TrafficCOP (Controller Optimizing Productivity)

Justin Simerly

Computer Science / Cognitive Science
University of Evansville

Sponsor: Dr. Anthony Beavers
Advisor: Dr. Deborah Hwang
4/30/2014

ABSTRACT

Traffic congestion data indicates that our current traffic controller systems are unable to meet the needs of the growing number of commuters, resulting in unnecessary costs and environmental pollution. There are a couple of traffic controller systems currently implemented or being tested that attempt to fix this issue, but a great deal more research and testing needs to be put into this expanding field in order to increase traffic productivity without decreasing the commuters' safety. In an attempt to help with this issue, the TrafficCOP (Controller Optimizing Productivity) project aims to design a traffic controller system in which each individual traffic controller will be realized through an artificial neural network that utilizes a learning algorithm with the intention of decreasing the aggregate average commuters' travel time while ensuring the safety of the commuters.

ACKNOWLEDGMENTS

I would like to thank Dr. Deborah Hwang for advice on specific questions throughout the full process of this project and Riley Mayr for his help with the design of images in my presentation and poster. In addition, I would like to give a big thank you to Dr. Anthony Beavers, who provided access to the Digital Humanities Laboratory for the use of its computers.

INTRODUCTION

Recent studies on traffic congestion have shown that the average delays (and costs) experienced by commuters has grown over the past couple decades. It has grown more apparent that our current traffic controller systems do not meet our increasing needs and thus, some traffic engineers have started to develop alternate methods. Although the introduction of adaptive traffic controller systems has made great progress in traffic flow, there is still an urgent demand to reduce the costs of traffic congestion. One possible solution would allow each intersection controller to implement a neural network architecture in order to recognize and respond to patterns of traffic. In an effort to create a better controller, this project attempts to model a traffic controller system using ideas such as adaptive systems and artificial neural networks and then compare it to the systems that are commonly used today. The results from this project show that TrafficCOP exhibited a significant increase of efficiency in cases where traffic congestion is high, indicating that the implementation of this type of traffic controller system should be considered in the future.

PROBLEM STATEMENT AND BACKGROUND

The issue of traffic congestion has become a growing concern in recent years as the roads are having to handle more commuters. In the 2012 Mobility Report it was reported that the average commuter endured 38 hours of traffic delay per year, almost a 150% increase from 16 hours in 1982. Not including the additional cost of environmental pollution, this delay has cost the US more than $120 billion, which is equal to over $800 per commuter. One of the revealing parts of this report, though, is that 40 percent of the delays occurred during hours that traditionally are not considered “rush hour”, indicating that our traffic control systems are unable to handle even normal traffic loads [3]. One of the largest areas of concern with traffic control are intersections; in fact it has been shown that a person on average will spend 6 months of their life waiting at a traffic light [9].

Traffic Controller Efficiency

Before continuing, it is important to establish how the efficiency of a traffic controller is measured. The common measurement of an intersection's efficiency is referred to as lost time, which is defined as the time during which an intersection is not using all of its resources to meet all the needs of its commuters. Due to the large impact of varying speeds of commuters on the calculations of lost time, there is a lot of variability in its calculation and thus it will not be utilized in this project. A more accurate measure of efficiency can be found with the amount of extra time a commuter experiences as it progresses through the intersection, defined as the commuter's wait time. It easily can be shown that the lower an intersection's lost time, the lower the aggregate wait time for all the commuters at that intersection. When comparing the efficiency

Int	Time	N-L	N-T	S-L	S-T	W-L	W-T	E-L	E-T
1	12	R	R	R	R	G	R	G	R
2	4	R	R	R	R	Y	R	Y	R
3	1	R	R	R	R	R	R	R	R
4	25	R	R	R	R	R	G	R	G
5	4	R	R	R	R	R	Y	R	Y
6	1	R	R	R	R	R	R	R	R
7	14	G	G	G	G	R	R	R	R
8	3	Y	Y	Y	Y	R	R	R	R
9	1	R	R	R	R	R	R	R	R

Table 1: An example of a phase plan.

Each indicator is signified as 'x-y' where x symbolizes the direction traffic is coming from (North, South, East, West) and y symbolizes the traffic path (Left, Through).

of multiple intersections, it is important to compare a commuter's wait time to its total time driving since commuters will have different distances that they travel [5]. Thus for this project, the efficiency of a traffic controller will be measured by the following equation:

(Σ (1 – (wait-time / total-time)) *for every commuter*) / number-of-commuters (Eq. 1)

Current Traffic Controller Systems

When it comes to 4-way intersection signals, there are several different varieties. However, the most common are stop signs, regular traffic lights, and smart traffic lights. Stop signs (which also can be represented by blinking red traffic lights) have the following rules: 1) All traffic must stop at the intersection. 2) The first commuter to arrive at an intersection gets the right-of-way. 3) In the case of a tie, the commuter to the right gets the right-of-way. [7] Stop signs have the benefit of being very low cost and are adept at handling small volumes of traffic. However, disadvantages of this system include that it requires every commuter stop, thus decreasing the intersection's efficiency.

In order to talk about the other two traffic controller systems, there is some terminology that must be explained. Each possible path that a car can take at an intersection is called a movement. At a regular four-way intersection, each of the four directions (North, South, East, and West) has three different options (Left, Through, or Right), and so there are twelve total movements available, which are shown in Figure 1. Movements can have three different states: stop, changing, or go. These states are communicated to the commuter by a signal indication (the actual lights), although not all movements have to be explicitly given by a signal. For example, when a light is green (indicating that the traffic may proceed in the "Through" direction) the "Right" movement also is allowed. One or more movements can be combined into a set of movements that is called a phase, and phases are displayed for a set amount of time referred to as an interval.

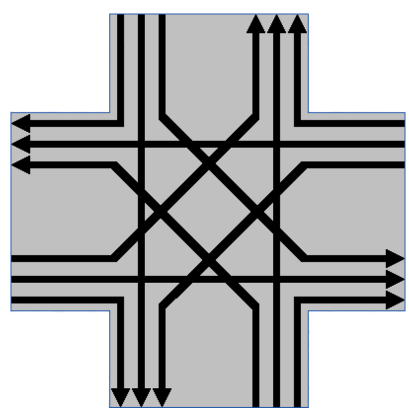

Figure 1: An illustration of all twelve possible movements for a four-way intersection.

Regular traffic lights, the next common type of traffic control systems, work by storing a table of intervals, an example of which can be seen in Table 1. Indicators for each movement (listed at the top of the columns) go through each interval for the given amount of time and repeat on a cycle. Over time, traffic engineers have developed a set of equations that calculate these intervals based on the expected volume of commuters per phase. Although this system of traffic signals allows cases where a commuter does not have to stop at the intersection, and thus improves efficiency, it also has cases where a commuter may be stuck at a red light with no other traffic at the intersection for a long time.

Many modern-day smart traffic lights use sensors either in the road or on the traffic light poles to sense when a car has arrived at the intersection. This type of smart traffic light system combines information from these sensors with a phase table similar to the one described above with the addition of being able to handle multiple possible paths through the phase plan. One common use of this system is to not turn on the left turn signal for a direction if there are not any commuters in the corresponding left turn lane. Due to their ability to adjust for certain cases, smart traffic light systems tend to be more efficient than regular traffic light systems. [5]

Adaptive Traffic Controller Systems

Unfortunately, each of the above signal control types are static, meaning that they have to be reprogrammed in order to adapt to changing traffic demands. Thus, there has been a strong push in recent years towards completely adaptive signal systems. Instead of relying on retrieved traffic data from previous observations, adaptive systems use real-time traffic data to determine the timings of intervals. Smart traffic lights are similar to adaptive traffic control systems since they use current sensors to change their output. However, this output is still greatly constrained to static data similar to what was shown in Table 1. Current traffic data can be collected by a growing number of technologies including smart-phone pinging, overhead sensors, and video cameras. As a result, the ability to know the location of every car on the streets is becoming more realizable. This makes adaptive signal controls very enticing and several have already been created and implemented. SCATS (Sydney Coordinated Adaptive Traffic System) was one of the first of such systems that was created over 40 years ago and has grown now to be able to cover as many as 16,000 intersections. It uses a centralized computer system to manage all the traffic controllers based on information received from various types of sensors. This approach requires no pre-calculations by the programmer and greatly increases the ability of the signals to adapt to the current traffic. [1]

SCATS and other centralized technologies have many positives to them; however, there are also some negatives, the biggest of which is that they force the complexity of handling massive amounts of data onto one system. While the centralized ideology of one main unit leading everything else is attractive, there has been much research recently into the merit of decentralized systems. This idea of moving the work from one centralized unit to each of the decentralized parts of a system has come up in a variety of different use cases including even traffic controller systems. In 2012, Carnegie Mellon started a pilot study on the effects caused by the implementation of the Scalable URban TRAffic Control (SURTRAC) by its Traffic 21 research initiative. Nine months into the study that involved nine traffic signals modified to sense their current traffic data, communicate with one another, and collaboratively adapt to real-time traffic conditions, researchers have seen some very positive results including reductions of 40 percent in vehicle delay time, 21 percent in projected vehicle emissions, and over 25 percent in travel time. The study also showed another strong positive with decentralized systems: each additional traffic light can be added with almost no manual adaptation of the current system (hence the “Scalable”). This allows for traffic engineers to be able to slowly build up traffic control systems and even easily connect multiple systems. Additionally, since each part runs on its own with minimal dependencies, decentralized systems lower the susceptibility to local failure and hacking. [8]

Artificial Neural Networks

When analyzing traffic data, particularly in real-time, one key part of a well-working traffic controller system is the ability to recognize patterns. While the traditional von-Nuemann architectures that our computers run on today often are poor performers at pattern recognition, a different type of computer architecture, called a neural network, has been shown to work exceedingly well in this area. Based on the structure of the human brain and its individual neurons, neural networks that are designed by humans are classified as Artificial Neural Networks (ANNs). The building block of an ANN is called a neuron, of which there are several models that have been created. For the type of neuron that will be used in this project, each of n inputs are first weighted to affect how much influence that input has on the output. Then the neuron functions by outputting the sum of all of its weighted inputs. When put in mathematical terms, the neuron's output is equal to the following:

$X_1W_1 + X_2W_2 + \dots + X_nW_n$ (X_i: input i; W_i: weight i) (Eq. 2)

The neuron's output then can become an input for any connected neurons. In order for neural networks as described above to work, there must be a learning method that changes the weights of a neuron's inputs and “points” the network towards a best solution. Reinforcement learning, which calculates a “return” after completion that represents how well the networks solution performed, is the best option for this project. [2]

PROJECT SPECIFICATIONS

The first specification for this project is that it should result in the development of a general traffic model in which different traffic signal systems that control a grid of intersections can be implemented. This required that the modeled commuters sufficiently model commuters in the real world, particularly that the commuters have varying speeds, have acceleration/deceleration, and slow down before making sharp turns. In order to obtain a complete analysis of this data, the overall trend of the traffic flow should be able to be modified in certain ways. These modifications include cases where the majority of traffic is going in only one direction, where there is a varying amount of traffic on the roads, and where there are different probabilities of a car turning at an intersection. Finally, a user running these models should be able to collect statistics about the system's efficiency.

After this model had been built, a traffic signal system that used networks for each traffic light in an attempt to lower the aggregate average wait time was designed. The system had access to data collected in real time which was used to implement a learning algorithm on the neural networks. For the purposes of these models, it is assumed that the the networks will always be able to detect the location of a commuter. While this is not very applicable to the real world at this time, the advancement of technology is moving closer to being able to do this and the purpose of the project is to show that this type of system would be beneficial to implement when our technology has reached this ability.

DESIGN APPROACH

This project was implemented with the modeling software NetLogo. The design of TrafficCOP used concepts taken from the analysis of safe movements and safe phases in order to develop a neural network that used input values collected from the observation of commuters in order to output the most efficient safe-phase given the current traffic conditions. Each traffic controller system was simulated with various traffic conditions to determine its respective efficiency and then compared.

NetLogo

A sizable part of this project involves making simulations that correctly model what actually happens in the real world. With this in mind, NetLogo was chosen as the modeling software. This program has been used by others before to analyze questions about traffic patterns, and its use of hundreds or even thousands of agents operating individually gives the user an accurate simulation of groups of objects acting independently. This program was already installed on computers in the Digital Humanities Laboratory, which has been setup to allow for running simulations in the background, giving the programmer the ability to easily collect data from long simulations. For collecting data, NetLogo gives the user the ability to be able to change input to the world through the use of user-built sliders, buttons, toggle switches, and input boxes to permit easy transitioning between different variables. In addition, data can be collected to organize data dynamically in a graph or table, allowing the user to quickly see how the simulation is progressing. NetLogo is made up of three agents that TrafficCOP uses: turtles, patches, and an observer. The observer is the main agent that sets up the NetLogo world and can be seen as the person writing the code. Turtles represent the commuters and move over the patches, which represent the roads. Special patches represent the light signals by communicating to commuters when to proceed through a given intersection. NetLogo gives patches the ability to know if a turtle is on it and thus each patch is the input for the neural network with the output shown through the light signals. NetLogo additionally allowed the programmer to give the user simple but meaningful control over any agent behavior, allowing a special flexibility that programs designed purely as traffic simulators do not provide. Perhaps most importantly, it was free, compared to some traffic simulation programs that can cost thousands of dollars. [10]

Safe Movements and Phases

TrafficCOP's design has the goal to keep commuters safe while increasing the traffic efficiency compared to traffic signal systems currently implemented. The previously mentioned traffic controllers' inabilities to adapt has led to the allowance of "breaking the rules" in certain cases such as right on red and yield turning left on a greet dot. However, such cases rely on commuters making their own decisions about the safety of proceeding through an intersection, thus decreasing the intersection's safety. Instead, TrafficCOP's goal is to explicitly communicate to commuters when it is safe for them to proceed through the intersection with their given movement. In order to keep the greatest efficiency possible as well, TrafficCOP must maintain the highest possible number of simultaneously safe movements at a time.

In order for a movement to be safe, it must be the only one using its exit and it must not intersect any other currently active movements. All four right-turn movements do not intersect any other movement, and thus are safe as long as no currently active movement is using its exit. This means that if an exit is not being used by a currently active movement, the right-turn movement that uses that exit can safely use it. From this, it can be deduced that an exit can always be used and since our given intersection has four exits, all four exits can be simultaneously used by four safe movements. Thus, there are always four simultaneously safe movements for a four-way intersection involving two-way roads. One set of four safe movements makes up a safe phase, and these combinations of movements are the only possible phases that TrafficCOP needs to consider when deciding on a best phase. All of the safe phases for a conventional four-way intersection are listed and labeled in Figure 2.

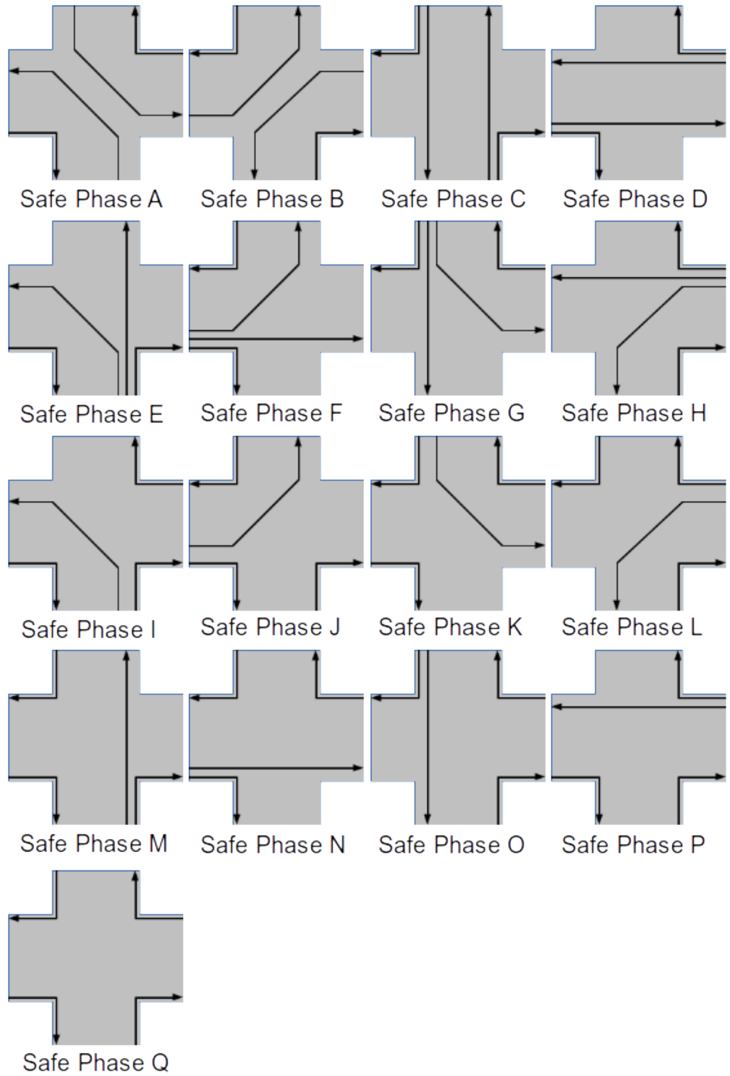

Figure 2: A list of all the possible safe phases for a four-way intersection.

TrafficCOP Design

Before the simulation starts, each movement determines all of the possible paths that could lead to the light that controls it and all of the possible safe movements that could exist with the original movement, including itself. Then the following input values are continuously calculated:

Given: commuters = the set of all commuters on a light's path distance = the distance from a commuter to the light			
Time-value	**Wait-value**	**Difference-value**	**Slowing-value**
Sum of (1 / distance) for every commuter	Sum of (wait-time / distance) for every commuter	Sum of ((max-speed – current-speed) / distance) for every commuter	Sum of (1 / distance) for every commuter that is slowing down due to its light being red

Figure 3: The four input values to TrafficCOP's neural network.

Referring back to *Eq. 2*, each movement calculates a W and X for each of its safe movements. Each X_i is equal to the wait-value of the corresponding safe movement. Each W_i is an integer between 0 and 1000 that is calculated in the following manner: 1) As long as the weight is greater than 0, 1 is subtracted from the weight. 2) If the color of the light of the corresponding safe movement is green, then an amount proportional to both of the movements' difference-values and slowing-values is added. The total output for a movement is then calculated using *Eq. 2*. After all of the movement's outputs at an intersection are calculated, the safe phase with the highest total output is chosen. Once an intersection choses its next safe phase, the lights change to match the safe phase (making sure to go through the proper sequence of yellow and red lights if changing from green to red). The new green lights are then locked for an interval based on the green lights' time-values. During this time, no new safe phases are determined. Once the green lights are unlocked, the process of choosing a safe phase repeats.

Simulation Setup

Traffic Controller	**Traffic Congestion**	**Road Configuration**	**Traffic Pattern**	**Turn Probability**
Stop Signs	5	1x1	Even	33%
Traffic Lights	25	2x2	Main Road	67%
Smart Traffic Lights	50	1x4	T Intersection	
TrafficCOP	75			

Table 2: All possibilities for each variable in the simulation

Each of the simulations was set up with 5 variables, with each the variables' possibilities shown in Table 2. The traffic controller variable would determine how the signals were controlled (by Stop Signs, Traffic Lights, Smart Traffic Lights, or TrafficCOP). Traffic congestion (with possible values of 5, 25, 50, and 75) signified, with some randomness, how many commuters would come from each direction in the span of 100 seconds. The road configuration represented how the grid of intersections was setup. For example, a 1x1 was just a single intersection, a 2x2 was four intersections in a block shape, and a 1x4 was one long road with four intersections on it. The traffic pattern was also one of three options; the first traffic pattern, 'Even', kept its variables as they were currently set. 'Main Road' set the north and south traffic congestions to be 1/5 their value so as to represent a main road running east/west with smaller side road(s) running north/south. 'T Intersection' only set the south congestion to be 1/5 its value. For all of these traffic patterns, a commuter that decided to turn would turn right 67% of the time (thus turning left the other 33%). The only exception to this was for 'T Intersection' which switched the left turn for cars from the west to be 67% in order to accurately simulate that the north, east, and west directions were the main roads. Finally, the turn probability of the commuters represented the probability that a commuter would turn somewhere in the model and it could either be 33% or 67%.

Each of the 288 combinations was simulated 10 times for the equivalent of 30 minutes. Intersections were placed the equivalent of a quarter-mile apart from one another and the max speed for commuters varied between approximately 30 and 45 mph.

RESULTS

The following tables show the results of averaging the efficiencies for the simulations mentioned above.

Traffic Congestion	Stop Signs	Traffic Lights	Smart Traffic Lights	TrafficCOP	**Totals**
5	80.90%	68.98%	83.79%	83.54%	**79.30%**
25	58.45%	61.42%	77.12%	79.41%	**69.10%**
50	54.51%	61.70%	73.39%	76.40%	**66.50%**
75	51.96%	60.59%	68.70%	73.67%	**63.73%**

Table 3: Traffic Controllers' Efficiencies Compared to Traffic Congestion

When looking at the effects of traffic congestion shown in Table 3, it makes sense that the larger amounts of traffic congestion would result in decreased efficiencies across the board. Stop signs are rather efficient for minimal amounts of traffic although this greatly declines with any more congestion. Although smart traffic lights are essentially even with TrafficCOP for the smallest amount of congestion, TrafficCOP's lead in efficiency grows substantially as the congestion increases, seeming to indicate that TrafficCOP, while fairly exceptional at handling small amounts of traffic, is very strong at handling large amounts of traffic.

Road Configuration	Stop Signs	Traffic Lights	Smart Traffic Lights	TrafficCOP	**Totals**
1x1	59.15%	68.36%	73.11%	78.17%	**69.70%**
2x2	60.44%	58.78%	76.22%	77.32%	**68.19%**
1x4	64.78%	62.37%	77.91%	79.27%	**71.08%**

Table 4: Traffic Controllers' Efficiencies Compared to Road Configuration

When looking at the effects of the different road configurations displayed in Table 4, although TrafficCOP performs highest on every configuration, its margin is rather slim for the road configurations that are made up of more than one intersection. There are two reasons that this might be the case: 1) TrafficCOP is currently designed to look at all of the commuters that are in the model, which may result in slightly skewed information for larger models. 2) Smart traffic lights often cause grouping of commuters as a red light stops many commuters and then the green light releases the group of them. This behavior may lead to the improved success of the smart traffic lights in situations where there are multiple intersections.

Traffic Pattern	Stop Signs	Traffic Lights	Smart Traffic Lights	TrafficCOP	**Totals**
Even	57.75%	61.87%	73.82%	75.91%	**67.34%**
Main Road	65.77%	66.04%	79.85%	81.46%	**73.27%**
T Intersection	60.86%	61.61%	73.58%	77.38%	**68.36%**

Table 5: Traffic Controllers' Efficiencies Compared to Traffic Pattern

Looking at the effects of traffic patterns shown in Table 5, there is a substantial increase in efficiency for all traffic controllers when the pattern follows a main road. This seems to make sense as it limits the amount of traffic that will cross other paths and might also heighten the effect of the grouping behavior that was discussed previously. The only traffic pattern that showed more than 2.5% of difference between TrafficCOP and smart traffic lights was T Intersections, which may be due to how the smart traffic light controller was set up since both north and south directions were treated equally although they did not have the same amount of volume.

Turn Probability	Stop Signs	Traffic Lights	Smart Traffic Lights	TrafficCOP	**Totals**
33%	61.44%	63.01%	74.05%	78.72%	**69.30%**
67%	61.48%	63.34%	77.44%	77.79%	**70.01%**
Totals	**61.46%**	**63.17%**	**75.75%**	**78.25%**	**69.66%**

Table 6: Traffic Controllers' Efficiencies Compared to Turn Probability

As can be seen in Table 6, there really was not a lot of of difference between results for the different turn probabilities. The increase seen for smart traffic lights may be due to the design of this traffic controller since if there are no commuters turning left, the opposite direction's through movement gets a larger green interval time. Especially for the lighter congestion simulations, not having many turns may have played a negative role by causing larger than necessary intervals for through movements. In addition, Table 6 shows the total efficiencies that each traffic controller system showed over all of the simulations.

CONCLUSION

Overall, TrafficCOP showed a increase in efficiency of 2.5% over the next highest traffic controller. This efficiency gain was especially prevalent in high congestion situations, making this design one that should be considered in the future for improving traffic delays in high congestion areas. Further research needs to be done into improving how this system performs for multiple intersections, but these initial simulations show that the idea of using neural networks to control traffic signals could be useful for traffic engineers.

FURTHER WORK

Original design specifications for this project including modeling a section of the Lloyd Expressway in Evansville, IN and comparing the measured efficiencies between the current traffic controller system and TrafficCOP. The complexity of this part of the project proved to be too large for the given time constraint although the programmer plans to continue working on it to its completion.

REFERENCES

1. *How scats works*. (n.d.). Retrieved from http://www.scats.com.au/how-scats-works.html
2. Kriesel, D. 2007, *A Brief Introduction to Neural Networks*, available at http://www.dkriesel.com
3. Lomax, T., Schrank, D., & Eisele, B. (2013). Inconsistent traffic conditions forcing texas commuters to allow even more extra time. Retrieved from http://d2dtl5nnlpfr0r.cloudfront.net/tti.tamu.edu/documents/tti-umr.pdf
4. Miller, J., & Page, S. (2007). *Complex adaptive systems : An introduction to computational models of social life*. Princeton, New Jersey: Princeton University Press.
5. *Signal operations*. (n.d.). Retrieved from http://www.webpages.uidaho.edu/niattproject/backgroundbasic1.htm
6. Stergiou, C., & Siganos, D. (n.d.). *Neural networks*. Retrieved from http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html
7. Tennessee Department of Safety and Homeland Security, Driver Services Division. (n.d.). *Tennessee comprehensive driver license manual*. Retrieved from website: http://www.tn.gov/safety/dlhandbook/DL_Manual2011.pdf
8. Walters, K. (2012, October 16). *Smart signals: Pilot study on traffic lights reduces pollution, traffic clogs*. Retrieved from http://www.cmu.edu/piper/stories/2012/october/smart-signals.html
9. Ward, L. "How Much Time People Spend Doing Stuff In Their Lifetime." . N.p., 29 Mar 2010. Web. 27 Sep 2013. http://www.thefactsite.com/2010/03/how-much-time-people-spend-doing-stuff.html.
10. Wilensky, U. (2013). *Netlogo*. Retrieved from http://ccl.northwestern.edu/netlogo/

BIOGRAPHY

Justin Simerly is a senior Computer Science / Cognitive Science double major attending the University of Evansville. His hometown is Jasper, Tennessee, and he is a member of Sigma Alpha Epsilon. After graduation, he plans to work at Ciholas Inc.