Traffic Model

New Mexico

Supercomputing Challenge

Final Report

April 4, 2018

SFHS122

Santa Fe High School

Team Members:

Elijah Sandoval

Omar Lux

Teacher:

Mr. Smith

N/A

According to the Association for Safe International Road Travel, approximately 3,287 people die in car-related incidents each day. That's around 1.3 million people annually, making car crashes the 9th largest cause of death globally. When we began deciding which problem we wanted to take on with our model, we realized that something to do with traffic or cars in general would make a lot of sense with the language we are using, netlogo. Seeing how massive of an epidemic car crashes are, it was an obvious choice to focus our simulation on them.

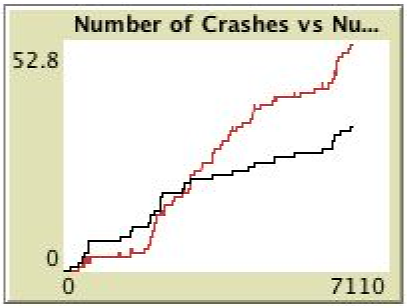
In order to be fair to the serious issue of car deaths, it was important to make the model as realistic as possible. We bagan the traffic model by creating a two dimensional intersection for our cars to interact with and move around on. Once this was set up, we created our cars by using Netlogo's turtles, and placed them in the proper lanes. One of the first problems we discovered was that since the Netlogo world is looping, we could only have the same cars on the screen forever, something that completely broke the realism the program needs. After a few days of trial and error, we found that we could have the cars disappear when they hit the border of the screen, and have a random number decide when and where to spawn new cars. Once the basics were set up, we spent around two months coding the traffic lights and how the cars reacted to them. We created the procedure of traffic light patterns using the tick system in Netlogo. When the program hits a certain time or number of ticks, the light will change. After we set up the lights, we spent a few weeks making sure that the cars could react to them. When they hit a white line, they check around themselves to see what color light they have. There is a random number generator that decides whether they will turn right or not. Left turns are governed by turning signals (turtles with their shape set to arrows that change their color at certain tick numbers). Cars wait at lights until they turn green, everything is very uniform at this stage. A few months into the project, we had our signals and how cars react to them, it was time to create the problem. We set up another random number generator to decide whether cars would run red lights or not. Because this rarely happens in most places, we set the probability extremely low. When this did happen, we created a system where if a crash occurred both parties would turn bright red and move to the side of the road to indicate it. We then created yet another randomizer that decided whether any fatalities occurred in the crash. If they did, the car would turn black rather than red. In order to best gather data, we created a graph within netlogo which tallies the number of deaths and compares it with the number of deaths.

The problem our group faced was to model traffic in realistic conditions.

Our solution was to model an intersection. We wrote code that enabled cars to discern between green and yellow arrows, as well as green, yellow, and red lights. The cars randomly choose a path to travel. We intentionally coded for crashes and deaths to occur occasionally.

We confirmed the accuracy of our model by analyzing real traffic intersections and comparing the realisticness to that in our model.

We collected data on the number of crashes versus the number of deaths.



Number of Crashes = Red

Number of Deaths = Black

Our most significant achievement was that we were able to simulate randomness by using the random number generator in NetLogo to make the turtles either turn or go straight.

<u>Source for crash statistics:</u>

http://asirt.org/initiatives/informing-road-users/road-safety-facts/road-crash-statistics