Rieser, M.; Nagel, K.

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

Journal article | **Accepted manuscript (Postprint)**

This version is available at https://doi.org/10.14279/depositonce-8351

This is a post-peer-review, pre-copyedit version of an article published in The European Physical Journal B. The final authenticated version is available online at: http://dx.doi.org/10.1140/epjb/e2008-00153-6

Rieser, M.; Nagel, K. (2008). Network breakdown ‘at the edge of chaos’ in multi-agent traffic simulations. The European Physical Journal B, 63(3), 321–327. https://doi.org/10.1140/epjb/e2008-00153-6

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

Abstract

Traffic is highly influenced by network structure and human behaviour. Small changes in the human behaviour can lead to huge changes in the load of a traffic network. Current transportation models do not, and most of them cannot, research such random behaviour but always calculate a steady state. In our multi-agent transport simulation, we frequently observe seemingly random ”network breakdowns”, huge traffic jams that spread over a big part of the network, making a normal traffic flow impossible. This paper describes the investigations that were performed on the results of our large-scale multi-agent transport simulations in an attempt to contribute to the better understanding of the dynamic processes in such simulations and, hopefully, better understanding and modelling of the real-world.

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 (e.g. [1]). 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, can lead to different behaviours than more smoothly embedded structures exhibit.

(2) The "forces" behind the network usage are based on human behavioural patterns. These are known to have long range temporal correlations for individual behaviour even in simple situations (i.e. sending print jobs to a printer [2], or replying to emails [3]).

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 [4, 5, 6, 7, 8, 9]. With respect to "understanding", there is a large body of work concerning traffic flow on links (e.g. [10, 11, 12]) and a smaller body of work concerning simple 2d traffic models (e.g. [13, 14, 15]). What is missing, however, is a better connection between the pragmatic real-world simulations and those models which are simpler but better understood. 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 in different setups, but this is the first systematic description of them. Breakdown behaviour in traffic models that go beyond the link were in fact reported earlier [13, 14, 15]. Those models, however, use a simple 2d geometry instead of a true network; they have no vehicle storage capacity between the nodes; and they use periodic boundary conditions. In contrast, the simulation model that is used for the investigations presented here 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. That minimal model still needs to be found.

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. 1), and only the end of the rush period can eventually resolve them. It is speculated that the mechanism is similar to the network breakdown mechanism displayed in simple 2d traffic models [13, 14, 15], 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" [16, 17]. It is speculated that this is because of adaptation reactions of the synthetic travellers on the *following* days, which disrupts the "normal" traffic pattern. The mechanism may be similar to the "decision avalanches" found in route choice experiments with real humans [18, 19].

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

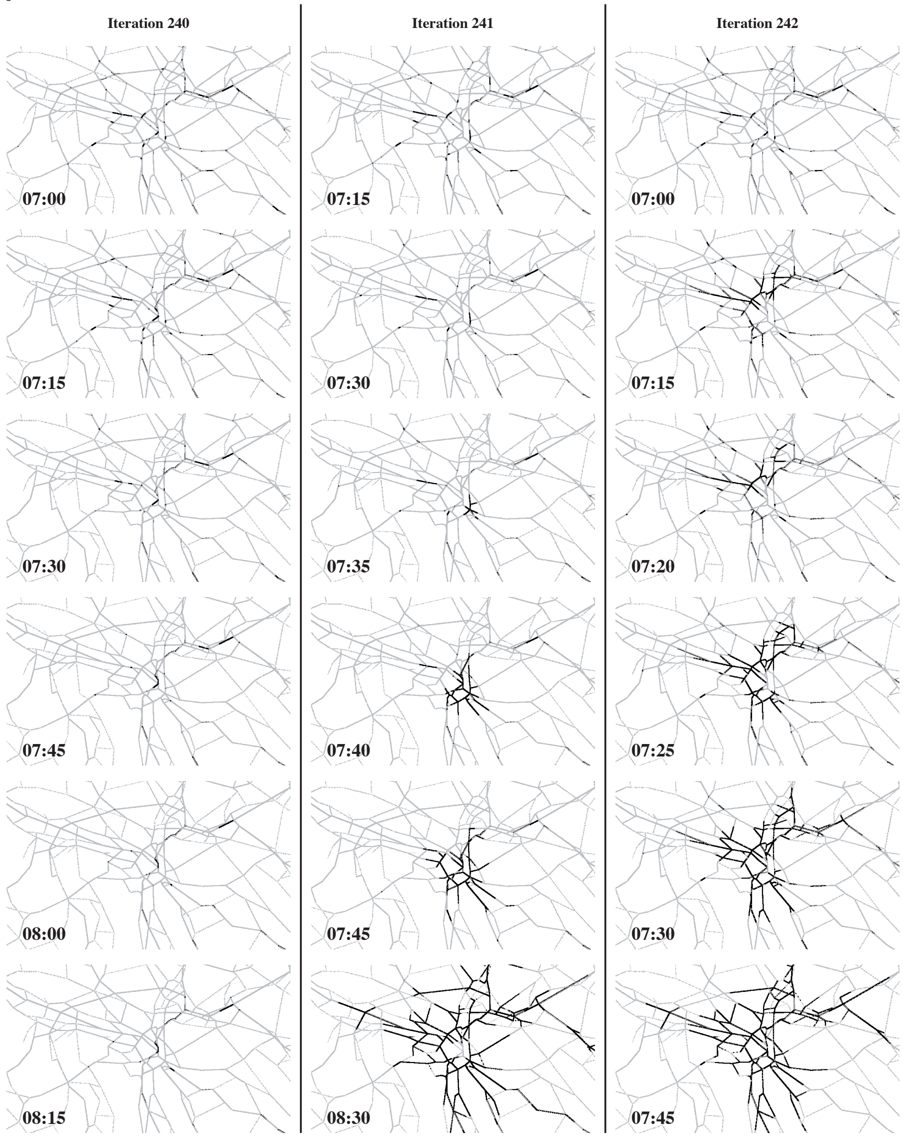
2.1 Overview

The basic ingredients 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 plus their individual dynamics. 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 of a set 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 trips connecting the activities. Activities contain information about the location, the type of activity and the planned start and end time for the activity. Trips 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 agent will traverse.

As traffic simulation, we use MATSim [5], our own implementation of a multi-agent transport simulation that is based on TRANSIMS [4]. 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 replanning process (sometimes also called agent learning, or "mental layer").

Fig. 1. Visual representation of the network breakdown. Vehicles stuck in a traffic jam are marked black.



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 [20], we now use intersections priorities according to capacities [21], and a deterministic rather than a randomized service rate (vehicles are served when some counter has exceeded the average waiting time).

2.3 Agent replanning (mental layer)

During agent replanning, 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 it 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 [22, 23], 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 [24]).

The utilities of travelling 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 travelling, 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 travelling, arriving at an effective (dis)utility of travelling 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 [22, 23].

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

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

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" [25]. 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.

A run was then started with the replanning 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 ("base run"). After iteration 50, the percentages of agents doing some kind of adaption is changed to 5% each to reduce fluctuations. The intial 100 iterations might, in consequence, be seen as transients, but removing them from the analysis did not change the results. The simulation is run to 4070 iterations (more than shown in Fig. 2). This run is the basis for the main analysis of this text.

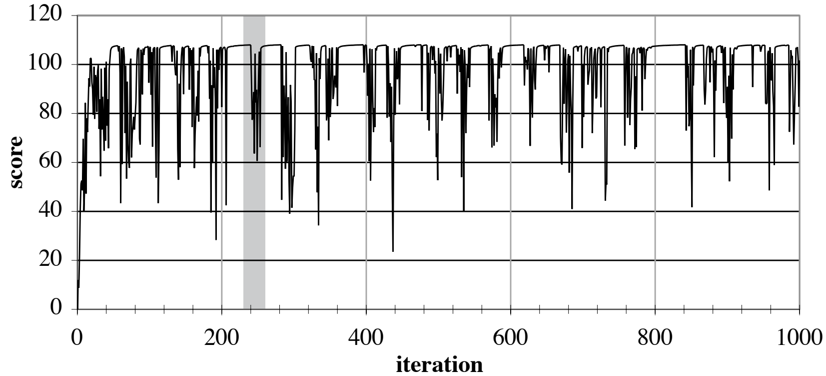
Two additional runs were started without any replanning. The base run's set of plans of iteration 230 were taken as initial plans for the two additional runs. This specific iteration was chosen because it was in the middle of a "quiet" period. Both additional simulations were run for over 1000 iterations.

The first of these additional runs takes the whole available set of plans per agent, but no new plans are generated (no "innovation"). In every iteration, each agent decides which of the plans should be selected to be executed by the traffic flow simulation, choosing according to probabilities based on the scores of the plans ("selection only" run). The probabilities are of the form $p \propto e^{\beta \cdot S_j}$, where S_j is the score of plan j, and β is an empirical constant. This is similar to a logit model from discrete choice theory [26]. Behaviourally, it corresponds to a situation where each member of the population, in every iteration, makes a random draw out of a fixed set of several options, with weights based on the agent's beliefs beliefs, and the beliefs are updated every time after the execution of the option.

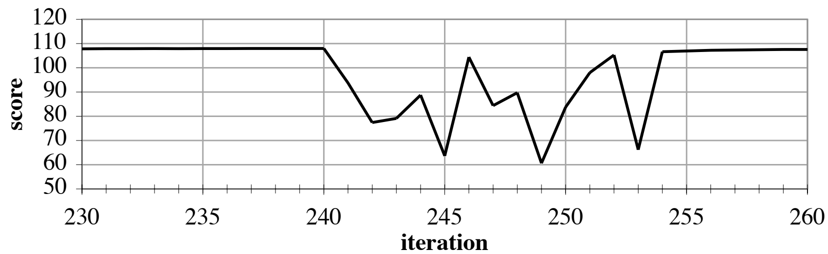
The second of the additional runs only keeps the currently selected plan of each agent, discarding all other plans. That is, in every iteration, all agents have the same identical plans. As the traffic flow simulation uses random numbers (e.g. to decide who may cross a node first when two vehicles arrive at the same time), and the random seed is set to a different value in each iteration, the results of the traffic flow simulation still vary from iteration to iteration ("fixed plans" run).

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. Fig. 2.a shows the average agents' score for the base run, where the slumps can be clearly seen. Fig. 2.b shows one of those slumps in more detail, corresponding to the highlighted region in Fig. 2.a.



a) Average agents' score.



b) Detail of the average agents' score.

Fig. 2. The average agents' score over the iterations

The reason for those slumps in the average score gets visible if the actual network state is visualized. Fig. 1 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 replan 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.

In order to gain some quantitative understanding, a spectral analysis was performed with the time series. Fig. 3 shows the smoothed periodogram of the agents' average scores, using the function `spec.pgram` from the software

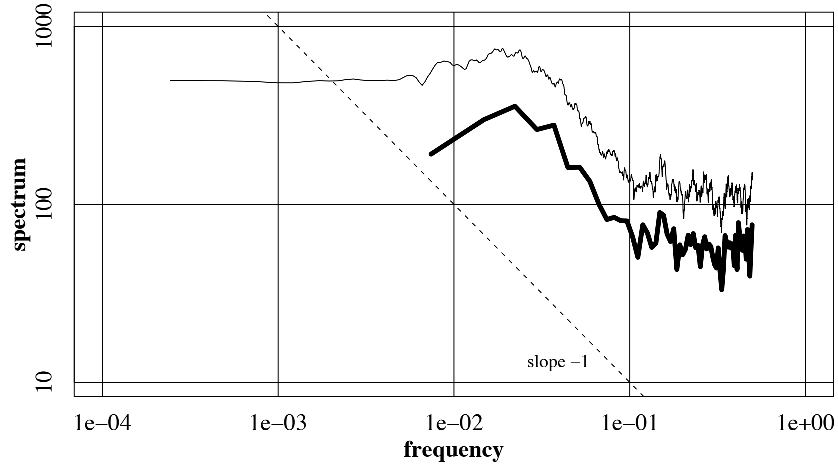


Fig. 3. Smoothed periodogram of the time series of the agents' average score. The different lines refer to different smoothing techniques, described in the text.

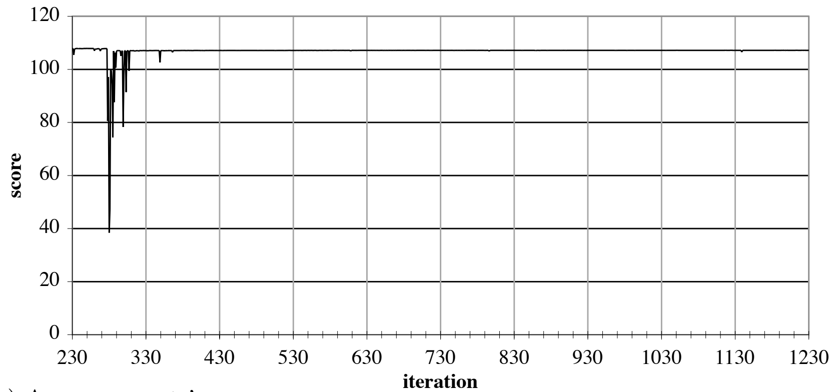
R [27]. The thin line uses Daniell smoothing (`spans=51` in R; a variant of a moving average), while the thick line cuts the time series in 30 pieces of equal length, computes the unsmoothed periodogram on these pieces, and averages the result. The result is also divided by two to offset the plots from each other. The dashed line shows the function $1/f$ for comparison.

One notices an $1/f$-like slope between the frequencies of 0.02 and 0.1. This $1/f$ behaviour is very noticeably different from the $1/f^2$ behaviour of a random walk that was, for comparison, submitted to the same treatment (not shown). To the left of the $1/f$-like slope is a peak, followed by a flat line. To the right of the $1/f$-like slope is a range with a slope that cannot be distinguished from zero.

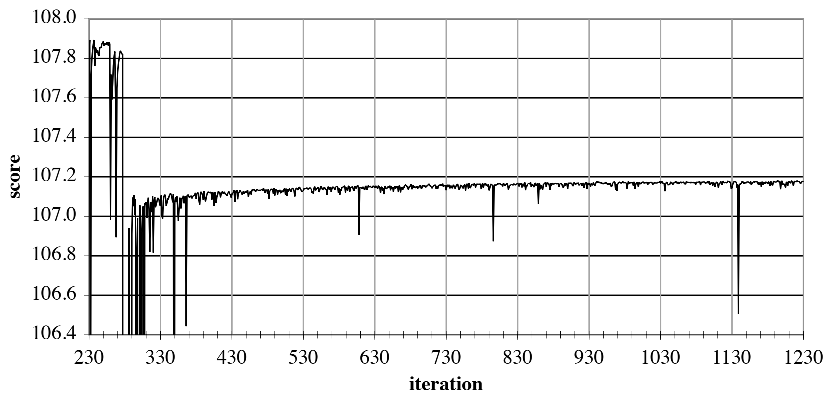
When comparing with the time series (Fig. 2), one may suspect the following:

- The nearly flat spectrum at large frequencies seems to correspond to the single avalanches. Fig. 2.b shows one such avalanche; from Fig. 2.a one observes that most if not all avalanches seem to have that "typical" behaviour. The fact that these avalanches take about 10 iterations corresponds to the fact that this flat regime extends to frequencies down to about $F \approx 0.1$.
- The peak at approximately $f \approx 0.02$ corresponds to the observation that avalanches seem to happen every approximately 50 iterations. The fact that cutting the time series into pieces does not shift the peak very much implies that the location of the peak is probably not a finite size effect. If this is correct, then even a much longer time series would display the same peak, meaning that long pauses between avalanches do not happen often enough to cause a "fat tail". – What remains unclear in this discussion

is if a larger *traffic system* (i.e. a city/region) might shift the peak to the left.



a) Average agents' score .

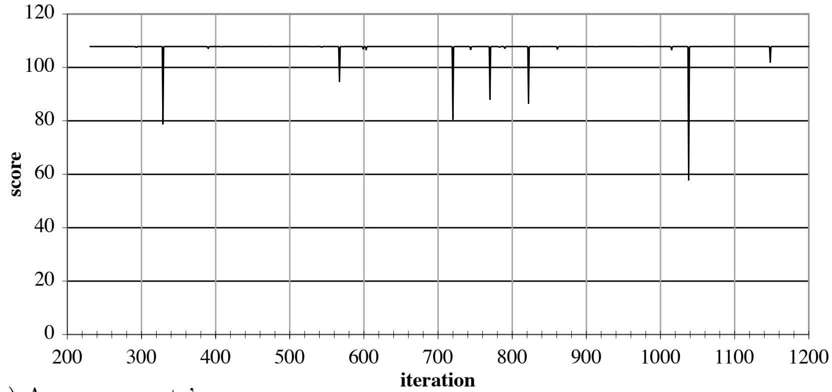


b) Detail of the average agents' score.

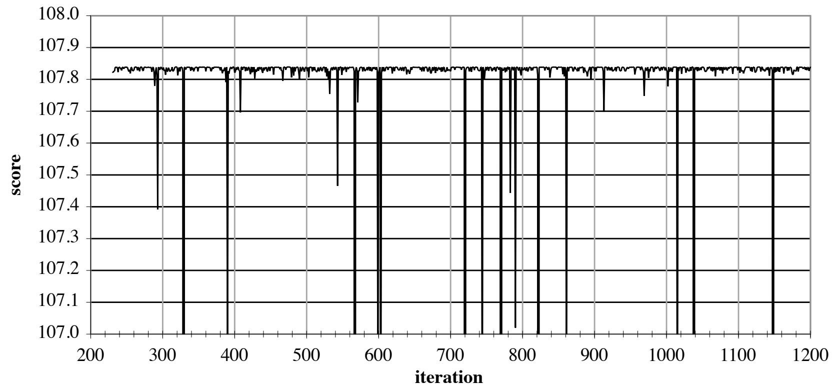
Fig. 4. The average agents' score over the iterations with agents selecting from a fixed set of plans.

4 Discussion

We explain the breakdown avalanches in traffic networks 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 the most crowded paths, the maximum capacity is exceeded and traffic jams occur. Because many other links are also at their limit, the initial traffic jam cannot



a) Average agents' score.



b) Detail of the average agents' score.

Fig. 5. The average agents' score over the iterations when only the random seed of the traffic flow simulation changes from iteration to iteration.

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" [28] since on most days, the (simulated) traffic system functions orderly, interrupted by occasional avalanches of breakdown. It is also reminiscent of self-organized criticality [16]. Yet, the fact that there is a peak in the spectrum implies that the documented behaviour 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.

The fact that the avalanches are intricately connected to the learning behaviour of the agents can be taken from the additional runs. Fig. 4.a shows the time series of the "selection only" run. One observes that there are one or arguably two large avalanches from iteration 272 to iteration 310, but after that, the system remains quiet. Fig. 4.b, at higher resolution on the y-axis, shows that the score has decreased over the initial avalanche, and is not coming back to the initial level during the thousand iterations that were simulated.

Fig. 5 finally shows what happens if one leaves the plans fixed as in the 230th iteration, but re-runs the microsimulation with different random seeds. One finds *isolated* network breakdowns that never result in avalanches.

Both additional runs together support the argument that only the learning of the agents, including the computation of new routes, triggers the frequent breakdowns: The “critical” set of plans from iteration 230, without any adaptation, keeps triggering breakdowns, but no avalanches. On the other hand, adaptation without innovation triggers one large avalanche, after which breakdowns are rare or non-existent. Only adaption with innovation triggers the frequent breakdown avalanches.

Traffic network breakdowns do not only occur in models, but also in reality. As mentioned in the introduction, Refs. [18, 19] discuss route choice experiments with human subjects that display similar avalanche behaviour. Experience shows that in some traffic system, relatively small disturbances (a construction site, degraded weather conditions, a single accident) 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 future, with the increasing deployment of “Intelligent Transport Systems (ITS)”.

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 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.

This work was funded in part by DFG (NA 682/1-1). We gratefully acknowledge significant amounts of computing time on the Beowulf cluster "simulant" provided and maintained by the Institute for Transport Research of the German Aerospace Centre (DLR) at Berlin-Adlershof. We also thank M. Paczuski and J. Davidsen for discussions regarding the analysis and interpretation of the time series.

References

1. Y. Shi, G.D. v. Albada, J. Dongarra, and P.M.A. Sloot, editors. *Computational Science – ICCS 2007: 7th International Conference*, volume 4487–4490 of *Lecture Notes in Computer Science*. Springer New York, 2007.
2. U. Harder and M. Paczuski. Correlated dynamics in human printing behavior. *Physica A*, 361(1):329–336, 2006.
3. J.P. Eckmann, E. Moses, and D. Sergi. Entropy of dialogues creates coherent structures in e-mail traffic. *Proceedings of the National Academy of Sciences*, 101(40):14333, 2004.
4. TRANSIMS www page. TRansportation ANalysis and SIMulation System. http://transims.tsasa.lanl.gov/, accessed 2008. Los Alamos National Laboratory, Los Alamos, NM.
5. MATSIM www page. MultiAgent Transport SIMulation. http://matsim.org/, accessed 2008.
6. DYNAMIT/MITSIM www page. http://mit.edu/its, accessed 2008.
7. DYNASMART www page. http://www.dynasmart.com/, accessed 2007.
8. R.J.F. Rossetti and R. Liu. A Dynamic Network Simulation Model Based on Multi-Agent Systems. In F. Klügl, A. Bazzan, and S. Ossowski, editors, *Applications of Agent Technology in Traffic and Transportation*, Whitestein Series in Software Agent Technologies and Autonomic Computing, pages 88–93. Birkhäuser, 2005.
9. J. Esser. *Simulation von Stadtverkehr auf der Basis zellularer Automaten*. PhD thesis, University of Duisburg, Germany, 1998. See also www.traffic.uni-duisburg.de.
10. D. Helbing. Traffic and related self-driven many-particle systems. *Reviews of Modern Physics*, 73:1067–1141, 2001.
11. D. Chowdhury, L. Santen, and A. Schadschneider. Statistical physics of vehicular traffic and some related systems. *Physics Reports*, 329(4–6):199–329, May 2000.
12. W. Knospe, L. Santen, A. Schadschneider, and M. Schreckenberg. Towards a realistic microscopic description of highway traffic. *J. Physics A: Math. General*, 33:L477–L485, 2000.

13. O. Biham, A. Middleton, and D. Levine. Self-organization and a dynamical transition in traffic-flow models. *Physical Review A*, 46(10):R6124–R6127, 1992.
14. F.C. Martínez, J.A. Cuesta, J.M. Molera, and R. Brito. Random versus deterministic two-dimensional traffic flow models. *Phys. Rev. E*, 51(2):R835–R838, Feb 1995.
15. T. Nagatani. Jamming transition in a two-dimensional traffic flow model. *Phys. Rev. E*, 59(5):4857–4864, 1999.
16. P. Bak, C. Tang, and K. Wiesenfeld. Self-organized criticality. *Phys. Rev. A*, 38(1):364–374, 1988.
17. P. Bak and K. Sneppen. Punctuated equilibrium and criticality in a simple model of evolution. *Phys. Rev. Lett.*, 71(24):4083–4086, 1993.
18. D. Helbing, M. Schönhof, and D. Kern. Volatile decision dynamics: experiments, stochastic description, intermittency control and traffic optimization. *New Journal of Physics*, 4(33):1–16, 2002.
19. R. Selten and M. Schreckenberg. Experimental investigation of day-to-day route choice behavior. In M. Schreckenberg and R. Selten, editors, *Proceedings of the Workshop on Human Behaviour and Traffic Networks*, pages 1–21. Springer, Berlin Heidelberg, 2004.
20. 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.
21. 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.
22. W. S. Vickrey. Congestion theory and transport investment. *The American Economic Review*, 59(2):251–260, 1969.
23. 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):161–179, 1993.
24. D. Charypar and K. Nagel. Generating complete all-day activity plans with genetic algorithms. *Transportation*, 32(4):369–397, 2005.
25. 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.
26. M. Ben-Akiva and S. R. Lerman. *Discrete choice analysis*. The MIT Press, Cambridge, MA, 1985.
27. R Development Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2007. ISBN 3-900051-07-0.
28. C.G. Langton. Life at the edge of chaos. In C.G. Langton et al., editor, *Artificial Life II*, Santa Fe Institute Studies in the Science of Complexity, Vol. 10, pages 41–92, Redwood City, CA, 1992. Addison-Wesley.