Kachow!

Creating a traffic flow optimized intersection that even Lightning McQueen would love

Rhali Attar

CS166: Modeling and Analyzing Complex Systems
Professor Ribeiro
College of Computational Sciences
November 10 2022

Contents

1 Introduction and Motivation

Pedestrians of Buenos Aires have very unique behaviors around traffic intersection without traffic lights: they assert their presence to cars by crossing the street regardless of whether cars are coming or not. While this seems messy and dangerous, it works surprisingly well as traffic always seems to flow, even in busy areas. The neighborhood of Palermo is notorious for this as the large majority of intersections do not have traffic lights and all streets are one way and two lanes, except one. This system for crossing the road makes sense here as the streets are small and not extremely busy, which means that the government does not have to invest in traffic lights. The motivation behind this research is to analyze Avenue Juan B. Justo, the only two way, four lanes street with traffic lights that divides Palermo which feed it cars from its one way streets (see Figure 1). After gathering data at the avenue, the intention of this study is to model two of its blocks to better understand the decision behind the introduction of traffic lights and how traffic flow could potentially be optimized.

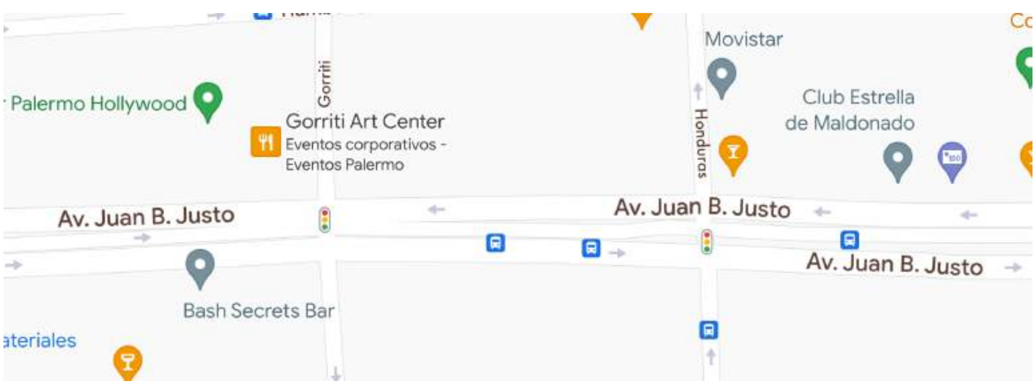

Figure 1: The intersections that will be explored in this paper

2 The Model

2.1 Method and Assumptions

The model that will be used for this study is a cellular automaton. It is built off of two classes: a street class that determines movement in individual streets and an intersection class that models the interactions between intersections of streets. The relevant parameters and variables of the code are the following:

- `lanes` (2d array): determines the number of lanes on a street and the direction the cars are moving in for each lane
- `speed_limit` (int): the maximum speed that cars on the street are allowed to reach
- `length` (int): the lenght of the street
- `lane_switch_probability` (float): the probability that a car will switch lanes at any given point
- `lights` (bool): whether the road as traffic lights at the end of it
- `green_light_time` (int): the time it takes for a traffic light to switch from green to red
- `red_light_time` (int): the time it takes for a traffic light to switch from red to green
- `traffic_light` (str): the current color of the traffic light (changing it when initializing a street will determine the initial color of the traffic light)

- `slow_down_probability` (float): the probability that a car will slow down at any given point
- `speed_up_probability` (float): the probability that a car will slow down at any given point
- `horizontal_streets` (array of Street objects): the streets that are horizontal to the traffic intersection
- `vertical_streets` (array of Street objects): the streets that are vertical to the traffic intersection
- `turn_left_probability` (float): the probability of turning right at the intersection
- `turn_right_probability` (float): the probability of turning left at the intersection
- `pedestrian_probability` (float): the probability that there will be a pedestrian at the intersection

At every time step, cars will move forward by their speed v according to the following rules:

Acceleration

Cars' speed will always be increased by one if there is enough space in front of them to do so. However, if there is a car in front of it that is moving at a slower speed, it will slow down to match its speed (or switch lanes, see below) - this means that it will stop if the car in front of it is stopped. Additionally, according to `slow_down_probability` and `speed_up_probability`, cars' speed will randomly be changed by one given that they are not going above their street's `speed_limit` and that they will not stop if they slow down.

Lane switching

If a can switch lanes to the left to overtake a car that is slower than it in front of it, then it always will. Cars will also randomly switch lanes to the right according to `lane_switch_probability`.

Stopping

If a car reaches a traffic intersection or the end of the street, it will immediately stop according to `lights` and `traffic_light`. If `lights` is False, a car will always stop as that models a situation where there aren't traffic lights which implies that there is a stop sign. Otherwise, the car will keep moving forwards if `traffic_light` is green or stop if it is red.

Accelerating after stopping

If a car stopped for any of the reasons above, it will start moving again according to `lights`, `traffic_light` and `pedestrian_probability`. If we are at an intersection and `lights` is False then a random car will be picked to start moving first. While there are rules about which cars should move first at an intersection, they are usually ignored which is why this is appropriate. The car will then only move based on the probability that a pedestrian crosses on its street (if there is a pedestrian on the intersection, we also have to check that it is on the correct street). Moreover, if `lights` is True, the car will only start moving again if `traffic_light` switches to green. The car's new speed v will be sampled from $V \sim \text{Unif}(1, \texttt{speed\_limit})$.

Turning at an intersection

Once a car reaches an intersection, it must choose where to go next which mostly depends on `turn_left_probability` and `turn_right_probability`. If it is about to turn, the car will have to check whether it can even turn in that direction which depends on whether the lane is is attempting to go to's is moving in the appropriate direction. If it can turn, the car is always assumed to join the left most lane as that is the law, but if it cannot turn it will keep going straight and stay on the same lane. An assumption that was made here to simplify the code is that cars immediately join the next street in the intersection that they are going to.

2.2 Critique

There is one assumption that was not mentioned above that is worth discussing. Indeed, the model assumes that cars' speeds are uniform distributed between 1 and the speed limit, which is probably not an accurate representation of reality as cars usually drive at a speed much closer to the speed limit. However, since the distribution was not known and cars are made to be able to accelerate if possible, this does not have such an important impact on our conclusions.

The rest of the assumptions of this model have been discussed above and assuming that people follow basic driving laws, they are all clearly appropriate except one important one: cars do not need to decelerate before stopping. This assumption was made as it significantly simplifies already very complex code, but it does have some impacts on our results. Indeed, it essentially assumes that a car can stop at any point, regardless of the speed they are moving at, which is not at all realistic as it completely removes the risk of car accidents. This therefore favors intersections without traffic lights even though they are more prone to accidents since they do not always stop traffic for a very large period of time. Additionally, cars are assumed to always stop at a stop sign, which does not always necessarily happen in Palermo. However, since this will be modeling an intersection with a large road, it is fair to assume that drivers would be more cautious and always stop. Lastly, it is important to note that this code is not completely perfect as, on a few rare occasions, cars seem to disappear. However, this does not significantly impact our results because it does not occur frequently enough to have a large impact on traffic flow.

3 Results

3.1 Gathering Data and setting up the simulations

I went to the both traffic intersections to collect data that would allow me to infer how to pick values for all the parameters that the model requires. Here is all the information I got:

Figure 2: Selfie at the intersection

Gorriti y Juan B. Justo:

- Gorriti green traffic light lasts 40 seconds
- Juan B. Justo green traffic light lasts 55 seconds
- Pedestrians: 73 in 5mins
- Gorriti turning right (per traffic lane change): 3, 3, 2, 1, 1
- Gorriti turning left (per traffic lane change): 3, 0, 3, 2, 3
- Juan B. Justo turning right (per traffic lane change): 3, 1, 2, 2, 5

Honduras y Juan B. Justo:

- Honduras green traffic light lasts 40 seconds

Figure 3: Selfie at the intersection

- Juan B. Justo green traffic light lasts 55 seconds
- Pedestrians: 114 in 5mins
- Honduras turning right (per traffic lane change): 1, 0, 0, 2, 2
- Honduras turning left (per traffic lane change): 3, 1, 1, 5, 1
- Juan B. Justo turning right (per traffic lane change)t: 2, 6, 4, 2, 2

Using this data, I was able to make calculations to pick some of the parameter values with the choice of one time step representing one second. Firstly, I calculated the probability of a pedestrian to show up at each time step (second) to be 0.243 for the first intersection and 0.38 for the second. Additionally, with the estimation of approximately 20 cars going through the intersection per traffic light change, I estimated the probability of turning right to be 0.115 for the first intersection and 0.105 for the second. I then also estimated the probability of turning left to be 0.1 for the first intersection and 0.11 for the second. Lastly, I found that the traffic lights that followed each other on the large avenue's colors are inverted (so when one is green the other is red).

Moreover, using google maps, I was able to find out that the length of Juan B. Justo between Gorriti and Honduras is 90 meters long. I decided to divide this by 2 to make the simulation quicker and therefore made all the horizontal streets (which represent this avenue) to have a size of 22 to represent that length. I also made the vertical streets to have a length of 10 as they are approximately 20 meters long according to google maps. Then, I found that the speed limits of Gorriti and Honduras is 40km/h $\approx$ 11 m/s and that the speed limit of Juan B. Justo is 60km/h $\approx$ 16 m/s. So I inputted those divided by two since each cell in my model is 2 meters long. I also made it so that one car gets added to the system per second as it is a relatively busy area. Lastly, I just estimated the probability of a lane change to be 0.2 and the probability of speeding up and slowing down randomly to be 0.1 as those values seem reasonable.

3.2 Results and Discussion

Firstly, I decided to keep the traffic light times constant and see how removing traffic lights at the intersections would affect the traffic flow so that the best configuration could be established. The results of this exploration can be seen in Figures 4-7 below.

Traffic lights at both intersections

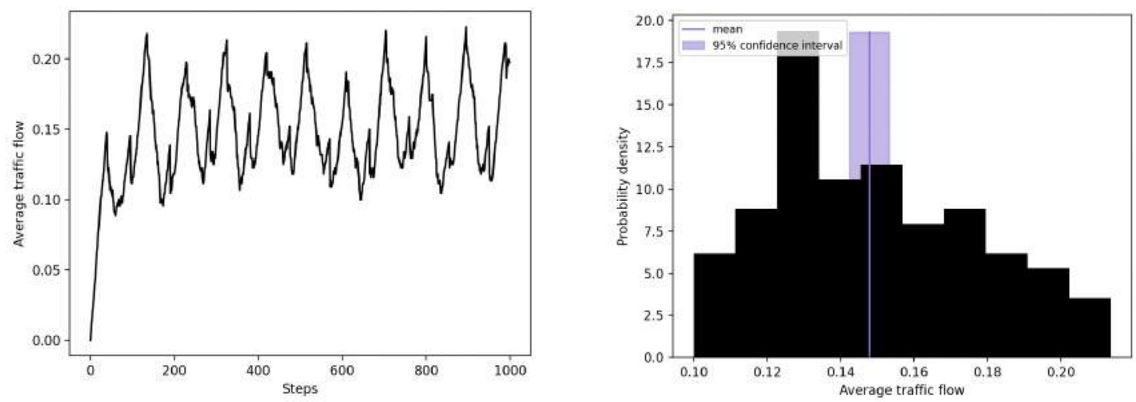

Figure 4: A line graph of how average traffic flow changes over time on the left and a distribution of average traffic flow after 200 steps

Traffic lights at the Honduras intersection but not at the Gorriti one

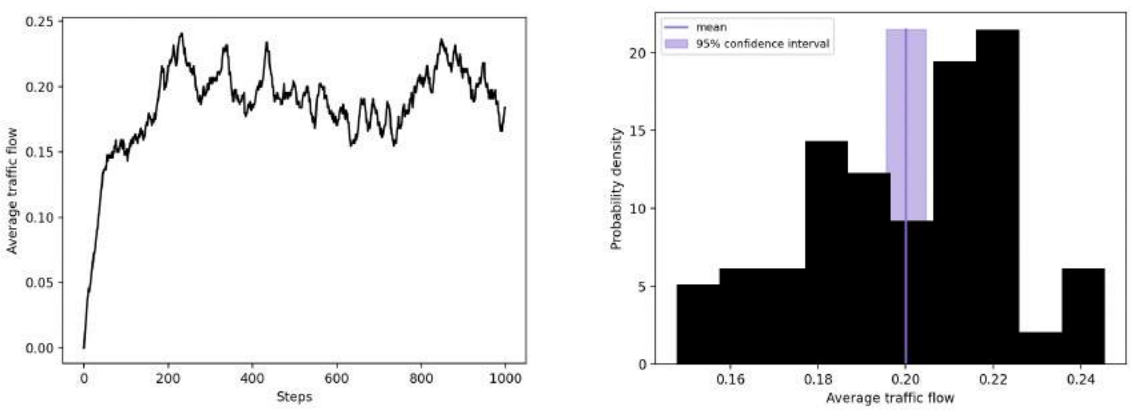

Figure 5: A line graph of how average traffic flow changes over time on the left and a distribution of average traffic flow after 200 steps

Traffic lights at the Gorriti intersection but not at the Honduras one

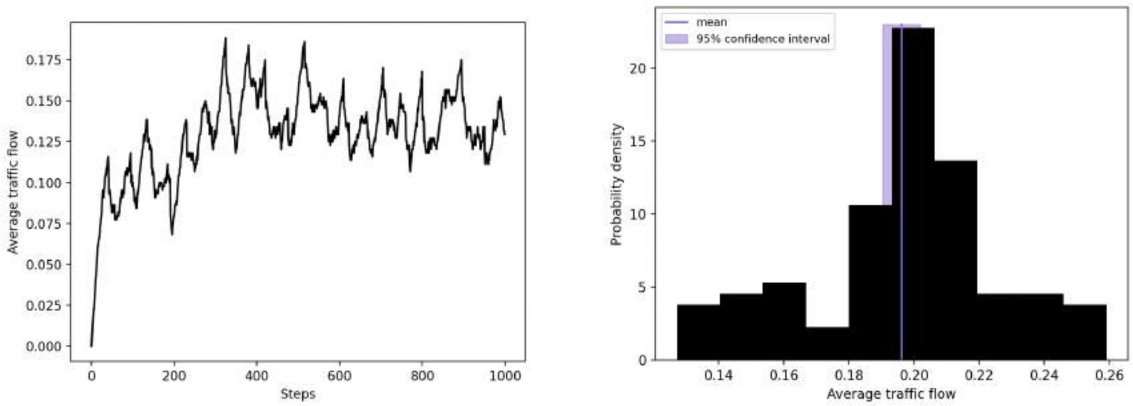

Figure 6: A line graph of how average traffic flow changes over time on the left and a distribution of average traffic flow after 200 steps

No traffic lights

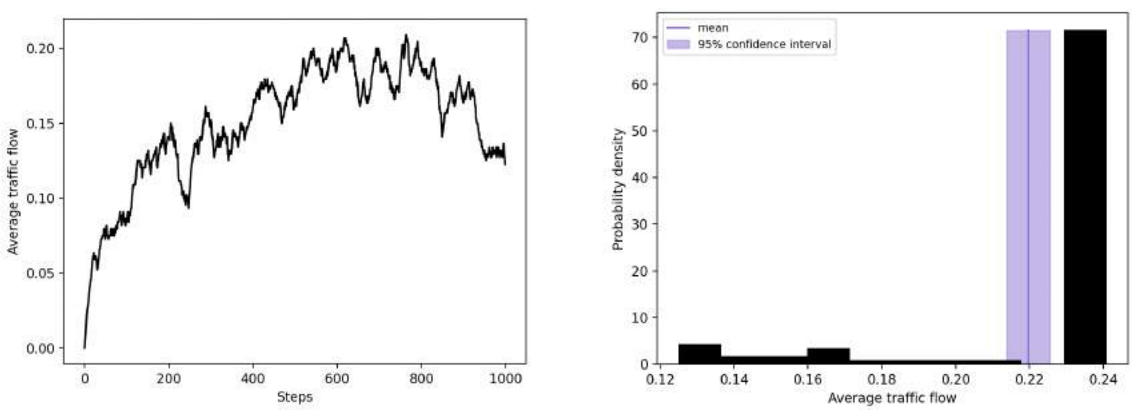

Figure 7: A line graph of how average traffic flow changes over time on the left and a distribution of average traffic flow after 200 steps

Analyzing the line graphs, we can see that the traffic flow always increases intensely before oscillating around the mean average traffic flow for these parameters.

Having traffic lights seem to make these oscillations much more regular. Looking at these graphs alongside the simulation animations allows us to understand that when the lights turn green on the main avenue, cars will start to flow into the area between the intersections, causing an increase in traffic flow, but when the lights turn red this area becomes emptier, reducing traffic flow. Since this happens at a regular rate, the regular oscillations in flow make sense. In a situation where there are no traffic lights though, we can see that the oscillations the pattern is much less regular due to the randomness of cars flowing through. Overall, we can see that having traffic lights at both intersection does seem like the optimal decision since traffic jams end up only being present at a few alternating traffic lights but when we remove them, the traffic jams are present at the end of every single street.

Now that this has been established, I decided to explore whether the lights system could be improved even further by changing the time it takes for lights to switch colors. I initially decided to see how various times affect average traffic flow with the constraint that the small streets and the large street's traffic lights take the same time (which is not reflective of the current situation). Figure 8 below shows the results of this simulation:

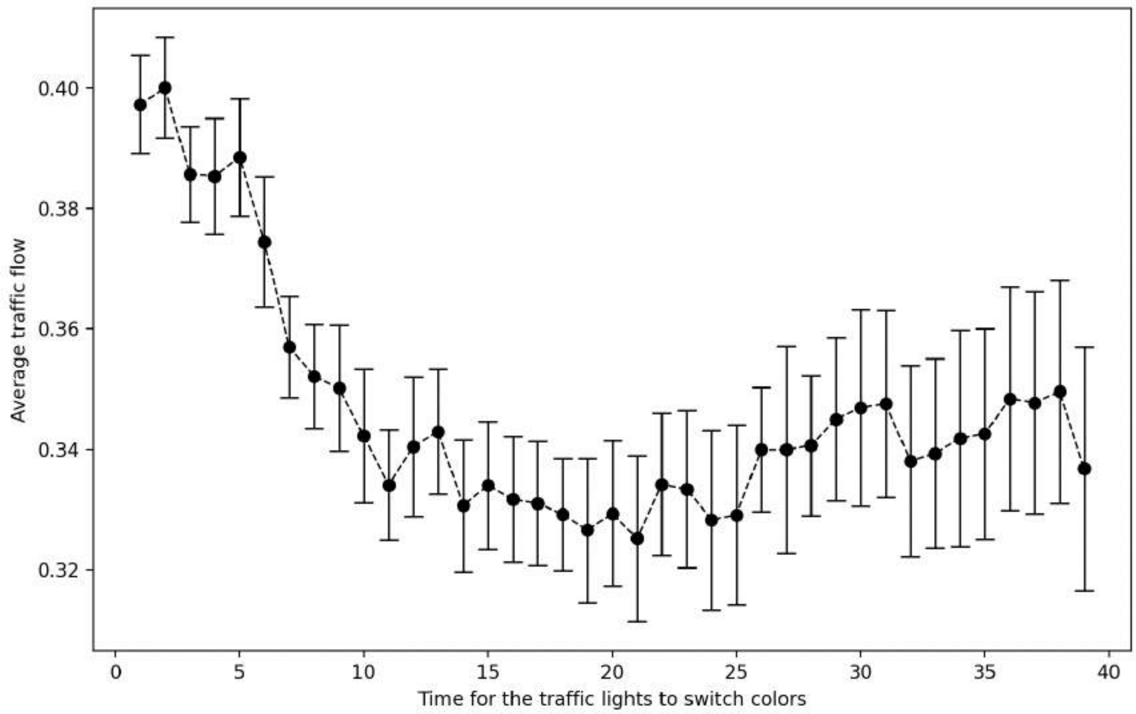

Figure 8: Line graph showing how varying the time it takes for traffic lights to change colors at intersections affects mean average traffic flow

We can see that the minimum is at 21 seconds but the mean average traffic flow is significantly higher than the results we have previously gotten. This therefore implies that having the traffic lights at the main avenue being green for longer than at the small avenue improves traffic flow. This makes a lot of sense because there is a higher likelihood of a car being in the main avenue than the small streets, making it busier, and therefore letting cars flow true for longer reduces blockage. I then decided to explore how varying the time for which the main avenue's lights are green affects mean average traffic flow when they are set up so that the main avenue's lights are green for 15 seconds longer than the small streets. The results are on Figure 9 below:

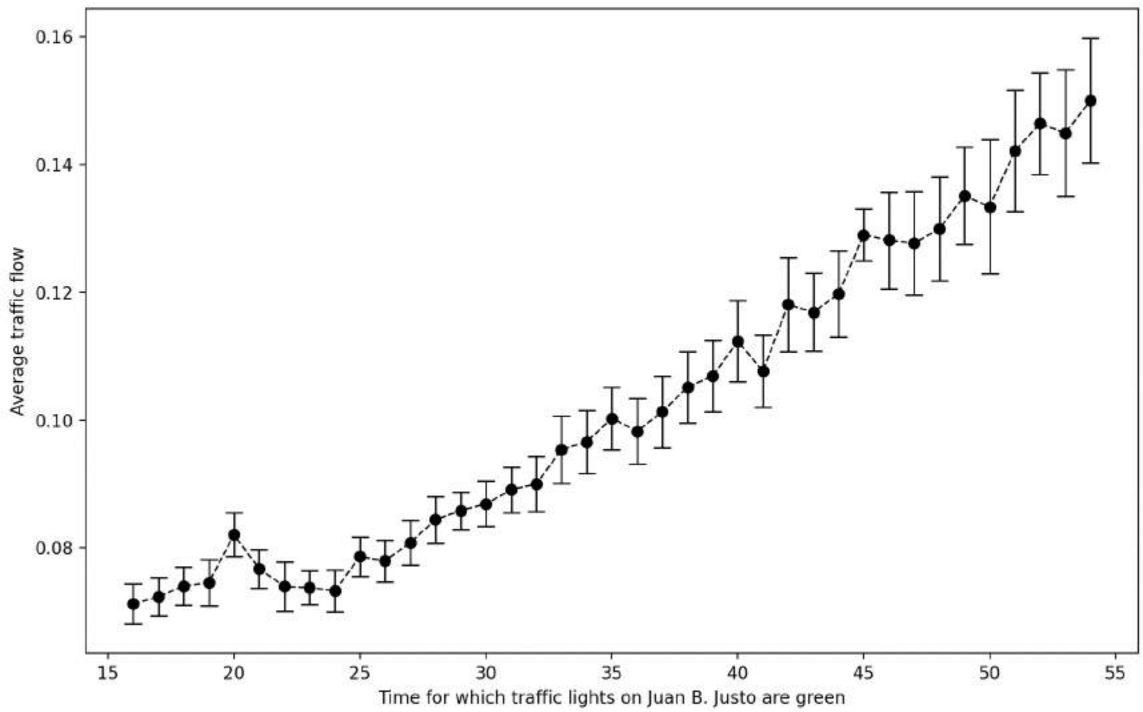

Figure 9: Line graph showing how varying the time for which the main avenue's lights are green affects mean average traffic flow

Here, the pattern changes significantly as there seems to be a very linear relationship. We can see that the minimum is at 16 seconds, however, this would be a very unreasonable because it would only give pedestrians 1 second to cross the main road. The second minimum at 24 seconds is better, but still not the greatest because it would only leave pedestrians 9 seconds to cross the main road. This therefore explains why the government chose the time that they did (55 seconds), as it was probably to leave pedestrians an appropriate amount of time to cross to main road. A study should definitely be done on the average amount of time it takes for a pedestrian to cross the street to make a specific decision on what time would be most effective.

3.3 Theoretical analysis

We can perform a Mean Field Approximation of whether a car on a street is moving (1) or stationary (0) based on the update rules above. Let's break this down into each of the constituent parts of the MFA:

Finding $\mathbb{P}(\mathbf{0} \to \mathbf{1})$

Firstly, this situation would never occur if the traffic lights are currently green because a car would never be stopped. However, if the traffic light is currently red, this situation would only happen if the traffic lights are turning green in the next state. Therefore, $\mathbb{P}(0 \to 1) = \mathbb{P}(\text{Car is stopped})\mathbb{P}(\text{Lights switch from red to green})$.

Finding $\mathbb{P}(\mathbf{0} \to \mathbf{0})$

Once again, this situation would never occur if the traffic lights are currently green because a car would never be stopped. Although, if the traffic light is currently red, this situation would only ever happen if the traffic lights stay red in the next state. So, $\mathbb{P}(0 \to 0) = \mathbb{P}(\text{Car is stopped})\mathbb{P}(\text{Lights remain red})$.

Finding $\mathbb{P}(\mathbf{1} \to \mathbf{1})$

This situation is more complicated as there are more cases to consider. Firstly, if the lights are green this will always happen with a probability of 1 if the lights will remain green. However, if the lights turn red in the next state then the car will only keep going if it not at the traffic light. Similarly, if the lights are red and will remain red, the car will only keep going if it is not at the traffic light and if there is not a car stopped in front of it. Lastly, if the lights are red and will switch to green, this will happen with a probability of 1. Putting this all together gives us: $\mathbb{P}(1 \to 1) = \mathbb{P}(\text{Car is moving})(\mathbb{P}(\text{Lights remain green}) + \mathbb{P}(\text{Lights switch from red to green}) + \mathbb{P}(\text{Lights switch from green to red})(1-\mathbb{P}(\text{Car at traffic light}))+\mathbb{P}(\text{Lights remain red})(1-\mathbb{P}(\text{Car at traffic light} \bigcup \text{Car in front is stopped})))$. We can make the intersection between two events a multiplication of their probabilities because these events are independent.

Finding $\mathbb{P}(\mathbf{1} \to \mathbf{0})$

This situation is quite similar to the one above. This situation would never happen if the traffic light is switching to green in the next state, regardless of whether it is currently red or green, because cars never stop when lights are green. However, if the lights are switching from green to red in the next state, the car would only stop if it is currently at the traffic light. Moreover, if the lights are remaining red in the next state, this situation would happen if the car is currently at the traffic light or if there is a stopped car in front of it. Putting this all together gives us: $\mathbb{P}(1 \to 0) = \mathbb{P}(\text{Car is moving})(\mathbb{P}(\text{Lights switch from green to red})(\mathbb{P}(\text{Car at traffic light})) + \mathbb{P}(\text{Lights remain red})(\mathbb{P}(\text{Car at traffic light} \bigcup \text{Car in front is stopped})))$.

Now, we need to find all of the probabilities that are mentioned above:

- $\mathbb{P}(\text{Car is stopped}) = p$

- $\mathbb{P}(\text{Car is moving}) = 1 - p$

- $\mathbb{P}(\text{Lights are green}) = \frac{\text{Time for which lights are green}}{\text{Time for which lights are green + Time for which lights are red}} = q$ (for simplicity).

- $\mathbb{P}(\text{Lights are red}) = 1 - q$

- $\mathbb{P}(\text{Lights remain green}) = \mathbb{P}(\text{Lights will be green}|\text{They are green}) = \frac{\text{Time for which lights are green}-1}{\text{Time for which lights are green}} = r$ (for simplicity). This is because lights will remain green as long as we are not in the last second before they turn red.

- $\mathbb{P}(\text{Lights switch from green to red}) = 1 - r$.

- $\mathbb{P}(\text{Lights remain red}) = \mathbb{P}(\text{Lights will be red}|\text{They are red}) = \frac{\text{Time for which lights are red}-1}{\text{Time for which lights are red}} = s$ (for simplicity). This is for the same reasons as above.

- $\mathbb{P}(\text{Lights switch from red to green}) = 1 - s$.

- $\mathbb{P}(\text{Car is at the traffic light}) = \frac{\text{Average speed}}{\text{Length of road}} = \frac{\bar{v}}{l}$. This is because $\frac{l}{\bar{v}}$ gives us the number of seconds it will take for the car to arrive to the traffic late, so the inverse gives us the rate of cars arriving at the traffic light.

- $\mathbb{P}(\text{Car in front is stopped}) = (\text{density of cars}) \cdot \mathbb{P}(\text{Car is stopped}) = dp$. This is because the probability that there is a car in front depends on the distribution of the cars throughout the road, which can be represented as the density of cars.

- $\mathbb{P}(\text{Car at traffic light} \bigcup \text{Car in front is stopped}) = \frac{\bar{v}}{l} + dp - dp\frac{\bar{v}}{l}$. We are assuming here that the car is at the traffic light has no cars in front of it which makes these two events independent.

Now that we have all of this, we can put it together to get our probability distribution:

$$\mathbb{P}(\mathbf{0} \to \mathbf{1}) = p(1-s)$$
$$\mathbb{P}(\mathbf{0} \to \mathbf{0}) = ps$$
$$\mathbb{P}(\mathbf{1} \to \mathbf{1}) = (1-p)(r+1-s+(1-r)(1-\frac{\bar{v}}{l})+s(1-\frac{\bar{v}}{l}-dp+dp\frac{\bar{v}}{l}))$$
$$\mathbb{P}(\mathbf{1} \to \mathbf{0}) = (1-p)((1-r)\frac{\bar{v}}{l}+s(\frac{\bar{v}}{l}+dp-dp\frac{\bar{v}}{l}))$$

We can now finally combine $\mathbb{P}(\mathbf{0} \to \mathbf{1})$ and $\mathbb{P}(\mathbf{1} \to \mathbf{1})$ to get our MFA difference equation (since we are trying to find the value of next average state which is the probability that it is 1):

$$p_{t+1} = -\frac{1}{l}(lp_t - s(dlp_t^2-(d+1)lp_t)+\bar{v}((p_t-1)r+(dp_t^2-(d+1)p_t+1)s-p_t+1)-2l)$$

Let's plug in the parameters for the first half of Gorriti so that we can find the fixed points such that the proportion of stopped cars always remains constant. On this street, we have $l = 10$, $r = \frac{39}{40}$, $s = \frac{54}{55}$, $\bar{v} = 2$, and, while d is constantly changing, we will set it to be the mean average traffic flow of the system which is $d = 0.15$ as we have found above. This gives us the following Cobweb plot:

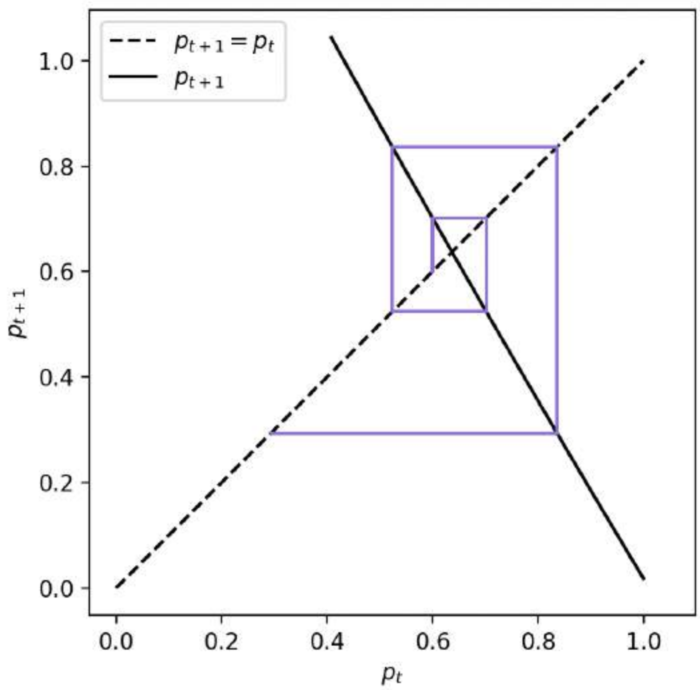

Figure 10: Cobweb plot of the MFA update equation for Gorriti

So, we can see that there is one unstable fixed point, which we can calculate to be at $p = -\frac{1}{864\sqrt{101536969}} + \frac{10627}{864} \approx 0.637$. Clearly this cobweb plot does not make much sense which is because this problem was intensely simplified with many means and probabilities which do not represent it appropriately. Another theoretical analysis of this problem that could be carried it out would be to model cars arriving at the intersection as a $G/D/1$ queue. Indeed, we have a deterministic service time which is $\frac{1}{1-q}$ or in other words, the average amount of time that cars wait at a red light. Our arrival rates would be a relatively complex distribution since cars are added to the system with a uniformly distributed speed, and they then proceed to change speeds stochastically until reaching the traffic light. This distribution could probably only be found empirically. The reason why this would be an interesting theoretical analysis to carry out is because it could allow us find the average waiting times of cars based on given parameters. We would then be able to find out how to minimize it so that cars do not have to wait very long at intersections.

4 Conclusion

Overall, this study informed us that the city of Buenos Aires either made very well informed decisions about the traffic intersections at Gorriti and Juan B. Justo and Honduras and Juan B. Justo or coincidently chose very effective traffic flow strategies. The model that was created for this paper could be used to simulate many more intersections though, and allow us to optimize traffic flow in the rest of the city!

5 HC and LO Appendix

#PythonImplementation

For this assignment, I created a (might I say) very impressive code to model a very complex and advanced intersection system. I used object oriented programming so that my classes can be very easily generalizable and self sufficient. Additionally, this enabled me to use inheritance between classes so that Street instances can be used to create Intersections instances which can be used to create multiple intersections. I strongly believe that, thanks to its generalizability, this code could definitely be used in various publishable research papers about modelling traffic flow, which I am very proud of.

#Modelling

I created a great model to explore traffic flow for almost any type of street and four way traffic intersections. In my paper, I made sure to explain the reasoning behind all the assumptions I have made when creating this model and then critiqued them appropriately.

#EmpiricalAnalysis

Using my model, I then went on to explore various ways in which average traffic flow can be optimized. To do this effectively, I ran the simulations a large number of times and created histograms with 95% confidence intervals. I then compared all of those, alongside the animations to critique the various intersection configurations and end up with a conclusion of which one is the most effective. Lastly, I would like to point out that I also collected actual data at the traffic intersections to choose parameters so that my model is as reflective of real life as possible.

#TheoreticalAnalysis

In the paper, I went through the process of making a Mean Field Approximation of the state of cars (whether they are moving or not). I thoroughly explained my thought process behind all the probabilities and then created a cobweb plot with my update function. I also explained how queuing theory could be used and the ways in which it would be useful.

#CodeReadability

Here is a link to a loom video that I recorded to go through all my code and explain it a little more https://www.loom.com/share/97164ea712c540369d41eeb41666b2c2 (the video has a time limit of 5 minutes so I unfortunately could not go in a lot of depth but I did my best to explain the essential parts of my code). Additionally, I left a large amount of comments to explain every single step of what my code is doing (especially in the update functions as they are the most important ones). I also made sure to structure my document such that all sections are properly explained and easier to understand. One last obvious way of making my code readable is through my use of clear variable names so that there are no ambiguity in what they are used for.

#Professionalism

I followed all the conventions of research paper writing to create a very clear and thorough paper about my investigation. I also made sure to use Latex so that any mathematics can be appropriately formatted and added various labels to make all plots easily understandable.

#multimedia

I have used three media alongside each other to effectively explain my code. Firstly, there is the code itself with a lot of comments which I explained with another media which is my own voice. By going through the code and explaining the essential parts of it, I was able to make it much more understandable and clear. Lastly, and most importantly, I made extremely clear animations to demonstrate the state of the system very clearly at every single step. I used colors to represent the speeds of cars (darker is faster, red is negative velocity, blue is positive velocity, and white is zero velocity), and the locations of roads (in black), individual lanes (separated by yellow lines), traffic lights and pedestrians.