A Microscopic CA Model of Traffic Flow?

Peter Wagner[1(✉)] and Johannes Rummel[1,2]

[1] DLR, Institute for Transport Systems, Rutherfordstrasse 2, 12489 Berlin, Germany
{peter.wagner,johannes.rummel}@dlr.de
[2] Technical University Berlin, Berlin, Germany

Abstract. Cellular automaton (CA) models of traffic flow are typically constructed to reproduce macroscopic features of traffic flow. Here, a few thoughts based on real car-following data are presented that show how to construct a discrete time/discrete space microscopic model of traffic flow. The question whether this can still be called a CA-model is left to the reader.

Keywords: CA models for traffic flow · Natural driving studies
Traffic simulation

1 Introduction

The title of this article seems strange, since a traffic flow model like the CA introduced in [9] (see e.g. [2,8] for reviews) may be called with right a microscopic model of traffic flow.

And the answer is: no, since CA models are created to reproduce macroscopic features of traffic flow, and not microscopic ones. Clearly, this boundary is fuzzy [5,7], especially when it comes to CA models with smaller spatial discretization δx than the paradigmatic $\delta x = 7.5\,\mathrm{m}$ used in the original work of [3,9]. To make this boundary clearer, a microscopic model of traffic flow discrete in space and time will be constructed here which is based on real car-following data.

1.1 Notation, CA-Rules, Data-Set

The time, space, speed, and acceleration discretization (in real metric units) will be called $\delta t, \delta x, \delta v$ and δa, respectively. Each vehicle i at time-step t is described by a position x_i, a speed v_i, and the acceleration a_i. Sometimes, to write clearer equations, the index is dropped and instead capital letters X, V, A are used for the vehicle in front $i-1$, while the follower is described by small letters (x, v, a), Then, the net space headway g is defined as

$$g = X - x - \ell \quad \text{or} \quad g_i = x_{i-1} - x_i - \ell$$

where ℓ is the length of the vehicle plus the distance when standing. Note, that this is correct only when working with homogeneous traffic, i.e. all vehicles have

G. Mauri et al. (Eds.): ACRI 2018, LNCS 11115, pp. 230–239, 2018.
https://doi.org/10.1007/978-3-319-99813-8_21

the same length. If not, then one must fix the co-ordinate system (front of the car, e.g.) and include the lengths of both vehicles.

Another useful short-hand notation is the speed difference Δv between the lead and the following vehicle:

$$\Delta v = V - v = \dot{g} \quad \text{or} \quad \Delta v_i = v_{i-1} - v_i$$

The CA model introduced in [9] is characterized by $\delta t = 1\,\text{s}$, $\delta x = 7.5\,\text{m}$, from which $\delta v = 7.5\,\text{m/s}$ and $\delta a = 7.5\,\text{m/s}^2$ follows. Its deterministic rule-set is defined, in a slightly different variant as in the original formulation as:

$$a = \min\{g - v, 1\} \tag{1}$$

$$v' = \max\{\min\{v + a, v_0\}, 0\} \tag{2}$$

$$x' = x + v' \tag{3}$$

The primed variables are the updated variables, and the stochastic term is left-out because an explicit white noise acceleration noise is a bad physical description of a heavy vehicle. The complicated looking second Eq. (2) simply restricts the speed to the interval $[0, v_0]$.

The data to be used in the following are an excerpt (from 3 August 2012) from a German project named simTD [1], which was a natural driving study although with the goal to do research on vehicle-to-vehicle communication. There, about 100 vehicles, most of them instrumented with sensors to measure position, speed, acceleration, distance, and speed-difference to the lead vehicles drove with about 1000 different drivers for three months in an area around Frankfurt/Main, Germany. A more detailed description of the data can be found else-where [12], here the data from the car-following episodes have been used. A first glimpse into the data can be gained from Fig. 1, where the distributions of the speed, the speed differences, the gaps, and the time headways are displayed.

2 From Scratch

To fix the spatial discretization at the vehicle length as done in [9] is definitely very elegant. Nevertheless, this is also creating the biggest problems for a truly microscopic model, since it makes acceleration way too big. To fix that, it is argued here that there is a kind of a minimum acceleration step δa, that is given by the acceleration noise (its standard deviation denoted σ_a) created by human drivers. This variable is difficult to measure, the following approach is used here to get hold of it, at least to a certain approximation.

2.1 Acceleration Noise

Acceleration noise will be defined as follows: in a deterministic world, the acceleration of a vehicle is determined by a function $a(v, g, \Delta v)$, and eventually additional variables which are difficult to measure or have not been measured. In

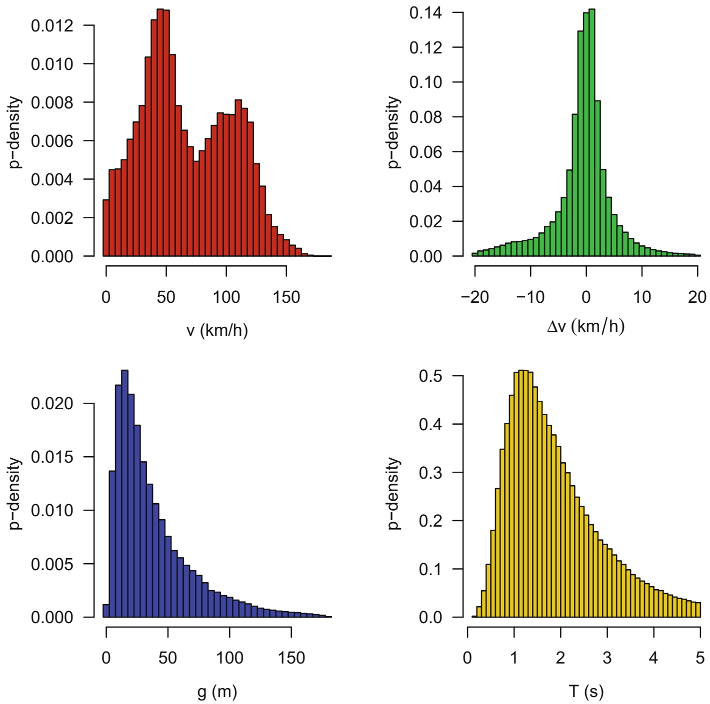

Fig. 1. The distribution of the variables speed (upper left), speed-difference (upper right), gap (lower left), and net time headway T (lower right). The headway data are filtered and contain only values where $v > 10\,\text{km/h}$, $g > 5\,\text{m}$, and $T \leq 5$.

a stochastic setting, even for the same set of variables, the acceleration at the same point is drawn from a distribution $p_{v,g,\Delta v}(a)$, and therefore it has a certain width, which can be quantified e.g. by the standard-deviation of the acceleration $\sigma_a(v, g, \Delta v)$. Neither the exact function $a(v, g, \Delta v)$, nor the distribution is known, and to make life even more complicated, it might be suspected that it depends on the state itself. However, both $a(v, g, \Delta v)$ and $\sigma_a(v, g, \Delta v)$ can be extracted from these data, to a rough approximation and within certain limits.

To do so, two sub-sets of the data are picked, a city and a freeway data-set. This is motivated by the two peaks in the speed distribution, so in a first step all speeds $v \in [40, 60]\,\text{km/h}$ (city) and $v \in [100, 120]\,\text{km/h}$ (freeway) will be selected. In addition, only data from close following situations will be used, which are defined by $\Delta v \leq 10\,\text{km/h}$ and $g \leq 80\,\text{m}$. Then, the phase-space $(\Delta v, g)$ is partitioned into boxes whose width is chosen so that in each dimension the same number of points is in each box. Within each box, now, the mean value of the acceleration and the corresponding standard deviation can be computed. This yields the result in Fig. 2.

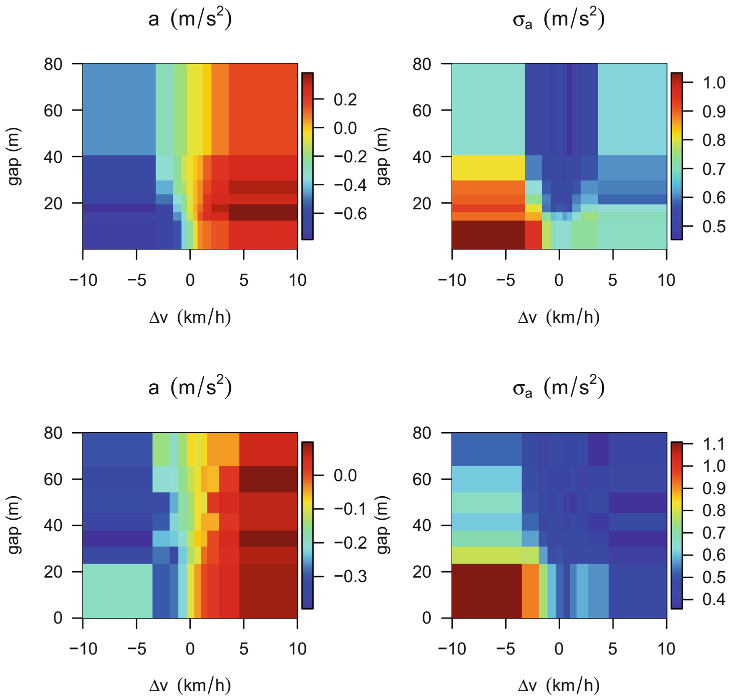

Fig. 2. $a(v = 50, \Delta v, g)$ (upper left) and $\sigma_a(v = 50, \Delta v, g)$ for city speeds, and for freeway speeds (lower row).

From Fig. 2, a minimum value of $\sigma_a \approx 0.4\,\text{m/s}^2$ may be read out. This will be chosen in the following equal to the minimum acceleration step $\delta a = 0.4\,\text{m/s}^2$.

2.2 Time Step Size δt

Having fixed $\delta a = 0.4\,\text{m/s}^2$, the time-step size is the next. Here, either the minimum reaction time of humans, which is of the order of 0.3 s might be used. Interestingly, this value is also close to the minimum time headway found in empirical data, see Fig. 1. Using $\delta t = 0.25\,s$ gives an additional safety margin and it divides nicely by 1 (what is not really necessary).

Having fixed δa and δt, which also shows that this construction process is bottom up, the remaining discretization values are determined, too. Since

$$\delta a = \delta v / \delta t \quad \text{and} \quad \delta v = \delta x / \delta t,$$

the spatial discretization follows to be $\delta x = \delta a \delta t^2$ which gives $\delta x = 0.025\,\text{m}$. This is of course way smaller than the value in [9]. For the speeds, this yields $\delta v = 0.1\,\text{m/s}$. For maximum speeds larger than 25.5 m/s, this will no longer fit into a 1 byte integer.

2.3 The Dynamics

There are two observations relevant here: the first is that human control is discrete. At so called action-points [10], the human controler changes acceleration (more precisely, the gas- and or brake-pedal position) quickly, and keeps it constant for 0.5...5 s, see [11] for the distribution of this times which may follow a gamma-distribution. This might have made another time-discretization δt, but it is more difficult to measure, since a small acceleration change is hard to discern from high-frequency acceleration noise. However, at least the size is similar to the 0.25 s that have been chosen above.

The second observation is that to model the car-following process (car-following is abbreviated CF in the following), at least the original model is too gross. Additionally, it mixes the car-following with a safety consideration, and since the small time-step size chosen above lead to plenty of modelling leeway, these two will be separated in the following. So, there is an emergency braking which happens if the normal car-following fails and that is with the typical Euler backward update formulated as follows:

$$\text{if } v > g/\delta t \text{ then } v = g/\delta t \tag{4}$$

Since in the discretization framework chosen here $\delta t = 1$, this is exactly the same as in the original formulation.

For the CF-process, a slightly more complex approach is chosen. Instead of the original dependence on distance g alone, the model acts as a linear controler with limited acceleration a_0, which is also pretty close to some of the adaptive cruise control of modern driver assistant systems. Furthermore, there is the need to introduce two pre-factors c_1, c_2 to the two terms in the equation, whose physical relevance will be discussed later on. Here, they simply scale the variables to a reasonable size (typically, $c_i < 1$).

$$a_0 = \min\left\{\max\left\{c_1\left(\frac{g}{T} - v\right) + c_2(V - v), -\beta\right\}, \alpha\right\} \tag{5}$$

and a_0 is bounded between $-\beta$ from below, and α from above, which again is written as the complicated min, max combination. This is however not the whole model. In addition, the action-points are included in the dynamics pretty much like in a Monte Carlo simulation. This acceleration a_0 is only made the new acceleration with a certain probability p_{AP}, so the additional step is needed:

$$a' = \begin{cases} a_0 & \text{with probability } p_{AP} \\ a & \text{else} \end{cases} \tag{6}$$

The rest, then, is just the same as in Eqs. (2), (3):

$$v' = \max\{\min\{v + a', v_0\}, 0\}$$
$$x' = x + v'$$

This is a simple linear controler, and it clearly has a stable fixed-point at $v = g/T$. The randomness injected by the action-point mechanism is too weak to change this. However, some randomness is needed, since it seems almost impossible to find car-following data where the lead vehicle's speed is constant. There are many ways how to do this, and the worst one (not to be followed here) is to add acceleration noise. Better is the approach of the so called 2D models [4]. There the preferred headway T is made stochastic. Here, we only allow three values of the headway $T = 3, 4, 5$ (in units of δt), and the driver switches between them each time a new AP is issued.

3 Phenomenology

Here, we simulate first the microscopic features, and then have a look at the macroscopic ones like the fundamental diagram.

3.1 Microscopic

For the microscopic description, it should be noticed that the behaviour, and especially the stability of the car-following process is strongly determined by the choice of the two parameters c_1, c_2. Clearly, they are inverse time-constants, named T_g and $T_{\Delta v}$. From the theory of linear controlers which is being used for adaptive driver assistant systems, it is well-known, that a platoon of those vehicles is only platoon-stable for a certain range of parameters, typically for small values. The model here is a little bit more complicate than a linear controler, since the acceleration is bounded, making it a non-linear controler. Therefore, for too small values of T_x, the acceleration distribution becomes bimodal with peaks at the limiting value a_0. Which is not realistic. Therefore, there is only a fairly small range of values for the T_x, where they are platoon stable and not bimodal. In the following, the parameters $T_1 = 26$ (6.5 s) and $T_2 = 6$ (1.5 s) are chosen, while $p_{\text{AP}} = 0.4$ is used. For a string of four vehicles, this yields then the results presented in Fig. 3.

The lower panel compares the distributions of Δv and of T with the empirical ones. While this fits well for the speed-differences, it does not so well for the headways. The simulation has used only three headway values $T = 3, 4, 5$ (0.75, 1, 1.25 s in real units), it seems that the distribution of real drivers is much wider than this three values.

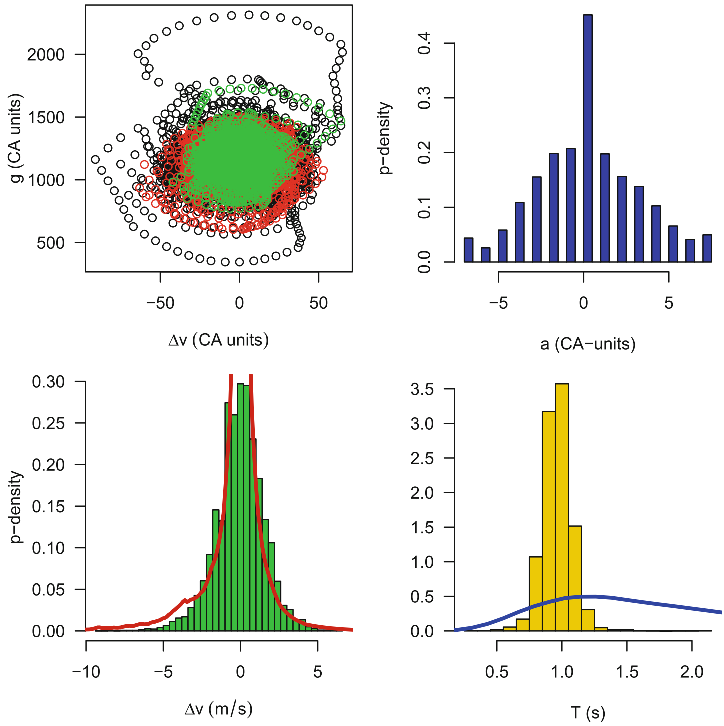

Fig. 3. The behaviour of the model in $(\Delta v, g)$-space, for the first follower (green), the second (red) and the third (black, upper left). The upper right displays the acceleration distribution in CA-units, still a slight peak at the boundary value is to be seen. The lower panel displays the speed-difference and headway distribution, and compares it directly with the empirical ones. (Color figure online)

3.2 Macroscopic

To compute a fundamental diagram, the simplest possible set-up is chosen. The vehicles run in a ring, which means that the density k is the control parameter of the fundamental diagram. Simulations are run for 1024 vehicles for 40 values of the scaled density $k/k_{\max}$ values ranging from 0.01 to 0.99. The system is started either in a homogeneous or in a jammed configuration, each simulation runs for 10^5 time-steps of which the first 25,000 time-steps are discarded. Only the speed distribution is sampled, where 40 simulation steps are left out between each sampling step to minimize correlations between subsequent states. The results shown in Fig. 4 the following: at small densities, only the free flow state exists. There is an intermediate range, where the system's state consists of a mixture of free flow and jams, and finally, for large densities, the system eventually becomes bistable.

No more detailed and exhaustive studies have been performed.

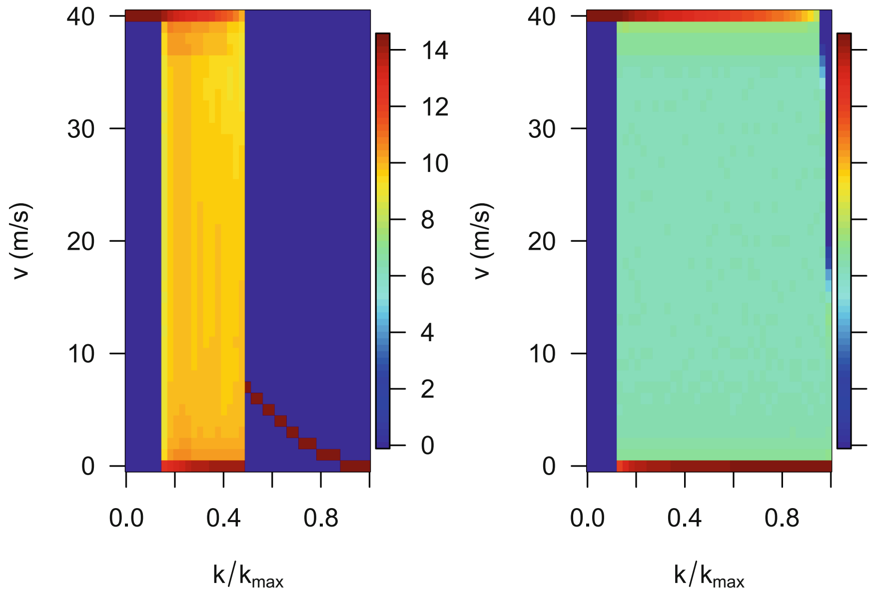

Fig. 4. The fundamental diagram of the model. Presented is the speed distribution $p(v)$ as function of the scaled density. The left plot is for homogeneous initial conditions, while the right plot for a system started in a jammed condition, the color scale is logarithmic to include more information about the distributions.

4 Execution Speed

There are a lot of science tales around the fantastic numerical performance of CA models. Partially, they stem from the time where the floating point units of computers were weak in comparison with the integer units, and therefore it was worth to look for integer-based models. With this model at hand that lingers between discrete and continuous, it is time to have a look into this issue. Note, that there is also a much more detailed contribution to this conference by Moreno Marzolla, which discusses in much detail what can be reached and which kind of tricks to apply in order to achieve good performance. Therefore, only a few short remarks to the issue of numerical performance may be added here.

One of the challenges with this problem is, that the performance depends very much on a lot of things that are difficult to control. Just for starters, when writing a program that implements the model in [9], and comparing it with a continuous one that implements SUMO's [6] default model, not much of a difference can be seen. This difference becomes even smaller, when real world things like copying data from the running simulation, or if the open system's simulations are needed. For some-one to dive into the depths of modern compilers and the tremendous possibilities of modern CPU's such as their SIMD (=Single Instruction Multiple Data) units, some differences can be found.

The implementation used here runs with 30 MUPS on a Pentium G4400 with Windows 10 and cygwin's g++ compiler in the version 6.4.0, which is on par with a implementation of the model of SUMO without any fancy tricks. The unit MUPS means Mega Updates Per Second and has been introduced in [9]. This is just by using a normal optimization setting of a year 2017 compiler. If the SIMD units of modern CPU's are utilized, this performance improves, in some cases by a factor of 10. Interestingly, there is not much difference between the CA implementation and the continuous one. But note, that this is only possible in the case of a closed system, where the numbers of particle (cars) does not change. It may also work, if the CA is implemented in a grid-based manner instead of a vehicle-based which is more efficient in terms of memory, but then a lot of simulation time is spent by updating empty cells. We have not found an implementation that can utilize SIMD or GPU's when simulating an open system car-based, which is the case for most applied settings. Then, those beautiful speed-ups vanish and one sticks with the 30 MUPS above.

5 Conclusions

In conclusion, the building blocks of a microscopic discrete space/discrete time CA model have been presented. When following the approach pursued here, this leads to a CA model with time-step size $\delta t = 0.25\,\mathrm{s}$ and spatial discretization $\Delta x = 0.025\,\mathrm{m}$. While still discrete, we think that such a model is not easy to discern from a continuous model, if this possible at all. When looking into the plots in this article only the acceleration is clearly discrete. It may be interesting to see what of the features included here can be left out to still have a valid description of microscopic driving. Another valid conclusion is also, that such simple models as the CA have their value, but should not be used when a really microscopic description of traffic flow is needed for the application at hand.

References

1. Safe and intelligent mobility - test field germany (2012). http://www.simtd.de/index.dhtml/enEN/index.html, http://www.simtd.de/index.dhtml/enEN/index.html. Accessed 7 Jul 2015
2. Chowdhury, D., Santen, L., Schadschneider, A.: Statistical physics of vehicular traffic and some related systems. Phys. Rep. **329**(4–6), 199–329 (2000)
3. Cremer, M., Ludwig, J.: A fast simulation model for traffic flow on the basis of boolean operations. Math. Comput. Simul. **28**, 297–303 (1986)
4. Jiang, R., et al.: Traffic experiment reveals the nature of car-following. PLoS One **9**(4), e94351 (2014). https://doi.org/10.1371/journal.pone.0094351. http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0094351
5. Kerner, B.S., Klenov, S.L., Wolf, D.E.: Cellular automata approach to three-phase traffic theory. J. Phys. A Math. Gen. **35**, 9971–10013 (2002)
6. Krajzewicz, D., Erdmann, J., Behrisch, M., Bieker, L.: Recent development and applications of SUMO - Simulation of Urban MObility. Int. J. Adv. Syst. Meas. **5**(3&4), 128–138 (2012)

7. Lee, H.K., Barlovic, R., Schreckenberg, M., Kim, D.: Mechanical restriction versus human overreaction triggering congested states. Phys. Rev. Lett. **92**, 238702 (2004)
8. Maerivoet, S., Moor, B.D.: Cellular automata models of road traffic. Phys. Rep. **419**(1), 1–64 (2005). https://doi.org/10.1016/j.physrep.2005.08.005. http://www.sciencedirect.com/science/article/pii/S0370157305003315
9. Nagel, K., Schreckenberg, M.: A cellular automaton model for freeway traffic. Journal de Physique I France **2**, 2221–2229 (1992). https://doi.org/10.1051/jp1:1992277
10. Todosiev, E.P., Barbosa, L.C.: A proposed model for the driver-vehicle-system. Traffic Eng. **34**, 17–20 (1963/1964)
11. Wagner, P.: A time-discrete harmonic oscillator model of human car-following. Eur. Phys. J. B **84**(4), 713–718 (2011). https://doi.org/10.1140/epjb/e2011-20722-8
12. Wagner, P., Nippold, R., Gabloner, S., Margreiter, M.: Analyzing human driving data an approach motivated by data science methods. Chaos Solitons Fractals **90**, 37–45 (2016). https://doi.org/10.1016/j.chaos.2016.02.008. http://www.sciencedirect.com/science/article/pii/S096007791630039X