1, we ran several simulations using road segments with a length of  $L = 60$  cells initialized with  $M_1 = M_2 = 30$  cars per road segment placed in a queue at the end. One car is chosen at random as the reference car. After  $12L$  iterations of traffic update, allowing us to reach a steady state, the cars start updating their local maps and exchanging information as explained above.

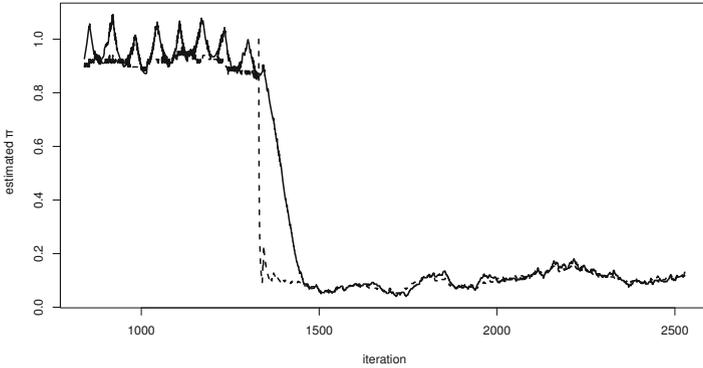


**Fig. 4.** Estimation of  $\hat{\pi}$  in a circuit with  $\pi = 0.2$ . The solid line corresponds to the local map and the dashed line to the global map. Increasing the memory length from 128 to 1024 would smooth out the variations and give a more accurate prediction of  $\pi$ .

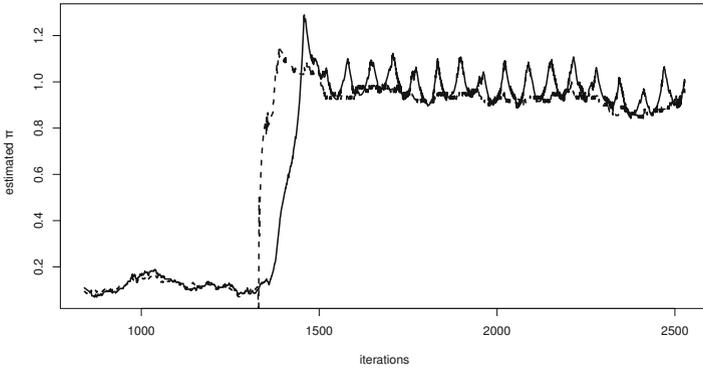
In this simulation we study how well the test car can estimate the probability  $\pi = p/q$  (see eq. (8)) that a car is not stopped at the end of a segment. The estimate value  $\hat{\pi}$  is computed as

$$\hat{\pi}(t) = \frac{\hat{\rho}_2(t)}{\hat{\rho}_{L-1}(t)} \tag{11}$$

This value is then compared with the value obtained from the global map using the same procedure, as well as the exact value of  $\pi$  used in the simulation. We consider here two situations: (i) a circuit where  $\pi$  is constant in time; (ii) a circuit where the value of  $\pi$  varies suddenly.



(a) Case with  $\pi = 0.9$  when  $t \leq 1320$  and  $\pi = 0.1$  for  $t > 1320$



(b) Case with  $\pi = 0.1$  when  $t \leq 1320$  and  $\pi = 0.9$  for  $t > 1320$

**Fig. 5.** Value of  $\hat{\pi}$  computed from the local map (continuous line) and from the global map (dashed line)

#### 4.1 Constant Flow

Here we analyze a circuit with  $\pi = 0.2$ , once then a steady flow has settled. In Fig. 4 we compare the value of  $\pi$  computed from the local map of the test car and that computed from the global map. As we can see, the measured values stay around the actual value of  $\pi$  for both cases. Variations, reflecting the intrinsic fluctuations of traffic and the limited length of the memory, are visible in both maps, though more pronounced on the local one, due to its incomplete information.

#### 4.2 Flow with Abrupt Transition

In the previous experiment, we have seen that the local map converges rather well to the global map in a steady state. It is also important to see if the system is able to detect in a reasonable time a sudden change in the traffic flow. These changes can arise in real situations, for instance when a car accident occurs and until the wrecked cars are removed.

To test this situation, we have run a simulation where the value of  $\pi$  starts at 0.9 and then decreases to 0.1 after 1320 iterations. We can see in Fig. 5(a) that the global map responds immediately to the change whereas the local map re-adjusts more slowly. It matches the global estimation in about 200 iterations.

We also performed the reverse experiment, i.e. increasing the value of  $\pi$  during simulation. Here  $\pi$  starts at 0.1 and increases abruptly to 0.9 after 1320 iterations. Again, the local map matches the global with a delay of about 200 iterations. Results are shown in Fig. 5(b).

### 5 Discussion

We have proposed a decentralized method allowing the cars traversing a road network to build an estimate of the traffic situation, by exchanging local and anonymous information.

We also showed that, in a simple CA model, the traffic conditions in a simple circuit can be described with only the knowledge of the car densities at the extremities of the road segments. Within this framework, we have tested the capability of the car to predict the global traffic conditions. The method is robust enough to detect flow variations caused by traffic accidents.

The method is very economic in terms of used memory, since only  $2 \times 128$  bits per road segment are stored. A local map for the whole Borough of Manhattan in New-York city could then be stored in less than 150 kB (11 avenues, 220 streets, and two lanes per road segment give us  $11 \times 220 \times 2 \times 2 \times 128$  bits). The computational cost is also very low: (i) most of the operations are element-wise and thus may run in parallel; (ii) bitwise **OR** operations are executed as a single instruction in all current processors.

In the current model of communication, we restrict communication with nearest neighbor cars and thus the information is propagated at the same speed than

the cars. However, in a concrete implementation of the communication system, information will be exchanged via a wireless protocol, which allows the communications to be both faster and with an extended range. We expect such a system to be even better because, as stated above, the union of all local maps in a system gives the exact global map. Thus, if the communication is more efficient, the local maps of every car will be more accurate. Radio-wave transmission could nevertheless be simulated with a multi-scale CA where the communication rule is applied with smaller time-steps than the traffic rule.

We have also tested the method using a more realistic street map and promising results are described in [1].

## References

1. Sartoretti, G.: Communication method for decentralized estimation of traffic density. Technical report, University of Geneva. Master Thesis (2012)
2. Chopard, B., Luthi, P.O., Quelo, P.A.: Cellular automata model of car traffic in a two-dimensional street network. *Journal of Physics A: Mathematical and General* 29(10), 2325 (1996)
3. Dijkstra, J., Timmermans, H.J.P., Jessurun, A.J.: A multi-agent cellular automata system for visualising simulated pedestrian activity. In: Bandini, S., Worsch, T. (eds.) *Theoretical and Practical Issues on Cellular Automata - Proceedings on the 4th International Conference on Cellular Automata for Research and Industry*, pp. 29–36. Springer (2000)
4. Dupuis, A., Chopard, B.: Cellular automata simulations of traffic: A model for the city of geneva. *Networks and Spatial Economics* 3, 9–21 (2003), doi:10.1023/A:1022044932736
5. Nagel, K., Schreckenberg, M.: A cellular automaton model for freeway traffic. *J. Phys. I France* 2(12), 221–2229 (1992), <http://dx.doi.org/10.1051/jp1:1992277>
6. Nishinari, K., Fukui, M., Schadschneider, A.: A stochastic cellular automaton model for traffic flow with multiple metastable states. *Journal of Physics A: Mathematical and General* 37(9), 3101 (2004)
7. Wolfram, S.: *Theory and Application of Cellular Automata*. World Scientific, Singapore (1986)