Calculating a Heuristic Route for Navigating Traffic

New Mexico

Supercomputing Challenge

Final Report

April 1, 2015

Team Number 118

Portales High School

Team Members

Nick Brown

Corey Gutierrez

Zachary Laurenz

Christopher Leap

Teacher

Jack Willis

Executive Summary

In big cities finding the quickest route from point A to B can be almost impossible because of the seemingly endless amount of routes. Because of this massive amount of routes even a computer would have to take time to find the optimal route. Despite this the amount of time needed to find the quickest time can be reduced using algorithms. There is a large amount of algorithms that can calculate the most efficient route.

The goal of this project is to find the best algorithms for and to possibly create our own to calculate the best route. The algorithms will have to factor in traffic and the distance that the route will take, as well as other. There will be several different algorithms that will all be rated on their operating efficiency (the time it takes the algorithm to run on average) and the efficiency of the route (how much time the route takes to navigate). Once these algorithms are created they can be utilized to look at real traffic and route optimization.

Summary of the Problem

Many traffic buildups can be attributed to an over-used, well-known road or intersection. Many people go to the same intersections because it's the most efficient route for an individual, not accounting for all of the other cars on the road. Because most route-navigation programs calculate the shortest route, the problem of traffic buildup is intensified. If, however, the system is omnipotent, it will know where traffic buildups are occurring and will be able to avoid these buildups, saving copious amounts of time.

How we are solving the problem

We divided this problem into four main parts:

1. Simulation of traffic
2. Route navigation
3. Route navigation using knowledge of stoplight timers and the location of other cars
4. Use of multiple instances of the program to network between cars and further optimize the heuristic route.

Simulation of Traffic

Our simulation has cars moving through a sample city making randomly calculated decisions at each intersection, all while obeying traffic laws and not running through each other, using collision detection. We are working to implement randomly generated goals for the cars to navigate to, replacing the random turn method.

Route Navigation

For route navigation, we have a car spawning in a randomly generated location and a destination that is also randomly generated. We then have the car navigating through the “city” using a minimized route.

We are working to implement the knowledge of the other cars’ locations and stoplight timers, as well as networking into our program.

Results

As of now, we have completed two of four parts of creating a solution for the problem. Over the course of the past few months, we have been laying the groundwork and building blocks for the later parts of the program. As of now, we created a program that makes a variable amount of cars navigate through a road while abiding all traffic laws and laws of physics. We have also prototyped a car that is able to navigate through an empty city with sub-optimal speed that can easily be implemented into our populated-city program. As we continue to work, we plan to use the timing for stop-lights and the location of other cars to help direct the main car into a more heuristic route. In the future, we also hope to be able to have multiple “intelligent” cars that network their destination and current paths to further optimize the route.

Most significant achievement

Our most significant achievement over the course of the project was gaining much more experience in programming, coding logic. We came in to the competition with programming knowledge, but we gained much more throughout the project. Programming a simulation like this involves massive amounts of thinking and reasoning, which we gained much experience in by participating. A huge part of learning to code is doing a ton of programming. Opportunities like this competition are great for gaining exposure to programming and coding logic.

References

- Accidents slow traffic on I-25, I-40 | Albuquerque News - KOAT Home. (2013, August 26). Retrieved from http://www.koat.com/news/new-mexico/albuquerque/accidents-slow-traffic-on-i25-i40/21657982
- Traffic Engineering — City of Albuquerque. (n.d.). Retrieved from http://www.cabq.gov/municipaldevelopment/our-department/traffic-engineering
- Netburn, D. (2012, August 31). Tech savvy: Smartphone apps can help drivers navigate traffic - Los Angeles Times. Retrieved from http://articles.latimes.com/2012/aug/31/business/la-fi-tech-savvy-traffic-apps-20120831
- "NetLogo Dictionary." NetLogo 5.2-RC2 User Manual:. Netlogo, n.d. Web. 31 Mar. 2015. http://ccl.northwestern.edu/netlogo/docs/dictionary.html

Acknowledgements

We would like to acknowledge Mr. Willis, our teacher and team advisor. We would also like to thank Matthew Curry and Jenelle Zamie, our Evaluation day judges, for their constructive criticism, as well as Teresa Roberts, for reviewing our proposal and giving us suggestions for how to start our program. We would also like to thank the Netlogo developers, for all of their hard work creating the Netlogo programming ide, and writing a well-made reference page, which was used often.