Stability Analysis Of Traffic Flow Models

Thesis in theoretical physics,
Eidgenössische Technische Hochschule in Zürich,
written by
Christopher Kayatz

Advisors:
Prof. Kai Nagel,
Prof. Matthias Troyer

August 26, 2001

Contents

Chapter 1

Introduction

Have you been in a jam on a highway that urged you to stop your vehicle but there was no visible reason that caused the jam?

Such jams are often called *phantom jams*, because they are not generated by reasons that we usually expect to cause a jam such as accidents, construction work, police action or obstacles of any kind. A phantom jam might be caused for instance by a traffic density that is too high. If any car in a very dense but still laminar traffic slows down unnecessarily, his behavior might cause a jam behind him.

In transportation science several kinds of jams can be of interest.

Alot of reserach is done on simulations of entire cities where jam occurance is examined depending on road networks, alternative public transportation, working hours and day planning habits etc.

This work is interested in jam occurance depending on the traffic density and the free parameters of the traffic models. We therefore examine a periodic street filled with cars according to a global density ρ.

The vehicular motion will be given by two traffic flow models. We will examine a model by Stefan Krauss and compare it to a discrete model, the Nagel-Schreckenberg cellular automat.

To compare and analyze traffic models, we will use the aspects explained in the following.

A very easy way to give a traffic flow model a "face" is to draw its *fundamental diagram*. This is a plot of the flow or flux, vehicles per second, towards the density: $q(\rho)$, where $q = \langle v(\rho) \rangle \rho$ using the average velocity $\langle v(\rho) \rangle$ at a given system density ρ. The model of Stefan Krauss (SKM) shows hysteresis, which we will investigate further in section 3.2.

Phantom jams depend intuitively on the global traffic density. This is

supported by simulations. We will analyze the breakdown behavior of the traffic models mentioned above as well as the recovery behavior. These are transitions from laminar to jammed traffic and from jammed to laminar traffic respectively. Hence we have two initalization possibilities. To get an idea of these transition behaviors we will measure the breakdown and recovery times (T_{bdown} and T_{rec}) depending on the traffic density ρ, system size (number of vehicles) N_{cars} and the free parameters in the SKM. T_{bdown} and T_{rec} seem to be of an exponential character. The questions we will ask ourselves are

What information do we get out of plots of T_{bdown} and T_{rec}?

How are these plots correlated to the plots of the fundamental diagram?

What kind of exponential behavior do T_{bdown} and T_{rec} show?

Are they diverging?

The SKM can be divided into three types I, II and III of traffic flow models. These types are distinguished by their choice of the parameters *acceleration a* and *braking capability b*. In agreement with Krauss and Janz, it is hard to distinguish the types I and II. Details about these configurations will be presented later.

The *outflow* from jams is a criterion to determine differences. Type I is said to have a *stable* outflow, whereas type II has an *unstable* outflow. Stable and unstable outflow refer to the downstream behavior below an artificial jam. In a stable outflow we find no further jams (daughter jams). In an unstable outflow we do. These daughter jams can appear and disappear on time scales T_{rec} and T_{bdown} respectively.

As a consequence, the interface (spacial transition between the artificial jam and free flow) of type II models is smeared out. We measured the interface width of both models in various configurations in order to categorize the models and the types I and II.

The leading questions are:

How does the interface width of different model types develop in time? ($\sim$ const., $\sim T$, $\sim T^{\frac{1}{2}}$)

How does it depend on the model type?

Chapter 2

Traffic Flow Models

There are various ways of implementing traffic on computers. Depending on the aim and purpose of the simulation different models apply better.

Before we explain the traffic flow models used in this work, we'll have a rough overview over other traffic models.

All traffic flow models (TFMs) are based on the thought of providing collision free car following. The apporaches being used can be categorized roughly as follows:

Macroscopic TFMs

The similarity between unidirectional traffic flow and the flow of a viscous liquid is obvious. Thus many models apply Navier-Stokes-style equations on traffic simulations. Collision free traffic is provided indirectly by the mass conservation equation and the existence of pressure.

Microscopic TFMs

Looking at individuals in traffic we can generate various rules of behavior for safe driving. In driving school we learn the *2 seconds headway*-rule, giving us a safe distance to the car in front of us. This rule, or slight adaptions of it, are the fundamental thought behind microscopic TFMs, updating every individual's speed and position at every time step. An individual driver usually adapts his speed taking only the leading car into account.

Continuous space vs. discrete space

Using microscopic TFMs gives us two possibilities in handling the positioning of vehicles: discrete space coordinates ($x_i \in \mathbb{N}$) or continuous space coordinates ($x_i \in \mathbb{R}$).

Continuous space coordinates allow a more realistic and smooth driving while discrete space coordinates provide faster updating and computing.

In discrete space coordinates, vehicles are placed into cells and "jump" from cell to cell in order to drive. The cellular automaton model uses this technique. The latter will be discussed and used in this work as an adaption of Krauss' model.

Of course one could construct hybrid models, using combinations of the two suggestions mentioned here. The reader can consult many articles and books on such topics.

2.1 Krauss' traffic flow model

The SK model is a microscopic continuous-space TFM. The SKM does not try to implement human decision making as realisticly as possible, it rather tries to implement what we observe when looking at individual behavior in traffic. In the following we want to give the assumptions relevant for the Stefan Krauss model, but skip the detailed derivation of the time discretization and the update rule.

2.1.1 Background, assumptions

Let us think of a single car with its leader so far ahead that it doesn't have to be taken into account for its velocity updating. In such a free headway situation we need to set a very general restriction, to give the velocity an upper boundary:

$$v \leq V_{\max}$$

This maximum velocity could be considered as the drivers desired velocity on a freeway or the maximum velocity given by traffic rules.

In dense traffic situations, individuals aim for collision free driving. Thus it is usually safer to drive within a certain safe velocity range:

$$v \leq v_{\text{safe}}$$

As in most microscopic models, v_{safe} of a following car has to take into account: the current velocity v_f of the car itself, the velocity of the leading car v_l and the gap g between the two,

$$v_{\text{safe}} = v_{\text{safe}}(vf, v_l, g)$$

While respecting these two velocity boundaries, vehicles try to obtain the hightest velocity within the given range. If possible they try to increase their speed according to a given acceleration a.

Using discrete time steps Δt, we have a temporary update rule

$$v(t + \Delta t) = \min\left(\mathrm{V_{max}}, \mathrm{v(t)} + \mathrm{a}\Delta\mathrm{t}, \mathrm{v_{safe}}\right)$$

We need to specify a few more things:

When updating the velocity for the next time step, it is realistic to bound acceleration and deceleration and implement an update rule that takes limited velocity changing capability into account. We call b the maximum deceleration and a the maximum acceleration capability:

$$-b \leq \frac{d}{dt}v \leq a$$

As we will see later, the restricted $\frac{d}{dt}v$ is essential in this model and distinguishes it from many others. The choice of a and b defines different types of macroscopic behavior and jam formation.

Gaps between vehicles are measured from bumper to bumper:

$$g = x_l - x_f - L_{\text{car}}$$

with x_l and x_f as the positions of the leading and following car and L_{car} the average length of a vehicle.

Using $d(v)$ as the braking distance of a vehicle driving with a velocity v, collision free driving between two following cars f and l means

$$d(v_f) + v_f \Delta t \leq d(v_l) + g$$

$d(v)$ depends on the braking capability b. The human reaction time (implemented via the $v_f \Delta t$ term) is set to be equal to the update time step Δt, which is typically one second.

2.1.2 Update rule

A detailed derivation of the final update rule can be found in [1] and [2]. The result is as follows:

$$\begin{aligned} v_{\text{safe}}(t+\Delta t) &= v_l(t) + \tfrac{2b(g-v_l)}{2b+v_f+v_l} \\ v_{\text{des}}(t) &= \min\left(\mathrm{V_{max}}, \mathrm{v(t)} + \mathrm{a}\Delta\mathrm{t}, \mathrm{v_{safe}(t)}\right) \\ v(t+\Delta t) &= \max\left(0, \mathrm{v_{des}(t)} - \mathrm{a}\epsilon\eta\right) \\ x(t+\Delta t) &= x(t) + v(t+\Delta t)\Delta t \end{aligned}$$

Additionally we included a noise term $a\epsilon\eta$ in the update rule that prevents the system from becoming homogeneous. The noise term makes individuals *not accelerate as fast as they could*, which does not affect safety. This idea becomes clearer when writing a formal velocity update with noise as

$$v(t+\Delta t) = v(t) + a\Delta t - a\epsilon\eta$$

ϵ is to be chosen in $[0\ldots1]$ indicating how much noise we allow, while η generates rational random numbers in $[0\ldots1)$.

2.2 The Nagel-Schreckenberg cellular automaton

The Nagel-Schreckenberg model uses discrete space coordinates and discrete velocities. An illustrative way to implement this is to use an array to represent a lane (street) which vehicles drive on. Each cell has the average length of a car in a jam, typically 7.5 metres. In the NS cellular automaton, velocities and positions are integers, which is an argument to use a cellular automaton when simulating very big systems due to faster updating possibilities.

Velocities should be bound by an upper velocity $V_{\max}$. $V_{\max} = 5\frac{\text{cells}}{\text{update}}$ corresponds to $135km/h$ and hence seems to be a realistic approach. A car is represented by an integer between 0 and $V_{\max}$ in the array, corresponding to its velocity. In the code, one can use -1 is an empty cell.

Cars "jump" from cell to cell according to their current speed. To preserve safety, cars should not drive further than the gap in front of them allows it. Assuming a sudden stop of a leading car at cell n, the following car will drive over the open gap and should halt in cell $n-1$. Note that this is an indirect implementation of infinite braking capability.

If vehicles are not hindered by leading cars, they should try to obtain maximum speed, and accelerate their speed by 1 velocity unit per update.

To give an impression of these thoughts, we can illustrate two following situations where cars are heading to the right. The numbers indicate the speed that the velocity update would choose for that specific situation:

step t	...	3				2			5						...
step $t+1$	...				2			3						5	...

Combining the cellular automaton rules of motion, we are able to formulate the following update rules:

$$\begin{aligned} v^*(t+1) &= \min(\text{gap}, \text{v(t)}+1, \text{V}_{\max}) \\ v(t+1) &= \max(0, \text{v}^*(\text{t}+1) - \eta_{\text{r}}) \\ x(t+1) &= x(t) + v(t+1) \end{aligned}$$

x represents the position in the array. η_r functions as noise and is drawn as follows:

$$\eta_r \in \{0,1\}, \qquad \eta_r = \begin{cases} 1 & \Leftrightarrow \text{random number} \leq \text{r} \\ 0 & \Leftrightarrow \text{else} \end{cases}$$

where r has to be chosen to regulate the amount of noise. In this work $r = 0.5$ was chosen everywhere.

2.3 How local jams occur

This section should explain roughly, how jams can occur. The thoughts will be given for Krauss' model but can be adapted to the cellular automatonas well.

In traffic situations with very low densities cars can drive at $V_{\max}$ without being hindered by their leading car. This means, that they can easily stop their own car (including reaction time) if their predecessor does a full brake. Using the braking distance $d(v)$, we have a safety condition for two following cars driving at $V_{\max}$

$$d(\underbrace{V_{\max}}_{\text{follower}}) + \underbrace{V_{\max}}_{\text{follower}} \Delta t \leq d(\underbrace{V_{\max}}_{\text{leader}}) + g$$

When increasing the car density ρ, the average gap $\langle g \rangle$ between cars decreases and the equation above becomes false. Hence it is necessary for cars to drive at a lower speed $v < V_{\max}$ to preserve safety. In terms of the SK model algorithm, v_{safe} decreases as g decreases.

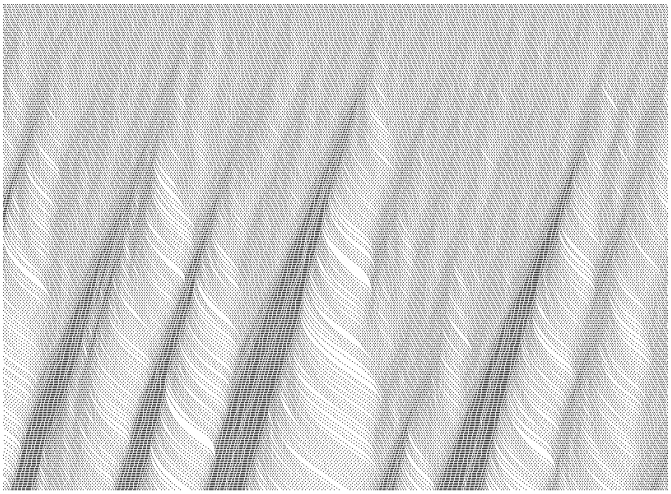


Figure 2.1: A space-time plot of Krauss model. Initialized to equidistant positions and uniform speed, the car following dynamics produce jam waves due to noise. $a = 0.2$, $b = 0.6$. The space axis is pointing to right right, the time axis is pointing downwards. The trajectory of a car can be found by connecting its positions (dots) between time steps. Space-time plots will be explained in section 3.1.

This change from unhindered to hindered traffic flow is sudden at a certain critical density.

For ρ above this critical density vehicles interact and adapt their desired speed to their leader's speed and the gap between them. This would thermalize into a state with all cars at equidistant positions $g_i = g_j$ and identical velocities $v_i = v_j$, for all leader-follower pairs (i, j), *if* we had not included the noise term $a\epsilon\eta$. The latter is responsible for density fluctuations and eventual breakdown of laminar flow.

In figure 2.1 we see a space-time plot of a configuration above this critical density. The system is initialized to equidistant positions and the velocities of the laminar noiseless solution. Velocity fluctuations add up in the upstream direction and cause the system to break down locally, meaning at least one car comes to a full stop.

We will deal with traffic breakdown and jam recovery in chapter 5.

2.4 Conventions

2.4.1 Constants and units

As all measurements will be averaged over *cars*, it is reasonable to identify the system size with the number of vehicles N_{cars}. For initialization, the code needs the constants ρ and N_{cars} and the parameters a, b, $V_{\max}$ and ϵ. With the average length of a vehicle L_{car}, the system length is calculated trivially by

$$L_{\text{system}} = \frac{N L_{\text{car}}}{\rho}$$

respecting periodic boundary conditions.

We drop dimensions of all kinds and agree to use

$$\begin{array}{llll} 1 \text{ spaceunit} & = 1 \text{ carlength} & = 7.5 \textit{ metres} & = 1 \\ 1 \text{ timeunit} & = 1 \text{ update} & = 1 \textit{ second} & = 1 \end{array}$$

Converting these relations back into SI units results in realistic velocity ranges: $V_{\max} \leq 5 \rightarrow V'_{\max} \leq 135\text{km/h}$.

a, b and $V_{\max}$ are constants, and we use typically $a = 0.2$, $b = 0.6$, $\epsilon = 1$ and $V_{\max} = 3$. *These parameters are being used in the following work for typical Krauss type I*[1] *systems unless specified otherwise.*

2.4.2 Terminology of "jammed"

In the following we will use the word jammed in various contexts. We will talk about jammed traffic, jammed start etc. The use of these expressions was adapted a bit, which we should declare here.

When talking about a *jammed system, jammed flow* or *jammed traffic*, we mean that there is at least one car in the system standing still at every time step and that jam waves exist. To call this state "coexistence" would describe more what we actually see, because for most jam situations not all cars in a periodic system are positioned within the jam, some drive freely between two jam waves - see figure 2.1. We still call such a situation *jammed*, because we do find halting cars at every time step.

We will also use the expression of a jammed start of a simulation. By this we mean an initialization of the system that was artificially forced into a jam before running.

[1] See chapter 4.

Chapter 3

Analysis of traffic flow

We will now introduce a few tools and graphs that will help us to qualify our traffic flow models. In a first section, we will look at normal space time plots, as you saw them already earlier in this paper.

Then we will describe and understand how to read fundamental diagrams and see how our two models can be distinguished and point out a few specialities about Krauss' model.

3.1 Space-time plots

Space-time plots are a very easy tool to get a first impression of vehicular movement. Technically it is also very easy to plot a lane into a UNIX shell using one line for every time step. therefore we draw the space axis pointing to the right and the time axis pointing downwards. A vehicle is represented by a dot. A trajectory of a car can then be found by connecting all dots representing that specific car from time step to time step. A driving car draws its trajectory into a "south east" direction in the space-time plot. A car standing still is drawn by a vertical line.

Traffic jams become easily visible this way: In a jam, cars stand still with relatively little space between them. A car jam is a dark wave travelling through the system. In a standard Type I Krauss system[1] and a cellular automaton, these waves travel upstream (in "south west" direction in the space-time plot).

Space-time plots are even more informative than video sequences, because they show an entire development through time *at once*. This allows us to compare different kinds of jam formation. We will see in chapter 4

[1] The different types of Krauss models will be explained later in chapter 4.

that the *kind* of model we choose and our parameter settings influence the development, shape and life time of jams.

Figure 3.1 shows two space-time plots: A Krauss' model on top and a cellular automat development on the bottom. Some comments:

Upper space-time plot, Krauss type I: We see a few rather big jam waves propagating through the system. This is what is ment, when we speak of a *phase separation*. The jam waves become more stable as we increase our system density. In between two jams we have a so called *outflow from jams*, where we find a laminar driving situation.

Lower plot, standard cellular automaton: The cellular automaton does not develop a clear separation of phases. Minijams come and go like fluctuations and have a comparably short life time. Between two jams we again find an outflow situation, but this outflow is not stable for this model. We can find several little jam waves starting within the chosen time window that were caused in the outflow of a previous jam.

In chapter 4 space-time plots will serve us to distinguish different types of traffic flow.

3.2 The fundamental diagram

3.2.1 Densities

We have to specify a bit what is ment when we talk about the *density* of a system. Clearly we do not always deal with absolutely homogeneous traffic situations, which makes it necessary to distinguish bewteen local and global densities.

In this work and if not specified otherwise, a simple ρ without indices declares the *global density*:

$$\rho = \langle\rho\rangle_L = \frac{N_{\text{cars}}}{L_{\text{system}}} = \frac{\text{amount of cars}}{\text{system length}}$$

Other densities of interest are for instance the density in a jam, ρ_{jam}, or the denity in the *outflow* of a jam, ρ_{out}. We will refer to these always using the corresponding index.

3.2.2 Fundamental diagrams

We found out earlier in 2.3 that above a certain density vehicles cannot drive at $V_{\max}$ anymore and have to slow down to preserve safety for collision free

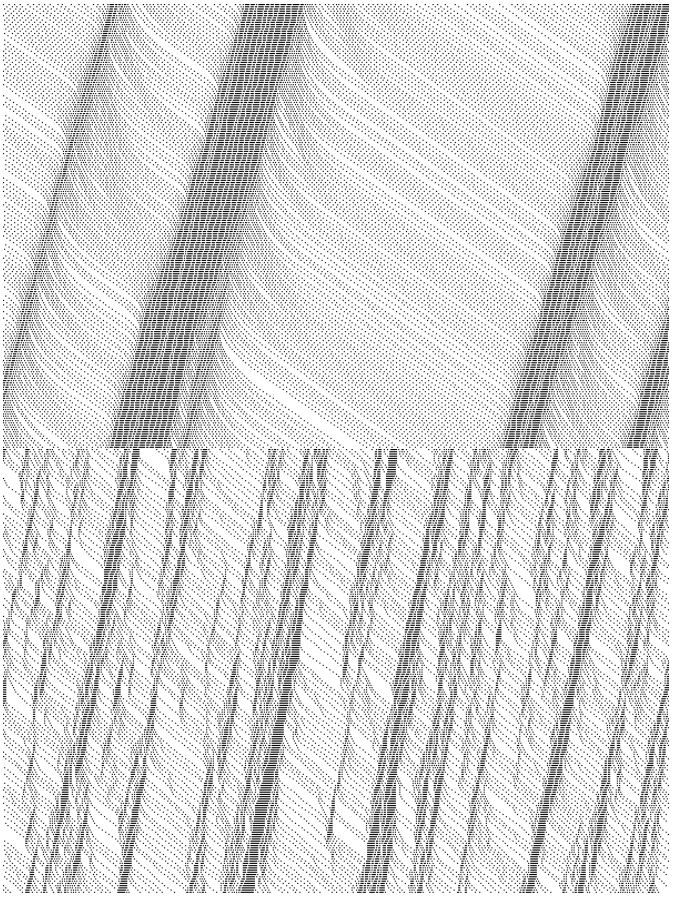


Figure 3.1: Space-time plots at $\rho = 0.3$. On top, a type I Krauss' model using $a = 0.2$, $b = 0.6$ and $V_{\max} = 3$ and on the bottom the cellular automaton with $V_{\max} = 3$.

driving. The average velocity $\langle v \rangle$ thus depends on the global density ρ. We expect that

$$\langle v \rangle \to \begin{cases} V_{\max} - \langle \text{noise} \rangle & \Leftrightarrow (\rho \to 0) \\ 0 & \Leftrightarrow (\rho \to 1) \end{cases}$$

Measurement of $\langle v \rangle$ give us a first impression of the efficiency of traffic flow. However, it is more applicable to know the vehicle flow, *how many cars per second*[2] pass by. In order to get there, we multiply $\langle v \rangle$ with the global density ρ to obtain the desired *flow*

$$q(\rho) = \langle v(\rho) \rangle \rho$$

Thus, $\langle v \rangle$ is the slope of $q(\rho)$ for low densities.

Assuming the so far deducted structure of $\langle v(\rho) \rangle$ we find for $q(\rho)$ that

$$q(0) = q(1) = 0 \quad \text{and} \quad q(\rho) \geq 0$$

The second statement holds for vehicles driving strictly forward. A plot of $q(\rho)$ is what we mean when talking about a *fundamental diagram.*

We will now talk a bit about the measurement of fundamental diagrams and then investigate specific cases for different models.

3.2.3 Different types of fundamental diagrams

Our systems have to be initialized in one way or another. The initialization has a direct influence on the flow of vehicles and their fundamental diagram.

Respecting thermalization, it is sufficient for some models and situations to measure the flow $q(\rho)$ for a certain period of time and calculate a *time*-averaged mean value.

This work uses models and initialization techniques that allow a system to *change state* after a certain waiting time which is unknown in the beginning[3]. We therefore need ensemble-averaged measurements at fixed moments in time. We say we take *snapshots* of the fundamental diagram.

We will initialize a system by placing vehicles at equidistant positions and with the expected mean velocity $\langle v(\rho) \rangle$. This requires sufficient thermalization before doing measurements.

[2] Single lane traffic

[3] To determine this waiting time (called recovery and breakdown time later) is a part of the aim of this work

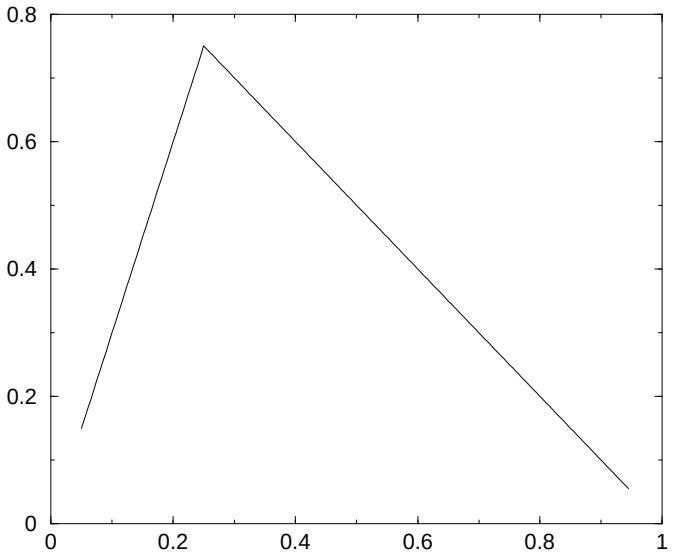


Figure 3.2: A fundamental diagram, Krauss' model, without noise, $a = 0.2$, $b = 0.6$, $\epsilon = 0$.

As we said, we expect our system to change state over time: A laminar system can break down into a jammed state and an artificial jam can recover into pure laminar traffic.

We have to be able to follow traffic flow through time and to obtain data for

$$q(\rho) = q(\rho, t)$$

This can be done by taking snapshots at certain moments in time. Our code should calculate these snapshots of the flow $q(\rho, i)$ at a given time step i.

To retrieve a reliable ensemble-averaged value over the corresponding part of the phase space, we have to start several parallel simulations with different random seeds.

3.2.4 $\epsilon = 0$ fundamental diagram for Krauss

Choosing $\epsilon = 0$ takes all disturbance by noise away. A reduction of velocity is only caused by shorter gaps between cars as density increases. The corresponding fundamental diagram is displayed in figure 3.2.

The slope on the steadily increasing part for low ρ equals $V_{\max}$. This fundamental diagram describes the so called "homogeneous" solution. It can be used for velocity initialization.

The point of maximum flow in figure 3.2 defines the outflow from jams $q_{\text{out}} = q(\rho_{\text{out}})$.

3.2.5 $\epsilon = 1$ fundamental diagram for Krauss

Since the shape of the fundamental diagram depends on the initialization, we will show both results for a laminar and a jammed start and explain them separatly.

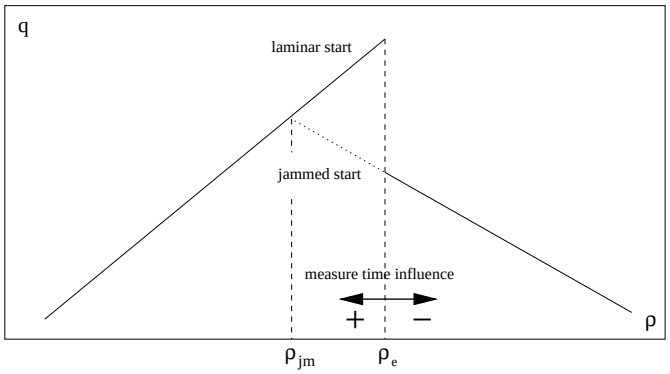
Starting a simulation with *jammed initial conditions* is done by halting an arbitrary car and waiting until all vehicles in the (periodic) system queue behind it. The jam is formed, when we measure

$$v_i = 0 \qquad \forall \text{ vehicles i}$$

for the first time.

Theoretical examinations

To understand what information we can retrieve from the fundamental diagram we will first have a look at a theoretical sketch. The following diagram contains both initial states laminar and jammed.



A theoretical sketch of a fundamental diagram belonging to a Krauss type I model.

For low densities (left hand side), both systems result in the same flow. The laminar system stays laminar, and the jammed started system recovers into laminar flow, because there are few enough cars to regain free flow.

As we increase our vehicle density ρ (going rightwards in the sketch) we do not get the same result for the flow for initial conditions laminar and jammed above a certain density ρ_{jm}. A laminar started system keeps its free flow and increases it with the typical slope of $V_{\max}$. The flow increases up to a density ρ_e ("e" like "edge", see figure above), where the laminar flow collapses and *jammed* and *laminar* systems show the same vehicle flow again. This is the typical *edge* we find in Krauss' fundamental diagram. We say that Krauss' fundamental diagram has a *reversed-λ-shape*. Some speak of hysteresis [2].

At the density ρ_{jm}, where the lines of laminar and jammed started system flows split, a jammed started (infinite) system is in general not capable of recovering anymore. The density where these two lines split (which is the maximum flow that an infinite jammed started system can reach after a finite run time) we call ρ_{jm}, because it is the "*m*aximum flow for *j*ammed starts".

The branch of the jammed start follows the lower dashed line in the fundamental diagram, until it is joined by the upper line at ρ_e where both systems have the same flow again.

It is possible for *finite* systems to recover onto the laminar branch after a so called recovery time T_{rec}, which increases rapidly with ρ. This phenomenon will be investigated later in this work.

Let us from now on talk about the *laminar branch* (top) and the *jammed branch* (bottom) in the fundamental diagram.

What happens at the edge at ρ_e in the figure?

The flow $q(\rho)$ does not join the jammed branch smoothly at ρ_e. A system initialized to a laminar state can collapse and form jam waves. By doing so, the flow breaks down and a system state $(\rho, q_{\text{laminar}})$ breaks down to $(\rho, q_{\text{jammed}})$ with

$$q_{\text{laminar}} > q_{\text{jammed}}$$

When a system breaks down to a jammed flow, the point (ρ, q) moves from the laminar branch down to the jammed branch. A system in between is not often found, and if so, it is usually at the very edge at ρ_e.

The position of this edge depends directly on the time when we take our snapshot. The longer our fundamental diagram snapshot measure time is, the more likely it is for a system to break down. And, denser systems are more likely to collapse. These two facts explain why the edge position depends on the snapshot time. Snapshots of later measurement times feature a "shorter" edge with the corresponding ρ_e further to the left, because more

systems collapsed and the corresponding data points at the edge are missing. They moved down to the jammed branch and

$$\rho_e = \rho_e(t)$$

The double arrow in the sketch signifies this with the $+/-$ symbols. It indicates the position shifting of ρ_e depending on larger $(+)$ or smaller $(-)$ snapshot measurement times.

In the following we will examine these observations on real measurements.

Measurements

Figure 3.3 shows what we just dicussed on an example of 5000 vehicles. The snapshot was taken after 30'000 updates, 20 such simulations were taken into account at each density. The ensemble average can then be calculated from 20 independent simulations with different random seeds. The jammed and the laminar branch were both drawn into the same plot to illustrate their relative positions.

Compared to the theoretical shape of the fundamental diagram we can notice a slight shift between the data of the laminar start and the jammed start for densities above ρ_e, in figure 3.3 above $\rho_e = 0.21$. The jammed started flow is a bit lower than the the laminar one.

This difference also changes in time. As we take our snapshot later in time, the jammed branch shifts upwards slowly until it overlaps with the laminar branch while the position of the laminar branch seems to be constant in time.

3.2.6 CA fundamental diagram

Finally we display measurements of the fundamental diagram of the cellular automaton in figure 3.4, 5000 cars after 30'000 updates. We displayed only the laminar start data, because it does not differ from the jammed started one.

Notice that the point of maximum flow is shifted towards smaller ρ compared to the point of maximum flow in the fundamental diagram of Krauss. Relatively lower car densities can cause a jam. The two plots do not use the same ρ scale though.

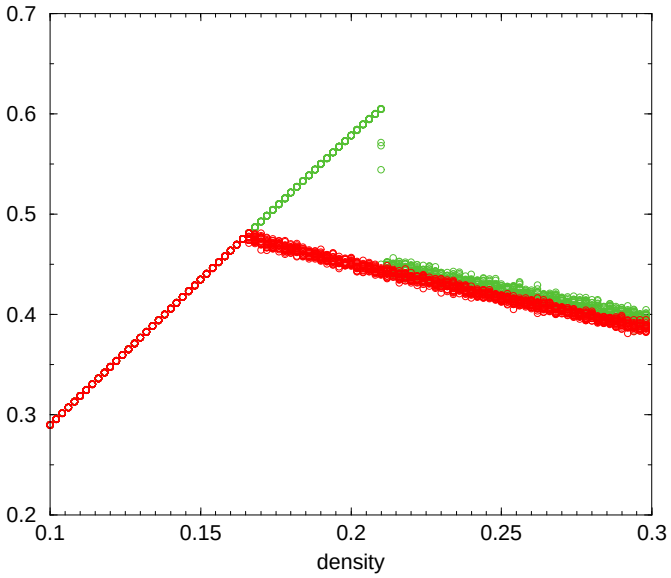


Figure 3.3: A fundamental diagram for Krauss' model, $\epsilon = 1$, $a = 0.2$, $b = 0.6$, $V_{\max} = 3$, 5000 cars, snapshot after 30000 steps.

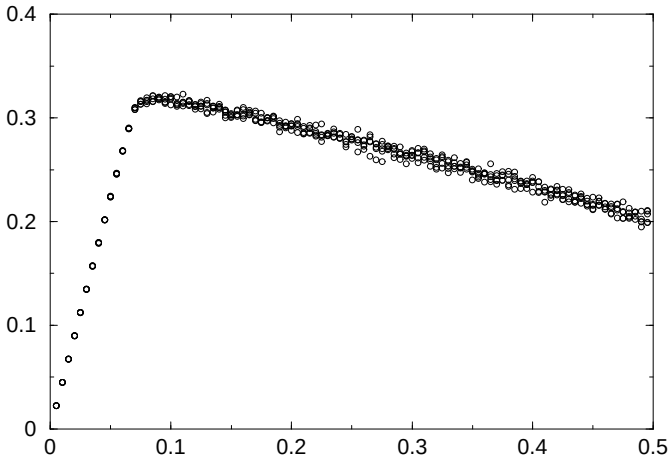


Figure 3.4: A fundamental diagram for a standard cellular automaton, $V_{\max} = 5$, 5000 cars, snapshot after 30000 steps.

Chapter 4

Features of Krauss' model

4.1 Free parameters

Krauss' model allows us to chose four parameters arbitrarily, which are the maximum velocity $V_{\max}$, the acceleration a, the braking capability b and the noise ϵ.

The maximum velocity $V_{\max}$ has an obvious global effect on the traffic flow $q = v\rho$. It is the slope in the fundamental diagram when every vehicle is driving at its desired maximum velocity $V_{\max}$. We will treat $V_{\max}$ as a global constant and not as a parameter to adjust the systems behavior.

Acceleration and deceleration

The acceleration and braking capability a and b respectively determine driving behavior and the interaction between cars. In laminar situations we therefore do not expect too many changes in behavior when adjusting a and b. In partly jammed regions a and b influence the behavior of cars when getting into a jam or leaving a jam, for example:

- Cars with low deceleration capability, $b \ll V_{\max}$, have to take an oncoming obstacle into account much earlier, because it takes them longer to slow down. This affects the upstream situation in front of a jam, cars "attach" smoothly to the jam. When slowing down, cars eventually have a velocity similar to the noise, $v \sim a\epsilon\eta$. The noise step in the update rule can push the velocity close to zero and cars halt even with a gap ≥ 0 ahead.

$$v(t) \sim a\epsilon\langle\eta\rangle \Rightarrow v(t + \Delta t) = v_{\text{des}}(t) - a\epsilon\langle\eta\rangle \sim 0$$

Hence, the average gap $\langle g \rangle$ in a continuous system is > 0. We obtain jams with densities $\rho_{\text{jam}} < 1$.

- Cars with an infitinte braking capability, $b = \infty$, do not need to adjust their velocity when approaching a jam. This is realized in the cellular automaton, where only the condition $v(t+1) \leq$ gap ahead ensures safe driving. The average gap between cars in a jam in a cellular automaton system is therefore zero. Jams are stiff and have no inner dynamics and $\rho_{\text{jam}} = 1$.

  More about making artificial jams in 6.2.1.

- A large acceleration is bound to large noise $a\epsilon\eta$. $a \sim V_{\max}$ results in an unpredictable chaotic movement, which might be a reason for the missing phase separation in this type of configuration.

Just as explained in these examples, the regulation of a and b allow us to distinguish several types of models within the Krauss model. Krauss distinguished three types of motion, enumerated I, II and III. The criteria to distiguish them are of a global kind, for example:

> *Does traffic spontaneously form structures? Does it seperate into two phases,* laminar *and* jammed*?*
>
> *How is the outflow from jams? Do we find daughter jams in the outflow from jams or can we call the downstream flow "stable"?*

Note that *all* three types of Krauss' model that will be explained in the following preserve the safety condition

$$d(v_f) + v_f \Delta t \leq d(v_l) + g.$$

Different settings just apply to *how* this condition is realized.

4.2 Types I to III according to Krauss

Type I: Low acceleration, low deceleration

This is probably the most "typical" configuration for Krauss' model.

The upper space-time plot in figure 3.1 shows an example of low acceleration and deceleration, we can see structure formation from a homogeneous initialization into two phases. Traffic can separate into *laminar* and *jammed* regions.

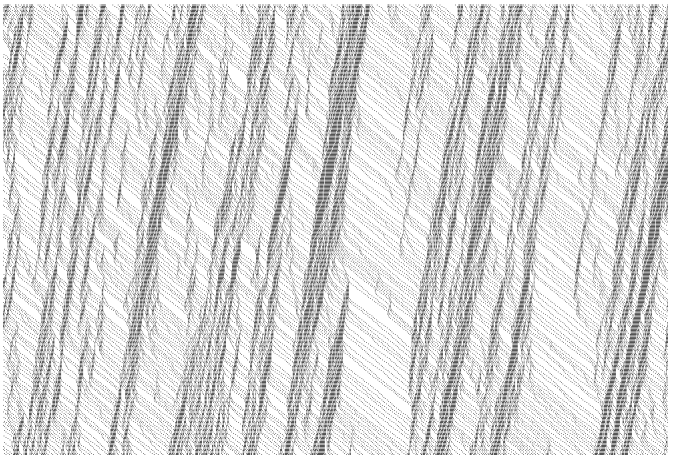


Figure 4.1: A space-time plot as an example of low acceleration high deceleration, type II. $a = 1$, $b = \infty$.

Type II: Low acceleration, high deceleration

This configuration is realized for instance in the standard cellular automaton as it was derivated earlier. $b \to \infty$ allows arbitrary deceleration. This type is difficult to distinguish from a type I.

However, the outflow from jams gives some arguments to determine a border between the types I and II, because type II does not show clear phase separation. The outflow from jams of a type II model is unstable. These issues are investigated in chapters 6 and 7. See figure 4.1 to get an impression of such a space-time plot.

Type III: High acceleration

A high acceleration setting for Krauss' model does not show phase separation at all. Flow is macroscopically homogeneous, and because jams do not occur, this type is not applicable to real traffic situations. An example is shown in figure 4.2.

4.3 The (a,b) parameter plane

For better orientation we can draw the parameter plane with axes a and b and draw the potential positions of the borders between the three types

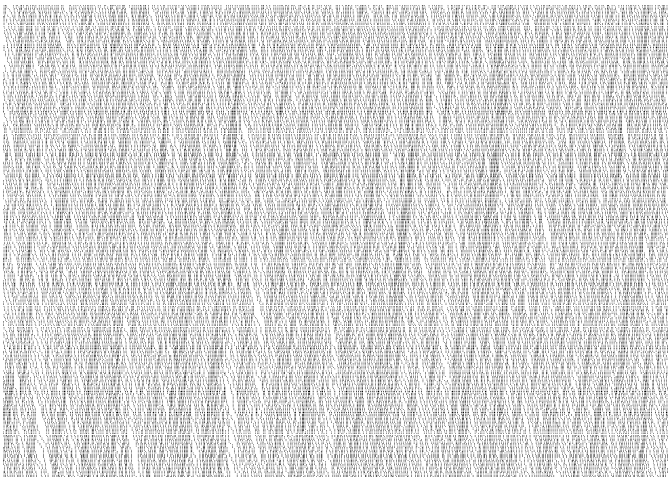


Figure 4.2: A space-time plot as an example of high acceleration (low deceleration), type III. $a = V_{\max} = 3$, $b = 0.6$.

described. Krauss did this separation as shown in figure 4.3, measuring the density difference between laminar and jammed flow and the proportion of stopped vehicles in free flow.

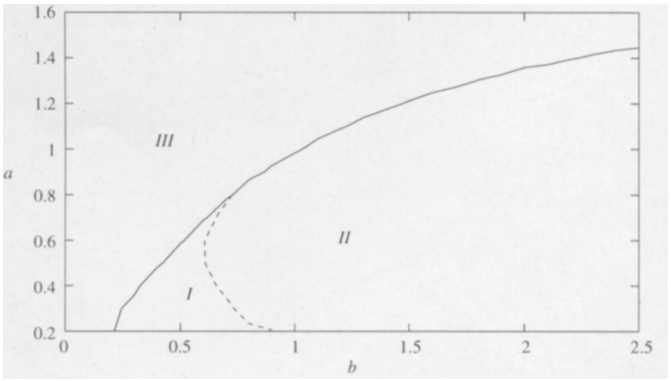


Figure 4.3: The parameter plane (a, b), taken from Krauss [1].

Chapter 5

Breakdown and recovery times

From the explanations of the preceeding sections it should be clear now, that traffic flow can be found in several states. The extremes are *laminar* traffic on the side of low vehicle densities, where a vehicle can drive freely without too much interaction of its leading car, and *jammed* traffic, where jam waves travel through the system and at least one car is standing still per time step. We want to analyze the kind of transition we obtain when going from very low densities to very high ones and back.

Measuring the *transition* or *waiting times* that pass until the system is likely to switch from an unstable initial state to another state can help us to describe the laminar↔jammed transition and whether a meta-stable state exists.

Such transition times are the *recovery* and the *breakdown* times.

5.1 The recovery time

In the range of low densities, $\rho < \rho_{\text{out}}$ traffic is laminar and absolutely stable. Even stopping an arbitrary car until the entire system is jammed behind it resolves back into laminar traffic flow. Such an artificial jam is called *megajam* and we call the stopped vehicle the *jam maker*.

Once we release the jam maker, the system will try to regain laminar flow. This becomes impossible when dealing with high vehicle densities, because in such a situation vehicles have too little freedom to rearrange themselves to obtain laminar flow globally. We can define a ρ-range, where the system *does* recover after a finite time T. We define T_{rec} to be the time

it takes to recover from an artificial megajam back into laminar flow.

For all measurements of this work, we will agree to measure the recovery time starting with the car positioned furthest upstream in the original jam moving again faster than a critical speed $v > V_{\max}/2$. This critical speed is arbitrary and we set it so high because finite gaps $\langle g \rangle > 0$ within the jam allow the most upstream positioned vehicle to move very slowly *within* the jam, which should be distinguished from that specific car reaching the downstream end of the jam and accelerating, see section 6.2.1 and figure 5.2. This way $T_{\text{rec}} = 0$ for very low densities because the system is resolved just as the last car moves again.

Alternatively, we could measure T_{rec} starting with all cars standing still in the artifical megajam. This adds a linear term scaling with N_{cars} to the recovery time (a thermalization time $T_j^{(\text{therm})}$), because the first $N-1$ cars need to first leave the megajam:

$$T_{\text{rec}} = T_j^{(\text{therm})} + T_{\text{rec}}^{(\text{eff})}$$

This approach we will use for an analytical discussion of T_{rec}, see figure 5.2 for further illustration.

For $\rho > \rho_{\text{out}}$ our T_{rec} increases exponentially and we have to face computational limits. To get reliable averaged measurements of T_{rec} above 10^7 iterations for 5000 cars takes several days on a Pentium III 700MHz.

5.2 The breakdown time

Similar to the definition of the recovery time, we define the *breakdown time* T_{bdown} as the time that passes between the laminar system initialization and the moment when we find the first car with speed $v = 0$. Because we do not interfere with the system (like we did in order to produce a megajam for T_{rec}) we measure T_{bdown} starting right after system initialization. As a consequence, we need to split T_{bdown} into an *effective* recovery time, and a thermalization term:

$$T_{\text{bdown}} = T_l^{(\text{therm})} + T_{\text{bdown}}^{(\text{eff})}$$

For low densities, T_{bdown} is extremely long because the laminar traffic is too stable to cause a jam ever. For $\rho \to 0$ we expect $T_{\text{bdown}} \to \infty$.

5.3 Theoretical expectations of the breakdown and recovery times

In figure 5.1 we have a very rough theoretical plot of the transition times connected to the fundamental diagram. It should help us to get a detailed understanding of the connection between transition time, fundamental diagram snapshot time and the shape of the fundamental diagram. We want to point out a few remarkable areas in the following. The enumeration corresponds to the numbers in figure 5.1.

(1) For $\rho \to 0$ we are dealing with stable laminar traffic. In the range of $0 \leq \rho \ll \rho_{jm}$ all cars have enough space to arrange themselves with gaps bigger than the minimal gap necessary to preserve safety. As a consequence, the jam-maker will not be able to reach the tail of the megajam before it resolves itself. Since we measure T_{rec} after the last car in the jam moves, we usually have a $T_{\text{rec}} \to 0$ with $\rho \to 0$.

(2) The Point $(\rho_{jm}, q(\rho_{jm}))$ qualifies the maximum flow of an infinite jammed started system. However, a *finite* system can recover from jammed into laminar state within a finite time range. In the figure this is illustrated by the dashed line indicating the jammed branch of the fundamental diagram using a relatively large snapshot time. This (reversed-λ-shape) edge development was indeed observed in measurements of small systems, see section 3.2. For extremely large snapshot times a system configuration (ρ, q) can move from the jammed to the laminar branch in the fundamental diagram and we obtain the edge (reversed-λ-shape) even for the jammed branch.

$T_{\text{rec}}(\rho_{jm})$ qualifies by the way a critical measurement time: We will most likely not find a reversed-λ-shaped fundamental diagram for a jammed start when simulating for a time shorter than $T_{\text{rec}}(\rho_{jm})$, because that is the minimal time required for a system to recover. We get a cut-off at $T_{\text{rec}}(\rho_{jm})$ below which a simulation is futile.

The longer we choose our snapshot time, the more likely it is for a finite system to recover onto the laminar branch. For large times, even denser systems can recover. As a consequence, the edge of the reversed-λ-shape grows in the direction indicated by the arrow in the lower left diagram of the figure when the snapshot time is increased.

(3) When increasing the snapshot time $T \to \infty$ the edge of the hook in the jammed branch of a *finite* system will move to the right (because

denser systems had the time to recover) until one of the following cases becomes true:

(a) T_{rec} has no divergence before it intersects with T_{bdown} coming from the right. Let us say they intersect at ρ_c. In the figure ρ_c would be positioned approximatly near ρ^*. See comments under (5). The system $[\rho_c, q(\rho_c)]$ is unstable in both states and is likely to continuously switch between laminar and jammed state in a time range $T_{\text{rec}}(\rho_c) = T_{\text{bdown}}(\rho_c) < \infty$.

$T_{\text{bdown}}(\rho)$ is limited by $T_{\text{rec}}(\rho)$ and we have an identical transition probability from one phase to the other. At ρ_c both states laminar and jammed are likely to be found.

(b) T_{rec} diverges. If T_{bdown} diverges also, it raises the question if both times have the same singularity or if they lie apart.

In between those two singularities or *on* that single singularity a system has an infinite recover and an infinite breakdown time. This means that the development in time depends only on the initialization. If the system is infinite, transition is not possible and the laminar and jammed states are separated by inifite boundaries.

(4) $T_{\text{bdown}} \to 0$ for $\rho \to 1$, because cars cannot move at all at $\rho = 1$ and a jam is found as the initial condition. The reason for this is that for $\rho \to 1$ we have $v_{\text{safe}} \sim \text{noise} = a\epsilon\eta$ which makes it very likely to find cars with $v = 0$ after the first update, even if they have a finite gap in front of them. In this case $T^{(\text{eff})}_{\text{bdown}}$ can be considered to be zero since cars are initialized using $v_i = \langle v(\rho) \rangle$, which thermalizes into *jammed* flow after only a few iterations.

(5) T_{bdown} grows exponentially for decreasing ρ. Either it diverges near a certain ρ^* or it intersects with T_{rec} at ρ_c, which we mentioned already.

As we approach the edge at ρ_e from above coming from $\rho > \rho_e$, T_{bdown} increases rapidly. The longer we choose our snapshot time, the more likely it is for a system to break down. We see this correlation in figure 5.1. The longer T_{bdown} is, the further left we move on the ρ-axis and we "eat up" the edge in the fundamental diagram (see the leftarrow in the lower right diagram in the figure).

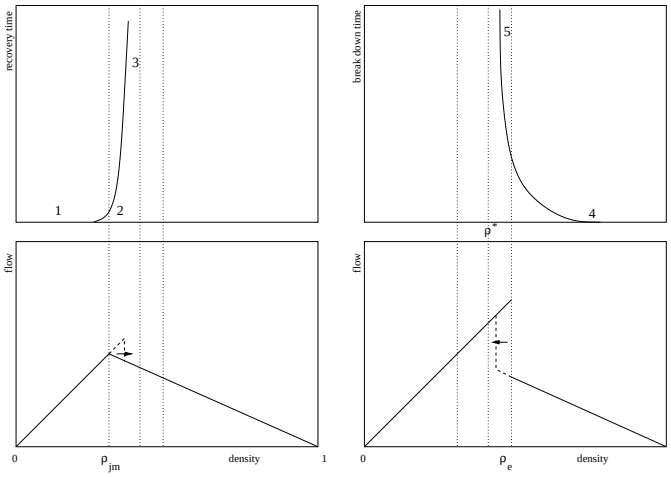


Figure 5.1: A theoretical comparison of recovery and breakdown time and the structure of a Krauss type I fundamental diagram using the same ρ-scale. The arrows indicate the edge displacement when increasing the fundamental diagram measurement snapshot time.

5.4 Measurements of T_{rec} and T_{bdown}

We will now look at the measurements and plots that were the starting point for the assumptions of figure 5.1.

Remember the comments we made on the *thermalization* times in T_{rec} and T_{bdown}. Let us declare once more:

- Measurements of the recovery time T_{rec} do *not* contain any thermalization time. We start measuring the effective recovery time as soon as the car positioned furthest upstream in the original megajam reached a speed of $\frac{1}{2}V_{\max}$. See figure 5.2.

- Measurements of the breakdown time T_{bdown} were measured *including* the thermalization time. For the simple reason that it is difficult to determine the thermalization time which also depends on ρ and N_{cars}. We obtain a hint of the possible thermalization time length *after* looking at T_{bdown}-plots. See section 5.5. To see that it is difficult to retrieve a guess about $T_l^{(\text{therm})}$, have a look at figure 2.1 where the system passes from a homogeneous initial state to jammed state steplessly.

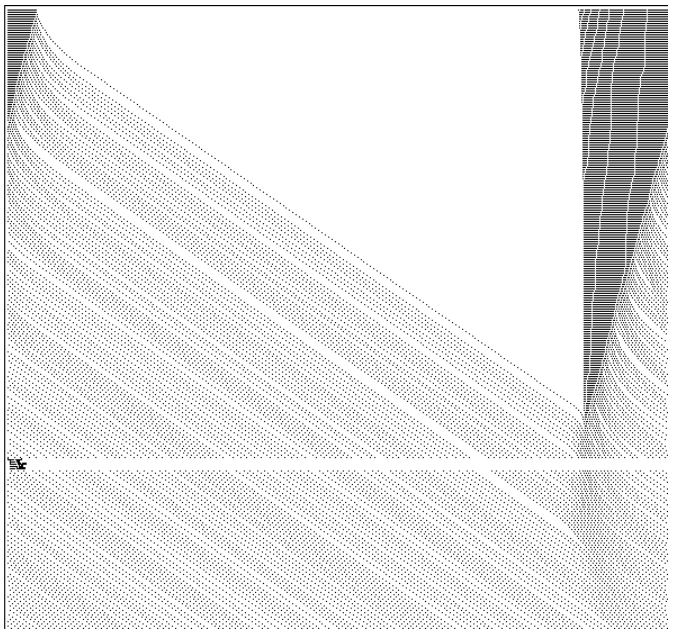
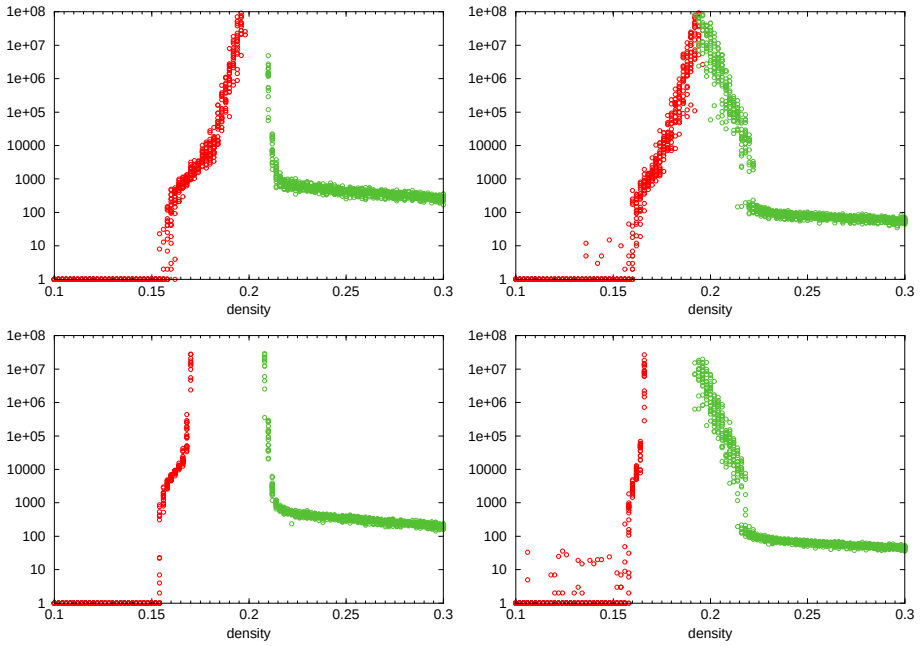


Figure 5.2: A space-time plot as illustration of the start when measuring T_{rec}. We start with a megajam. The jam goes over the boundary on the right and comes back in on the left side. The interrupting horizontal white space indicates when the last car in the mega jam (left most in the figure) reached $\frac{1}{2}V_{\max}$. *Until* that point, $T_j^{(\mathrm{therm})}$ passes. *Afterwards*, we start measuring $T_{\mathrm{rec}}^{(\mathrm{eff})}$ which is zero in the figure because cars move already at a speed $> \frac{1}{2}V_{\max}$.



$N = 625,\ b = 0.6$	$N = 625,\ b = \infty$
$N = 5000,\ b = 0.6$	$N = 5000,\ b = \infty$

Figure 5.3: Recovery and breakdown time measurements for two system sizes and model types. For both configurations (type I, $b = 0.6$ on the left, type II, $b = \infty$ on the right), the absolute positions of T_{rec} and T_{bdown} shift to the left towards lower densities when increasing the system size N_{cars} (upper row vs. lower row). On the other hand, the absolute positions of these waiting times shift towards lower densities for both system sizes when increasing $b = 0.6 \to \infty$, because jam occurrence is favored for increasing b. This corresponds to a transition from type I to type II.

5.4.1 T_{rec} measurements

The measurement samples for T_{rec} are the ones coming from low densities from the left in all plots in figure 5.3.

Comments on the measurements

Measurements started at $\rho = 0.1$. Whenever a system was found to have resolved itself, the simulation went on to find resolving jams in a little bit denser system.

The criterion to call a jammed-start system "resolved" is, that the slowest car reached a velocity

$$v > 0$$

This does not mean that the jam wave disappeared totally when looking at the corresponding space-time plot. Jam waves or faint traces of it in a system that was found to be resolved may still exist, but cars do not halt anymore. The above criterion is arbitrary. Choosing stricter criteria such as for instance $v > \frac{1}{2}V_{\text{max}}$ results in a steeper slope of $T_{\text{rec}}(\rho)$ for increasing ρ, because it takes a system longer to recover to "fast" laminar flow than to a state where no car is found to stand still anymore.

Discussion of the results

T_{rec} *at low densities.* Because we excluded the thermalization time in the measurement of T_{rec} we find

$$T_{\text{rec}}(\rho) \sim 0$$

for densities $\rho \to 0$ in the range of absolutely stable laminar flow, as we expected before.

For the right column in figure 5.3 with $b = \infty$ we find a few measurements with $T_{\text{rec}} \gg 0$ for relatively low densities. Choosing $b = \infty$ we run a Krauss model of type II which is supposed to be structureless. Also, we saw the fluctuation-like behavior in the space-time plots, shown in figure 4.1. These fluctuations make us find several cars with velocities zero, which the simulation interprets as a "not resolved" jams. Hence the longer time we need to wait before finding a fully resolved system. In accordance with figure 5.3, these disturbances are more likely to be found for larger systems.

T_{rec} *for increasing densities.* T_{rec} grows very steeply as we increase the global density. All four plots of T_{rec} display a certain "knee" at the height

of about 10'000 steps. The origin of this knee is not fully understood. It may indicate a higher time scale that has to be respected. The position of the knee depends on the system size N_{cars} and the type of model.

T_{rec} increases as we increase the system size. This is intuitive, because bigger systems act more inertly and have more difficulties in recovering from artificial jams.

A type II model using $b \to \infty$ allows cars to attach quicker to the upstream end of the jam, because they do not need to slow down in advance. As a consequence, cars leaving the downstream end of a jam reach the upstream end (over the periodic boundary) quicker and thus delay jam recovery. Consequently, T_{rec} is larger (or shifts towards lower densities) for larger b.

5.4.2 T_{bdown} measurements

The measurement samples for T_{bdown} are the ones coming from high densities from the right in figure 5.3.

Comments on the measurements

Like for the measurements of the fundamental diagram, for this measurement, systems are initialized to a laminar state at equidistant position and speed of the homogeneous solution.

A system will be called broken down as soon as we find any car with a speed

$$v = 0$$

The time will be measured starting right after the initialization and as we mentioned before, T_{bdown}-measurements include the thermalization time.

Discussion of the results

T_{bdown} for high densities. Figure 5.3 displays T_{bdown} on a linear-log scale. For high densities ($\rho \to 0.3$), T_{bdown} follows a straight line in the diagram, which corresponds to an exponential curve on a linear-linear scale.

Results for T_{bdown} show a clear border, in the specific case of figure 5.3, at $\rho = 0.215$. Below that border, we find a steep shape increasing for decreasing ρ. Above it, we have a "flatter" (linear-log plot) shape and relatively less variation. This should give us reason to believe that two different components cause the shape of the measured T_{bdown}. We believe

that theses two components are the thermalization time (mentioned before) and the effective breakdown time

$$T_{\text{bdown}} = T_l^{(\text{therm})} + T_{\text{bdown}}^{(\text{eff})}$$

For densities $\rho \sim 1$ the system breaks down immediatly and we find cars with $v = 0$ after a single update. We expect the thermalization time $T_l^{(\text{therm})}$ to dominate the effective breakdown time $T_{\text{bdown}}^{(\text{eff})}$ for $\rho \to 1$.

Using the fundamental diagram for various snapshot moments we see that the system can hardly be found in a laminar state for $\rho > \rho_e(0)$.[1] As we approach $\rho \to 1$, the probability of the existance of a laminar system decreases and we expect $T_{\text{bdown}} \sim 0$ for densities ρ below 1.

T_{bdown} *for decreasing densities.* Below the border we just mentioned, T_{bdown} measurement results increase steeply and we expect the effective recovery time to dominate the thermalization time.

Linear-log plots and linear-loglog plots did not make these results more graspable. As we increase the system size, the effective break down time sinks. This is expected because bigger systems have more possibilities in space to break down.

5.5 Analytic approaches to T_{rec} and T_{bdown}

In this section we want to try to find the basic analytical structures of the waiting times T_{rec} and T_{bdown}. We will include thermalization times in both the recovery and the waiting time.

5.5.1 T_{rec}

We divided T_{rec} so far into two components,

$$T_{\text{rec}} = T_j^{(\text{therm})} + T_{\text{rec}}^{(\text{eff})}$$

which we try to specify separatly.

The thermalization time $T_j^{(\text{therm})}$ is the time that passes between the compilation of the artificial megajam and the moment when the most upstream car in that megajam begins moving again. In figure 5.2, this corresponds to the time between the first line (time = 0) and the white gap,

[1] $\rho_e(T)$ is the density, where the position of the right edge of the reversed-λ-shape is determined by taking a snapshot after T updates. $\rho_e(0)$ is the rightmost possible position of ρ_e. ρ_e is also where the laminar branch breaks down onto the jammed branch in Krauss' fundamental diagram. See chapter 3.2.

where the last car in the jam moves with $\frac{1}{2}V_{\max}$ (which is used as an arbitrary critical velocity). During the thermalization time we just have to wait until all N cars accelerate. The thermalization time is therefore connected linearly to the system size as

$$T_j^{(\text{therm})} \propto N$$

The effective recovery time is a bit more tricky to deduct. In very thin systems of length L_{system}, where we find absolute laminar flow, the following equation holds:

$$L_{\text{system}} \geq N_{\text{cars}} V_{\max}$$

We know that in dense systems, cars need to slow down to preserve safety and the overall mean velocity is decreased and jams can occur. To measure the recovery time, we force our system into one megajam before letting it run and we can distinguish the laminar and jammed phases. Let us look at a system with a single megajam wave travelling through the system. Hence we have some cars situated in the megajam (probably standing still) and free flow between the downstream and the upstream end of the jam across the periodic boundary. Assume that N_{cars} and thus L_{system} are very large so we can ignore the regions where cars accelerate at the downstream and brake in front of the upstream end of the megajam. In a system with N_{cars} vehicles, we define N_j to be the number of vehicles situated in the megajam. They occupy a space

$$\text{jam length} = \mathrm{N_j L_{car}} + (\mathrm{G})$$

(G) represents a correction, because we know that in realistic jams, $\langle g \rangle > 0$. We drop this correction in the following assuming that we *do* find $\langle g \rangle = 0$.

On the other hand we find $N - N_j$ cars in free flow between the downstream and the upstream end of the megajam (again, across the boundary). Those $N - N_j$ cars are spread over the free part of the system at the outflow density ρ_{out} with average gaps $\langle g \rangle_{\text{out}}$ between them.

They occupy a space

$$\text{length of free flow} = (\mathrm{N - N_j})(\mathrm{L_{car}} + \langle \mathrm{g} \rangle_{\text{out}})$$

These two (laminar and jammed) regions cover the entire system and by adding both lengths we find the equation

$$L_{\text{system}} = \frac{NL_{\text{car}}}{\rho} = N_j L_{\text{car}} + (N - N_j)(L_{\text{car}} + \langle g \rangle_{\text{out}})$$

Solving for N_j gives us the number of cars in the jam as

$$N_j = N_j(N, \rho) = N \frac{\frac{L_{\text{car}}}{\rho} - (L_{\text{car}} + \langle g \rangle_{\text{out}})}{-\langle g \rangle_{\text{out}}}$$

In the outflow of the jam the following equation holds:

$$\langle g \rangle_{\text{out}} = L_{\text{cars}} \left(\frac{1}{\rho_{\text{out}}} - 1 \right)$$

which gives us simply

$$N_j(N, \rho) = N \frac{\rho_{\text{out}} - \rho}{\rho(\rho_{\text{out}} - 1)}$$

To obtain an analytical estimation of $T_{\text{rec}}^{(\text{eff})}$ we will seek the probability that the jam with N_j cars resolves itself. If the number of cars in the jam N_j stays constant, which means that as many cars attach to the jam at the upstream end as detach from the downstream end, the jam will not resolve itself. We define P_1 to be the the probability that one car can leave the jam at the downstream end *without* a car attaching at the upstream end. With a probability $P_1 < 1$ the jam shrinks down to a size of $N_j - 1$ cars. The probability P_{N_j} that *all* cars can leave the jam (letting the jam disappear) is then

$$P_{N_j} = (P_1)^{N_j} = \exp\left(\ln(\mathrm{P_1})\mathrm{N_j}(\mathrm{N}, \rho)\right)$$

and after inserting N_j

$$P_{N_j} = \exp\left(\ln(\mathrm{P_1})\mathrm{N}\frac{\rho_{\text{out}} - \rho}{\rho(\rho_{\text{out}} - 1)}\right)$$

Finally we obtain an estimation for the recovery time:

$$T_{\text{rec}}^{(\text{eff})} \propto \frac{1}{P_{N_j}} = \exp\left(-\ln(\mathrm{P_1})\mathrm{N}\frac{\rho_{\text{out}} - \rho}{\rho(\rho_{\text{out}} - 1)}\right)$$

Note that $-\ln(\mathrm{P_1}) > 0$. We do not find a diverging recovery time and for finite N we have:

$$T_{\text{rec}} \begin{cases} \to & 0 & \Leftrightarrow \rho \to 0 \\ \sim & 1 & \Leftrightarrow \rho = \rho_{\text{out}} \\ \to & \exp\left(-\ln(\mathrm{P}_1)\mathrm{N}\right) & \Leftrightarrow \rho \to 1 \end{cases}$$

For increasing N the recovery time shows the following behavior:

$$\lim_{N\to\infty} T_{\text{rec}} \begin{cases} 0 & \Leftrightarrow \rho < \rho_{\text{out}} \\ \infty & \Leftrightarrow \rho > \rho_{\text{out}} \end{cases}$$

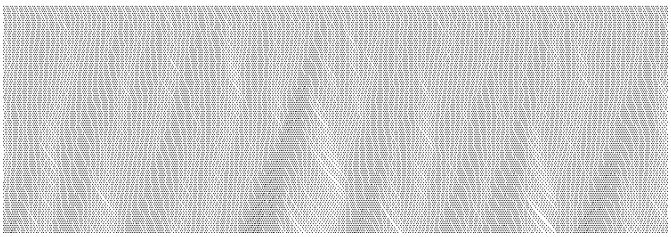
which is an infinite step function. Consequently, infinite systems in theory do not recover megajams at all.

5.5.2 T_{bdown}

Just as we did for the recovery time, let us divide the theoretical analysis of T_{bdown} into two parts as follows:

$$T_{\text{bdown}} = T_l^{(\text{therm})} + T_{\text{bdown}}^{(\text{eff})}$$

Some words on the thermalization time: For this theory cars are initialized to the homogeneous solution, which is an unnatural state. A few updates shake the system enough to show enough irregularities and for high densities the system can collapse into a few jams. The time needed to bring the system into a natural inhomogeneous state is the thermalization time $T_l^{(\text{therm})}$. The process of this thermalization is illustrated below:



In the above space-time plot we can see how the individual vehicle dynamics develop clusters. We call them clusters because denser and thinner stripes of vehicle clusters can be seen but we don't find real jam waves with halting vehicles yet. When increasing the system size, the average cluster size does not get wider, we just find proportionally more clusters. In this thermalization process correlation can be expected to reach as far as the

width of such an average cluster, which should motivate us to believe that the thermalization time $T_l^{(\mathrm{therm})}$ does not depend on the system size N_{cars}.

From measurement results we saw that T_{bdown} followed a straight line in the linear-log plot for densities $\rho \to 1$ and we also assumed that in those ranges the thermalization time dominates the effective break down time. Hence it is reasonable to look for a decreasing exponential behavior of $T_l^{(\mathrm{therm})}$ for increasing ρ such as

$$T_l^{(\mathrm{therm})} \propto e^{(-c(\rho-\rho_0))}, \qquad c > 0$$

which leaves the determination of the unknown c and ρ_0 to be solved.

Now to the effective breakdown time. In stable laminar flow cars are not slowed down by the randomization step. The velocity of the next update step is chosen by

$$v_{\mathrm{des}} = \min\left(\mathrm{V_{max}}, \mathrm{v(t)} + \mathrm{a}\Delta\mathrm{t}, \underbrace{v_{\mathrm{safe}}(t)}_{\mathrm{H}}\right)$$

followed by a randomization step

$$v(t+\Delta t) = \max\left(0, \mathrm{v_{des}(t)} - \mathrm{a}\epsilon\eta\right)$$

determining the final chosen velocity $v(t+\Delta t)$.

Let us assume we are in a rather dense, homogeneous traffic situation, where free flow at V_{max} cannot be realized by individuals and *all* vehicles are driving more or less equidistantly at a speed $v_{\mathrm{des}} < V_{\mathrm{max}}$. We can do this by temporarily switching off the noise by setting $\epsilon = 0$, or (which is what we do) by initializing the system to the homogeneous solution.

We will now examine an arbitrary car and its velocity updates for the next few time steps (including noise). We want to find out which velocity will be chosen to be minimal in the above equation for v_{des}. V_{max} cannot be chosen, because we demanded to be in a rather *dense* system. Secondly, we demanded to be in a (rather) homogeneous equidistant situation. This prohibits us from accelerating as

$$v_{\mathrm{des}} = v(t) + a\Delta t$$

because it harms safety:

$$v(t) + a\Delta t > v_{\mathrm{safe}}(t) = \langle v(\rho) \rangle$$

Hence the update rule will choose

$$v_{\text{des}} = v_{\text{safe}}(t) = \langle v(\rho) \rangle$$

as minimal velocity, which is marked by the underbracing "H" above.

We now implement the randomization step we can expect the final updated velocity to be

$$v = v_{\text{safe}} - a\epsilon\langle\eta\rangle$$

and we see that the randomization step can slow down cars in dense systems.

therefore we want to define $P_{\downarrow}$ to be the probability that a car decreases its velocity by $a\epsilon\langle\eta\rangle$ below its own current v_{safe}. This random step affects the vehicle's follower because it reduces the gap between them and the followers $v_{\text{safe}}^{(f)}$ is decreasing below the leader's $v_{\text{safe}}^{(l)}$. With the same probability $P_{\downarrow} < 1$ the follower is expected to decrease its velocity by $a\epsilon\langle\eta\rangle$ *below* its $v_{\text{safe}}^{(f)} < v_{\text{safe}}^{(l)}$.

The sequence of such events can cause a more upstream positioned car to stand still after at least s velocity retardations through noise:

$$s = \frac{\langle v \rangle}{a\epsilon\langle\eta\rangle} = \frac{2\langle v \rangle}{a}$$

The probability P_0 for that incident is

$$P_0 = (P_{\downarrow})^s = e^{\left(\ln(\mathrm{P}_{\downarrow})\mathrm{s}\right)}$$

Using the homogeneous solution as shown in figure 3.2 as estimation, we can roughly approximate the expected velocity for high densities ρ with a linear decaying line:

$$\langle v(\rho) \rangle = -m\rho + d, \qquad m > 0$$

Finally we obtain the probability P_0

$$P_0 = e^{\left(\ln(\mathrm{P}_{\downarrow})\frac{2(-\mathrm{m}\rho+\mathrm{d})}{\mathrm{a}}\right)}$$

and we find the following rough estimation for the breakdown time:

$$T_{\text{bdown}}^{(\text{eff})} \propto \frac{1}{P_0} = e^{\left(\ln(\mathrm{P}_{\downarrow})\frac{2(\mathrm{m}\rho-\mathrm{d})}{\mathrm{a}}\right)}$$

Note that $\ln(\mathrm{P}_{\downarrow}) < 0$.

What is still missing is the N_{cars}-dependency of $T_{\text{bdown}}^{(\text{eff})}$. The most probable way to implement this is to expect $P_{\downarrow} = P_{\downarrow}(N_{\text{cars}})$ because the explained random velocity reduction has more chances to occur in larger systems.

Chapter 6

The laminar-jammed interface

In chapter 4 we mentioned that the outflow from a jam is a way to distinguish Krauss' model types I to III. We want to investigate that approach to analyze traffic flow in detail.

6.1 Interfaces in traffic

When talking about *interfaces*, we mean the region in space where two phases of traffic touch each other or diffuse into each other. Rough comparisons (only as illustration) to thermodynamical examples could be for example the two-dimensional interface on the surface of a water nucleus, separating the liquid from the gas phase. Or, in an n-dimensional Ising model, the $(n-1)$-dimensional surface of a cluster of up-spin-particles surrounded by down-spin-particles.

The interface we will talk about here is the region where laminar traffic morphes into a jam and vice versa. To deterimine the boundaries of this interface, we have to set some arbitrary rules that define, where the jam ends and where free flow starts. We will see later on that the interface stretches over a certain region in space. Our vehicles run on a one-dimensional street, thus the interface is something between a point and an interval on the street, where the diffuse spatial transition from jammed to laminar takes place.

In this work, we will only investigate the downstream interface where cars *leave* the jam. The upstream interface, where cars come into the jam is also interesting[1], but will not be looked at here. We will analyze different

[1] Remarkably that the upstream interface of a jam is stable, when the fundamental

models by asking ourselves the following questions:

Is the outflow from jams (downstream interface) stable?

How does the interface width develop in time?

How does the interface width depend on the type of model or the braking capability b?

6.2 Initializations and definitions

As we mentioned we will only examine the downstream interface. We need to create an artificial jam and examine the flow of vehicles leaving it.

6.2.1 Making artificial jams

We investigate a bit how we produce artificial jams and what aspects have to be taken into account.

Since we want to examine interface widths that can grow to arbitrary lengths we have to drop periodic boundary conditions. When vehicles leave the artifical jam, it shrinks and we have to add further cars at the upstream end of the jam. In section 3.2.5, talking about the jammed branch in the fundamental diagram we mentioned that we create an artifical jam by halting any car and waiting until the entire system is queued behind it. We called it a jam, when we found $v_i = 0$ $\forall$cars i for the first time.

Due to the noise step

$$v(t + \Delta t) = \max\left(0, \mathrm{v_{des}(t)} - \mathrm{a}\epsilon\eta\right)$$

forward motion is retarded strongly when $v \sim a\epsilon\eta$ and it is possible to find cars in jams with a finite distance to their leader and they still do not drive closer. A car will move forward in the jam only if the random number generator produces a number that is small enough.

Let us define two types of jams:

Let us agree to call a jam *complete* when we halt an arbitray car for an infinitely long time so that $\rho_{\mathrm{jam}} = 1$ and the average gap to a leader $\langle \mathrm{gap} \rangle = 0$. Technically we can initialize this jam of course by aligning cars positioned bumper to bumper in a queue instead of waiting for this situation to occur.

---

diagram fulfills $\frac{d^2q}{d\rho^2} \leq 0$ everywhere.

A jam shall be called *incomplete* when it was formed by halting any car in a periodic system until $v_i = 0$ $\forall$cars i is true for the first time, like we mentioned above. Such a jam shows a typical density of $\rho_{\text{jam}} = 0.94$ and an average gap $\langle \text{gap} \rangle = 0.06$. We can compile such an *incomplete* jam by placing cars in a line according to these two values. However, halting a car and waiting results in a different density distribution within the artificial jam, namely denser at the downstream end, because cars there have had more opportunities to close the gap in front of them.

To be able to run interface width measurements, we have to keep feeding cars into the artificial mother jam at the upstream end and try to keep the original jam at a constant width. Technically it is easier to keep a constant number of cars (i.e. 50) between the most upstream car in the system, and the car defining the upper end of the interface.

The choice of jam used to examine the questions above is so far not relevant. Complete jams represent a stable situation in time while incomplete jams represent the more realistic situation:

In *spontaneous* jams of Krauss' space continuous model we observe gaps $\langle \text{gap} \rangle = 0$ only locally, rarely in jams that were several car lengths long. For global traffic densities $\rho < 1$ that show phase separation and form jam waves, any car is found in a jam only during a finite stretch of time and cars in the jam do not have the opportunity to align to $\langle \text{gap} \rangle = 0$.

6.2.2 Defining the interface

At the downstream end of the artificial jam, cars accelerate. The interface is the transition in space from jammed to laminar traffic. Technically it is impossible, due to fluctuations of positions and velocities, to nail down the interface to a single point. We need to specify an upstream and a downstream end. *The upstream end* of the interface shall be the point where cars can leave the jam to try to obtain a higher velocity. As daughter jams can occur for some configurations, this might be possible at the downstream end of several jams. The car, positioned at most upstream, who is just about to accelerate and leave its jam shall define the upstream end of the interface. We detect this car by finding the most upstream positioned car in the artificial mother jam that fullfills

$$v \geq v_{\text{critical}} \sim \frac{1}{2} V_{\max}$$

The critical velocity v_{critical} is arbitrary and difficult to choose. Using $\frac{1}{2}V_{\max}$ has shown to be a good approach. Alternatively we could detect the most upstream car whose gap g to its leader fullfills

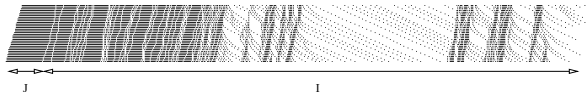


Figure 6.1: A sequence of a cellular automaton space-time plot developping an interface. The arrow above the "J" marks the artifical mother jam that is kept at a constant number of cars. The arrow above the "I" marks the interface, limitied by the first car in the mother jam moving, and the last two cars downstream driving at about V_{critical}.

$$g \geq g_{\text{critical}} \sim \text{const.} \times \text{V}_{\text{max}}$$

The downstream end of the interface shall be the point where the *two* most downstream positioned cars reach a relatively high critical velocity V_{critical}. Due to noise, cars never reach V_{max} itself. The test condition to find the downstream end of the interface can be

$$v \geq V_{\text{critical}} \sim V_{\text{max}} - \langle \text{noise} \rangle = \text{V}_{\text{max}} - \text{a}\epsilon\langle\eta\rangle$$

This condition should be fulfilled for the *two* cars positioned furthest downstream. We can say they managed to escape from the jam without forming further jams and therefore *remove* the car positioned furthest downstream from the system (next time the condition is checked for the *new* two cars positioned furthest downstream, which were before the second and third cars positioned furthest downstream). The downstream end of the jam is now simply marked by the car positioned most downstream in the system.

Figure 6.1 summarizes these issues with a space-time plot, indicating the mother jam and the boundaries of the interface.

6.3 Simulations

6.3.1 Stability of outflow and interface width development

The interface development in time is a criterion to qualify different types of traffic flow models as we saw in chapter 4. There we talked about stable and unstable outflow from jams.

Remembering that we remove the most downstream positioned car if it and its follower reach V_{critical}, we expect a model with a *stable outflow* to

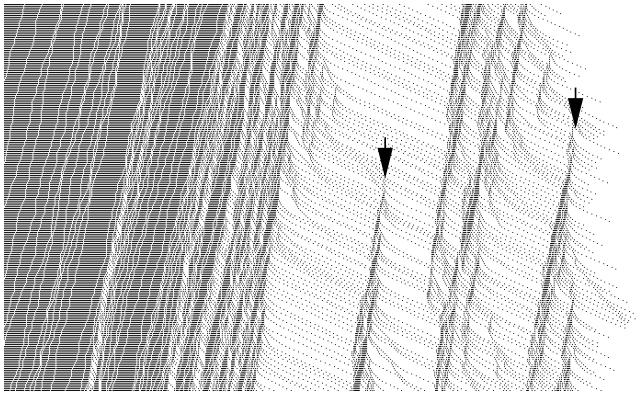


Figure 6.2: A space-time plot of a cellular automaton developing daughter jams in the outflow of previous jams. The arrows indicate their starting points.

have a *constant* interface width. If no daughter jams develop, the interface is very simple, as it just describes the region where cars accelerate from standing still at the downstream end of the mother jam until they reach V_{critical}.

An *unstable* outflow is signified by an *increasing* interface width. If a daughter jam develops, the interface gets wider. Only the two most downstream positioned cars will be respected when testing if we can remove a car. The development of a daughter jam therefore just shifts the edge of potential cars getting free further downstream, which widens the interface.

In figure 6.2 we see a space time plot of a cellular automaton with daughter jams. The quasi-diffuse appearance of the interface is typical for the that kind of model.

6.3.2 Results

Let us now look at simulation results of interface width measurements using complete jams, $\langle \text{gap} \rangle = 0$ and a motherjam kept at 100 cars.

We obtain the measurement of the interface width as plotted in figure 6.3.

In agreement with Krauss' definition and description of the types I and II, we obtain a constant interface width (stable outflow) for a normal type I Krauss model using $a = 0.2$, $b = 0.6$ and a growing interface (unstable outflow) for a type II model implemented by a standard cellular automaton. The shape of the plot for the interface width development of the cellular

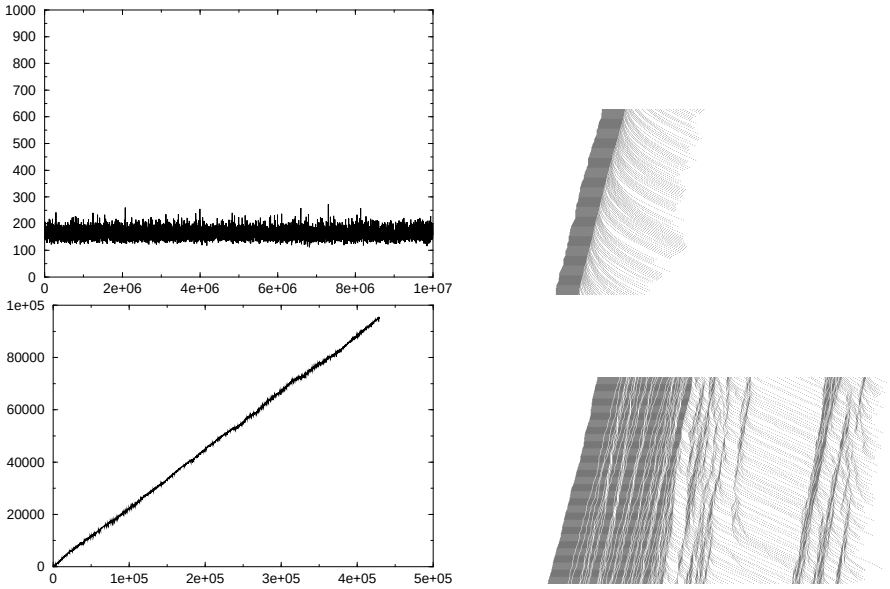


Figure 6.3: Interface width development over time. The simulation run time axis is pointing to the right, the interface width axis is pointing upwards. On top for a type I Krauss model, on the bottom for the cellular automaton. The left charts show the development on large time scales, on the right we can see corresponding space-time examples.

automaton allows us to be certain that it grows *linearly* in time. This is not too surprising: The region that determines whether the interface grows or not is the outflow of the most downstream positioned daughter jam. That region is somehow self similar in time. If another daughter jam develops further downstream, this region will just be shifted on below to the new daughter jam, but show the same behavior there as before. The probability of a daughter jam developping at the downstream end of the system is probably constant in time, hence the linear growth.

$$\text{interface width} \propto \text{time}$$

The corresponding space-time plots of stable and unstable outflow can be seen in figure 6.3.

Chapter 7

Krauss model vs. cellular automaton

The parameter space of Krauss' model contains the cellular automaton. Using

$$\begin{aligned} a &= 1 \\ b &= \infty \\ \text{noise} &= a\epsilon\eta \\ \eta_r &\in \{0,1\} \\ \eta_r &= \begin{cases} 1 & \Leftrightarrow \text{random number} \leq \text{r} \\ 0 & \Leftrightarrow \text{else} \end{cases} \end{aligned}$$

we can convert to the Nagel-Schreckenberg cellular automaton. The configuration $a \ll b = \infty$ makes the cellular automaton similar to a type II Krauss model with modified noise.

By initializing systems with vehicle velocities and positions as integers

$$v_i, x_i \in \mathbb{N}$$

we obtain standard cellular automaton dynamics[1].

To see how the cellular automaton fits into the theory of Stefan Krauss, we would like to try to investigate the dynamics via the break-down/recovery behavior and their jam interface widths.

[1] For every car that has had a gap $g = 0$ *once*, $v_i, x_i \in \mathbb{N}$ will be fullfilled automatically.

7.1 Jammed↔laminar transition behavior and waiting times comparisons

The preceeding section introduced tools and aspects to qualify the transition from laminar to jammed flow. The waiting time plots indicate alot about a system's behavior by the shape and relative and absolute positions of T_{rec} and T_{bdown}.

The following discussion of the waiting times refers to figure 7.1.

Krauss type I

Both recovery and waiting time increase steeply at two corresponding densities. The upper boundary of the T-axis ($T_{\max} = 10^8$ updates) represents the longest simulation run time examined for 5000 cars. The density range between the two curves where neither T_{rec} nor T_{bdown} could have been determined represents systems whose phase for a *finite* time range $T \sim T_{\max}$ depends only on the initialization: Such systems need more time than $T_{\max}$ to recover from a megajam and hence are expected to stay jammed for simulation times below $T_{\max}$ and laminar systems are not expected to break down below that time. In earlier sections we observed that the absolute positions of T_{rec} and T_{bdown} shift slightly towards low densities when increasing the system size N_{cars} because larger systems are more inert to change state in time.

Krauss type II (continuous)

The graphs of the waiting times for a Krauss type II show a different behavior just by the increased $b = \infty$. We know from space time plots, that type II systems show no phase separation but have fluctuation-like minijams. These minijams are responsible for the measurement "irregularities" in low densities, where we find several examples of $T_{\text{rec}} \gg 0$ even though we expect $T_{\text{rec}} \to 0$ for $\rho \to 0$. The fluctuations make the system detect cars with $v \sim 0$ which is interpreted as a still existing jam.

For increasing ρ, T_{rec} is steeper here than in the graph of type I and the knee for T_{rec} at $\rho \simeq 0.16$ is "flatter". But all in all the relative position of high T_{rec} has not changed drastically between types I and II.

The break down time T_{bdown} is very different. We mentioned that T_{bdown} is probably dominated by its thermalization time for large densities. This region ($\rho > 0.215$) shows the same structure for both types I and II. Below $\rho = 0.215$, T_{bdown} is much "flatter" for type II and the density range of

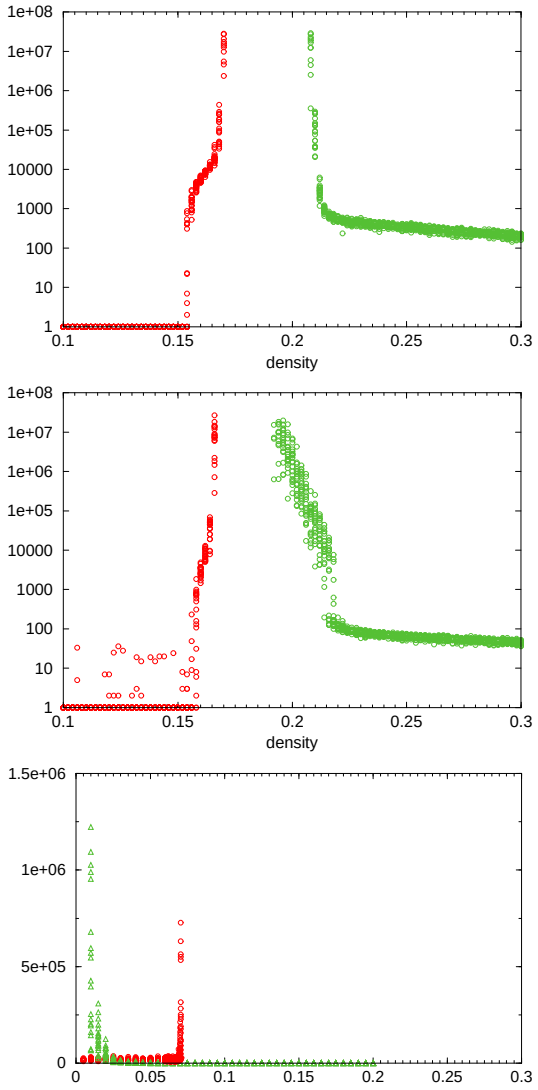


Figure 7.1: Breakdown and recovery times for $N_{\text{cars}} = 5000$. On top: Krauss type I continuous ($a = 0.2$, $b = 0.6$, $V_{\max} = 3$). Middle: Krauss type II continuous ($a = 0.2$, $b = \infty$, $V_{\max} = 3$). Bottom: cellular automaton, discrete ($V_{\max} = 5$). In the upper two diagrams, the left peak belongs to T_{rec} and the right peak to T_{bdown}. In the lowest plot this is reversed due to an overlap of the waiting time plots.

undetermined system waiting times is more narrow. Whether T_{bdown} follows a straight line in the linear-log plot is difficult to determine.

The type II waiting time diagram makes one believe that T_{rec} and T_{bdown} could intersect for finite system sizes.

cellular automaton (discrete type II)

The cellular automaton is part of the Krauss type II regime of models. The system used to obtain the waiting times for this model adapted the noise using a probability $r = 0.5$.

Compared to the continuous type II, we do not find the irregular measurements of T_{rec} for $\rho \to 0$. It seems to be easier to find cars with sufficiently large velocities to call the system "resolved". Maybe this is a hint that the cellular automaton fluctuation-like minijams are shorter in time (*increasing* the likelihood of find a recovered system) or that the ones for continuous type II are more likely to overlap in time (*decreasing* the likelihood to find a recovered system).

While T_{rec} increases at a similar position as in the cases of the previous types, T_{bdown} is found much further left, near low densities. The fact that they overlap is a consequence of the choice of critical velocity for resolved systems and broken down systems. In both cases we examine the slowest car, which has to be above a certain $v_{\uparrow}$ to call the system recovered from a jam and below a certain $v_{\downarrow}$ to call it broken down from laminar. The waiting time overlap is a consequence of

$$\frac{1}{2} = v_{\uparrow} > v_{\downarrow} = 0$$

A waiting time overlap can by the way also occur for the types I and II when simulating very small systems.

The fact that T_{bdown} is positioned so far towards low densities is most probably a consequence of the fluctuation-like minijams that make the system detect cars with speed zero, even though such jams resolve quickly.

7.2 Downstream interface comparison

Another possibility to distinguish types I and II according to Krauss, is to examine their interface outflow behavior which we introduced before. Type I has a stable outflow from jams while type II has an unstable one.

Figure 6.3 shows space-time plots of the corresponding outflows.

The Krauss type I system ($a = 0.2$, $b = 0.6$ and $V_{\max} = 3$) shows clear phase separation and we find no further jam development downstream.

The cellular automaton develops downstream daughter jams and outflowing vehicles converge in new jams. As we stated previously, jams in the cellular automaton appear and disappear within short time ranges. This fluctuation-like behavior dissolves clear downstream edges of jams and we observe a diffusion of the interface, which is probably linear.

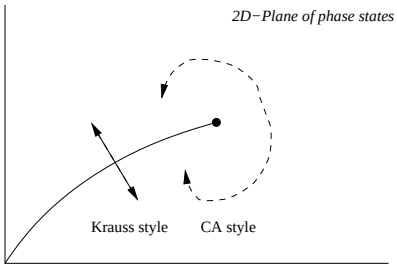
The border between types I and II in the (a, b)-plane defined by the dashed line in figure 4.3 is very vague. It should be possible to re-draw that line alternatively by doing precision measurements on detecting growing interface widths when decreasing b at a constant a.

Analogue to phase transitions

Looking at the space-time plots, one wants to compare the laminar↔jammed transition to a regular (first order) phase transition. This comparison is not totally precise, but it is an interesting and fertile mind experiment.

The two phases *jammed* and *laminar* can be definded precisely and the analogon to water (gas↔water) is obvious. The transition between jammed and laminar flow though is different to the ones of nucleation and Ising models. For instance, the first order liquid-gas nucleation is determined by surface tension. Such a force cannot be found amoung cars, because they are not attracted by electromagnetic or aequivalent forces. However, the following thoughts point out a few remarkable *similarities* to first order phase transitions:

Let us think of a 2-phase system with two control parameters and a first order phase separation line ending in a second order critical point. The corresponding phase plane could look like the following diagram (using arbitrary axes):



(For example, water without a solid phase, with control parameters (axis

labels) temperature and pressure.) The text in the diagram will be explained:

The first order phase transition is designated by a sudden reformation in the new phase when crossing the critical point, and by a phase coexistance *at* that point. *Krauss systems* of type I show clear phase separations. Jam waves are stable and can be looked at as a condensation in the new phase. Also, if we assume the existance of a singularity ρ_* of the breakdown time, ρ_* could function as a critical point above which spontaneous jam formation occurs. These arguments should be a motivation to draw the left dashed line in the diagram above and to understand Krauss' model as a realization of a quasi first order phase transition between the phases laminar and jammed.

The *cellular automaton* on the other hand, being a type II Krauss system, shows no clear phase separation (figure 3.1). Jams occur as density fluctuations (nucleoli) with short life times. A cellular automaton system, initialized to a single megajam, dissolves into many small short-life jams distributed over the entire system. Something similar is observed when going "around the critical point" in the 2-phase plane (right dashed line in the diagram): water changes continuously into the gas phase and both phases are mixed but not separated.

Chapter 8

Computational issues

8.1 Code

Structure

In this work we examined two traffic flow models, namely Krauss' model and the cellular automaton. The cellular automaton was developed *out* of Krauss' model with the adaptions mentioned in chapter 7. This does not result in the fastest possible implementation of the cellular automaton ever (and the expression "cellular" somehow loses its purpose) but this adaptive implementation is helpful for model comparison.

We will therefore only discuss the implementation of Krauss' model. The core of the program is a

```
class car
```

containing all information for individual vehicles, position x, velocity v and the gap to the their leader g. Cars are organized in an array of type `car` using the Blitz++ library, which offers quick manipulations on arrays:

```
blitz::Array<car,1> cars;
```

The Blitz array does not need the array length at compile time compared to standard arrays. All information about the simulation and functions operating on cars are put together in a class `system`:

```
class system
{
public:
```

```
        system();

        blitz::Array<car,1> cars;

        void reset(real);
        void update(bool);
        void measure();
...

        int N;
        real rho, L_system;
        bool collapsed;
        bool resolved;
...
}
```

As we mentioned before, the system size is identified with the number of cars. All measurements are averaged over cars, which makes it reasonable to keep the number of cars constant as a precision reference. Also, the simulation performance does not depend on the global density as it is the case for many programs. The density is the argument in the function `reset` mentioned above.

For this work an external setup file containing information about the simulation was used. That way we could avoid recompiling before running a new configuration. The setup file contains information about:

- The system settings such as N_{cars}, ρ
- The density range that should be examined and its resolution
- How long the simulation should run and at what intervals fundamental diagram snapshots should be saved
- Krauss' parameters, such as a, b, $V_{\max}$, ϵ and the random seed
- Settings on how to start the simulation (laminar or jammed)
- Settings whether the simulation should halt when a jam was resolved or a laminar start broke down (required for waiting time measurements)
- Settings, if the space-time plot should be piped into the standard out

As random number generator the standard drand48 was used. Via the setup files different simulations could be started in parallel with different initializations to obtain an ensemble-averaged value.

Performance

The code structure illustrated above results in 1.25 million updates per second on a 700 MHz Pentium III. About 8% of the CPU time is consumed by measurement and storage of data.

8.2 Use of the Beowulf cluster

We talked about ensemble averaging of measurements.

To obtain reliable data for fundamental diagram or waiting times, at least ten independent simulations should be run with different random seeds. We are also interested in comparing a fundamental diagram with the corresponding waiting times. To get all this data takes alot of CPU time, and we need to think of some kind of parallelization.

A real parallelization of the code is not reasonable, because we are dealing with a single lane road. However, the independent simulations leading to one ensemble averaged measurement can be run spread over several CPUs running in parallel.

Due to our data structure getting all all of its parameters from an external setup file, this is a very easy thing to do on the Beowulf cluster provided by ETH. Users don't have to spread their CPU time over different nodes. This is handled by a central queueing system respecting jobs from users according to the requested wall time. All nodes mount the users home directory before starting. therefore we created a directory structure with a different setup file in each directory. This is easily done using a few simple batch files.

To examine traffic flow depending on snapshot moment, system size and breaking capability b we need to look at the fundamental diagram and waiting times of various systems. 20 independent simulations (different random seeds) of four different system sizes were examined. This results in 640 simulations requesting an average wall time of approximatly 24 hours. On a single CPU machine this would take 2 years to compute, which makes the use of a cluster inevitable.

Chapter 9

Summary

The work of this paper can be seen as a digestion and re-evaluaton of the dissertation of Stefan Krauss.

We made a few comments on traffic models in general and described the model of Stefan Krauss and the cellular automaton by Nagel and Schreckenberg.

In a few middle sections we presented several tools that help us to analyze traffic. We showed common techniques such as space-time plot analysis giving good pictures over development in space and in time. We explained the fundamental diagram of various model types used in this work and pointed out their specialities. We mentioned the uncommon ensemble-averaging of space-time plot instead of the general approach to calculate time-averaged plots.

The Krauss model can keep one busy for quite a while. Not only because it hides a few uncertain features, but also because it allows variation of at least two parameters that change the behavior drastically. We pointed out the differences between the types I, II and III and gave arguments to distinguish them.

We examined the transitions times in detail. These are the recovery time that it takes to resolve an artificial megajam at a given overall density, and the breakdown time that passes until we find the first vehicle with speed zero after homogeneous initialization. Especially the recovery time has not been investigated much. We explained how to read plots of transition times and explained their tight connetion to the fundamental diagram. Then we showed a few measured plots of these times and pointed out their behavior when changing to another model (by changing the breaking capability) and when changing the system size. We tried to give reasonable analytic ap-

proaches to the waiting times via the corresponding recovery and breakdown probability.

Another tool to analyze traffic is the interface width development in the outflow of a jam. Some model types show a stable outflow from jams while some others show an unstable one. These observations can be grasped by measuring the interface width dependency in time. Krauss model types I and II are distinguished by their interface behavior. We believe that further investigation on the actually very simple interface width could help to draw a more precise line between the types I and II.

In a later chapter, we tried to compare a type II Krauss model to the standard cellular automaton and point out their differences, summarizing the observations of the previous chapters.

Appendix A

Bilbiography

[1] Krauss S. *Microscopic modeling of traffic flow: Investigation of collision free vehicle dynamics.* Dissertation, DLR-Forschungsbreicht 98-08, Köln, 1998.

[2] Janz S. *Microskopische Minimalmodelle des Strassenverkehrs*, Diplomarbeit im Fachbereich Physik, Mathematisch-Naturwissenschaftliche-Fakultät, Köln 1998.

[3] Nagel K. *Personal conversation and email*

[4] Wagner P. *Email conversation*

Appendix B

Acknowledgements

I want to thank a few people who directly contributed to this work.

This work would not have been possible with the detailed creative help of Kai Nagel who was my direct mentor for this thesis, which I want to thank him for very much. He always supported me with new input, provided me with the required theory and helped me to keep orientation and focus in the very vast field of traffic flow theory.

I want to thank Matthias Troyer, who was my professor in computational physics, for giving me the computational tools required to get where one should be to master everyday problems in computational physics.

A great thank you goes to sis Nurhan Cetin. Her presence, all the coffee breaks and all the time I spent with her were so supporting. Furthermore she helped me so adorably as mental fuel, reality check, living Linux manual and energy source.

Bryan Raney, among my favorite office mates in the group, I want to thank for his continuous patience with me and my work. As the best English speaking person I want to thank him for his language support, for answering all my questions concerning C++ and awk and for keeping up with me during the entire time.

Martin Strauss I want to thank for the great time we had when working parallely sharing the office. I want to thank him for testing the most recent ideas popping up in my head and for creating a very creative ambience in

our room.

Furthermore but nevertheless so very important for providing such caring background I want to thank Jan, my mother and father and Darja.