Traffic Flow – Investigating traffic flow using a cellular automaton inspired model

Josh Swift – Student No. 4287866

Abstract

A cellular automata model can be used with a simple algorithm controlling the acceleration and deceleration of vehicles along a road to produce a model of traffic flow. Despite the simplicity of the model, large scale patterns such as backwards-propagating traffic jams are emergent. A variation of this model produced in MATLAB was used to investigate the effect of changing parameters such as the density of cars or number of lanes on the flow rate of traffic along a road. The most basic system, which has one lane and a maximum velocity of one, was found to undergo a phase transition at a density of 0.08 cars per site. This phase transition was found to occur only for systems with a maximum velocity greater than one.

Introduction

A cellular automaton (CA) is a regular grid of cells with a finite number of states per cell. These cells evolve in time according to a simple set of rules. This set of rules can cause the cellular automaton to display emergent behaviours that would not be expected of such a simple system. Cellular automata are commonly used to produce and analyse systems such as traffic flow and load bearing. [1] Cellular automata are an effective model for physical systems like this because these systems are quantised and ruled by local interactions. In the case of traffic flow this interaction is cars braking or overtaking due to the car directly in front of them.

The CA model of traffic flow is described in Nagel and Schreckenberg's *A cellular automaton model for freeway traffic* [2]. Their model, which will henceforth be labelled as the N&S model, is described as a 'one-dimensional array of *L* sites with open or periodic boundary conditions'. Open or periodic boundary conditions refers to whether the end of the road is open, in which case cars leave the road and new cars are generated at site 1, or periodic, in which case cars effectively move in a circle. As mentioned previously, a CA cell has a finite number of states. In the N&S model these states are whether a cell is occupied by a car, and if it is, what velocity this car has. This is done by labelling empty sites with a zero, then counting from one representing velocity zero to a value of v_{max}+1 for cars with velocity v_{max}. This representation would be the one used in a program, but to represent it visually empty sites can be shown by dots and occupied sites represented with their velocity. The one lane system with a maximum velocity of one can be represented by an array of zeros and ones. Similarities can be drawn between this model and the Ising spin model, though the traffic model's equivalent 'spin' (whether a site is occupied or unoccupied) only depends on the sites in front for occupied sites and the site behind for unoccupied sites. This system uses the simplest CA model, as each cell only has two possible states and it is a one-dimensional array of sites. An example of this one lane configuration is shown in Figure 1 below.

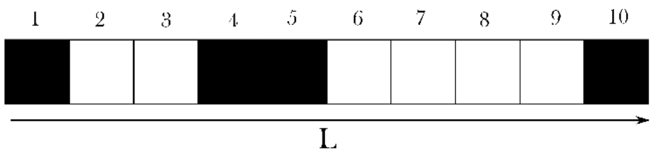

Figure 1: An example of the simple one lane array with a road length of 10 and 4 cars

The configuration evolves according to the following set of rules for each car:

1. Acceleration: if the velocity v of a vehicle is lower than v_{max} and if the distance to the next car ahead if larger than v + 1, the speed is increased by 1 (v = v + 1).

2. Deceleration: if a vehicle at site i sees the next vehicle at site i + j (with j ≤ v), it reduces its speed to j - 1 (v = j – 1).

3. Randomisation: with probability p, the velocity of each vehicle (if greater than zero) is decreased by one (v = v – 1).

4. Car motion: each vehicle is advanced v sites.

v=velocity of car i=position of site
j=position of comparison site
v_{max}=maximum velocity

In this case, it is the third stage, the randomised braking that causes emergent behaviour in the system for velocities greater than one. At intermediate densities, where the cars are typically within around v_{max} sites. Usually, a lead car will randomly decelerate, and this will cause a chain of cars behind it to brake in reaction to this. This is the beginning of a traffic jam. Cars behind the jam will approach the stationary traffic at close to maximum velocity and be forced to stop just behind it. Simultaneously, if there is space, the car at the front of the jam will begin to accelerate away. This process repeats, and the net effect is the backwards propagation of the traffic jam. The jam is typically only found to dissipate for lower densities as the rate at which cars leave the jam must be higher than the rate at which they join. Conversely, this means that for higher densities this jam is a semi-permanent feature of the periodic road, since it can propagate backwards endlessly.

The Model

The first stage is to generate the initial configuration of the road. This is done by randomly permuting a list of numbers ranging from 1 to L where L is the road length. A number is then drawn from the list for each car on the road, starting from the first number in the list and proceeding in order. This 1xN array is then concatenated with a 1xN array of starting velocities, which were generally all zero, and a 1xN array of starting lane. For configurations in which there was only one lane, this last array was not used, leaving the initial setup to be described by a 2xN array. N is the number of cars on the road.

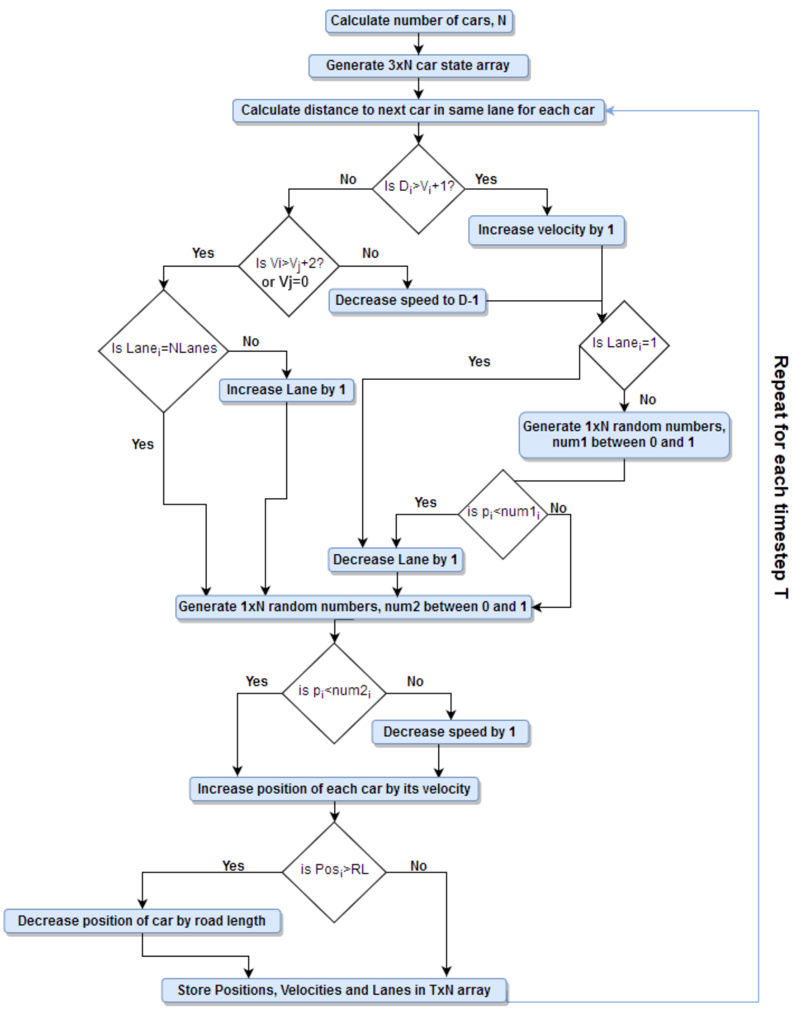

Figure 2: Flowchart of the traffic flow simulation process

Rather than using the N&S model described above, the array that was used in the program only stored the positions, velocities and lane for each car in a 3xN matrix (2xN for one lane). The program uses Boolean logic on the contents of each column to calculate the distance between the cars, which is used to process in parallel the four stages outlined above. The sequence of processes is shown in Figure 2.

For configurations with multiple lanes, the process of determining whether to switch lanes is combined with the braking stage. This is because in this model, cars overtake when they approach a car from behind with a velocity two greater than that of the lead car, or if the car in front has a velocity of zero. This overtaking logic seems very narrow, but it was found to cause emergent behaviour that appears close to a realistic traffic model.

Because the initial positions are randomly selected, the cars can be initially arranged in positions that cause traffic where there otherwise would be none. This is most important for low density configurations. To resolve this, the above simulation was run for a number of timesteps equal to the road length. This allowed the system to reach a steady state, meaning that systems in which traffic is rare will tend to not show traffic in the second iteration of the simulation. For larger road lengths (above 500), the system was not found to stabilise noticeably more beyond half the road length. Because of this, the initialising time period was set to 1000 to save computing time.

The second run-through begins from where the first ended and runs for T timesteps. The end array in which the positions, velocities and lanes of each car are stored is updated once per timestep, and it is used at the end to calculate the time averaged of each car. This average velocity is multiplied by the density to give the flow rate, which is the primary metric for measuring how effectively the road is being used.

Another detail of note is that the periodic nature of the road means that when the distance between cars is calculated, the cars at the 'front' and 'rear' of the road also must calculate their distances. This is done by sorting the car array by the contents of the position row. For switching lanes, cars in the target lane (next lane for overtaking and previous lane for returning) are selected and the position and velocity of the car that is moving is added to this list as if it were in this lane. The cars with the highest and lowest position in this list then have the road length L removed or added to it respectively. This allows the car at the front and to the rear to interact with one another as they should.

Several methods were used for visualising the data from the simulation. One method for viewing the data was a space-time plot. This method very clearly shows the backwards propagation of the traffic jams, but it requires colour coding to show the velocities of cars properly. The space time diagram also cannot deal with multiple lanes on the same diagram without significant loss of data due to overlapping. The more aesthetic plotting method is an animated plot of the road. The velocities of cars and the forming of jams is immediately apparent, but as it is a frame by frame plot of the simulation, it is poor at showing long term behaviours of the system. The animation plot appears identical to the corresponding row of space time plot for any point in time.

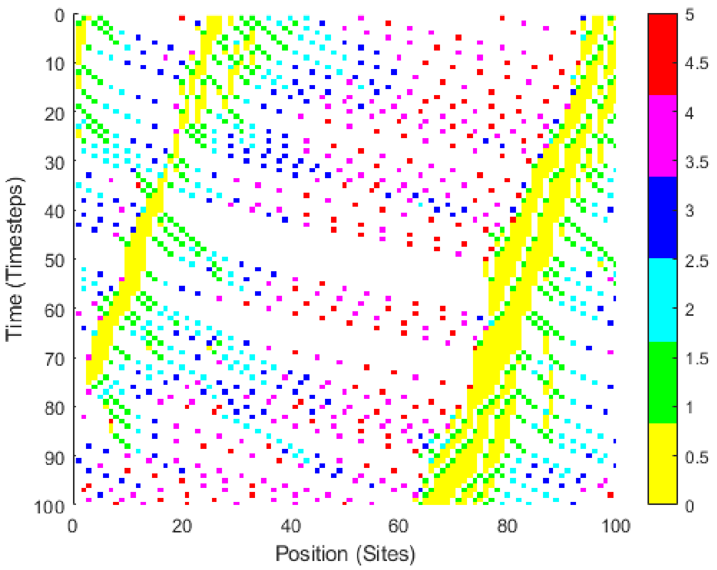

Figure 3 is an example of the colour coded space time plot. It has a density high enough that traffic is

Figure 3: An example of a space-time plot of a one lane periodic road with ρ=0.2. Note the dense yellow regions which are the traffic jams.

essentially a permanent feature along the road. The colour scale on the right shows the colours that correspond to car velocities. Cars are shown approaching the jam with a velocity of either 4 or 5, and they remain in the jam until all the cars in front have pulled away.

Figure 4: A snapshot of the animated plot of a five-lane road with ρ=0.7.

The animated plot corresponds to a snapshot at one point in time. The animated plot is much more effective for displaying multiple lanes, as can be seen in Figure 4. Despite the high density, the cars do not seem to be making full usage of all five lanes due to the overtaking rules. The second and third lanes are mostly vacant, and the first lane is approaching saturation. The fourth and fifth lane are empty due to the low usage of the third lane. Note that the animated plot does not use the same colour scheme as the space-time plot. In this plot, the darkest blue corresponds to the stationary cars and yellow represents the cars moving with maximum velocity.

The system shows sensitivity to all variables other than the road length, since the number of cars is controlled by the density, meaning that varying the road length alone does not affect the density. Note that when the density is referred to, this refers to the number of cars per site as a length rather than an area. Because of this, for roads with multiple lanes, a density of one does not correspond to total occupancy of the road. The program can change the maximum velocity, probability of cars randomly braking and probability of returning to the slower lane on a per car basis. This means that the effect of speeding cars on the flow rate could be performed, though this was not part of the investigation. As explained above, the road length is used to run an initial simulation for the transient state. To save computation time, and aid in the visualisation process, this means that the road length can be set to a reasonable low value. This value was usually set to a value between 100 and 1000.

The first investigation made with the model was the effect of changing the density along a one lane road. This data and the graph produced from it was to be compared with the results in N&S [2]. The road length was set to 250, with a low-resolution time period of 100 and a high-resolution of 10000 timesteps. The number of different densities for the low-resolution trial was 250 and for high resolution the number was 100. This was done to gain a vague idea of the finer shape of the density vs. flow rate graph while also gaining more precise data intermittently. The density values for both tests were equally spaced. A graph was produced from the data received from the simulation.

The linear portion of this graph, from a density of 0 to 0.08 cars per site, is caused by the cars having very little interaction. This is the emergence of a steady state form, with cars spaced so far apart that interactions are infrequent. This portion of the graph can be modelled by the equation:

$$Q = \rho \times v_{max} \tag{1}$$

where Q is the flow rate, ρ is the density of cars, and v_{max} is the maximum velocity along the road. The downwards sloping portion of the graph is caused by an effective 'over-saturation' of the lane. The graphs in Figure 5 are the same shape that was found in the N&S study, so the two models produce the same result.

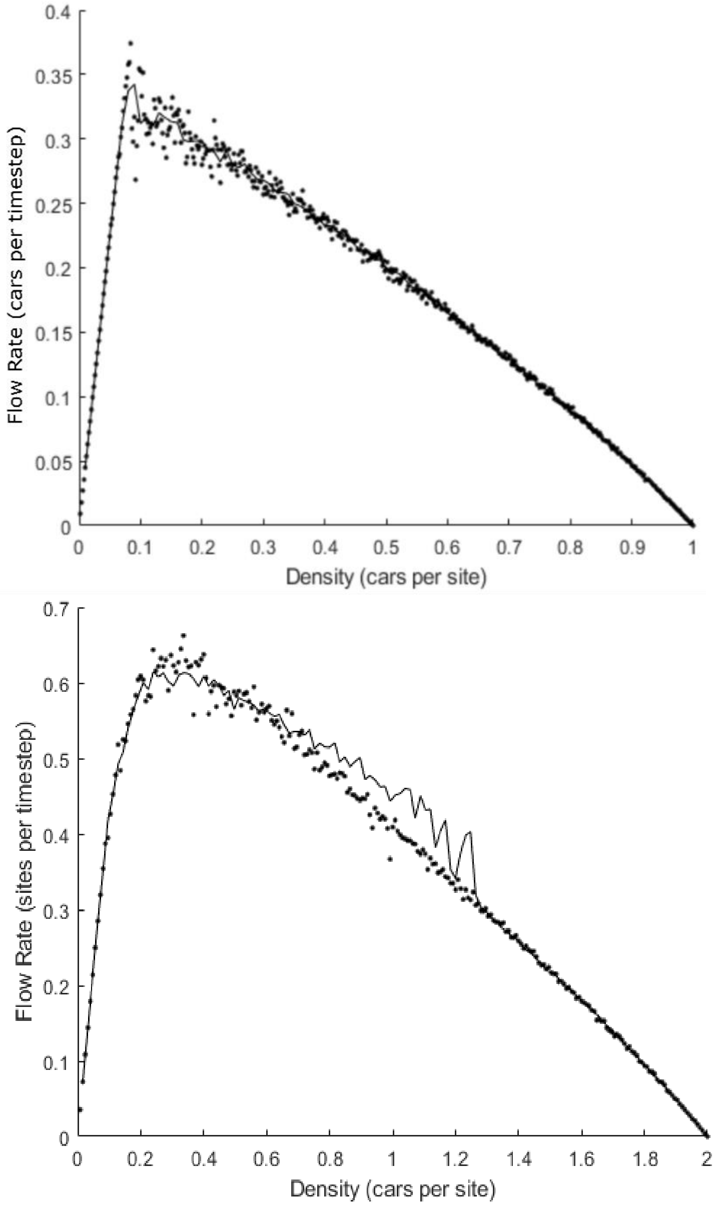

Figure 5: The effect of varying the density of cars along both a one-lane and two-lane road. The lines correspond to a time period of 10000 (1000 for the second figure) while dots are a period of 100 timesteps.

The next stage was to compare the scales of the model with real traffic in order to produce quantitative results rather than timesteps and sites. First off, the site width is set to about 8 metres which is the rough length occupied by each car. The maximum velocity of 5 sites per timestep is then equivalent to 40 metres per timestep. If this is equated to a rough UK speed limit of 80 mph (~35 metres per second) then a timestep turns out to be equal to

$$40\,\frac{metres}{timestep} \times \frac{1}{35}\,\frac{seconds}{metre} = 1.14\,\frac{seconds}{timestep}. \tag{2}$$

Therefore, the values predicted by this model should approximate the real-life values. If they do not, then there is an issue with the model. The data that was used to produce Figure 5 was then compared with real traffic data from the N&S paper.[2] These three figures all

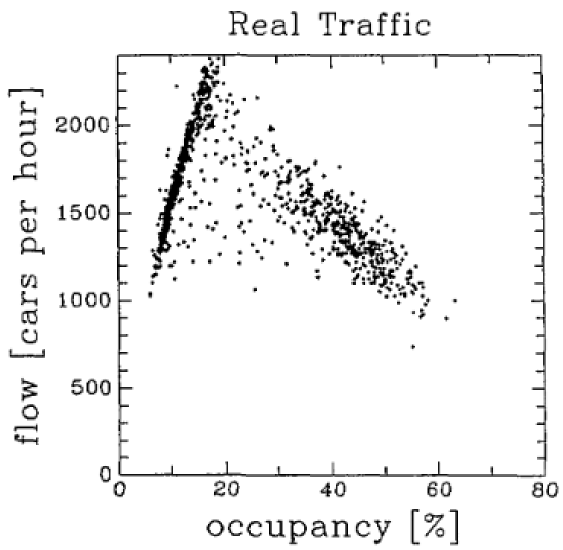

Figure 6: Flow rate of real traffic for varying occupancy ratios. Original source: [2,3]

have the same general form, with a linearly increasing region of unrestrained flow, and a decreasing region where jams begin to form. The peak of the real traffic flow rate is seen to occur for a higher density of 0.2, which matches the critical density of 0.2 in the lower graph. From the result of Equation 2, the vertical axis of Figure 5 can be scaled into units of cars per hour to match Figure 6. This scaling is not dependent on the density, so the shape of Figure 5 is not affected. The new flow rate peaks for the simulated data are calculated to be:

$$0.37 \times 1.14 \times 60 \times 60 \approx 1520\ cars\ per\ hour. \tag{3}$$

$$0.61 \times 1.14 \times 60 \times 60 \approx 2500\ cars\ per\ hour. \tag{4}$$

From these two flowrate peaks the second graph, the two-lane model, is an almost perfect match for the trend in Figure 6. This is likely due to roads with multiple lanes being more common in the study since multiple lane roads such as motorways are a much more reliable source of information than country roads.

Investigation

One obvious way to use the model was to change the parameters and observe the effect on the flow rate along the road. One such investigation was done on the effect of ‘smart cars’ on the flow rate. In this model, smart cars do not randomly brake like human-driven cars, so they are essentially perfect drivers. This means that a smart-driven car should never be the direct cause of a traffic jam, so the flow rate would be expected to increase the greater the proportion of smart cars to total number of cars on the road.

In this model, removing the randomised braking from specific cars is trivial as the probability for each car is stored in a 1xN array. Therefore, the investigation can be performed simply by setting a varying number of these cells to zero. This was done on road configurations

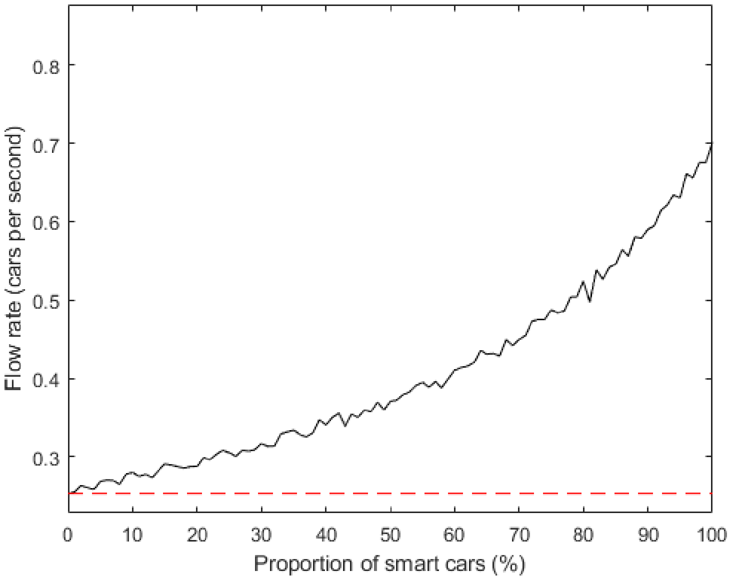

Figure 7: Graph showing the effect of replacing human-driven cars with smart cars on the flow rate. The road has one lane and a car density of 0.2 cars per site

just beyond saturation, the point at which it is not possible for all cars to move at the maximum velocity. This system is of interest because systems close to saturation are the most prone to the removal of randomised braking.

Figure 7 clearly shows a large increase in the flow rate due to the addition of these smart cars. When approximately 70% of the cars on the road are ‘smart’, the flow rate along it has doubled. The red dashed line shows the flow rate with no smart cars.

The difference between normal cars and smart cars is more pronounced because a substantial portion of the reduction in flow rate is caused by the random braking of the human-driven cars. Replacing cars with smart cars has a small effect on the formation of jams above a density of $1/v_{max}$, because at this critical density cars are forced to brake at some point. Only jams that would

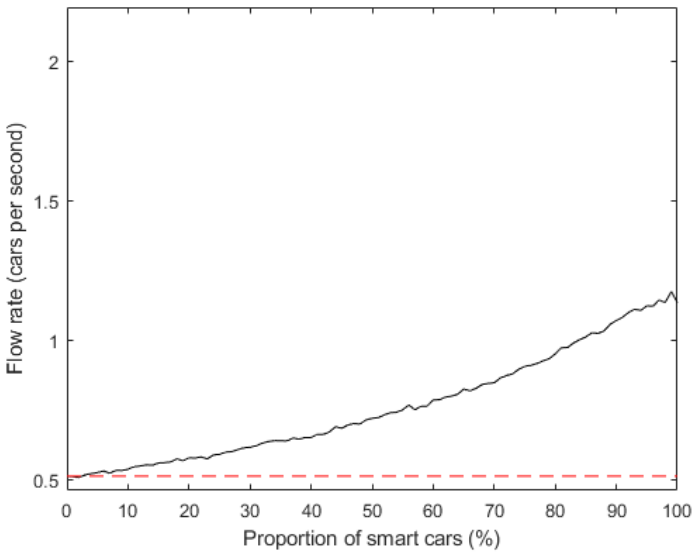

Figure 8: Graph showing the effect of replacing human-driven cars with smart cars on the flow rate. The road has two lanes and a car density of 0.5 cars per site.

form because of a car randomly braking are removed by this car becoming a smart car.

Next, a trial was run on a two-lane system with a density of 0.5 which corresponds to an ideal spacing of 4 sites per car. The system shown in Figure 8 behaves very similar to the one plotted in Figure 7. The flow rate of this system doubles at 90% smart cars. Systems with densities lower than this saturation density show large improvements as smart cars are introduced to the system. This result is of notable importance for real life

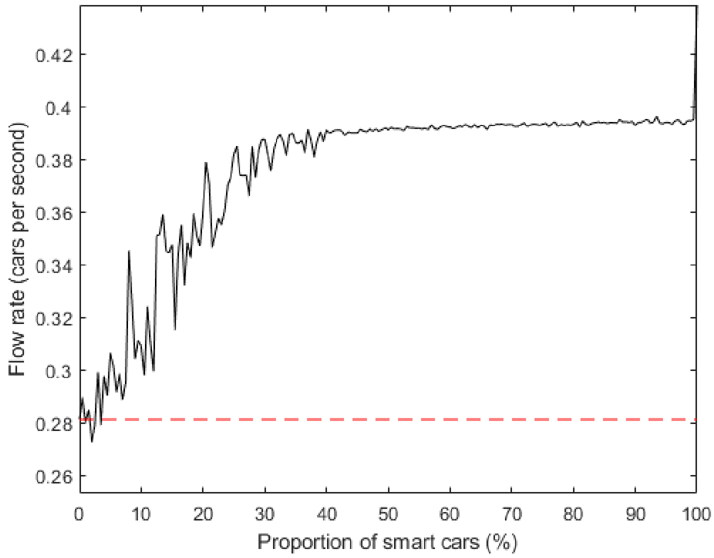

Figure 9: A one lane system with a density of 0.1 cars per site.

applications, as they often improve flow rate dramatically with only a small number of cars being added.

The flow rate in Figure 9 can be seen to increase by around 33% or so with only 25% of cars being smart cars. This is relevant with self-driving cars becoming more popular in real life, as we can infer from this model that traffic flow rate will increase as more people begin owning self-driving cars. Obviously, this change in the proportion of self-driven cars will be small at first, which is why this large flow rate increase with only a quarter of the cars being smart is notable. Not much data on the effect that autonomous cars have on real traffic, but there have been other studies with similar models. A study using reinforcement learning [4] concluded that, although the autonomous vehicles were able to improve the flow rate substantially for several road configurations, they were not able to arrange themselves into the most optimal configuration.

Summary

The model was found to be a reasonably realistic model of traffic flow. Simulations of traffic flow were run for varying numbers of lanes and car densities, with the initial positions along the road randomised. The evolution of the cars positions was governed by a simple set of rules that were found to produce emergent features like traffic jams.

The one lane configuration was found to have a phase transition at a density of 0.08 ± 0.01 cars per site, at which point further increases in density cause the flowrate along the road to decrease rather than increase. The two-lane road also behaved similarly, with a transition at around 0.20 ± 0.01 cars per site. It also, unsurprisingly, had a peak flowrate close to double that of the single lane. The produced flowrate vs. density graph was remarkably close in shape to the data measured in real traffic. These two graphs, the simulated and the real plots, were used to match the arbitrary timesteps and site lengths to more appropriate metres and seconds/hours. The ratios relating these quantities was then used to change the quantities produced by the model into real quantities.

For low densities this system was found to have a steady state configuration which appeared as cars being equally spaced out with little interaction between them. For higher densities the converse was true, most of the cars tended to be stuck in jams at any given point in time. These traffic jams were essential permanent features of the road, and they propagated backwards along the road.

Systems with higher densities were found to either not have discernible steady state solutions or take such a long time to reach them that it was unfeasible to simulate this. Because of this, a large road length of 250-1000 was used instead. The higher density systems were not a particularly useful system to model as cars spent a significant amount of their time in traffic jams. The low-density systems are of much more interest as the behaviour of individual cars is much more relevant.

Further possible extensions that could be made include making the model capable of handling features like traffic lights and junctions to analyse their effect on traffic flow. The current structure of the model also does not currently facilitate a non-periodic road without a complete rewriting of the functions that calculate the distances between cars.

[1] *Ada Yuen and Robin Kay, Applications of Cellular Automata* (2009)

[2] *Kai Nagel and Michael Schreckenberg, A cellular automaton model for freeway traffic, J. Phys. I France 2,* 2221 (1992)

[3] *Hall F.L., Brian L.A., Gunter M.A., Transpn. Res. A 20A* 3,197 (1986)

[4] *E.Vinitsky et.al., Benchmarks for reinforcement learning in mixed-autonomy traffic* (Oct 2018