

A realistic simulation for Highway Traffic by the use of Cellular Automata

Enrico G. Campari¹ and Giuseppe Levi²

¹ Physics Department of Bologna University
and Istituto Nazionale per la Fisica della Materia
viale Berti-Pichat 6/2 I-40127, Bologna, Italy
campari@df.unibo.it

² Physics Department of Bologna University
and Istituto Nazionale di Fisica Nucleare
viale Berti-Pichat 6/2 I-40127, Bologna, Italy
levi@bo.infn.it

Abstract. A two lane, non-periodic highway with on-ramps and off-ramps is simulated using a simple cellular automata model. The main experimental features of highway traffic are reproduced, indicating the usefulness of this approach for the study of traffic and it is shown the need of ramps to produce congested traffic. A fractal dimension for the variables used for traffic analysis, that is car density, car speed and flow has been measured with a Box Counting algorithm. Finally, it is suggested how to perform real-time traffic forecasts.

1 Introduction

Every day an enormous number of car drivers, commuting to work or going to the countryside for the weekend, get stucked into traffic. Many of the readers of this article, as car drivers, might have noticed how sometimes, driving on an highway, the traffic suddenly slows to a crawl. For a while cars move very slowly and then are again free to accelerate. There are no accidents or roadworks to cause the jam, which apparently has no explanation.

This and other aspects of highway traffic, like the creation of car density waves, together with the great economical costs and health problems linked to traffic, spurred the scientists attention into the study of the subject. In the last decades, experimental data were collected and analysed, mainly in USA, Germany and Japan [1–5]. At the same time, models of traffic flow were built. The firsts were based on collective properties of traffic [6]. Then there were microscopic models, like the so called optimal velocity model by Bando and coworkers [7]. Both were based on the computer resolution of non-linear differential equations. Despite some succes, a real progress in the study of traffic came only with the introduction of computer simulations based on cellular automata. This method is simpler to implement on computers, provides a simple physical picture of the system and can be easily modified to deal with different aspects of traffic. For these reasons, cellular automata models are getting more and more popular.

In synthesis, a cellular automaton consists of a regular lattice with a discrete variable at each site. A set of rules specify the space and time evolution of the system, which is discrete in both variables [8,9]. These rules are usually limited to first neighbours interactions, but this is not necessary however, and the particular application of cellular automata to traffic is one case where longer range interactions are used.

2 Cellular Automata Model

The model proposed in this article simulate a two lane highway with on and off-ramps as a long straight road divided into cells which can be occupied by *cars* moving in the direction of increasing cell number. Cars enter the highway at the road beginning and at on-ramps at a given rate by means of pseudo random number generation. Once in the road, they move according to a set of rules. These evolution rules were derived from those used by Schreckenberg, Nagel and coworkers [10–12]. These authors simulated a multilane highway with the use of few simple rules, which proved how cellular automata models are effective in simulating traffic. Although really interesting, the above cited model and other similar are unrealistic. Their main pitfalls are the use of periodic boundary conditions and the lack of ramps. Leaving aside that every highway has a beginning and an end, the use of periodic boundary conditions forbids simulating a variable car density along the highway, which is the common situation. As widely known, traffic in most highways presents two peaks, one in the morning and a second one in the afternoon [13], in addition to an absolute minimum during the night. These basic experimental facts simply cannot be reproduced in the framework of periodic models without ramps.

Ramps absence also hinders the well known fact that traffic jams usually nucleate in correspondence of ramps or other obstacles to car flow such as road-works or junctions. Even if it is proved that above a certain car density along the road, density fluctuations naturally occur and grow into jams, as far as ramps are always present it is there that jams first nucleate [14]. Furthermore, this happens at a lower density with respect to a rampless highway.

In order to consider a more realistic road and to be able to simulate transient phenomena, we developed a cellular automata model for a two lane highway leaving periodic boundaries conditions and introducing on-ramps [15], which were modeled as a single entrance cell in lane 1. Their number could vary from simulation to simulation and were typically located 2500 cells apart. In this article we present an improved version of that model where both on and off ramps are present, data collection at a fixed point from a car detector is simulated and data analysis is performed in order to look for the fractal dimension of traffic.

In the model, the highway length can be varied from simulation to simulation and in order to compare the results of the simulations with experimental data, we considered each cell to correspond to a length of 5 m and each evolution step for the cells to occur in 1 s. The model is asymmetric: one lane, named in the following lane 2 or *left* lane, is used for overtaking while the other, named

lane 1 or *right* lane, is for normal cruise. For all on-ramps, at each time step (ts) the program checks if the cell is empty. If empty, a random number in the $[0,1]$ interval is generated and if it is less than the threshold chosen for that simulation, a car is generated at that cell with a speed of 2 cell/ts. At the road beginning, cars are generated both on lane 1 and lane 2 with the procedure described above, with starting speed 2 cell/ts. Every car entering the highway has a given destination or off-ramp. Approaching its exit, the car slow down and eventually move into the right lane. When the off-ramp is reached the car is removed from the road.

For the sake of simplicity, cars can have only two values of their maximum speed: 6 or 9 cell/ts, corresponding to a maximum speed of 108 or 162 km/h. The first kind of vehicles corresponds to trucks. In this kind of simulations, cars have an unbound braking ability in order to avoid car accidents.

The evolution rules away from ramps are schematically described as follows in *pidgin algol* language:

```

MAIN:
Begin:
Init variables;
Load Highway-configuration;
Load Run-Parameters; /* RP */
Open Output-file;
  If requested by RP then
    Extract random initial state with initial car density given by RP ;
    Do Velocity-Normalization;
  Endif
/* MAIN LOOP: */
Repeat for Ntimes; /* Number of times specified in RP */
Forall on ramps:
  With Propability P Do: /* Entrance Probability is given in RP for each entrance */
    If entrance cell is empty then
      generate new vehicle with velocity := 2;
    endif
  End With Propability
End Forall
Forall cars in Lane 2:
  If are verified the conditions to reenter in lane 1 then
    Change Lane;
  Endif
End Forall
Forall cars in Lane 1:
  If (car distance from exit) < 2 then
    remove car;
  endif
  If are verified the conditions to overtake then Change Lane;

```

```

    If car.velocity < car.maxv then
      car.velocity := car.velocity + 1;
    Endif
  Endif
End Forall
Do Velocity-normalization;
Forall cars in Lane 1:
  With Propability P Do:
/* Random deceleration with probability P given in RP */
car.velocity := car.velocity - 1;
  End With Propability
End Forall
  Forall cars in Lanes 1 and 2:
    Car.position := Car.position + Car.velocity;
  End Forall
Measure-and-Store physical quantities;
/* eg.: mean car density and velocity, flux etc. */
End Repeat /* End of main Loop */
Close ouput files;
End /* end of main program */

```

Velocity-normalization :

DEFINE Forward.Gap as the MINIMUM between the number of free cells from a car to the next one in the same lane along the car direction of motion and the distance of the car from the exit;

Begin:

```

  Forall cars in Lanes 1 and 2:
    Measure Forward.Gap;
    If car.velocity < car.maxv then
      car.velocity := car.velocity + 1;
    Endif
    If car.velocity > Forward.Gap then
/* This model (as the Schreckenberg one) avoids accidents */
      car.velocity := Forward.Gap;
    Endif
  End Forall
Return ;
End

```

The input parameters to the program at the beginning of each simulation are: highway length, ramp number and position, starting car density for each lane, car generation probability, random deceleration probability, fraction of slow cars, maximum speed of vehicles (*maxv*) and simulation length in time ticks.

At each iteration the state of the highway is computed and at selected times recorded in a file. This means computing for each cell mean car speed and density,

averaging over the 100 cells before and after the given cell. From these quantities the flow is readily computed as density by mean speed. Furthermore, for each car, position, speed and gaps are recorded. The use of averages in the calculation of car speed and density is due to the use of a discrete model. Where in a real road cars can have a continuous range of speeds and distances, here there are only discrete quantities. A meaningful comparison with experimental data becomes possible only after the averaging process, which increases the number of possible values for speed and density. It is to say that in most cases experimental data from highways are averages over time and/or space [14, 16], so that the use of averages is a legitimate procedure.

In addition to the state of the road at chosen times, the traffic evolution as a function of time at selected positions is recorded. This should be particularly useful when the program will be used to simulate real highways and used to make traffic forecasts. We recently patented a system consisting of car detectors to be installed along an highway, whose output will be used to calibrate a cellular automata program implementing that particular highway. The simulations will provide a detailed knowledge of the road characteristics. As a result, it will be possible to forecast if a car jam is going to happen somewhere along the highway on the basis of the current state of traffic and try to take some action in order to prevent it. This could be done reducing car flow inside the highway or having some *official* car driving at reduced speed. This second kind of action could help preventing the complete stop of cars which is the main cause of jam formation [17].

The computer program used for these simulations was written in Fortran90. In the program, software packages written at CERN (CERLIB) are used, mainly for data analysis and graphic outputs (*paw* software) [18]. Eighth hours traffic along a 100 Km long highway can be simulated in 3 to 5 minutes (depending on traffic) with a personal computer equipped with a 1 GHz processor, a 266 MHz bus, 256 Mbyte RAM memory and Linux operating system.

3 Representative results from computer simulations

Three main kind of traffic are found in highways. These will be called free flow, synchronized flow and congested flow. A wide variety of subconditions is found in synchronized and congested flow, which are sometimes named under different terms, like wide moving jams.

Free flow is a condition where fast cars can easily overtake slow ones, there are large gaps between cars and traffic flows easily. This phase occurs at low car density (below 0.1 car/cell in the computer simulations of this work, but the value is not sharply defined) and is characterised by the property that an increasing car density along the highway produce an increasing flow. Experimental data show that in terms of flow this condition lasts for car flow increasing from zero up to 2000-2500 cars per lane per hour [14].

At the higher densities (above 0.5 car/cell in the computer simulations, but again the value is not well defined) it is found the so-called congested flow. It

is a condition of heavy traffic, with flow decreasing with increasing car density and easy formation of jams. Overtaking is difficult and car speed can be nearly zero.

Intermediate between the former and the latter is the so-called synchronized flow [14]. This condition is defined as a state of traffic in multilane roads in which the vehicles in different lanes move with almost the same speed. In this region flow can be high in spite of an increasing density and an average speed of cars only about 1/2 that of free flow. In synchronized flow the linear correlation between flow and density is lost and the two quantities becomes totally non-correlated, that is a density increase can be accompanied by either a decrease or an increase in flow rate. At each traffic condition corresponds a different value of the average distance between cars.

These conditions of traffic (or phases, since they resemble in some aspects phase transitions in matter [14]), correspond to three distinct regions in a flow vs. density plot, also named fundamental plot. It is obtained plotting car flow as a function of car density for all positions along the highway and for all times. Every reasonable model of traffic must be able to correctly reproduce this plot, as is the case of the cellular automata model here described. In Fig. 1 such a plot, as obtained from one of the simulations performed with this program is reported.

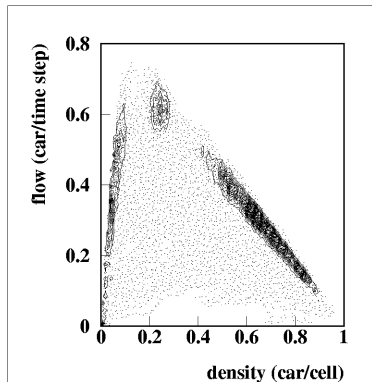


Fig. 1. Fundamental plot of traffic. The population contour lines reveal how most points (> 99%) concentrate into 3 regions. The first, on the left, corresponds to free flow and is characterized by a positive slope. The central contoured region corresponds to synchronized flow. The last contoured region, with negative slope, is due to the presence in the simulation of congested traffic.

Noteworthy, this cellular automata model is able to simulate dynamical features of traffic which were never observed with continuous and/or periodic models. Two examples are shown in Fig. 2 and in Fig. 3. In the first plot, the onset and backward propagation of a wide jam is reported. As found in real highways,

jams forms when car density becomes high and are nucleated by ramps (and also by junctions or roadworks, features not introduced at present in the model).

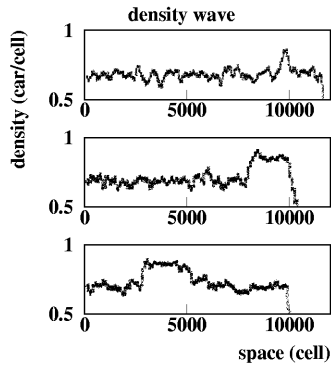


Fig. 2. In this simulation a density wave is generated at a certain time at about cell 10000 (top figure) and propagates backward (2400 and 10400 time steps later in the mid and bottom figure, respectively).

The second plot is particularly interesting to us. It shows how a single on-ramp is not enough to produce congested traffic. In the simulation relative to Fig. 3, a congested flow road with an entrance at the beginning and an exit at its end turns in time into free flow, despite the maximum possible number of cars entering the road at its beginning. In other words, only the presence of several ramps sufficiently closely spaced between them, can produce heavy traffic. This is a common experience for drivers that at the highway beginning or after a queue due to an accident, find fluid traffic and low car density.

3.1 Fractal Analysis

The results from the simple algorithm reported in the previous paragraph indicate how the complexity of traffic is not due to a great number of drivers, each behaving differently. It is rather the result of the repeated application of simple rules, which are surely not corresponding to a linear interaction between cars. Traffic is in fact a non linear system and the problem of writing down the differential equations corresponding to the evolution rules of the cellular automata problem is still open. This, together with the repeated fragmented look common to all plots obtained from the data of the various simulations performed, suggested us to look for self-similarity in traffic. To our knowledge such a research has never been done on experimental data and only once on simulated data from a unreal single lane round-about without entrance or exit [19].

Car density and flow as a function of space were analysed with the box counting algorithm [19–21]. This is probably the most widely used algorithm to

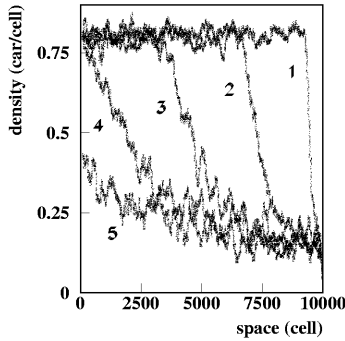


Fig. 3. Density as a function of space at different times in a simulation without ramps except at the beginning (car entrance) or the end (car exit) and a starting density of 0.8 car/cell. In this lane density decreases as a function of time: curve 1 refers to time=1000 time steps after the beginning of the simulation, curve 2 to time=5000 ts, curve 3 to time=10000 ts, curve 4 to time=15000 ts, curve 5 to time=20000 ts.

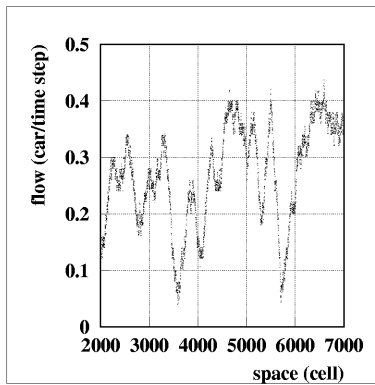


Fig. 4. Car flow as a function of space for a 5000 cell long treat of highway. The curve is repeatedly fragmented with self similar look over about 2 orders of magnification along the x-axis.

extract a non integer dimension from some data. Let consider for example Fig. 4. The plot area is divided into 25 boxes with side length $1/5$ of the abscissa and ordinate range of variation. This can be done again and again using n^2 boxes of decreasing side length ($1/n$ of the whole length). An estimate of the fractal dimension D of the curve can be obtained from a linear fit of $\log(S)$ vs. $\log(1/n)$, where $S(n)$ is the number of squares which contain at least one point of the curve. With a simple program implementing this algorithm, the value of S is readily obtained. The capability of the program to correctly yield the fractal dimension of several point sets was tested with fractal objects taken from mathematics and with non fractal curves. A further analysis, based on the discrete Fourier transform was accomplished, revealing that our traffic data do not have periodic components.

Values of the fractal dimension D between 1.4 and 1.6 were found in the examined data set over a range of about 2 orders of magnitude in highway length. This is not much in comparison with other simulations of fractal systems, but here we are working with a discrete system, so that it is not reasonable to expect a self-similar behaviour measured over a range greater than that of the experimental data.

The spread in D values coming from simulations performed with different traffic conditions could be an indication of a different fractal dimension for the various phases of traffic. We are at the present time trying to better our data set and to use other algorithms to measure D in order to clarify this point. We also plan in the next future to accomplish a similar analysis on real data, which will be collected on italian highways, to check the simulated results.

4 Conclusions

The use of cellular automata algorithms seems to be the most effective way to simulate global and dynamic features of highway traffic. Wave nucleation and propagation, transition between phases of traffic, lane inversion (a greater car density in left than in right lane): all most significant features experimentally known from highways seem to be reproducible. These simulations even suggest to look for features of traffic which were never noticed before, like an eventual fractal dimension. Finally, as a result of the confidence in the goodness of the model, it would be possible to test the effect of new traffic rules or drivers behaviour. For example, as suggested in [17] the usual driver behaviour is *competitive*, that is with continuous accelerations, reduced safety distance, etc. What about the effect of a non competitive behaviour? The *experimental* results reported in [17] and our personal experience, seems to show that there should be a noticeable difference. This could be easily simulated by a modified version of the model proposed here.

References

1. B. S. Kerner, H. Rehborn, Phys. Rev. Lett. **79** (1997), 4030.

2. B. S. Kerner, H. Rehborn, Phys. Rev. E **53** (1996), 4275.
3. M. Bando, K. Hasebe, A. Nakayama, A. Shibada, Y. Sugiyama, Phys. Rev. E **51**(1995), 1035.
4. A. D. May, *Traffic flow fundamentals* (Prentice-Hall 1990, New Jersey).
5. L. Neuber, L. Santen, A. Schadschneider, M. Schreckenberg, Phys. Rev. E **60** (1999), 6480.
6. M. J. Lighthill, G. B. Whitham, Proc. R. Soc. Lond. A **229** (1955), 281.
7. M. Bando, K. Hasebe, A. Nakayama, A. Shibada, Y. Sugiyama, Phys. Rev. E **51** (1995),1035.
8. S. Wolfram, Nature **311**, (1984), 419.
9. S. Wolfram, Theory and applications of Cellular Automata (World Scientific, Singapore, 1986).
10. K. Nagel, M. Schreckenberg, J. Physique **2** (1992), 2221.
11. K. Nagel, D. E. Wolf, P. Wagner, P. Simon, Phys. Rev. E **58**,(1998) 1425.
12. W. Knospe, L. Santen, A. Schadschneider, M. Schreckenberg, Physica A **265** (1999), 614.
13. O. Kaumann, R. Chrobok, J. Wahle, M. Schreckenberg, in: *8th Meeting of the Euro Working Group Transportation*, Editor M. Dielli (2000).
14. B. S. Kerner, Physics World **12**, (1999), 25.
15. E. G. Campari and G. Levi, Eur. Phys. J. B **17**, (2000), 159.
16. D. E. Wolf, Physica A **263**, (1999), 438.
17. <http://www.eskimo.com/billb/amateur/traffic/traffic1.html> .
18. <http://wwwinfo.cern.ch/asd/paw/>.
19. J. Woinowski, in: *Fractal Frontiers '97* Editors M. M. Novak and T. G. Dewey (World Scientific 1997, Singapore), pp.321-327 .
20. Paul Addison, *Fractals and Chaos* (IOP 1997, London).
21. Lui Lam, *Nonlinear physics for beginners* (World Scientific 1998, Singapore), pp.237-240 .