

Simulation of multi-lane highway using cellular automata model

Erik Åsgrim

December 14, 2022

Problem description

A multi-lane highway is going to be simulated using a cellular automata model, in order to study how the amount of lanes affects the flow rate for different car densities. We would also like to study the fluctuations of the flow rate, and investigate whether varying the amount of lanes can affect the fluctuations of the flow rate.

Model

The model of the highway was implemented as a cellular automata on a road with some integer distance L where each car on the road occupies 1 length unit of the road. The road does not have an end nor a beginning meaning that the distances to the surrounding cars were calculated as period distances. In order to make the dynamics of the simulation more realistic all cars were given personal maximum velocity $v_{i,max}$. The value of $v_{i,max}$ for each car was randomly generated as a rounded value taken from a Gaussian distribution around some mean value μ , where μ would represent the speed limit on the road. A Gaussian distribution of the values of $v_{i,max}$ thus seems logical since in real life most cars drive at approximately the speed limit even though a few cars drive much faster or slower.

The rules for the cellular automata were performed in two different parts at each time step. First, rules determining what lane changes are going to be performed are implemented. This was done by first calculating desired lane change of each car and then performing the desired lane change only if the lane change can be performed safely. Secondly, rules determining how the various cars are going to change their velocity are implemented. Each rule was implemented simultaneously on each car before moving on to the next rule. This means that the ordering of the rules is very important, as latter rules can 'override' previous decisions. In order to make the rules more compact we also introduce the following notation

- x_i is the position of the i :th car
- v_i is the velocity of the i :th car
- $v_{i,max}$ is the maximum velocity of the i :th car.

- v_{back} is the velocity to the next car backward in the **new lane** we would like to change to.
- $d_{forward}$ is the distance forward to the next car in the same lane we are currently in.
- $d_{forward,left}$ is the distance forward to the next car in the lane to the left.
- $d_{forward,right}$ is the distance forward to the next car in the lane to the right.
- $d_{backward}$ is the distance backward to next car in the **new lane** we would like to change to.

Using the above notation the rules of the cellular automata were implemented as follows:

Lane changes

1. If $v_i < v_{i,max}$ and $d_{forward,left} \geq d_{forward}$ and there exists a lane to the left, desire a lane change from right to left
2. If $d_{forward} > v_{i,max}$ and $d_{forward,right} > v_{i,max}$ and there exists a lane to the right, desire a lane change from left to right
3. If $d_{forward,right} > d_{forward}$ and no previous lane change desire exists, desire a lane change from left to right with 5% probability
4. If $d_{backward} > v_{back}$ perform desired lane change (car behind should not need to brake because of lane change)

Velocity changes

5. If $v_i < v_{i,max}$ increase velocity as $v_i \rightarrow v_i + 1$ (try to accelerate to maximum velocity)
6. If $v \geq d_{forward}$ decrease velocity as $v = d_{forward} - 1$ (avoid collisions)
7. With some probability p decrease velocity as $v_i = v_i - 1$ (cars might brake randomly)
8. Update position as $x_i(t+1) = x_i(t) + v_i$

Most rules above are quite logical, and result in a behaviour of the cars that appears natural. Rule (3) might however need some clarification. Without rule (3) mostly the left lane becomes occupied for higher car densities. In order to solve this rule (3) creates a bias towards changing to the right lane which solves the problem.

Problem

Method

Results

Discussion