COMPUTATIONAL METHODS AND TOOLS

Motorway Insertion Model

Faculty of Environmental Sciences,
EPFL, Lausanne

Autumn 2024

Written by:

Adrien Turner

Contents

DEVIATIONS

In the proposal, I stated that "the broad aim of this project [was] to understand the overall effect of adding a lane to a motorway". The broad outline has not varied however it is true that the focus has changed and now tilts towards the specific effect of insertions. The presence of another lane is implicit throughout the study however not much emphasis is put on what effects of adding one would be. Instead we try to understand specifically how and when insertions effect traffic. Also as stated in the proposal we won't actively code a second lane, the idea is to allow a new car to appear in our lane if certain criteria are met. This way we imitate what would happen had we coded a slow lane from where cars would transition into the fast one, all the while maintaining a one-dimensional model.

1 Introduction

The idea of this model is to simulate 70 km of a motorway (120km/h legal speed limit) over 5 minutes and understand the effects of lane changes. More specifically we wish to understand the consequences of insertions on traffic flow. To do so we will model the fast lane of a motorway and see what happens when cars insert. This model is a cellular automaton; a one dimensional grid where each cell can either be occupied by a car going at a certain speed or empty. A set of rules are put in place to dictate the movement of cars. This kind of modeling differs from other types of models (such differential equation based ones) in its mathematic simplicity. Indeed, each 'actor' in the model follows basic rules and evolves without the need for extensive and complex computation. The idea is to allow a new car to appear in our lane if certain criteria are met. Focusing on the fast lane of our motorway, we will observe the effects of insertions in different traffic contexts.

2 The model

2.1 Model structure and basic vehicle movement

The starting point of this model lies in the work performed by Kai Nagel and Michael Schreckenberg in their 1992 paper 'A cellular automaton model for freeway traffic'. The idea is to mimic the lane of a motorway as a one dimensional array of a given length. Each cell (of the array) is either empty or occupied by a car. At any given time t a car in position i goes at speed v, i is simply its coordinate in the array and the speed is measured in cells advanced per time step. For example if a car is in cell 24 with speed 4, at the next iteration (or t+1) it will be in position 28. Computationally, at time t the value of the 24th cell is 4 and more generally if there is a car at position i, the value of the i-th cell is the car's speed. With each new time step a car advances by v spaces. Note that v is never greater than vmax = 5 cells/time step, the maximum authorised speed on the road.

let road be a 1D array

$$\text{road}[i] = v \quad \text{where} \quad 1 \leq i \leq \text{length of the road}, \quad v = \begin{cases} -1 & \text{if there is no car,} \\ 0 \leq v \leq v_{\max} & \text{the car's speed if there is one.} \end{cases}$$

This structure neatly links speed, position and time allowing for smooth and elegant calculations. We are at liberty to choose the length of our road, the number of time steps, the length of each cell and the duration. For simplicity, we choose to fix one cell as 7 meters in the physical world and one time step as 1 second. This way if we run a simulation with a road of length 10000 for 300 time steps it equates to observing 70 km of a road for 5 minutes. From these initial dimension condition, we can also derive a what the maximum speed of the road can be in km/h. As things stand vmax is 5 cells per time step. Using 1 cell = 7 m and 1 time step = 1 s,

we get vmax = 5 cells per time step = 5*7 m per s. Vmax is therefore 35 m/s or 126 km/h. As the legal speed limit on the motorway we are observing as 120km/h it is not unreasonable to assume 126km/h would be the speed many people choose as their maximum.
We now have:

$$\text{road} \in \mathbf{R}^{1\times 10000}$$

$$1 \leq i \leq \ 10000, \quad v = \begin{cases} -1 & \text{if no car,} \\ 0 \leq v \leq 5 & \text{otherwise.} \end{cases}$$

$$1 \text{ time step} = 1 \text{ second}$$

$$1 \text{ cell} = 7 \text{ meters}$$

$$v_{\text{max}} = 5\,\text{cells per time step} = 126\,\text{km per hour}$$

In their model Nagel and Schreckenberg put in place basic rules to appropriately modify the speed of a car. With a few modifications, this model implements theses rules as the backbone for traffic movement. Here are the rules in detail:

1. Acceleration
Let v be the speed of a car. Provided v is lower than v_{max}, if the distance to the car immediately in front is greater than $v + 1$, then v is increased by 1.

$$v = v + 1$$

2. Security Deceleration
If a car in cell i sees a car less than v spaces ahead, it decelerates to the distance j between the two cars minus 1. In other words, if there is a car at position i and a car at position $i + j$, the distance between the two is $(i + j) - i = j$. If j is smaller than v (the speed of car i), then:

$$v = j - 1$$

3. Stochastic Deceleration At every iteration, each vehicle decelerates by 1 with a probability P. This represents one of the human aspects of the model, as most traffic incoherence is due to random (human) acceleration or deceleration. Therefore, with probability P, the speed v becomes:

$$v = v - 1$$

Note that (unlike in the Nagel & Schreckenberg model) this last rule is applied only to cars going at a speed of 2 or more. Since this model represents a motorway, we assume that a car going at a speed of 2 or less is not in a stable situation and will reaccelerate as soon as possible. Therefore, the stochastic deceleration is not applied to these vehicles.

Once the speeds of all cars have been modified (or not modified) by the three rules above, we move the cars to their new positon. A car at position i moves to position i + v.
We perform a check of these rules at each time step and move accordingly. Other functions are put in place to keep a continuous flow of oncoming traffic.

2.2 Insertion

As stated previously the main aim of this model is to understand the effects of insertions. We must therefore implement some insertion process to the model. Considering we are in a fast lane, this model mimicks a second lane by simply adding new cars to the existing lane. The idea is to randomly generate a cell of the road then check several criteria regarding its current state and state of the cells around. If these conditions are met a car is 'inserted'; that is, a cell that previously had value -1 becomes non-empty. We will then be able to vary the amount of insertion and observe how the behaviour of the road changes.
After having generated our random cell we start by checking if it is free and then calculate the speeds and positions of the cars immediately behind and ahead. We then insert accordingly:

- Let b be the distance to the car behind and v_b the speed of that car.
- Let f be the distance to the car ahead.

$$\text{If } (b \geq 4 \text{ and } v_b = 5) \text{ or } (b \geq 3 \text{ and } v_b = 4) \text{ or } (b \geq 2 \text{ and } v_b < 4),$$

$$\text{Then:}$$

$$\text{If } f \geq 5, \text{ insert with speed 4.}$$

$$\text{If } f = 4, \text{ insert with speed 3.}$$

$$\text{If } f = 3, \text{ insert with speed 2.}$$

$$\text{If } f = 2, \text{ insert with speed 1.}$$

More generally, if v_b is large we need a similarly large b to insert. Empirically, if a car is going slowly we can afford to allow less space when inserting. The speed of insertion then depends on the space in front. If there is space you can go fast, if not one must be more cautious. It is possible (and even likely) that the car and the car behind it must immediately adjust their speed after insertion (according to the rules of **2.1 Basic vehicle movement**).

Note: these rules are only valid for a road with a vmax of 5. If we were to increase that value, these rules should be modified slightly (not necessarily linearly).

If at any point an insertion criterion is not verified we add one to our initial random value and start again. This means that when the traffic gets very dense, it is possible that no insertions can be made. This is another stochastic and human element of the model which we can accept as it is not unreasonable to assume that sometimes a car that wishes to insert can in fact not do so.

2.3 Parameters and output

Like all models, this model (particularly the C code) can been seen as a black box. We give the box some numbers and it gives us back some other numbers. In this case we have already fixed certain parameters for the model:

- `LENGTH = 10000`
- `ITERATIONS = 300`
- `VMAX = 5`

- `P 0.1` the probability of randomly slowing down

We must also give the program a value of `I`:

- `I` the number of insertions made every 10 seconds (or time steps)

And a value for `F` the initial frequency of the cars on the road:

- there is a car every `F` spaces on the road

The model is such that one can feed multiple values of `F` and `I` to the model and it will run the simulation accordingly as many times as necessary. The other parameters are fixed as they do not change from one run to another. *The `F` and `I` parameters are fed to the C-model via a python program, the others are implemented directly in C (more details in the README file).*

With theses 6 parameters, the model will output 4 values:

- `I` gives back the number of insertions per 10 s
- `avgspeed` the average speed of all cars on the road after 5 minutes
- `spac` the current spacing between cars (measures density in the same way as `F`)
- `F` also returns the initial spacing

2.4 Insertion Based Speed and Ideal Average Speed

Equipped with the model described above, we will generate two distinct set of values.
The first one **Insertion Based Speed** (IBS), is the average speed of cars after five-minutes under the influence of I insertions per 10s. We will feed the model several values of I but a fixed F. This way we will observe how the flow evolves on a theoretically identical road under more or less intense insertion conditions. IBS shows us how the speed of a given road is affected if we change the frequency of insertions. We choose F = 20 in the calculations. This corresponds to 1 car every 140m (moderate traffic density).
Ideal Average Speed (IAS) is an idealised situation where a certain density of cars is uniformly distributed along the road. Crucially in this measure, we remove insertion effects. We want to understand how the speed changes only under the influence of density. IAS shows us how the flow of a road changes given a density.
To calculate IBS, F=20 and I = 0,...,40. To calculate IAS, F=3,...,30 and I=0. The values if IAS are generated once and then stored for easy access in a csv file. As explained above, we give the model I=0 and we vary F. The csv file is stored in the format [*value of F, corresponding avg speed after simulation*].

3 Results

First off, we will plot the average speed of the road after 5 minutes with an increasing amount insertions per 10s. Recall that 1 s = 1 time step so the unit of the x-axis is equivalent to insertion per 10 time steps.

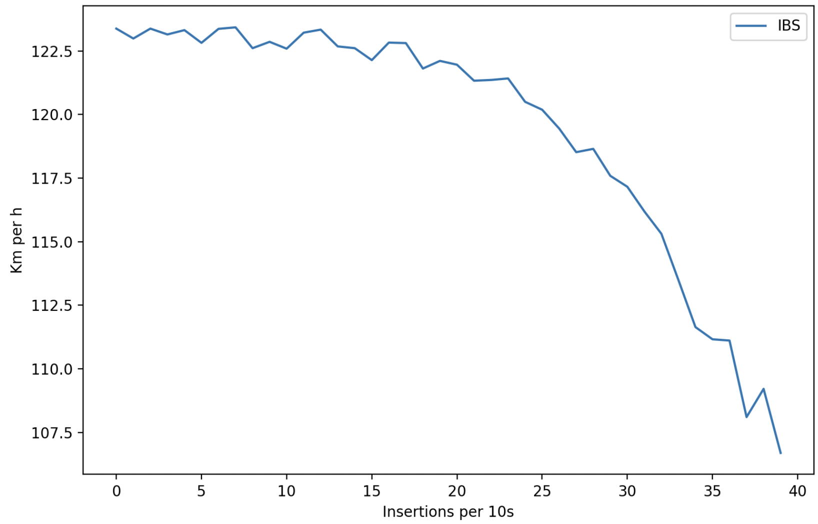

Figure 1: Average car speed on the road after 5mins of simulation as a function of number of insertions per 10s or IBS as a function I.

We immediately notice two distinct general trends of evolution of the average speed.

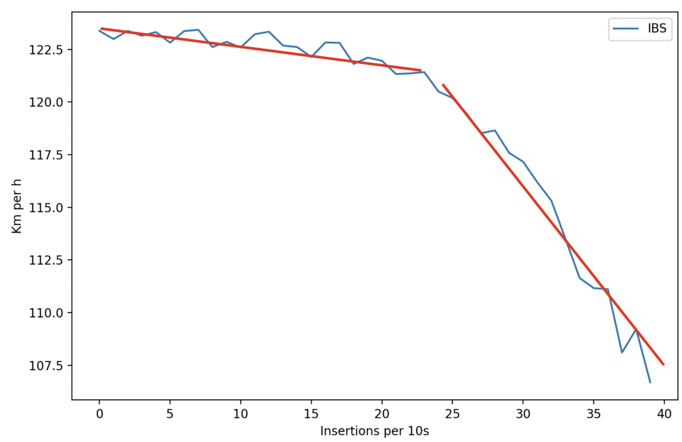

Figure 2: General trends of *Fig. 1*

To understand the effect of insertions we want to compare the average speed under the insertion scenario with the ideal average speed associated to the density seen at the end of the 5 minute simulation. To each value of `I` on the above graphs, there is an associated `spac` value provided by the C code (see **2.4**). This corresponds to a certain density of cars (at the end of the simulation). We want to know what the average speed of a road with that density would be if it were nicely distributed and unaffected by insertions. Conveniently we have a file (`ias_data.csv`) that contains precisely that. `spac` is not necessarily an interger so to access the corresponding IAS value in the file we must floor the value. We find the value in `ias_data.csv` corresponding to $\lfloor$`spac`$\rfloor$.

Just to recap, we have an `I`. That `I` has a corresponding `avgspeed` (currently plotted on the graph) and a `spac`. We use this `spac` to find the theoretical speed associated to that density. We now have two speeds associated to our `I`, the insertion influenced one (IBS) and the ideal one (IAS). Now we plot.

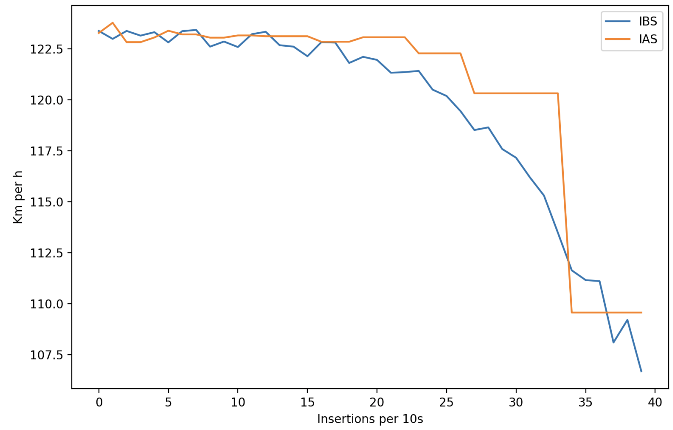

Figure 3: IBS and IAS as a function of I

The function is quite jumpy due to the flooring. To ease analysis here is a slightly smoothed version.

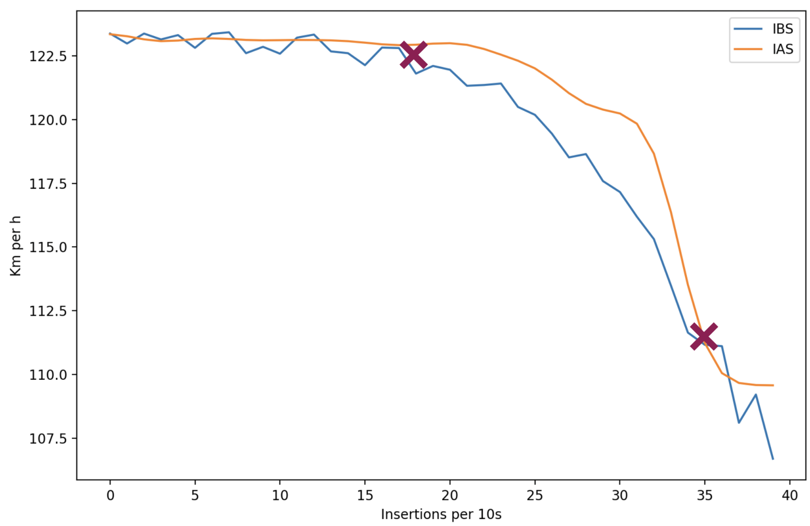

Figure 4: Smoothened out version of *Fig. 3* with two critical points

4 Interpretation, discussion & outlook

General flow of a road is affected by three aspects: random acceleration/deceleration, insertions and traffic density. The first aspect we accounted for in the model but it is not the main aim of this study. The two others are precisely what we want to observe. If we run the the simulation for too long, the density of the traffic becomes too important and drowns out the insertion effects. This is why we choose to run the model for 5 minutes, any longer would produce more significant density effects. Yet we want to study the effect of insertions without the effect of high density being an overwhelming factor. We choose the average speed after 5 mins as our 'unit' of flow. This is a reasonable measure at it gives an indication of how long it would take a given car to cover the 70km and a that point in time. This is the most practical way of understanding flow or traffic efficiency. All the driver really wants to know is how long it will take him to get from A to B.

As shown in Fig. 2 the IBS speeds decrease significantly at arround I = 22. This yields the question: is this sudden drop due to insertion effects or are we simply observing the density increase and force a decrease in overall speed ? To answer this question, we can use Fig. 4. As it was extensively explained above, it plots in addition to Fig. 2, the theoretical average speed on the road according to the density at a given point (IAS).
We identify three zones on this graph. Prior to the first critical value, the two graphs are more or less the same. This means that up to approximately 18 insertions per 10s, the flow is essentially unaffected by the insertions. Granted the speed of IBS decreases but not shockingly more than IAS, so we can not blame the insertions.
The second zone is very interesting. In between the two critical values, we observe that the two graphs diverge and the ideal speeds are consistently above the insertion calculated ones. In between 18 and 35 insertions per 10s, the IBS underperforms compared to the IAS. What does this tell us ? It tells us that traffic is slowed down due to insertions effects. The density at that time allows for a faster average speed and yet our model shows the actual speed is slower.
In the third zone (after the second critical value), the two graphs converge again. We can conclude that the density becomes so significant that the insertion effects are no longer relevant. Even within the 5 mins, the density effects suffocate the insertion ones.
It is clear that the higher the values of I are, the larger the traffic density will be by the end of the 5 minutes. We add more cars every 10s so naturally we will have more when the simulation terminates.
For all values of I under 18, the traffic flow is unaffected by insertions. We can explain this by observing that to insert we randomly generate a position. As the traffic density remains rather low for I<18, the spacing in between the cars is quite large. This allows for an effective buffering of the insertion effects. Potential car deceleration due to an insertion is not propagated too much throughout the road. This also somewhat justifies our choice of F = 20 (**2.4**) as we have observed that prior to this density the effect of insertions is not blatant.
In contrast, as the insertion levels get higher the acceleration/deceleration effects are more and more dramatic. A car that might insert too slowly will force the car behind to slow down, forcing the car behind that one to slow down as well. This creates a chain effect and ultimately a traffic jam. It is also important to point out that in very high density traffic it is possible that no insertions can be made as the rules regarding insertions are never satisfied.
Hence, we seemed to have found a zone where insertions are most harmful to flow. We have shown that insertions precipitate the existing negative effect of density and prematurely slow traffic down. This is a rather subtle observation and perhaps not of immediate revolutionary use but it could be taken into account when deciding on the creation or not of more roads and the overall dynamics of automobile mobility.

We must remember that this model is a cellular automaton and is perhaps not as precises as differential equation based models. We could double check the conclusions from this project by using another model. What is more, the data plotted was taken from a single 5 min simulation. To increase accuracy, we could make several such simulations then plot the average values of all of them. As a last element, we could modify the maximum speed to try and observe weather or not insertions have a similar effect on slower roads.