Analysis of a Cellular Automaton Model for Car Traffic with a Slow-to-Stop Rule☆

Adam Clarridge*,[a], Kai Salomaa[b]

[a] *McMaster University, Hamilton, Canada*
[b] *Queen's University, Kingston, Canada*

Abstract

We propose a modification of the widely known Benjamin-Johnson-Hui (BJH) cellular automaton model for single-lane traffic simulation. In particular, our model includes a 'slow-to-stop' rule that exhibits more realistic microscopic driver behaviour than the BJH model. We present some statistics related to fuel economy and pollution generation and show that our model differs greatly in these measures. We give concise results based on extensive simulations using our system.

1. Introduction

An almost universal daily annoyance in most North American cities is getting slowed down or stuck in traffic. Many people spend hours each day in traffic, slowly losing their money and sanity while generating unnecessary pollution. Unfortunately, in many cities the addition of more highways to reduce the growing amount of congestion is far too expensive since the land is already developed. Because of these limitations, if traffic management is to be improved, it is important to understand the dynamics of car traffic flow extremely well to facilitate the planning and prediction of high density traffic. The earliest traffic flow models were based on fluid dynamics, but more recently cellular automata (CA) based models have been gaining popularity. This is partly because simulations are easy to develop and run very quickly (especially on designated parallel hardware), but also since cars in traffic operate under their own power and do not emulate particle flow based on the laws of physics particularly well.

The first study using CA for car traffic simulation was conducted by Nagel and Schreckenberg [1], who develop a simple stochastic CA model to simulate single-lane highway traffic. Essentially, the model says that all cars follow the

☆A preliminary version of this paper appears in the *Proceedings of the 14th International Conference on Implementation and Application of Automata* (2009).

*Corresponding Author
Email addresses: adam.clarridge@gmail.com (Adam Clarridge), ksalomaa@cs.queensu.ca (Kai Salomaa)

same basic transition rules, and then move v sites at each time interval. They increase their velocity v by 1 up to some limit as long as there are no cars v spaces ahead of them, slow down to speed $i-1$ if they see a car i spaces ahead of them, and randomly slow down by one speed unit with some probability p. The authors observe nontrivial, realistic simulation, particularly the transition from laminar traffic flow to start-stop waves as density increases.

In this paper, we focus on the microscopic behaviour of cars in the BJH model (a modification of the Nagel-Schreckenberg model) - specifically the fact that they decelerate in a very unrealistic way. Since cars only decelerate to avoid collisions, it is a frequent occurrence that cars drive up to a jam at maximum speed and slow down to a stop in a single time step. In order to more closely simulate the behaviour of human drivers, we propose a modification to the BJH model where cars begin slowing down earlier by an amount which is a function of their speed, the speed of the car ahead, and the distance to the car ahead. In our simulations we investigate alternate methods for introducing new cars to the section of road used for simulation, and also attempt to measure the average fuel efficiency of the cars by recording the number of acceleration steps each car takes.

In Section 2, we give a summary of selected papers relating to CA-based traffic modelling, and we present a brief comparison of our model with other (similar) recent extensions of the NaSch model. We describe our model in detail in Section 3 and present simulations that compare it with the BJH model. We experiment with an alternate method for introducing new cars to the road during simulation in Section 4. Finally, we summarize and conclude the paper in Section 5.

2. Summary of CA-based Traffic Simulation Models

The Nagel-Schreckenberg (NaSch) model [1] has been studied quite extensively in several papers [2, 3, 4, 5, 6].

Another model developed by Benjamin, Johnson, and Hui (BJH model) [7] is quite similar to the NaSch model, but with the addition of a 'slow-to-start' rule. That is, a vehicle which has come to a complete stop moves forward at its first available opportunity with probability $1-p_{slow}$, and on the time step after that with probability p_{slow}. The authors used this model to study the effect of junctions on highways, finding that setting a speed limit near junctions on single lane roads can greatly decrease the queue length to enter the road.

Since almost all major highways have two lanes or more, several researchers have constructed multi-lane models for highway traffic. The first work in this area was done by Rickert et al. [8], who designed a working model based on the NaSch model. They noticed that checking for extra space when switching lanes ('look-back') is an important feature of their model in order to get the realistic behaviour of laminar to start-stop traffic flow. Wagner et al. [9] design a two-lane simulation which accounts for a faster left lane which is to be used for passing. Using simple rules, they are able to obtain the realistic behaviour that at higher overall densities, the left lane has a higher density than the right one.

They remark that this correct macroscopic behaviour is fairly easy to obtain using a CA model, and cite some failed attempts to simulate multi-lane traffic using other types of models. Knospe et al. [10] study heterogeneous two-lane traffic and find that even at low densities, a very small amount of slower cars effectively cause both lanes to slow significantly. Also, they note that a system with mostly slow cars and a small percentage of fast cars is almost identical to a system with all slow cars. Finally, Nagel et al. [11] summarize the existing lane-changing CA models and propose a general scheme according to which realistic lane-changing rules can be developed.

Esser and Schreckenberg designed a complete simulation tool for urban traffic in [12]. The model accounts for realistic traffic light intersections, priority rules, parking capacities, and public transport circulation. The simulation of large traffic networks can be performed in multiple real-time. Several other researchers have devised related schemes [13, 14, 15, 16, 17, 18, 19].

After most of the work for this paper had been completed, a slightly more extensive search of the literature yielded a few strongly related papers. We feel that it is important to address some of the differences in our work, since there are several papers which present modifications to the NaSch model.

One of the first papers to propose a modification of the NaSch model was by Emmerich and Rank [20]. They devise a scheme which describes the change in velocity by a matrix M, whose indices correspond to the velocity of the current car and the gap (distance to the next car) and whose entries correspond to the speed of the car on the next time step. This model indeed provides a very general braking scheme, but does not depend on the velocity of the car ahead. A model by Knospe et al. [21] suggested many modifications to the NaSch model, the most relevant to our work being the braking according to an 'effective gap'. This term is defined to be a function of the 'anticipated' velocity of the car ahead, the 'security gap' (a fixed quantity set at simulation time), and distance to the car ahead. This is intuitive, however one possible criticism is that the 'anticipated' velocity of the car ahead is a function of that car's distance to the car ahead of it. Since drivers cannot always see two cars ahead, this behaviour may lead to unrealistic car decisions in some situations. An interesting study by Makowiec and Miklaszewski[22] also modifies the microscopic interaction of cars in the NaSch model, but with a 'looking behind' rule. They claim that on Polish motorways, there are several different maximum vehicle speeds (due to tractors, bikes, horse carts, and modern cars on the same road). Usually there is no place to pass, so the normal maximum speed of a vehicle travelling at that speed actually increases when the distance to the car behind it is small. Mallikarjuna and Rao [23] and Lan and Chang [24] both experiment with the general idea of modifying the size of cells in order to simulate heterogenous traffic, including motorcycles and trucks instead of just cars. Bham and Benekohal [25] developed a very detailed model which they claim is validated at the microscopic and macroscopic levels using two sets of empirical data. Chakroborty and Maurya [26] compared this and other models against several macroscopic benchmarking criteria, and gave their own model as well which passed all of their benchmarks.

Our model is perhaps simpler to implement than most of these models, since we use only the speed of the current car, the speed of the car ahead, and the distance between them in order to determine the velocity on the next time step. Our modification to the BJH model, though perhaps minor, produces interesting output and we provide at least some indication of the fuel economy or pollution generation statistics, a characteristic that is lacking in most of the literature.

3. A Modified Version of the BJH Model

Here we investigate a modification of the well-known Benjamin-Johnson-Hui (BJH) CA model [7] for single-lane highway traffic. This model is able to correctly capture several of the macroscopic characteristics of real traffic using very simple and computationally fast cellular automata, and as a result, has been studied extensively and incorporated into several complex traffic simulators. The primary reason we choose this slightly older model rather than one of the recent more complex models is that it has extremely simple transition rules which are easy to understand, so our extension will be clearer.

The BJH model is based on the NaSch model, which we will now describe in detail. The NaSch model is defined on a one-dimensional cellular space of N cells, usually with the toroidal (periodic) boundary condition. On a particular time step each cell either contains a car or is empty, and each car has an integer velocity v between 0 and v_{max} inclusive. Given some global configuration of cars at various velocities, the NaSch model dictates that cars are advanced along the road on the next time step according to the following rules, which are performed in order and in parallel for all cars. The quantity d is the distance in cells to the next car ahead.

1. Acceleration: if $v < v_{max}$ and $d > v{+}1$, then velocity increases ($v \leftarrow v{+}1$).
2. Slowing down (collision avoidance): if $d <= v$, then velocity decreases appropriately ($v \leftarrow d - 1$).
3. Randomization: if $v > 0$, with probability p_{fault}, velocity decreases by one ($v \leftarrow v - 1$).
4. Motion: the car advances v cells.

These velocity rules implicitly do not allow collisions or overtaking.

The BJH model is a fairly straightforward extension of the NaSch model - the authors attempt to more accurately simulate the behaviour of drivers which have come to a complete stop in traffic jams on the highway. Cars which have velocity 0 either accelerate at their first available opportunity (as soon as there is an empty space ahead of them) with probability $1 - p_{slow}$, or on the time step immediately after that with probability p_{slow}. Otherwise, they follow the NaSch model. This scheme is intended to reflect the fact that drivers take longer to accelerate from a complete stop, perhaps because they do not immediately notice the car ahead of them moving, or because of the slow pick-up of their car's engine. So the BJH model is essentially the NaSch model with the addition of a 'slow-to-start' rule. An example of cars following the BJH model on a small road

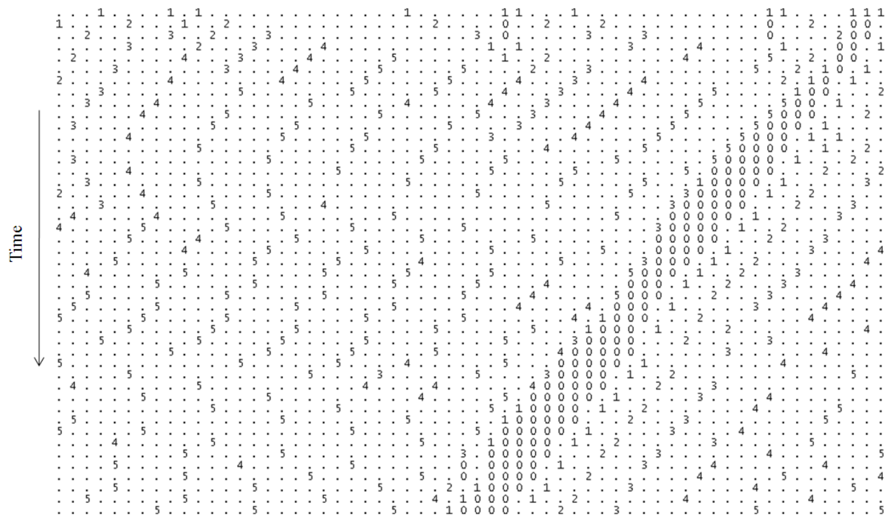

Figure 1: A small example of cars following the BJH model. The dots refer to empty cells, and the numbers represent the velocities of cars. Here the density $\rho = 0.2$, $p_{fault} = 0.1$, and $p_{slow} = 0.5$. Cars drive from left to right.

is given in Figure 1, and a more complete picture on a larger road for a longer period of time is given in Figure 2. In these examples, the initial configuration is a random placement of ρN cars ($0 < \rho < 1$) with velocity 1, where N is the size of the road in cells.

We noticed that cars following these models behave in an unrealistic fashion when approaching a jam; if a car B ahead has velocity 0, then a car A may drive up to B at velocity v_{max} only to brake down to 0 velocity in one time step in the cell right behind B. This microscopically inaccurate behaviour may not be a big issue since these models are only meant to be macroscopically realistic in some ways, but we believe it could be interesting to explore the addition of a 'slow-to-stop' rule. That is, we want to modify the BJH model so that cars look farther ahead than v cells and slow down earlier in certain situations. People typically pay attention to the velocity of the car directly ahead of them, so we use this information to aid in the decision of how much and when to slow down. A car's change in velocity is then a function of its current velocity, the velocity of the car ahead of it, and the distance between them.

In our model, the cars' velocities are adjusted at each time step according to the following rules. Recall that d is the distance to the next car, v is the velocity of the current car, v_{next} is the velocity of the next car, p_{slow} is the probability that the slow-to-start rule is applied, and p_{fault} is the probability that the car slows down randomly. We fix $v_{max} = 5$.

1. Slow-to-Start: As in the BJH rule, if $v = 0$ and $d > 1$ then with probability $1 - p_{slow}$ the car accelerates normally (this step is ignored), and with

Figure 2: A 'zoomed-out' view of a larger simulation of the BJH model. Black dots refer to cars, while white space is empty road. Here the density $\rho = 0.15$, $v_{max} = 5$, $p_{fault} = 0.1$, and $p_{slow} = 0.5$. The road is 1000 cells wide and the last 1000 evolutions out of 2000 are shown (to reach a steady state). Cars drive from left to right, and time 0 is at the top.

probability p_{slow} the car stays at velocity 0 on this time step (does not move) and accelerates to $v = 1$ on the next time step.

2. Deceleration (when the next car is near): if $d <= v$ and either $v < v_{next}$ or $v <= 2$, then the next car is either very close or going at a faster speed, and we prevent a collision by setting $v \leftarrow d - 1$, but do not slow down more than is necessary. Otherwise, if $d <= v$, $v >= v_{next}$, and $v > 2$ we set $v \leftarrow \min(d - 1, v - 2)$ in order to possibly decelerate slightly more, since the car ahead is slower or the same speed and the velocity of the current car is substantial.
3. Deceleration (when the next car is far): if $v < d <= 2v$, then if $v >= v_{next} + 4$, decelerate by 2 ($v \leftarrow v - 2$). Otherwise, if $v_{next} + 2 <= v <= v_{next} + 3$ then decelerate by 1 ($v \leftarrow v - 1$).
4. Acceleration: if the speed has not been modified yet by one of rules 1-3 and $v < v_{max}$ and $d > v + 1$, then $v \leftarrow v + 1$.
5. Randomization: if $v > 0$, with probability p_{fault}, velocity decreases by one ($v \leftarrow v - 1$).
6. Motion: the car advances v cells.

These rules prevent collisions and overtaking. We now attempt to justify the second and third of these rules, which differ from the BJH model.

Consider the following scenario: a car with velocity 5 has a car 5 spaces ahead of it with velocity 0. The BJH model would change the car's velocity to 4, and assuming the car ahead still has not moved, the car would be forced to decelerate to 0 on the next time step. Our model's second rule decelerates the car to 3 in this case so that it is two spaces away, then on the next time step to 1 so that it is one space away, then finally to 0. We believe this is much more realistic behaviour, since cars which see a stopped car ahead of them would certainly attempt to slow down gradually. In less extreme situations, our model behaves the same way as the BJH model in terms of collision avoidance. Note that we are assuming for both models that the car ahead does not move and the randomization rule has not been applied.

Now consider another situation: a car with velocity 5 has a car 6 spaces ahead of it with velocity 0. The BJH model would not change the velocity of the car, resulting in a very sharp deceleration on the next time step as it decelerates from 5 to 0. Our model's third rule decelerates the car to 3 so that it is 3 spaces away on the next time step, then the second rule decelerates the car to 1 so that it is two spaces away, then the car continues at 1 to the last space, then stops. Again, we believe that this type of gradual deceleration is typical of real drivers, and again we have assumed in this scenario that the car ahead does not move and that the randomization rule has not been applied.

Although both examples involved cars ahead which were stopped, the deceleration rules apply whenever a car is going significantly faster than the car ahead of it. While the car ahead with velocity 0 is the most illustrative case, the above examples could also be considered for different 'car ahead' speeds of 1 or 2.

An example of cars following our 'slow-to-stop' model is given in Figure 3. In this example, the simulation parameters are exactly the same as in Figure 2.

One would think that on a real highway with a fairly low car density, where a small jam is visible from a distance, drivers would slow down enough beforehand to allow the stopped cars to continue. The 'slow-to-stop' rule causes drivers to go slower when approaching jams, and as we conjectured this added foresight seems to help to slow down cars enough before the jam so as to let it dissipate on its own over time. There are fewer long jams with many cars at a complete stop, and instead there appear to be many slowdowns to avoid these situations, which we think is fairly accurate behaviour at medium traffic densities.

In Figure 4 we give the so-called 'fundamental diagram' for our model.

We were interested to discover the impact on fuel economy that the 'slow-to-stop' rule would have on the BJH model, so the average number of acceleration cycles and loops driven per car were recorded. The number of accelerations per car was recorded by simply incrementing a counter at each time step by an amount equal to the number of cars whose velocity increased by 1 on that time step. The number of loops driven per car was counted by incrementing a counter each time a car reached the end of the road and started back at the beginning of it. These two quantities provide at least a rough idea of fuel economy. For the simulation parameters used in Figures 2 and 3 averaged over 10 iterations, it was found that the average number of acceleration cycles per car for the BJH model and the slow-to-stop model was 134.3 and 216.7 respectively, and average number of loops driven per car was 3.7 and 3.4 respectively. It is very interesting that although the 'slow-to-stop' cars had several more acceleration cycles (about

Figure 3: A 'zoomed-out' view of a larger simulation of our 'slow-to-stop' model. Black dots refer to cars, while white space is empty road. The simulation parameters used to produce this output are the same as those used for Figure 2. Cars drive from left to right, and time 0 is at the top.

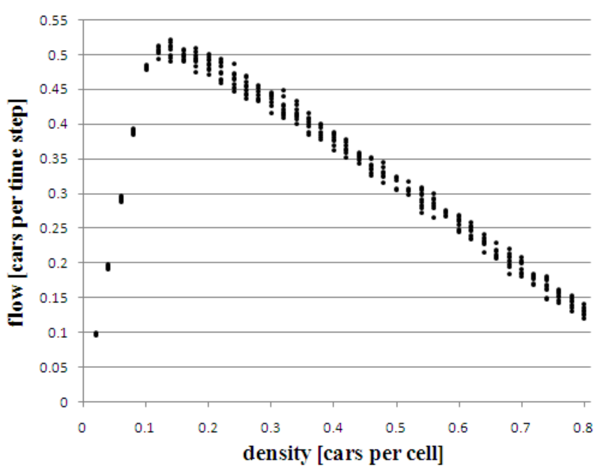

Figure 4: The 'fundamental diagram' for our model. Each point represents the result from the latter 1000 iterations out of 2000 iterations (to reach steady state) on a road of length 1000 starting from a random configuration. Car density was set from 0 to 0.8, in intervals of 0.02, and ten simulations were performed for each density. p_{fault} was set to 0.1, and $p_{slow} = 0.5$.

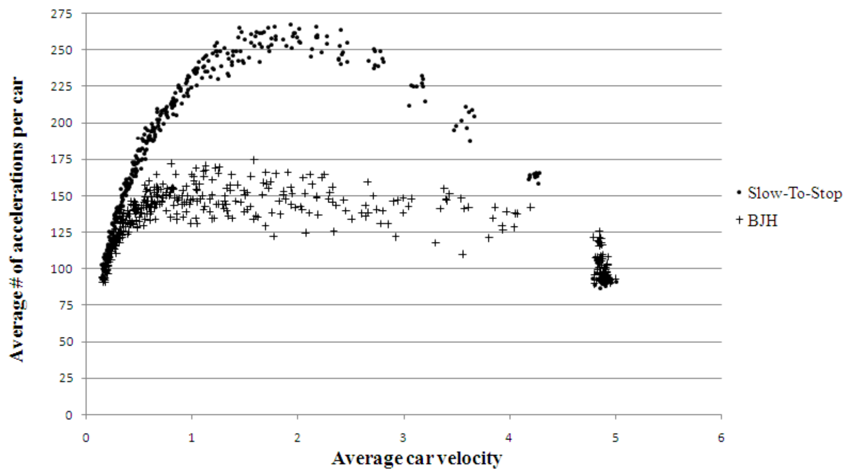

Figure 5: A fuel economy diagram comparing our model with the BJH model. Each point represents the result from the latter 1000 iterations out of 2000 iterations (to reach steady state) on a road of length 1000 starting from a random configuration. The same simulation parameters as in Figure 4 were used, but average car speed and average number of acceleration cycles per car were recorded instead.

61% more), cars travelled a very similar distance in the same amount of time. Since 'slow-to-stop' cars tend to slow down more often, the two models probably had similar distance results because in the BJH model cars spend more time in complete jams, whereas in our model cars tend to slow down rather than stop completely.

This type of fuel economy indicator (comparing average number of acceleration cycles per car among simulations with a similar average car velocity) can be seen more clearly in Figure 5. We can see that for very low or very high average car velocities (resp. very high or very low ρ values), the two models have fairly similar fuel consumption characteristics, but in the middle range our slow-to-stop model causes cars to accelerate much more often. We think this is probably more realistic, since in the BJH model cars are mostly either at a complete stop, or are going at maximum speed (as in Figure 2).

4. Simulation using Alternate Car Arrival Rules

Up to this point our simulations have been carried out using a 'circular' road - that is, there are a fixed number of cells at any given time which are in the nonempty state in a one-dimensional cellular automaton of fixed size and periodic boundary conditions. One possible drawback of this approach is that periodic trends such as the unending backward propagation of a traffic jam (see Figure 2) may not occur in reality since roads are typically not circular. If we want to simulate a stretch of single-lane road more accurately, we should allow cars to enter the road (with arrival times following some random distribution) and leave it once they have travelled through it. In this section, we address some alternative methods for simulating the movement of cars on a stretch of single-lane road with a particular traffic density, and repeat some of our experiments from the previous section.

We want cars to enter the road at a rate equal to the traffic density, and we want these arrivals to be random. One possibly intuitive way to achieve this type of behaviour in continuous time is by having the cars arrive at the start of the stretch of road as a Poisson process with rate λ, where λ is calibrated to achieve a desired average traffic density. Since a Poisson process has Exponential(λ) interarrival times, it is easy to randomly generate the arrival time of the next car. However, we need the arrivals to happen at discrete time steps. We propose the following scheme to achieve this goal.

We precompute enough pseudorandom samples of an Exponential(λ) random variable in order to obtain arrival times for the entire simulation, then round each to the nearest time step. Note that these pseudorandom samples are readily computed as $F^{-1}(U)$, where U is a Uniform(0,1) random variable and F is the distribution function of an Exponential(λ) random variable. A queue can be used to 'store' cars which arrive on the same time step until there is space for them to enter the road. The question of how to transfer cars from the arrival queue to the road (i.e. where to place them, and what initial velocity to give them) is an important one. We explore two different schemes for doing this transfer.

The most simple scheme we conceived is the following. If at time t the queue is nonempty and there is not a car in the first cell of the road, then at time $t+1$ the car at the head of the queue will enter the first cell, and shall have velocity v, where v is the minimum of v_{max} and the number of empty cells between the first cell and the next nonempty cell. In Figure 6, we can see the results of a short simulation using this method. It is clear that the simulation quickly falls into a pattern where the first car is added with velocity 0. The queue never has a chance to empty for reasonable rate λ, and we do not see any interesting behaviour on the road because there are not enough cars on it. If we increase λ we simply have the same problem, and if we decrease it then the road does not have a large enough density to be interesting.

In an attempt to avoid the problem of having the queue fill up while cars enter the road with zero velocity, we can try to relax the restriction that only one car may enter the road per time step. At time step t, if the nearest car is d spaces away from the start of the stretch of road, we shall add a new car to the road at time step $t+1$ at a position which is the minimum of $d-1$ and v_{max} (either the space right behind the car, or the space v_{max} cells away from the start). The reason for the latter restriction is to prevent cars from entering the road at a position they could not possibly have reached in one time step. The car is given an initial velocity equal to the minimum of v_{max} and the number of empty cells between it and the next car. This process of adding cars is repeated as many times as possible, as long as the queue is still nonempty, at each time step. In Figure 7, we can see the results of a short simulation using

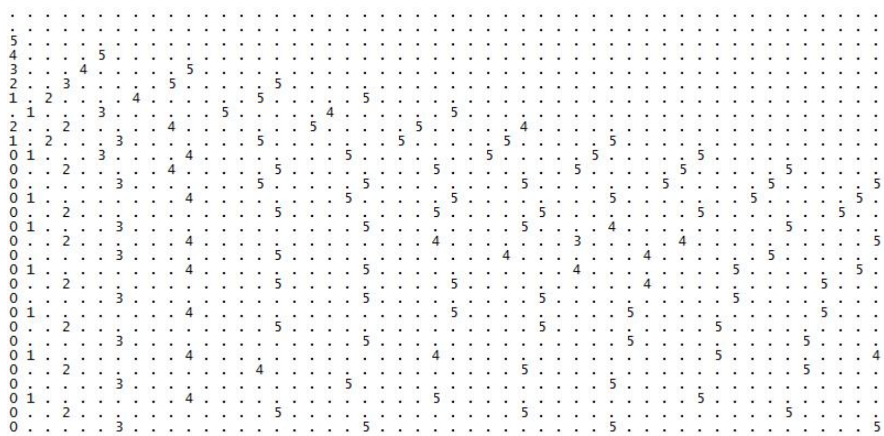

Figure 6: A small example of cars following the 'slow-to-stop' model, with Poisson arrivals instead of a circular road. The dots refer to empty cells, and the numbers represent the velocities of cars. Here the realized density $\rho = 0.10$, $p_{fault} = 0.1$, and $p_{slow} = 0.5$. The rate of the Poisson process $\lambda = 0.8$. Cars drive from left to right, and time steps go from top to bottom.

this method, and again we have the same problem of the simulation falling into a pattern started by adding a car to the first cell with velocity 0. Essentially, whenever we have several cars arriving on the same time step, a jam occurs, and only has a very short distance to propagate backwards before the beginning of the road is reached and the queue begins to fill up. The queue never has a chance to empty for reasonable rate λ, and we do not see any interesting behaviour on the road because there are not enough cars on it. There is the same issue with increasing or decreasing λ - it is not possible to see any interesting or realistic behaviour because we cannot achieve high traffic densities.

It seems that since there are problems with converting the continuous time Poisson process into a discrete arrival model, it makes sense to try a discrete process. In particular, we will investigate using a Bernoulli process to determine whether or not to add a car to the road in position 0, the leftmost cell in the array, at each time step (whenever position 0 is empty). Essentially a Bernoulli process is a sequence of coin flips, possibly with a biased coin. At each time step, the outcome of a flip determines whether or not a car is added to the road. We let this probability that a car is added be p_{add}. The initial velocity of the car is determined by a vector of probabilities, one for each possible starting velocity:

$$\{p_{vel_i}\}_{i=0}^{v_{max}}$$

where

$$\sum_{i=0}^{v_{max}} p_{vel_i} = 1.$$

These probabilities define a partition of the interval [0,1], so the car's velocity

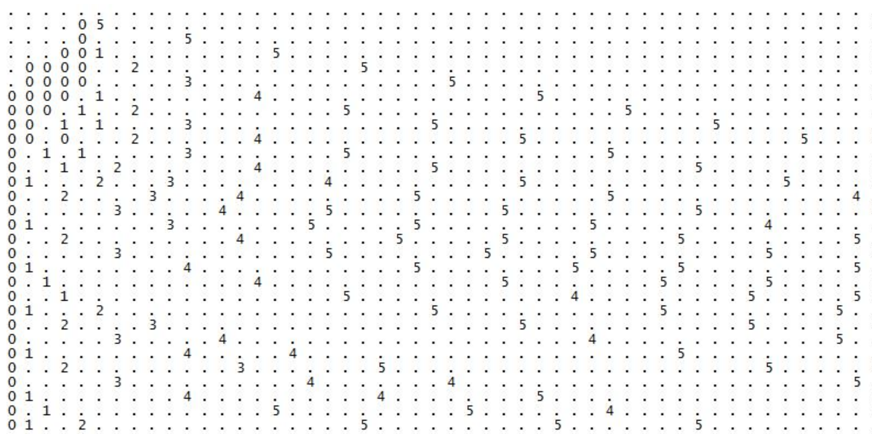

Figure 7: Another small example of cars following the 'slow-to-stop' model, with Poisson arrivals instead of a circular road. Multiple cars are allowed to enter the road on the same time step. The dots refer to empty cells, and the numbers represent the velocities of cars. Here the realized density $\rho = 0.10$, $p_{fault} = 0.1$, and $p_{slow} = 0.5$. The rate of the Poisson process $\lambda = 0.5$. Cars drive from left to right, and time steps go from top to bottom.

is determined by sampling a Uniform(0,1) random variable and recording in which section of the partition the sample lies. If the distance to the next car is less than v_{max}, then the velocity is selected from among the velocities less than v_{max}. In Figure 8 we show a zoomed out view of a typical simulation using Bernoulli process arrivals. This simulation realized an 'interesting' average traffic density of 16.3%. It is clear that traffic jams occurred, although most seem to form fairly close to the start of the road. A downside of this new approach is that we cannot see how these particular jams grow or dissipate since they quickly propagate back to the start of the road. However, here it seems that the Bernoulli process arrival method for simulating traffic using CA has some potential validity. Further investigation is certainly needed. For example, we did not try very many configurations of the p_{vel} vector, and it is not well understood how this would affect the simulation, or which configurations would most closely simulate real traffic.

In Figure 9, we give the fundamental diagram for our model using Bernoulli process arrivals. Using this method we cannot achieve densities higher than about 17%, but this should be sufficient for some applications.

In this section we experimented with some alternate methods for simulating a single-lane highway using CA. Instead of simply populating the cellular array with cars and applying periodic boundary conditions, we tried to add new cars to the start of the road using random arrival times. At first it was hypothesized that cars arriving as a Poisson Process may provide a more realistic simulation than the standard method which essentially assumes a circular road, but after some testing we found that the specifics of how cars are added to the road (i.e. at what initial position and velocity, and how many at each time step) are

Figure 8: A 'zoomed-out' view of a larger simulation of our 'slow-to-stop' model using Bernoulli process arrivals. Black dots refer to cars, while white space is empty road. Here $v_{max} = 5$, $p_{fault} = 0.1$, $p_{slow} = 0.5$, $p_{add} = 0.8$, $p_{vel_2} = p_{vel_3} = 0.25$, and $p_{vel_4} = 0.5$. The road is 1000 cells wide and the last 1000 evolutions out of 1400 are shown so that the road has time to populate with cars, since it starts out empty. The realized average car density for the simulation shown is 16.3%. Cars drive from left to right, and time 0 is at the top.

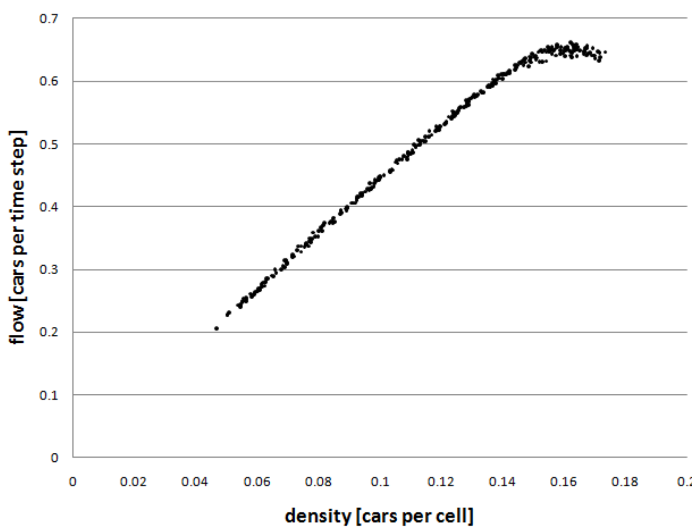

Figure 9: The 'fundamental diagram' for our model using Bernoulli process arrivals. Each point represents the result from the latter 1000 iterations out of 1400 iterations on a road of length 1000. The probability of adding a car on a given time step, p_{add}, was set from 0.2 to 1 in intervals of 0.02, and ten simulations were performed for each p_{add}. $p_{fault} = 0.1$, $p_{slow} = 0.5$, $p_{vel_2} = p_{vel_3} = 0.25$, and $p_{vel_4} = 0.5$.

difficult to determine so that roads with high traffic densities can be simulated. Not only has this Poisson process arrival model shown itself to be difficult to calibrate and discretize, but it may also require a significant amount of work to actually program onto a Field Programmable Gate Array (FPGA) [27] device for fast CA simulation. Structures such as the queue and the list of arrival times were easy to implement in our high-level simulator, but are not so easily fabricated in CA-specific parallel hardware. Our second approach involved using a discrete (Bernoulli) process to determine whether or not to add a car with random velocity to the road. This appeared to be a more useful model since it was possible to obtain high enough traffic densities so that 'jamming waves' occurred.

5. Conclusion and Future Work

We have presented a modification of the well-known BJH model for single lane car traffic, designed to simulate the braking behaviour of cars more correctly. We have provided the fundamental diagram for our model as well as some supplemental simulation results, and have recorded a statistic proportional to fuel economy and the amount of pollution generated. We have also investigated some different ways to simulate adding new cars to a stretch of road instead of using CA with circular boundary conditions, and had some interesting results using Bernoulli process arrivals. The simulator[1] we have constructed is fairly simple to understand and modify, and could be a useful tool for future researchers to incorporate into their work in this area. It performs an iteration of cars moving on a road in $O(L)$ time, where L is the length of the road - of course, a parallel implementation could do this in constant time.

Comparison with empirical traffic data is needed in order to tell if our model provides realistic figures for fuel economy and general driving and jamming characteristics. We believe it may be interesting to compare traffic data from North American traffic networks, since there currently appears to be a shortage of this type of comparison in the literature.

References

[1] K. Nagel, M. Schreckenberg, A cellular automaton model for freeway traffic, Journal de Physique I 2 (12) (1992) 2221–2229.

[2] K. Nagel, Particle hopping models and traffic flow theory, Phys. Rev. E 3 (5) (1996) 4655–4672.

[3] M. Sasvári, J. Kertész, Cellular automata models of single-lane traffic, Phys. Rev. E 56 (4) (1997) 4104–4110. `doi:10.1103/PhysRevE.56.4104`.

[1] Email the first author for a working copy.

[4] A. Schadschneider, The nagel-schreckenberg model revisited, Eur. Phys. J. B 10 (1999) 573–582.

[5] A. Schadschneider, M. Schreckenberg, Cellular automaton models and traffic flow, J. Phys. A Math. Gen. 26 (1993) 679–683.

[6] P. Wagner, Traffic simulations using cellular automata: Comparison with reality, in: Traffic and Granular Flow, World Scientific, 1996.

[7] S. C. Benjamin, N. F. Johnson, P. M. Hui, Cellular automata models of traffic flow along a highway containing a junction, Journal of Physics A: Mathematical and General 29 (12) (1996) 3119–3127.

[8] M. Rickert, K. Nagel, M. Schreckenberg, A. Latour, Two lane traffic simulations using cellular automata, Physica A: Statistical and Theoretical Physics 231 (4) (1996) 534–550.

[9] P. Wagner, K. Nagel, D. E. Wolf, Realistic multi-lane traffic rules for cellular automata, Physica A: Statistical and Theoretical Physics 234 (3-4) (1997) 687–698.

[10] W. Knospe, L. Santen, A. Schadschneider, M. Schreckenberg, Disorder effects in cellular automata for two-lane traffic, Physica A: Statistical and Theoretical Physics 265 (3-4) (1999) 614–633.

[11] K. Nagel, D. E. Wolf, P. Wagner, P. Simon, Two-lane traffic rules for cellular automata: A systematic approach, Phys. Rev. E 58 (2) (1998) 1425–1437. `doi:10.1103/PhysRevE.58.1425`.

[12] J. Esser, M. Schreckenberg, Microscopic simulation of urban traffic based on cellular automata, International Journal of Modern Physics C 8 (5) (1997) 1025–1036.

[13] P. M. Simon, K. Nagel, Simplified cellular automaton model for city traffic, Phys. Rev. E 58 (2) (1998) 1286–1295. `doi:10.1103/PhysRevE.58.1286`.

[14] D. Chowdhury, A. Schadschneider, Self-organization of traffic jams in cities: Effects of stochastic dynamics and signal periods, Phys. Rev. E 59 (2) (1999) R1311–R1314. `doi:10.1103/PhysRevE.59.R1311`.

[15] E. Brockfeld, R. Barlovic, A. Schadschneider, M. Schreckenberg, Optimizing traffic lights in a cellular automaton model for city traffic, Phys. Rev. E 64 (5) (2001) 056132. `doi:10.1103/PhysRevE.64.056132`.

[16] R. Barlovic, E. Brockfeld, A. Schadschneider, M. Schreckenberg, Optimal traffic states in a cellular automaton model for city traffic, in: Traffic and Granular Flow '01, Springer-Verlag, 2003.

[17] P. Deo, H. J. Ruskin, Simulation of heterogeneous motorised traffic at a signalised intersection, in: ACRI, 2006, pp. 522–531.

[18] I. Spyropoulou, Modelling a signal controlled traffic stream using cellular automata, Transportation Research Part C: Emerging Technologies 15 (3) (2007) 175 – 190.

[19] S. Sakai, N. Nishinari, S. Iida, A new stochastic cellular automata model and its jamming phase transition, Journal of Physics A: Mathematical and General 16 (4) (2006) 15327–15339.

[20] H. Emmerich, E. Rank, An improved cellular automaton model for traffic flow simulation, Physica A: Statistical and Theoretical Physics 234 (3-4) (1997) 676–686.

[21] W. Knospe, L. Santen, A. Schadschneider, M. Schreckenberg, Towards a realistic microscopic description of highway traffic, Journal of Physics A: Mathematical and General 33 (2000) L477–L485(1).

[22] D. Makowiec, W. Miklaszewski, Nagel-schreckenberg model of traffic - study of diversity of car rules, in: International Conference on Computational Science (3), 2006, pp. 256–263.

[23] C. Mallikarjuna, K. Rao, Identification of a suitable cellular automata model for mixed traffic, Journal of East Asian Society for Transportation Studies 7 (2007) 2454–2468.

[24] W. Lan, C. Chang, Motorbike's moving behavior in mixed traffic: Particle hopping model with cellular automata, Journal of East Asian Society for Transportation Studies 5 (2003) 23–27.

[25] G. H. Bham, R. F. Benekohal, A high fidelity traffic simulation model based on cellular automata and car-following concepts, Transportation Research Part C: Emerging Technologies 12 (1) (2004) 1–32.

[26] P. Chakroborty, A. K. Maurya, Microscopic analysis of cellular automata based traffic flow models and an improved model, Transport Reviews 28 (2008) 717–734(18).

[27] C. Maxfield, The Design Warrior's Guide to FPGAs, Academic Press, Inc., Orlando, FL, USA, 2004.