CS166 Location-Based Project

Traffic Intersections

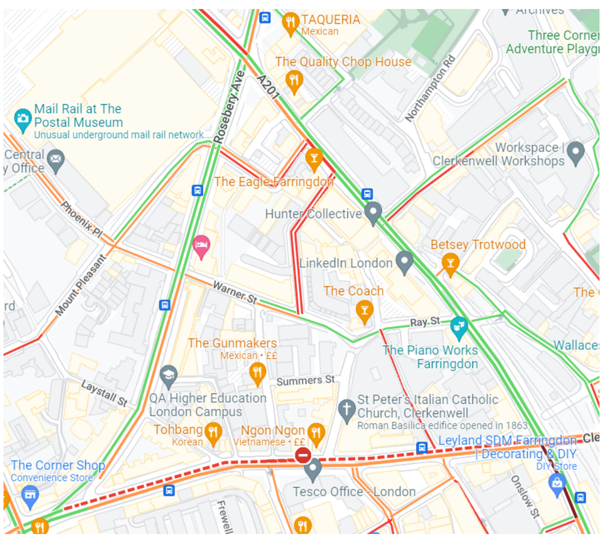
Pedro Haschelevici

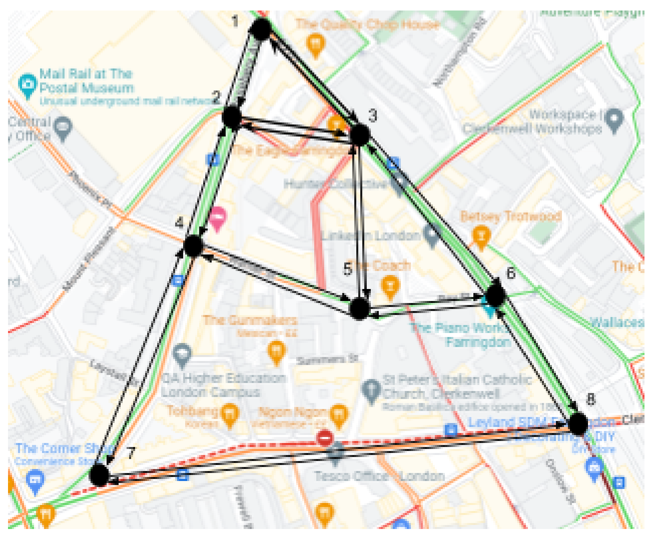
Introduction

For this project I built a simulation to model the traffic flow in the blocks surrounding the area that I live. The goal of this project is to collect insights on the traffic patterns of the area and hopefully suggest improvements to it. I based my model on the Nagel and Schreckenberg 1992 traffic simulation using cellular automata, and extended it to include intersections and traffic lights.

Data Collection and Map Adaptation

For this project I decided to do a mixture of data collection and the use of google maps. I first selected the area I wanted to work with and made a simplified graph model of it:





It made it a lot harder that most of the streets here are two-way, the cars drive in the opposite direction (compared to most of the rest of the world), and, the position and direction of the streets makes absolutely no sense.

With this simplified model done, I went and collected some rough data on the max speed that the cars have in each street, the usual density of the streets, and how likely people are to reduce their speed (spoiler alert: not very likely). With this, I set the probability of slowing down to 0.2, and tunned the other parameters to feel more like the real world. The detailed parameters can be found in the code.

Model Design

For the model, I followed the same rules as the Nagel–Schreckenberg model, which represents traffic as a cellular automata with a few simple rules. The street is modeled as a series of a fixed number of discrete cells. Each cell can either contain a car or not. Each car has a speed between 0 and the speed-limit, which is a parameter of the model. Streets are one way. The model is discrete time. At each step, each car updates as follows.

1. Acceleration: each car increases its speed by 1.
2. Slowing down: each car's speed is decreased so that it is not higher than the speed limit and not higher than the number of spaces to the next car.
3. Randomization: each car decreases its speed by 1 with probability p-slow, a parameter of the model.
4. Motion: each car is moved forward a number of cells equal to its speed.

To adapt this model to my needs, I had to make a few modifications. First, the streets have a beginning and an end, unlike in the Nagel–Schreckenberg model where they circle back. To mark the end of the street, I put a fake stopped car, so that it simulates the driver slowing down when approaching an intersection (or at least they should). In order to adapt it to two-way streets, I simply used 2 independent lanes to describe it.

When a car reaches the intersection, the behavior of that car is governed by the intersection, not the lane. The intersection checks where the car is coming from, and randomly assigns a lane (different than the one it came from), if the first position of that lane is free, the car goes there, if not the car waits for the next turn.

With this model as a base, I also made a second type of intersection with traffic lights. It works very similarly to the first one, but with a simple twist. It keeps track of the timesteps, and updates the traffic light after every 5 steps. Once a car gets to the intersection and the lane it's going to is selected it not only checks the lane, but it also checks if the traffic light is green for that particular combination lane from and lane to.

Simulation Analysis

For the analysis, I looked at 2 main metrics (average traffic flow, and end density), over 2 different scenarios (with traffic lights, and without traffic lights). For both scenarios I ran the simulation 1000 times and up to 100 timesteps. Let's look at the results without the traffic light first.

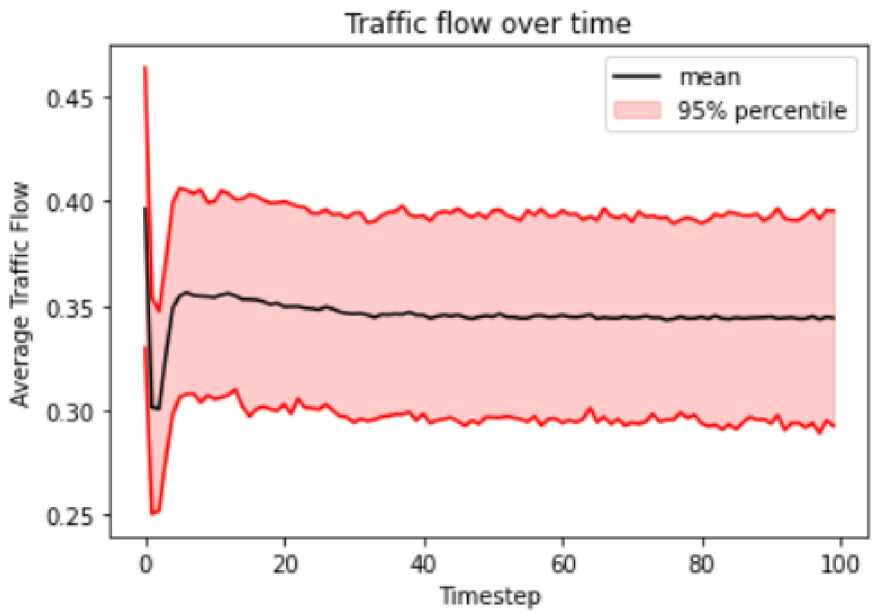


Fig 1: Average traffic flow over the timesteps with a 95% confidence interval calculate using the percentiles for 1000 simulations without traffic lights

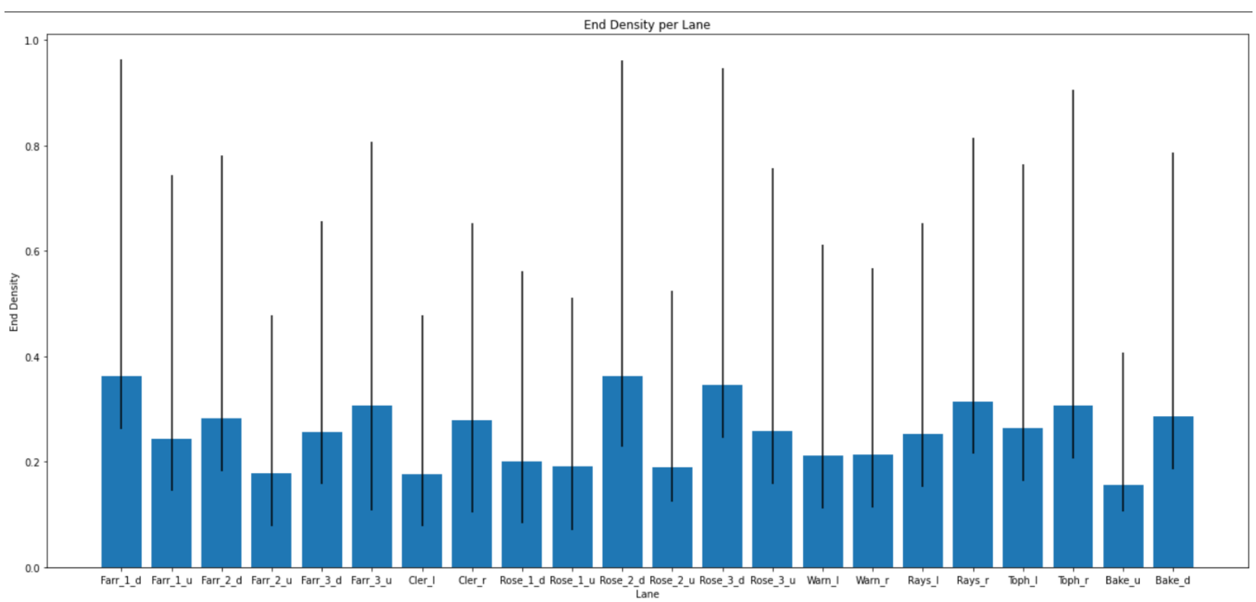


Fig 2: Average density in the end of the simulations for each lane for 1000 simulations without traffic lights. The error bars represent the 90% confidence interval calculated using the percentiles

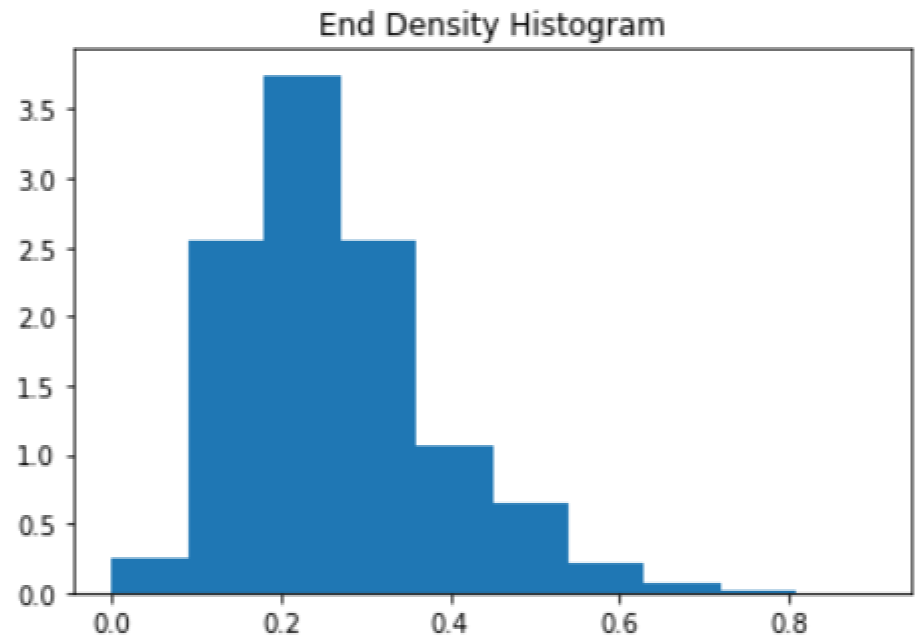


Fig 3: Normalized Histogram of the end densities of the lanes for 1000 simulations without traffic lights.

We can see from Figure 1 that, without the traffic lights, the average traffic flow tends to converge towards 0.35 with a brief period of rapid decrease and increase on the first 10 timesteps. My guess is that this is due to the initial density of the streets being different from the densities in which they tend to stabilize after some time. We can also be very confident that the average traffic flow will converge to somewhere between 0.3 and 0.39, which is a pretty good flow for a big city.

In the end densities on the other hand, we can see a much bigger variation, with the busiest streets reaching almost full capacity in some simulations, but, mostly, as we can see from Figure 3, ending around 0.25, which is very good for a big city.

Now for the simulation with the traffic lights:

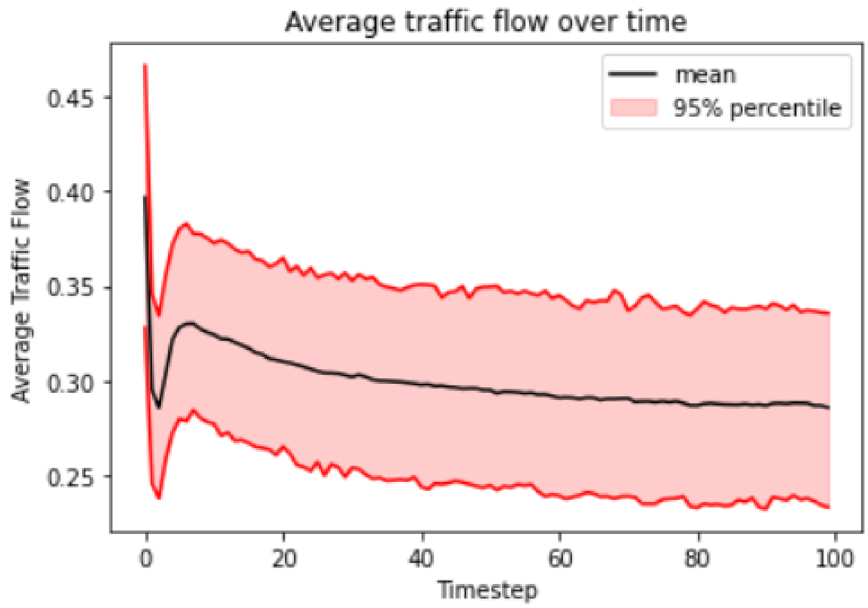


Fig 1: Average traffic flow over the timesteps with a 95% confidence interval calculate using the percentiles for 1000 simulations with traffic lights

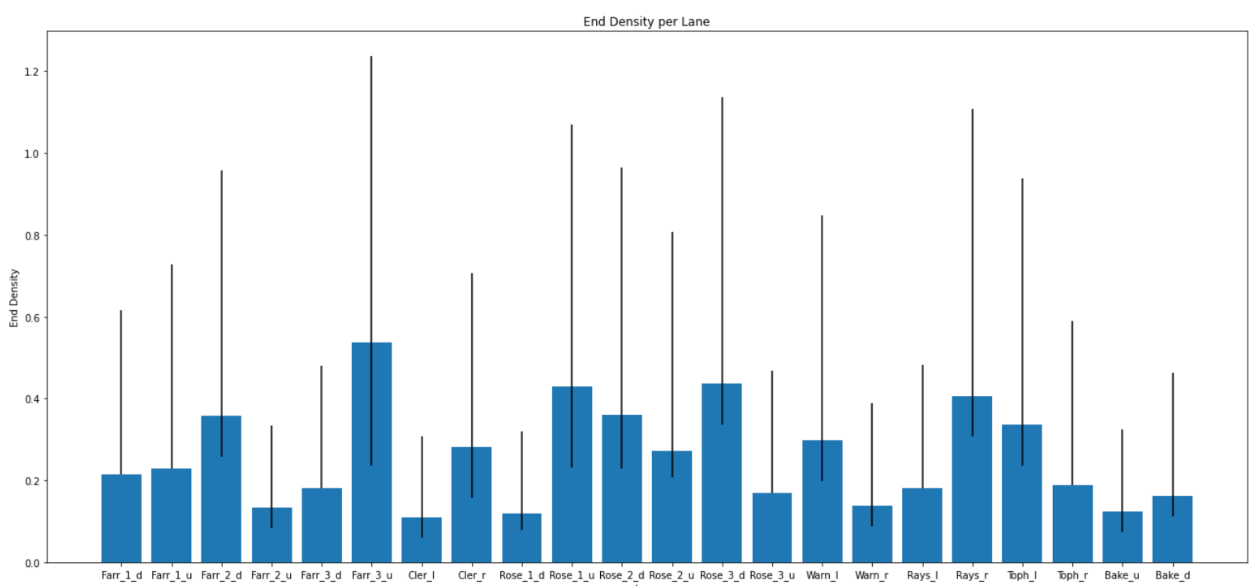


Fig 2: Average density in the end of the simulations for each lane for 1000 simulations with traffic lights. The error bars represent the 90% confidence interval calculated using the percentiles

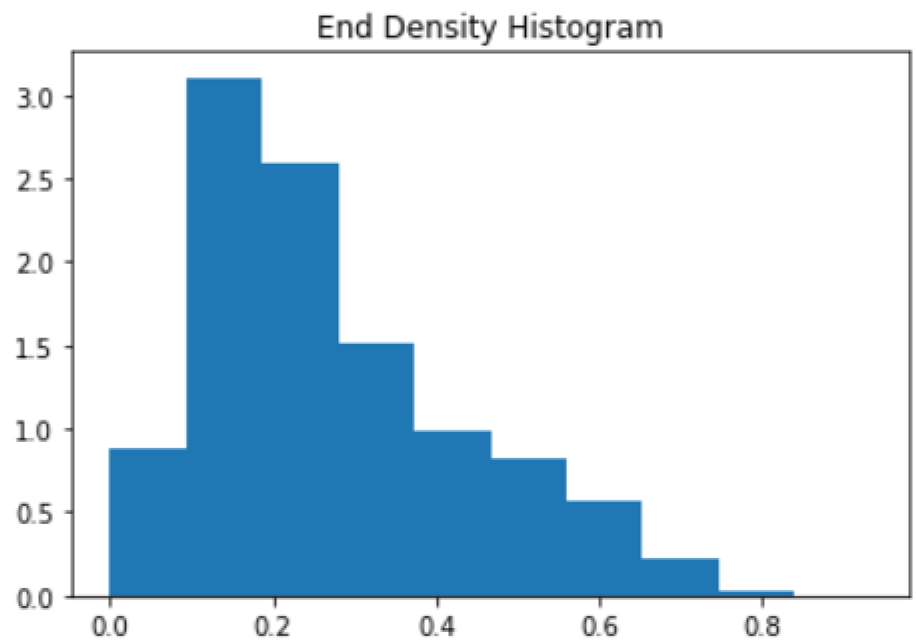


Fig 3: Normalized Histogram of the end densities of the lanes for 1000 simulations with traffic lights.

The results with the traffic lights are pretty similar to the ones without it, the overall behavior is the same, and, we can see in figure 1, the same transition period in the beginning. However, as we can also see in figure 1, the result is, on average, worse than the previous simulations with this one converging to around 0.3. On figure 2, we can also see that the variation is even bigger than the previous one on the density. It looks like the traffic lights made the densities less balanced between the streets.

While this simulation might provide us with some interesting insights, it is a very big simplification of the actual world. I would love it if the cars in London actually slowed down when approaching the intersections, or calmly turned in order when changing lanes.

References

Nagel, K., & Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. Journal de Physique I, 2(12), 2221–2229. https://doi.org/10.1051/jp1:1992277

Full code on https://github.com/phasc/cs166/tree/main/lba