Network breakdown "at the edge of chaos" in multi-agent traffic simulations

Marcel Rieser[1,2] and Kai Nagel[1]

[1] Transport Systems Planning and Transport Telematics (VSP), TU Berlin, Salzufer 17-19, Sekr. SG 12, 10587 Berlin, Germany, {rieser,nagel}@vsp.tu-berlin.de
[2] Institute for Transport Planning and Systems (IVT), ETH Zurich, 8093 Zurich, Switzerland, rieser@ivt.baug.ethz.ch

1 Introduction

Much of our technological infrastructure (traffic, electricity, telephone, water) operates on networks. Many of the phenomena on infrastructure networks display spatio-temporal structure. For example, traffic congestion only occurs at certain places and times; electricity is most used on very cold or very hot days, and in specific neighborhoods; telephone networks are heavily used on special days (e.g. Christmas) or after exceptional events (e.g. catastrophes).

Both the theory and the modelling/simulation of spatio-temporal systems made important progress in the last decades. However, the following two aspects differentiate infrastructure systems from "standard" spatio-temporal systems:

(1) The spatial substrate of the dynamics of infrastructure systems is a network instead of "flat" 2d or 3d space. Even though these networks are embedded in space, strong inhomogeneities in the network structure, such as broad degree distribution of the nodes or their clustering, lead to different behaviors than more smoothly embedded structures exhibit.
(2) The "forces" behind the network usage are based on human behavioral patterns. These are known to have long range temporal correlations for individual behavior even in simple situations (i.e. sending print jobs to a printer [1], or replying to emails).

In consequence, progress in "standard" spatio-temporal systems alone will not suffice to understand infrastructure systems.

Much progress has been made in the pragmatic modelling and simulation of aspects (1) and (2). This is particularly true for traffic [2, 3, 4, 5, 6, 7]. What is missing is a better connection between those pragmatic real-world simulations and a deeper understanding of the dynamic processes in those simulations. The ultimate hope is that better understanding leads to (even) better pragmatic real-world models, and to an improved functioning of the system itself.

This paper attempts to contribute to this by reporting occurrences of network breakdown in large-scale multi-agent transport simulations. These occurrences have been observed by us for many years, but this is the first systematic description of them. The simulation model that is used for those investigations is one of the "pragmatic" real world models mentioned earlier rather than a "minimal" model that contains nothing but the ingredients necessary to generate the observed phenomenon. Investigations with such a minimal model should follow.

Although the model is not minimal, it is speculated that the following mechanism is at work:

1. The basic "physical" dynamics consists of daily traffic in a congested metropolitan region.
2. The simulation is run for many days in sequence ("iterations"), and the synthetic travellers that produce the traffic can adapt from iteration to iteration.
3. It is observed that many of these "days" traffic runs smoothly. But sometimes, traffic "breaks down", leading to macroscopic, network-spreading traffic jams (see Fig. 2), and only the end of the rush period can eventually resolve them. It is speculated that the mechanism is similar to the simple network breakdown mechanism displayed in simple 2d traffic models [8, 9, 10], although the microscopic dynamics of the breakdown in true networks (as opposed to the flat 2d space used in those references) looks rather different.
4. Such a network breakdown is usually caused by a microscopic fluctuation (a queue that is a couple of vehicles longer than normal), which has macroscopic consequences. It is, however, observed that traffic then remains problematic for the next couple of iterations (days), somewhat akin to an "avalanche" [11, 12]. It is speculated that this is because of violent adaptation reactions of the synthetic travellers, which therefore disrupt the "normal" traffic pattern. It is speculated that the mechanism is similar to the "decision avalanches" found in route choice experiments with real humans [13, 14].

This paper will describe the investigations that were performed. It therefore consists of the following sections: Section 2 describes the simulation setup. Since this is not a minimal model, this section is rather long. Section 3 will describe our observations which are discussed in Section 4. Section 5 contains our conclusions.

2 Simulation Setup

2.1 Overview

The foundations of traffic simulations are a (road) network and some description of the traffic. In multi-agent simulations, the traffic is described by the sum of all agents. Every agent has zero or more trips planned to travel from one place to another. All the trips of all the agents describe the total traffic demand. The network consists usually of a series of nodes, and of links connecting the nodes. The links can contain additional attributes describing physical aspects of the link like the number of lanes, the length of the link, or the maximum speed allowed on the link.

In our case, agents have at least one plan, of which exactly one is selected and executed during the traffic simulation. A plan contains a list of activities, and of legs connecting the activities. Activities contain information about the location, the type of activity and the planned start and end time for the activity. Legs contain information about the planned departure and expected travel time, the travel mode (car, bike, public transport, . . .), and the exact route through the network, given by the list of nodes the agents will cross on its leg. For the purposes of this paper, “legs” and “trips” are the same.

As traffic simulation, we use MATSim [3], our own implementation of a multi-agent transport simulation that is based on TRANSIMS [2]. One scenario run consists of multiple iterations, each iteration consisting of a run of the traffic flow simulation (sometimes also called “physical layer”) and a run of the agent re-planning process (sometimes also called agent learning, or “mental layer”).

2.2 Traffic flow simulation (physical layer)

The traffic flow simulation is a comparatively simple so-called queue simulation. This is essentially a queueing model simulation, with the important difference that links can be full, causing spillback into upstream links. Input parameters into the queue simulation, besides the traffic network topology, are, for each link: free speed on the link (vehicle speed in the absence of congestion), flow capacity (maximally possible exit flow), and storage capacity (maximally possible number of vehicles on the link.) Compared with the original version of the queue simulation [15], we now use intersections priorities according to capacities [16], and a deterministic rather than a randomized service rate (where vehicles are served when some counter has exceeded the average waiting time).

2.3 Agent replanning (mental layer)

During agent re-planning, a fixed percentage of agents make a copy of an existing plan and modify it. In the next iteration, this modified plan is executed and scored. Possible modifications are:

route adaption: choose different routes through the network to travel from one activity to the next one.
time adaption: choose different activity durations and thus different departure times for trips

Additional modifications could be thought of (e.g. reordering the sequence of activities, dropping or adding activities, choosing a different location for an activity), but are not yet implemented. To limit the memory usage, there is a limit of the number of plans an agent can remember. Once an agent reaches this limit, the plan with the lowest score will be deleted.

2.4 Scoring Plans

The correct scoring of the simulated plans is crucial to the success of the simulation. As the agents try to optimize their daily routine, the score must reflect the dis-utilities of travel as well as the utilities of performing activities. Minimizing the travel time by choosing an alternate departure time does not help the agent if she arrives too early or too late at an activity location (e.g. shop has already closed on arrival, or the agent arrives too late at work).

The utility function used is derived from the traditional utility function based on the Vickrey bottleneck model [17], but is modified to be consistent with complete day plans:

$$U_{plan} = \sum_i U_{act,i} + \sum_i U_{trav,i} + \sum_i U_{late,i} \tag{1}$$

The utility of performing an activity is assumed to increase logarithmically:

$$U_{act,i}(x) = max(0, \alpha \cdot ln(\frac{x}{t_0})) \tag{2}$$

where x is the duration that one spends at the activity. Time spent waiting at an activity because of arriving too early (e.g. before a shop opens) is not included in x. We take $\alpha = \beta_{dur} \cdot t^*$, where β_{dur} is uniformly the same for all activities and only t^* varies between activity types. With this formulation, t^* can be read as "typical" duration for an activity, and β_{dur} as the marginal utility at that typical duration:

$$\left.\frac{\partial U_{act,i}}{\partial x}\right|_{x=t^*} = \beta_{dur} \cdot t^* \cdot \frac{1}{t^*} = \beta_{dur} \tag{3}$$

t_0 can be seen as a minimum duration of an activity, but is better interpreted as a priority: All other things being equal, activities with large t_0 are less likely to be dropped than activities with small t_0 (for details, see [18]).

The utilities of traveling and of being late are both seen as dis-utilities which are linear in time:

$$U_{trav,i}(x) = \beta_{trav} \cdot x \tag{4}$$

where x is the time spent traveling, and

$$U_{late,i}(x) = \beta_{late} \cdot x \tag{5}$$

where x is the amount of time an agent arrives late at an activity. In our simulations, β_{trav} is set to -6 EUR/h, and β_{late} is set to -18 EUR/h.

In principle, arriving early or leaving early could also be punished. There is, however, no immediate need to punish early arrival since waiting times are already indirectly punished by foregoing the reward that could be accumulated by doing an activity instead (opportunity cost). In consequence, the effective (dis)utility of waiting is already $-\beta_{dur}$. Similarly, that opportunity cost has to be added to the time spent traveling, arriving at an effective (dis)utility of traveling of $\beta_{trav} - \beta_{dur}$. No opportunity cost needs to be added to late arrivals, because the late arrival time is already spent somewhere else. These effective values are the standard values of the Vickrey model (Arnott et al. 1993).

Because the scoring function uses monetized costs and gains, the function could be easily extend to include tolls or other external effects.

At the end of each traffic simulation, the score σ for each simulated plan is calculated. The calculated (new) score U_{plan} is then assigned to the plan according to:

$$\sigma = \begin{cases} U_{plan} & \text{if the plan was executed the first time} \\ (1-\lambda)\cdot\sigma + \lambda \cdot U_{plan} & \text{if the plan already has a score} \end{cases} \tag{6}$$

λ is the learning rate with which the agents adapt the new score. If the learning rate is lower than 1, the new score will not completely overwrite an existing score, but only influence it (exponential smoothing).

2.5 Scenario

The simulated scenario is located around Zurich, Switzerland. Both the road network and the traveller population are based on realistic data. The set of activities, however, was reduced to “home – work/education – home” [19]. This leads to plausible morning rush hour traffic. A total of over 260’000 agents were simulated, corresponding to all people commuting by car in the aforementioned region.

Two different runs where simulated, one with a learning rate of $\lambda = 1$ (no memory of old scores) and one with a learning rate of $\lambda = 0.1$ (exponential smoothing).

Both scenarios start with the re-planning set so that 10% of the agents do route adaption and 10% do time adaption, while the others simply chose the plan with the best score from their plan memory for simulation in the next iteration. After iteration 50, the percentages of agents doing some kind of adaption is reduced to 5% each.

3 Observations

As in most iterative simulations, the system first takes several iterations in which the average agents' score improves steadily, until it levels at some value. But even then, the agents' average score does not stay at that level, but has more or less severe slumps from time to time. Figures 1.a and 1.c show the average agents' score for the two scenarios, where the slumps can be clearly seen. Figure 1.b shows one of those slumps in more detail, corresponding to the highlighted region in fig. 1.a.

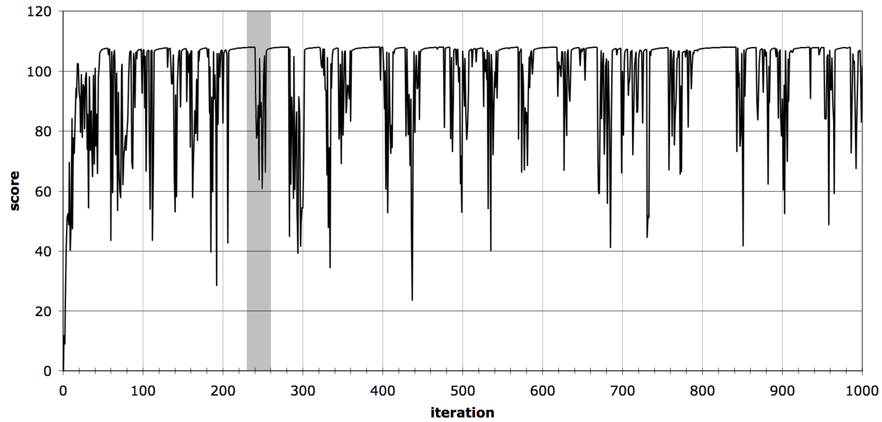
While the slumps in fig. 1.c for the scenario with the learning rate set to 0.1 are clearly smaller than in 1.a, they still occur frequently. The reduced amplitude of the slumps is due to the exponential smoothing, which is applied before the data is extracted for the plot. Since the amplitude is reduced by about the same factor as the learning rate, it is expected that the main contribution to the amplitude reduction is simply a result of the smoothing rather than a consequence of a modified dynamic behavior of the system.

The reason for those slumps in the average score gets visible if the actual network state is visualized. Fig. 2 shows a visual representation of the network in three consecutive iterations at different times. While in iteration 240 only a few minor traffic jams can be observed which have no further consequence ("fluctuations on the micro-scale"), a major traffic jam starts building in iteration 241 at 7:35am in the center of the city. The tailbacks of this initial traffic jam spread wider and wider into the network in the following minutes, until most parts of the network are jammed. We call this situation a network breakdown, where no more traffic is possible. In the following iteration, iteration 242, most agents have chosen another plan. Those agents that could re-plan seem to have mostly chosen a route leading through north of the previous center of the traffic jam. The consequence is that in this iteration this route is overloaded: Already at 7:15am, there are some severe traffic jams in the northern part of the city, which slowly extend themselves until at 7:45am another network breakdown can be seen.

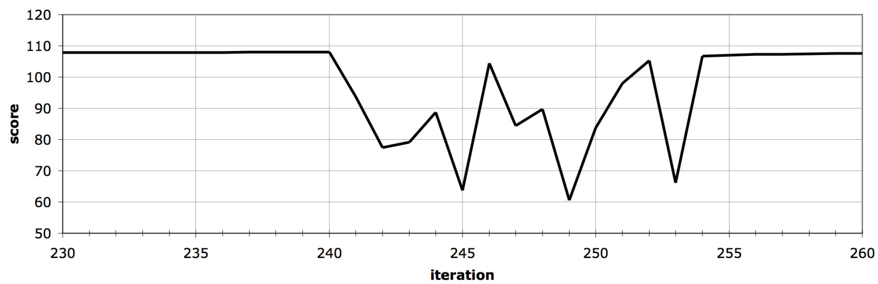
Figure 3 shows the Fourier transform of the agents' average scores. Technically, the original time series was loaded into xmgrace (http://plasmagate.weizmann.ac.il/Grace/), the Discrete Fourier Transform (DFT) was applied to the data, and as output option "Magnitude" was chosen. According to a simple test with synthetic data this means $S(f) = \sqrt{A(f)^2 + B(f)^2}$, where $A(f)$ and $B(f)$ are the amplitude coefficients. We are unable to find universal agreement if this or the square of it should be the (generalized) power spectrum.

With a learning rate of $\lambda = 0.1$, a fairly convincing $S(f) \sim 1/f$-dependency can be recognized. For $\lambda = 1$, there is only a short slope section, from $f \approx 0.02$ to $f \approx 0.1$.

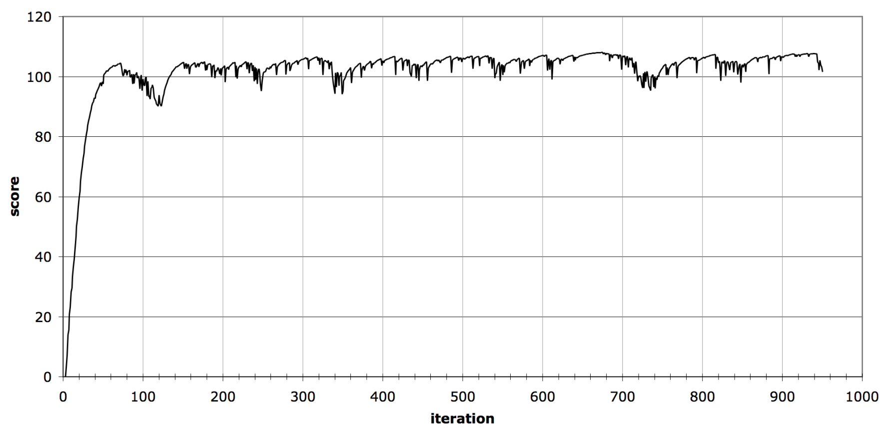
For comparison, a random walk was simulated and then submitted to the same treatment; the result is plotted in blue. One notices that also the random walk displays $1/f$-behavior under these circumstances, implying that the



a) Average agents' score with learning rate $\lambda = 1$.



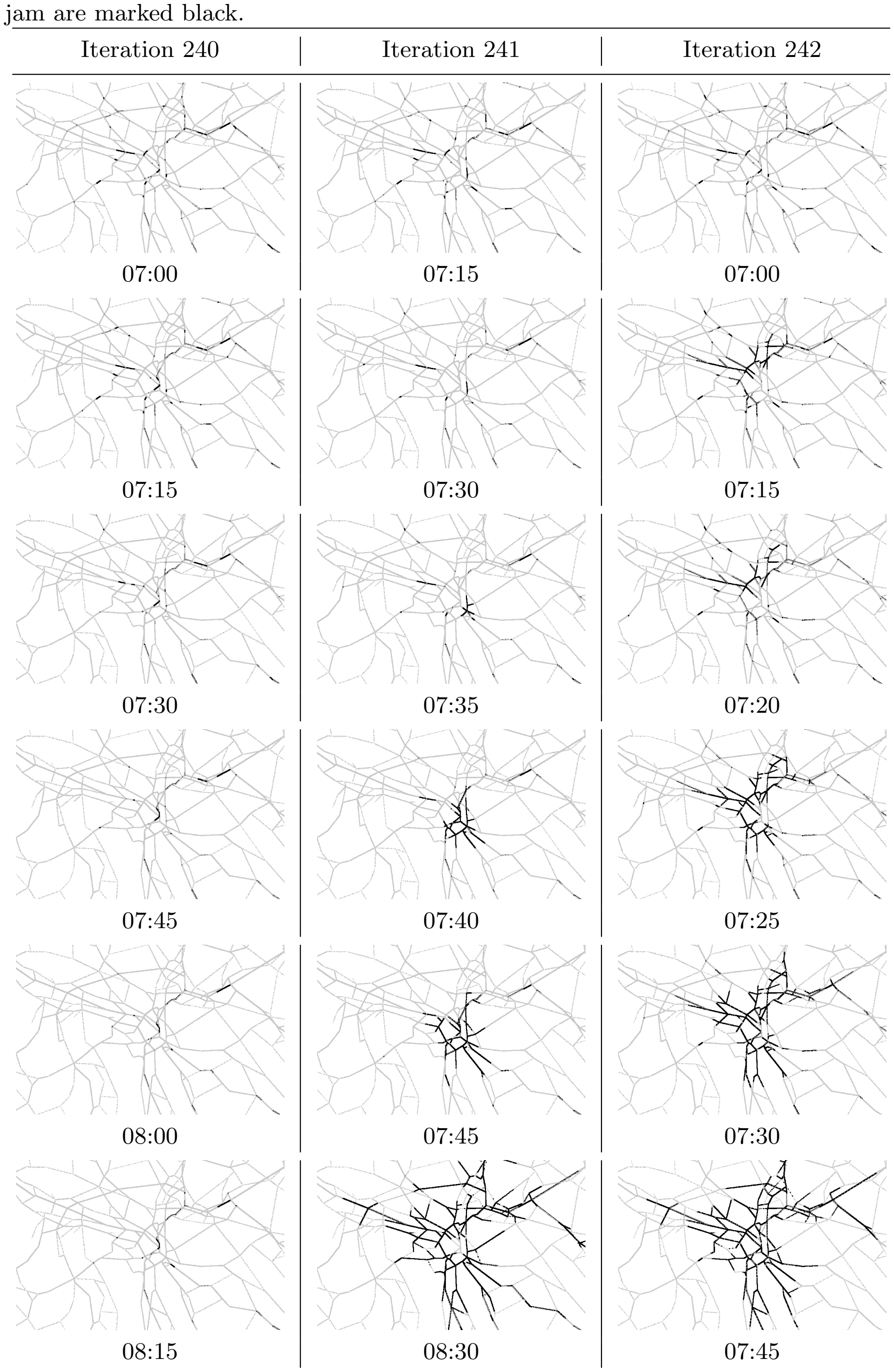
b) Detail of the average agents' score with learning rate $\lambda = 1$.



c) Average agents' score with learning rate $\lambda = 0.1$.

Fig. 1. The average agents' score over the iterations

Fig. 2. Visual representation of the network breakdown. Vehicles stuck in a traffic jam are marked black.



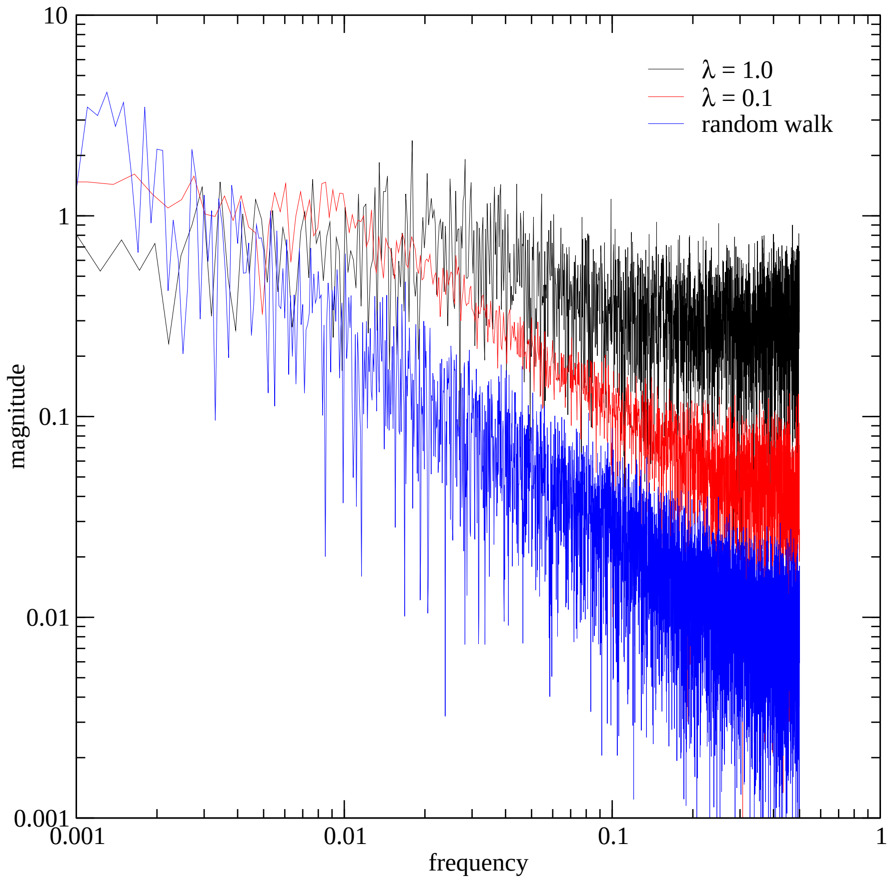


Fig. 3. Fourier transformations of the time series of the agents' average score. For comparison, the corresponding treatment of a random walk is shown (in blue).

$\lambda = 0.1$ time series displays random-walk-like characteristics in the spectrum. The distinct difference between the random walk spectrum and the spectrum of the $\lambda = 0.1$ score time series is that the $\lambda = 0.1$ spectrum flattens out for frequencies smaller than ≈ 0.01. This means that there seem to be no correlations beyond ≈ 100 iterations, which is consistent with the observation that there are no periods larger than 50 between network breakdown avalanches (Fig. 1.a). Tests with different lengths of time series imply that this is a true finite time cut-off.

Since the $\lambda = 0.1$ spectrum comes from a time series that is essentially a filtered (exponentially smoothed) version of the $\lambda = 1$ spectrum, it makes sense to assume that the random walk behavior of the $\lambda = 0.1$ spectrum is caused by the filtering.

4 Discussion

We explain the breakdown avalanches with the agents' striving to further optimize their plans. They push the network capacities up to the limits, so that as soon as some additional cars want to travel along these paths, the maximum capacities are transgressed and traffic jams occur. Because many other links are also at their limit, the initial traffic jam cannot be absorbed by the surrounding links, leading to the expansion of the congestion until one huge traffic jam occurs in which the cars are in a deadlock. We dub this "at the edge of chaos" [20] since on most days, the (simulated) traffic system functions orderly, interrupted by occasional avalanches of breakdown. It is also reminiscent of self-organized criticality [11]. Yet, the fact that there is a peak in the $\lambda = 1$ spectrum implies that the documented behavior rather has a weakly periodic structure: A network breakdown pushes the system away from the "critical edge" by a certain amount; it then needs a certain time to approach the edge again; a breakdown occurs and the system is pushed away; etc.

Traffic network breakdowns do not only occur in models, but also in reality. As mentioned in the introduction, Refs. [13, 14] discuss route choice experiments with human subjects that display similar avalanche behavior. Experience shows that in some traffic system, relatively small disturbances (a construction site, or degraded weather conditions) can have a huge impact, clogging up the system for hours. It is therefore a bit doubtful if time and effort should be invested to remove these effects from the simulation, as traditionally is done. We expect that data to verify or falsify these effects in the real world will become available in the near future.

5 Conclusions

It was shown that a real-world traffic model with learning iterations displays network breakdown "avalanches". These avalanches are separated by relatively long periods of calmness in which the system operates rather smoothly. Already during those calm periods, small scale fluctuations in the network performance (i.e. localized jams) can be observed. Sometimes, these fluctuations trigger a large scale breakdown, which, after initiated, quickly spreads through the network. Once an iteration displays network breakdown, it is highly likely that successive iterations also display network breakdown, leading to the above-mentioned avalanches. The slope of the power spectrum of the time series of the agent scores may be flatter than that of a random walk, but the results are inconclusive.

The intuitive explanation for the observed phenomena is that (a) agents optimize for themselves until the system is pushed "to its limits"; (b) the system "fights back" not by gradual degradation but instead by erratic complete breakdowns.

The observed results are not only of theoretical importance, but also of practical relevance. Existing transport planning software usually calculates a steady traffic flow distribution. These models cannot reflect the instabilities observed in such networks. Multi-agent simulations are not yet able to give answers when the network is surcharged and collapses or how the erratic behaviour of the system could be reduced. But by looking at the simulation history and not just analyzing one single iteration, multi-agent traffic simulations could at least help to determine how likely or how often network breakdowns may occur.

References

1. U. Harder and M. Paczuski. Correlated dynamics in human printing behavior. *Physica A*, 361(1):329,336.
2. TRANSIMS www page. TRansportation ANalysis and SIMulation System, accessed 2005. Los Alamos National Laboratory, Los Alamos, NM.
3. MATSIM www page. MultiAgent Transport SIMulation, accessed 2007.
4. DYNAMIT/MITSIM, accessed 2007. Massachusetts Institute of Technology, Cambridge, Massachusetts. See mit.edu/its.
5. DYNASMART www page, accessed 2007.
6. R.J.F. Rossetti and R. Liu. Towards a multi-agent traffic simulation model. In *Workshop on agents in traffic and transportation at Autonomous agents and multiagent systems (AAMAS'04)*, New York, NY, July 2004.
7. J. Esser. *Simulation von Stadtverkehr auf der Basis zellularer Automaten.* PhD thesis, University of Duisburg, Germany, 1998. See also www.traffic.uni-duisburg.de.
8. O. Biham, A. Middleton, and D. Levine. Self-organization and a dynamical transition in traffic-flow models. *Physical Review A*, 46:R6124–R6127, 1992.
9. F.C. Martinez, J.A. Cuesta, J.M. Molera, and R. Brito. Random versus deterministic 2-dimensional traffic flow models. *Phys. Rev. E*, 51(2), 1995.
10. T. Nagatani. Jamming transition in a two-dimensional traffic flow model. *Phys. Rev. E*, 59(5):4857–4864, 1999.
11. P. Bak, C. Tang, and K. Wiesenfeld. Self-organized criticality. *Phys. Rev. A*, 38:368, 1988.
12. Bak P and Sneppen K. Punctuated equilibrium and criticality in a simple model of evolution. *Phys. Rev. Lett.*, 71(24):4083–4086, 1993.
13. D. Helbing, M. Schonhof, and D. Kern. Volatile decision dynamics: experiments, stochastic description, intermittency control and traffic optimization. *New Journal of Physics*, 4(33), 2002.
14. R. Selten and M. Schreckenberg. Experimental investigation of day-to-day route choice behavior. Springer, Berlin Heidelberg, 2004.
15. C. Gawron. An iterative algorithm to determine the dynamic user equilibrium in a traffic simulation model. *International Journal of Modern Physics C*, 9(3):393–407, 1998.
16. N. Cetin, A. Burri, and K. Nagel. A large-scale agent-based traffic microsimulation based on queue model. In *Proceedings of Swiss Transport Research*

Conference (STRC), Monte Verita, CH, 2003. Earlier version, with inferior performance values: Transportation Research Board Annual Meeting 2003 paper number 03-4272.
17. R. Arnott, A. De Palma, and R. Lindsey. A structural model of peak-period congestion: A traffic bottleneck with elastic demand. *The American Economic Review*, 83(1):161, 1993.
18. D. Charypar and K. Nagel. Generating complete all-day activity plans with genetic algorithms. *Transportation*, 32(4):369–397, 2005.
19. M. Balmer, B. Raney, and K. Nagel. Adjustment of activity timing and duration in an agent-based traffic flow simulation. In H.J.P. Timmermans, editor, *Progress in activity-based analysis*, pages 91–114. Elsevier, Oxford, UK, 2005.
20. Langton C G. Life at the edge of chaos. In Langton C G et al., editor, *Artificial Life II*, Santa Fe Institute Studies in the Science of Complexity, Vol. 10, Redwood City, CA, 1992. Addison-Wesley.