Traffic Simulation

Rita Kurban - 15 February 2018

Part 1: Traffic jams on a circular road

Rules

For Part 1, I implemented the single lane, variable speed traffic model as described in Nagel, K., Schreckenberg, M. (1992).

The update step of this model is based on the following rules:

1. Accelerate +1 if the distance to the next car is bigger than the current speed.
2. Slow down to distance - 1 if the distance to the next car is smaller than the speed.
3. Randomly decrease the velocity by one if the car moves.
4. Move each car forward.

Analysis

Traffic flow rate is defined as the average number of cars passing between two particular cells per unit time. I ran this simulation to analyze how the overall average traffic flow rate varied with traffic density and created a few visualizations which illustrate how the state of this model changes over time.

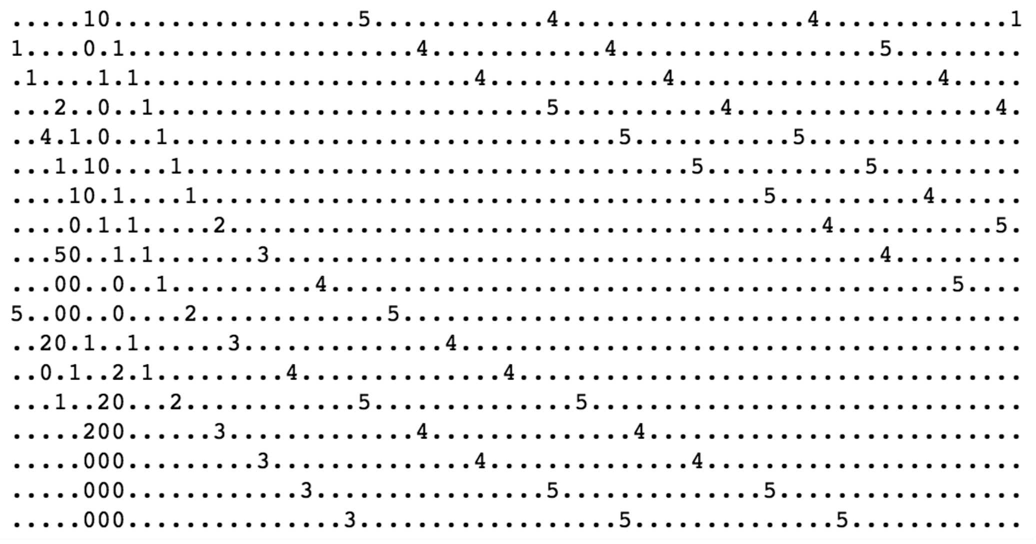
On the low densities, the traffic is smooth, and it is possible for cars to move freely. The rows represent the steps of the simulation in chronological order, the cars move from left to right.

```
..............1............................5..........................
................2..............................4......................
...................3...............................4..................
......................3................................4..............
.........................3.................................4..........
.............................4..................................5.....
.................................4..................................4.
...5..................................5...............................
........5..................................5..........................
............4...................................5.....................
.................5...................................5................
.....................4...................................4............
..........................5...................................5.......
...............................5...................................5..
..5.................................5.................................
.......5................................4.............................
...........4.................................5........................
...............4..................................5...................
```

Figure 1. Traffic simulation with initial density 0.01

The velocity can only decrease due to random events specified by the parameter *prob*.

However, as the density increases, the probability of traffic jams increases as well. The cars start to slow down following the second rule. At the density of 0.1 first visible traffic jams begin to appear (Figure 2).



```
.....10.................5............4.................4.............1
1....0.1....................4............4..................5.........
.1....1.1.......................4............4..................4.....
...2..0..1...........................5...........4..................4.
..4.1.0...1...............................5...........5...............
...1.10....1...................................5...........5..........
....10.1....1.......................................5..........4......
....0.1.1.....2.........................................4...........5.
...50..1.1.......3..........................................4.........
...00..0..1..........4...........................................5....
5..00..0....2.............5...........................................
..20.1..1......3..............4.......................................
..0.1..2.1.........4..............4...................................
...1..20...2............5..............5..............................
.....200......3.............4..............4..........................
.....000.........3..............4..............4......................
.....000............3................5..............5.................
.....000...............3..................5..............5............
```

Figure 2. Traffic simulation with initial density 0.1

The situation becomes even worth as the density increases. Figure 3 illustrates how the overall average traffic flow rate varies with traffic density for different densities in the range *np.arange(0.0, 1.05, 0.01).*

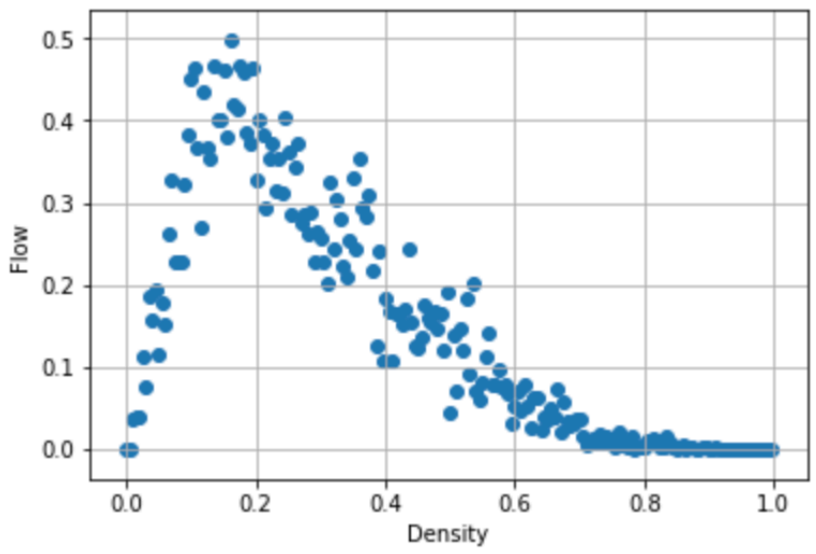


Figure 3. The overall average traffic flow rate with different traffic densities

As we can see from the plot, the traffic flow rate first sharply increases because more cars are being added to the simulation, picks at around 0.18, and gradually decreases until it reaches 0. It can be easily described by the fact that with higher densities traffic gets more congested. And so, the initial density of 0.18 is a threshold value which enables the highest number of cars to move without much congestion.

Part 2: Multi-lane highways

Parameters and Rules

In the second part, I implemented the multilane, symmetric, uni-directional, variable speed model which was described in Rickert, M., et al. (1996).

It has the same parameters as the single lane model (length of the road, initial density, maximum velocity, and the probability of the velocity do decrease by 1), as well as a set of new parameters: the probability to switch the lanes, the number of lanes, *L, L0,* and *L0(back)* which state how far ahead the cars should look forward on their and forward and backward on the other lane. They also use the *gap* which is defined as the distance to the next car - 1, or the number of empty cells in front of the car.

All the cars are being updated simultaneously. To do so, we first check if it is necessary to switch the lane. The first rule is that the car should look forward and check if the gap is bigger than *L* (how far to look ahead on your lane). In case this gap is smaller, the car performs the same procedure for the second lane. If there are enough empty cells there (>*L0*), it proceeds to the third rule, where it looks backward and decides if a change of the lane would disturb the traffic on the second lane. If the gap is bigger than *L0(back),* the car should switch the lane. The last rule introduces the probability of switching the lane if all the rules were met. If a randomly generated number is smaller than *switch*, the car will change the lane. After that, the cars' velocities are being updated according to the single lane model rules.

Assumptions

This model relies on several assumptions. First of all, it assumes that cars on different lanes move in one direction which is not necessarily the case. Secondly, it assumes symmetry: cars have the same likelihood of switching to the right and to the left lane which is not the case for some countries. In case of multiple lanes (>=3) the cars first check the lane to their right and only in case it's impossible to switch there, check the line to their left. My reasoning behind that is that in many countries the left lane is considered to be the passing lane and it is often illegal to use the "far left" on a major highway as a traveling lane. Thirdly, the cars always check for the cars above and behind them which makes crashes impossible for this implementation. It also uses periodic boundary conditions which means that the road is circular and there is no way for new cars to enter the simulation. The model also assumes there are no traffic lights, junctions, and turns which makes it a little bit unrealistic. Despite this assumptions, it is still useful for understanding fundamental relations between microscopic rules and macroscopic measurements and can give insights into how traffic changes over time.

Analysis

Similar to the single lane model, on the low densities, the traffic is smooth, and it is possible for cars to move freely. They also rarely change their lanes.

```
      -
........................5.............................................
......................................................................

.............................5........................................
......................................................................

.................................4....................................
......................................................................

.....................................4................................
......................................................................

.........................................4............................
......................................................................

..............................................5.......................
```

Figure 4. Traffic simulation with initial density 0.01

As the density increases, the probability of traffic jams increases as well. It means that cars start changing the lanes and/or slowing down.

```
............5...3.........4..2...2.....4...........3.....4......2.....
...............4....3...10...2..........4...........................5.

..............2....3........2...3...3.......5..........4......5....3..
..4................4..2.00.....2.............5........................

.4...............3.....4.......3..2.....4.......4..........4..........
......4..............2.10.1......2...............4................4...

......5..............4....4....2...3.......5......4...........5.....
.5........4...........100...2.......3.................5...............

...........5..............5.....5..2....3.........5.....4...........4.
.....4.........5......00.1.....3.......3..................4...........

...5...........4..............4...2..2......4.........4......5........
.........4..........5.0.1..2......3........4..................4.......
```

Figure 5. Traffic simulation with initial density 0.1

As the density increases, cars cannot switch to a different lane anymore, and so the traffic gets congested. The overall pattern stays the same and is illustrated in Figure 6. The flow increases to approximately 0.2 which shows a slight increase in the flow compared to a single lane. However, this change is insignificant because we calculate the flow as the average of the flows on two lanes. One interesting observation is that the variance between different runs decreases for multiple lanes which can be explained by a more even distribution of cars in such cases.

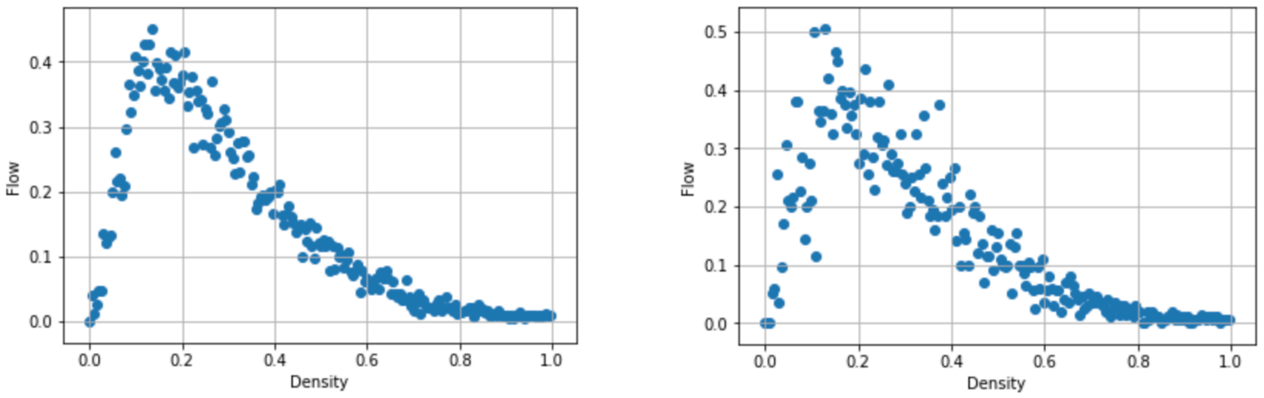


Figure 6. a) The overall average traffic flow rate with different traffic densities for two lanes compared to b) Figure 3.

Key Questions

As I've mentioned in the assumptions, this model supports multiple lanes. Figure 7. demonstrates an example visualization a 5-lane traffic model. From the figure, we can see that the density is around 1% as we have 6 cars over 500 cells. Cars move freely, and, again, the flow rate doesn't change significantly with the number of lanes. It starts to decrease after the initial density of 0.2 until it reaches 0 when the density is close to 1. At the same time, the variance decreases even further because the program averages the flow over multiple lanes and the flow itself is evenly distributed across these lanes.

```
........................................................................
........................................................................
..........................2.............................................
..1........................4......................................1.....
.4.....................3................................................

........................................................................
........................................................................
.............................3..........................................
....2...........................5...................................2...
......5...................3.............................................

........................................................................
........................................................................
.................................4......................................
......2..............................5.................................3
..........4..................3..........................................
```

Figure 7. 5-lane traffic simulation with initial density 0.01

This model follows strict rules, and that's why it's very difficult to apply it to chaotic Hyderabad. People in Hyderabad don't take into account the lanes and can switch them back and force as well as drive in between them. One significant feature of Indian rides is that they have different agents: not only cars but also people, motorcyclists, cows, etc. Our simulation assumes rational behavior and makes traffic accidents impossible. However, according to Hyderabad traffic police, more than 354600 accidents happened in 2017 alone and this number remains very high over time. Another important feature of Hyderabad traffic is its dynamic change: the density is much higher in the morning and the evening. Unfortunately, the simulation doesn't support a dynamic change in the density because the number of cars remains constant. One more issue is the quality of Indian roads. Despite the fact that millions of rupees are spent by the highway agencies in extensive pothole patch repairs, there are still a lot of roads that are in terrible condition. Of course, this fact influences the overall situation on the streets and significantly decreases the flow rate. In the stretch goal, I extended my model to account for bad roads with speed limits and implemented traffic lights.

Stretch Goals

To make my model more realistic, I implemented "bad roads." This feature works in the following way: the user has to specify the probability of having bad roads. After that, the model randomly chooses an index where the bad road starts. Then, it calculates the length of the bad road by multiplying the parameter with the length of the road. The resulting number is the last cell of the bad road. After that, the model decreases the maximum velocity of every cell on the bad road which means that the cars cannot speed up there. As a result, I get an array of maximum velocities for every single cell. It is divided into "good road" which has the maximum velocity of 5 and "bad road" with the maximum velocity of 2.

I also implemented traffic lights by adding a boolean variable *Green* and splitting the speed limits array into two new arrays: one for green lights, one for red.When the traffic lights are green, the cars can move freely. However, after a certain period, it turns red and so the cars have to wait which leads to the formation of traffic jams which start to disappear once the lights turn green again. This situation is partially illustrated in Figure 8. In this simulation, the lights were changing every five steps. As a result, you can see how the jam from two traffic lights which are formed at the top of the picture start to resolve once the light changes. You can also see that all the cars to the right of traffic lights move very slowly and only start to speed up when they enter the good road to the left of the traffic lights.

```
.....................000.........1.100.........1.1...2..........2..2..
........4.........1..200....0...2..2.0.........2...2..2.......1....2..

3....................000..........1000..........1..2...2.........1....
3...........4.......2000.....1....20.0...........2...2..2.......2.....

....4................00.1.........000.1...........2..2...2.........2..
...3..........2.....000.1......2..0.1.1............2...2..2......1....

3........5...........0.1..2.......00.1.1............2..2..1...........
......3.........2...000...2......2.1.1..2............2...2..2......2..

....4.........5.......1..2..2.....0.1.1..2............2..2..2.........
3........3........2.00.1....2.....10...2..2............2..1...2.......

........4.......2.......2..2..2....1.1..2..2............2..2..2.......
....4........4.....10.1..2....2...0.1....2..2............2..2...2.....
```

Figure 8. Traffic simulation with speed limits and traffic lights

Future Work

In the future, I will also implement obstacles and lane merges. To do so, I will add them to the speed limit arrays. In comparison to traffic lights which are present on all the lanes, the obstacles will only be present in some lanes (accidents) so that other cars will have to go around them. This will require modifying the lane changing code.

Another aspect of the simulation that will make it more similar to the real-life Hyderabad traffic is implementing bad drivers' behavior. By adding more randomness into their behavior (such as unexpected breakdown), playing around with the lane switching strategies (such as not looking backward), and increasing speed limits, it might become possible to model accidents.