Assignment 2

Jacob Puthipiroj

February 22, 2019

Nagel-Schrenkerg (Single Lane) Models

Cellular Automata models of traffic flow began with the work of Nagel and Schrenkenberg (1992). In the Nagel-Schrenkenberg model, the road is divided into cells, each possibly containing a maximum of one car. To model a circular road, the boundaries are considered periodic; cars exiting the road on the right automatically reenter the space on the left. Furthermore, time is discretized into steps, and at each time step, cars follow an update rule:

1. **Acceleration:** If the car is not at its maximum velocity, increase its velocity by one.

2. **Deceleration:** If the distance to the car in front is less than the current car's velocity, decrease the speed to exactly the distance to the car in front to avoid a collision.

3. **Randomized Slowdown:** If the car is not static, reduce their velocity by 1 with probability p. This is to model random behaviors on roads, such as hard braking.

4. **Motion**: The cars now move according to their new probabilities.

By convention, states of cars on the road are displayed between steps 3 and 4. An example is shown below.
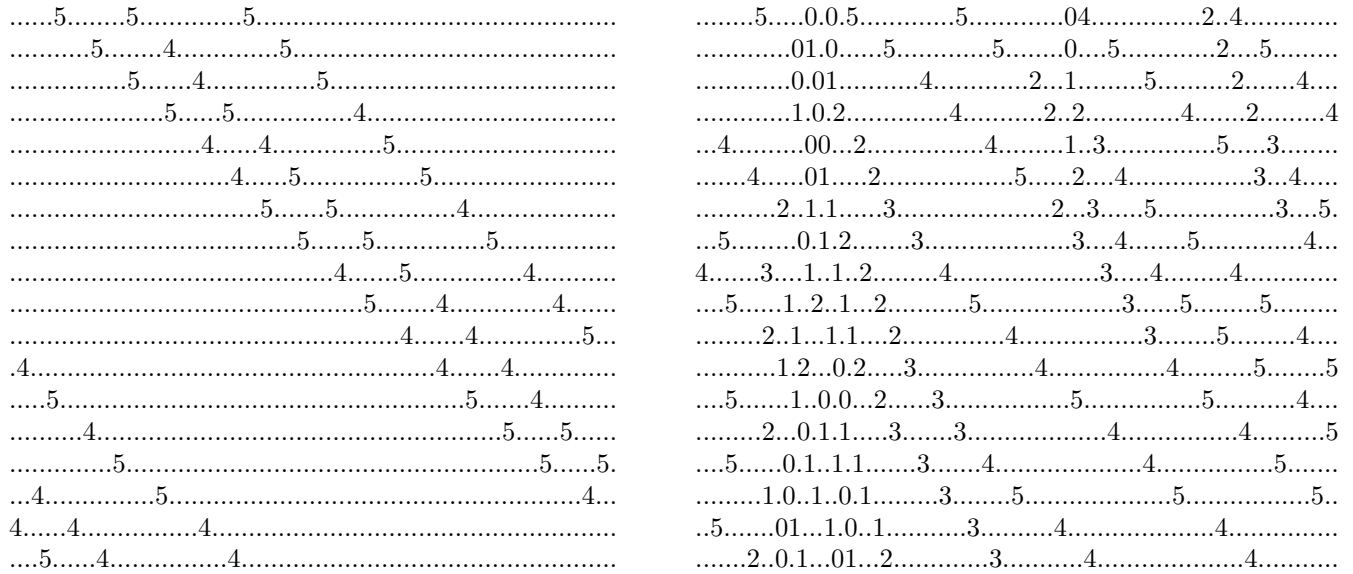
```
......5........5.............5....................          ........5.....0.0.5.............5.............04..............2..4...........
..........5.......4............5..................          ............01.0.....5.............5........0...5............2....5........
...............5......4...............5...........          ............0.01..........4............2..1.......5.........2......4....
....................5.....5..............4........          ............1.0.2.............4..........2..2.............4.......2.........4
.........................4......4...............5.          ...4..........00...2................4.........1..3................5....3........
..............................4......5..............5.       ......4.....01.....2................5......2...4................3...4.....
...................................5......5.............4.   ...........2..1.1.....3................2...3.....5..............3...5.
........................................5.......5.........5. ...5.........0.1.2.......3.................3....4........5.............4...
..............................................4......5....4. 4......3....1..1..2........4.................3....4......4............
...................................................5.....4....4. ....5......1..2..1...2........5.................3.....5.......5........
........................................................4......4.....5.. .........2..1...1.1....2.............4............3........5........4....
.4.............................................................4......4.. ...........1.2...0.2.....3...............4..............4.........5........5
.....5............................................................5......4. ...5......1..0.0...2.....3..................5...............5.........4....
.........4................................................................5.....5..  .........2..0.1.1.....3......3................4.............4.........5
.............5............................................................5......5. ....5......0.1..1.1.......3.......4.................4.............5.......
...4.............5.............................................4... .........1.0..1..0.1........3......5...................5.............5..
4.....4.........4..............................................  ..5.......01...1.0..1..........3.........4...................4...........
....5......4............4..........................             .......2..0.1...01...2............3..........4...............4..........
```

Figure 1: Both figures show cellular automata simulations with a periodic (circular) road of length 80, a maximum velocity of 5, and $p = 0.5$. While cars slow down randomly. on the left, the sparsity of the open road prevents a traffic jam from forming. On the right, with identical parameters but on a denser road, cars within the jam slow down and prevent the jam from clearing. Instead, the jam slowly moves left throughout the simulation.

Nagel and Schrenkenberg used flow —defined as the number of vehicles passing a reference point per unit of time— to compare between various cellular automata models. Intuitively, a model with a very high density should have roads so congested that few cars can move freely through the reference point, and thus have low flow, while models with too low a density should also have very few cars moving through the point. Thus the density of maximum flow should lie somewhere in between.
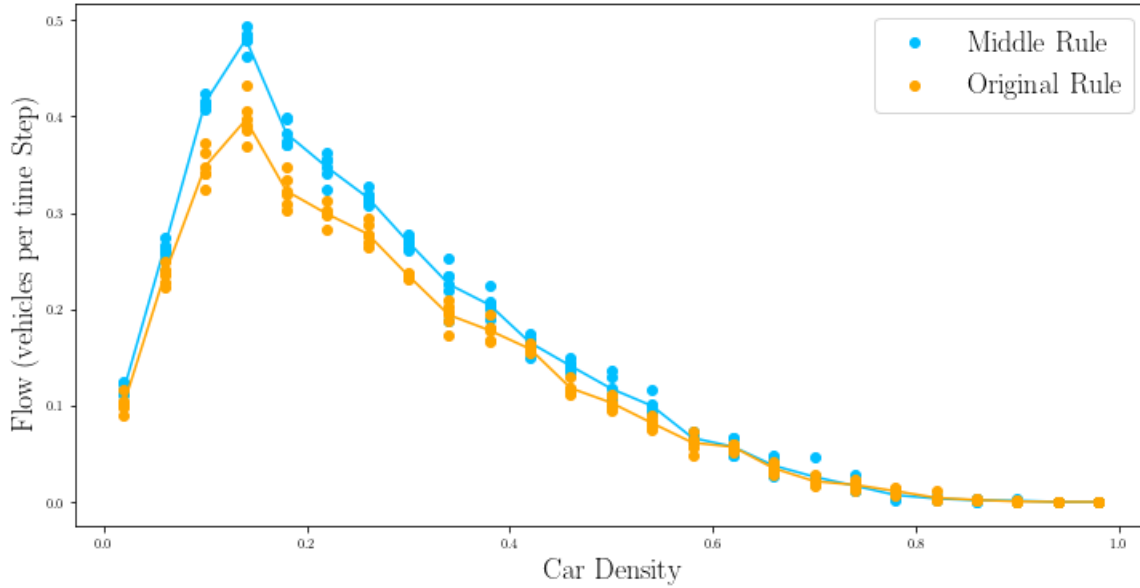
Figure 2: This shows cellular automata simulations with a length of 100, a maximum velocity of 5, $p = 0.5$, Flow, as a function of car density follows an inverted V shape, peaking sharply at a density of around 0.15, before gradually becoming less efficient. When cars follow the 'always in the middle' rule, the model achieves a peak flow of 0.5 cars per time step.

A further augmentation of the algorithm is for cars in the model to aim to be around the same distance from the car in front as to the car in the back. This is slightly more complicated, but yields better flow results, and is as follows:

1. **Acceleration/Deceleration:** Check the distance to the car in front, and to the car at the back.

   (a) If the distance to the car in front is greater than the distance to the car in the back, accelerate the car by one if the car is not already at the maximum speed, and if the car will not exceed the distance to the car in front.

   (b) Otherwise, reduce the speed by one if the car is not already static, or reduce the speed to the distance to the car in front, whichever is less.

Rickert (Multi-Lane) Models

In this section, we implement Rickert's (1996) extension to the Nagel-Schrankenberg model. Compared to a single lane road,
Visualizations over time.
Analyze how much more traffic can flow through a multilane compared to a single lane model (Plot), latexify

2

Model Comparison

How applicable is the model to Buenos Aires? - Buenos Aires is a city with wide, open streets.
How can bad driver behavior be encoded in our algorithm? - Define bad driver behavior
Extend the model to include speed limits, traffic lights, obstacles, and lane merges.

Future Work

References

Appendix

Single Lane Traffic Flow

The code used in this section is available here.

MultiLane Traffic Flow