A Quasi-Sequential Cellular Automaton Approach to Traffic Modeling

Team # 799

February 7, 2005

Introduction

Entering and exiting a toll plaza can be hazardous experience for a motorist. The stress of merging from a small number of lanes to a large number of lanes and then back again, all the while looking for the shortest lane that will minimize total time in and out of the plaza, can be traumatic. Is there an optimum number of tollbooths that minimizes the stress of a motorist?

Figure 1: Photograph of the New Jersey Turnpike (I-95) At Night

Figure 1: Photograph of the New Jersey Turnpike (I-95) At Night

Toll roads have existed since ancient times. Charon, the mythical ferryman, charged the dead to cross the river Acheron. In Arabia, tolls were exacted from caravans traveling through certain parts of the country. Ancient Germanic tribes charged travelers for use of their mountain passes. In the 14th century, boats in the Netherlands were charged for crossing rivers. The first turnpike, a road where travelers pay for upkeep, was built in Hertfordshire, England in 1663. The first U.S. turnpike was built to connect Lancaster to Philadelphia in 1790. The development of more modern turnpikes did not begin until the 1940's. Since then the total number of vehicles on the road has steadily increased due to the decline in price of the automobile. The increase of 'private' car travel has lead to traffic and congestion, especially at toll plazas. The question of organizing traffic in a way that increases mobility for all has frustrated mathematicians and engineers for years [14].

One approach at organizing the traffic at toll plazas is to find an optimum number of tollbooths; that is, a number that minimizes the average total time spent by a motorist waiting upon entering and exiting the plaza and minimizing the congestion due to the merging. A key factor in finding this optimum number is modeling traffic at the plazas in an accurate fashion.

A myriad of models have been created to describe traffic flow. These models can be broken down into two basic types. The first type treats space and time as a continuum. In the case of modeling traffic flow, both cars and time would be continuous in nature. One of the first continuous studies of traffic flow began in 1935 by Greenshield; it "measured actual flows and velocities for one lane of traffic, and fitted a straight lie to a plot of velocity against concentration." His continuing efforts led to a linear model between traffic density and average speed. Other models also exist, incorporating higher order terms and ideas from fluid dynamics. The second type, involving discrete models, is a more modern approach to the classic flow problems. Instead of using continuous equations like partial differential equations, discrete models treat space as a lattice and time discretely. One of the most common discrete models is **cellular automata**, where space is modeled by a lattice and each lattice site represents a state of the system. Eventually, all of the lattice sites are updated and there states change. In the case of traffic flow, the states of the lattice sites represent whether a car is present at that spatial location or not.

There are many advantages to continuous and discrete models, but in the case of modeling congested traffic at tollbooths, discrete models work far better. In general traffic modeling, one is usually not concerned with stop and go action. However, near a tollbooth, cars must stop to pay cash before moving on. As each car affects the other cars in its direct neighborhood, it is not reasonable to model cars as a continuum. Discrete time also allows us to control the movement of the cars at each individual time step. As these cars are reacting to different stimuli, one may not

know a car's path a priori. Thus, discrete time allows us to change a car's path on the fly. Finally, discrete models in general are much easier to understand and to implement on modern computing resources. While these models may seem simple at first, one recalls that a system of several agents obeying simple rules can exhibit very complex behavior.

Assumptions

In order to properly model traffic at a toll plaza, one must make some fundamental assumptions about the nature of such traffic.

To begin, consider the behavior of a driver as he or she nears a toll plaza. While it may have been convenient for the driver to remain in one lane while on the main stretch of highway, the driver will most likely try to maneuver him or herself such that he or she can get through the toll plaza as fast as possible. However, the driver usually makes his or her decision on where to maneuver based on how congested the traffic is immediately in front of them. Once the driver is within a certain distance of the tollbooths, say one hundred feet, he or she will remain in that lane until they have paid the toll. Once the driver has paid the toll, the driver will then accelerate out of the toll slowly back to highway speeds. If the driver is in a tollbooth lane that is not a highway lane, they will do their best in order to get onto a highway lane.

In our model, we assume there is no cooperation between drivers. Instead, it is “every man for himself.” In fact, in this manner, traffic near a toll plaza is like a collection of games played by certain agents, the drivers. The drivers, based on their immediate environment, make certain decisions to win the game: making it through the tollbooths as quickly as possible. While the drivers are not directly competing against each other, they are affecting each other and are hence fierce indirect obstacles/opponents.

Summarizing our assumptions about the nature of the drivers:

- Upon nearing a toll plaza, a driver maneuvers based on local congestion in order to minimize travel time.
- Within a hundred feet of the toll plaza, a driver slows down (if not already moving at a low speed) and remains in their lane.
- Once a driver pays the toll, they maneuver to a highway lane and accelerate to highway speeds.
- Drivers do not cooperate. (“Every man for himself.”)

We also make the following two assumptions about the cars and toll plaza:

- Vehicles are assumed to be of constant length 17.5 feet.
- It takes about four seconds for a tollbooth employee to process a motorist [4].

One question remains: what speed does a motorist move at while approaching a tollbooth? Within one hundred feet of the tollbooth, we assume a car slows down to an average speed of about 5-10 MPH. However, at farther points from the tollbooth, traffic moves along in a much different manner. There have been many macroscopic models derived comparing average speed to density. These models include a linear model developed by Greenshield, a fluid-flow model developed by Greenberg, and a higher-order model developed by Jayakrishnan. While these models are very useful for highway modeling with large amounts of cars, we are dealing with a finite number of discrete cars in a small area. For this reason, we base the speed of the cars in our model on what is suggested in most driver’s manuals: car separation should be one car length for every ten miles per hour of driving speed.

With the above considerations, we are now able to dive into our model.

A Quasi-SCA Model of Toll Plaza Dynamics

Case Study One: Equal Number of Incoming Lanes and Booths

In this section, we first consider the case when the number of toll booths is equal to the number of incoming lanes. In the next section, we shall discuss how this model can be improved to accomodate for more tollbooths.

Preliminaries

Traffic has been modeled by a variety of discrete methods. Perhaps the most popular of these methods is **cellular automata**. **Cellular automata** are discrete dynamical systems whose behavior is completely specified in terms of its local region. They can be thought of as stylized universes. Space is represented as a uniform grid, with each cell containing some pieces of data. Time advances in discrete steps and the laws of the universe are expressed in some look up table relating each cell with a neighborhood of nearby cells so the system can compute its new state. In such a universe, the system's laws are local and uniform.

The basic one-dimensional **cellular automata** model for highway traffic flow is the CA rule 184 , as classified by Wolfram [11], [17], [19]. Cellular Automaton 184 is a discrete time process with state space ? ? {0, 1} ^{Z} and the following evolution rule: if ? ? {0, 1} ^{Z} is the state of at time n then the state ? at time n + 1 is defined by

? := 1, if ?(x) = ?(x + 1) = 1
1, if ?(x) = 1 - ?(x + 1) = 0 ? x ? Z (
0, otherwise

where ?(x) denotes the value of ? : Z ? {0, 1} at the coordinate x. In this model, the cars are essentially marching to the right in a rather uniform manner. Furthermore, all the nodes execute their moves in parallel. Many models of traffic flow are based on this basic one-dimensional cellular automata model. Many incorporate probabilistic deviations to the movement of the cars, such as the FI model developed by Fukui and Ishibashi.

Toll plaza dynamics, while similar to traffic dynamics, are quite a bit different. First of all, toll plazas can not be approximated as covering an infinite domain. Second of all, drivers must make decisions based on who moves in front of them. In this sense, we use ideas from Sequential Cellular Automata (SCA) [15] instead of the classical schemes. Third of all, cells are updated in a slightly different manner than classical **Cellular Automata**. In order to properly model car movement, "cars" are moved through cells one at a time. In this manner, we see that our model is more like a board game of sorts. For these reasons, we dub our model a "Quasi-SCA Model of Toll Plaza Dynamics."

To begin, we divide a multilane highway into equally partitioned lanes. Each cell is approximately 25 feet long and contains information on whether it contains a car and if it does, certain information

25 feet long and contains information on whether it contains a car and if it does, certain information about the car. Furthermore, there are specialized cell characteristics for different regimes, as shown in Fig. 2. In our model, we also move forward in discrete time steps. For convenience, this time step is set to be two seconds in length.

Figure 2: Possible Regimes

To implement our model, we exploited the object-oriented features of C++. A car class was created, with certain variables associated with it. The first of these is called Occupied. This is

a boolean characterizing whether or not the car is real or null. The second is called Congestion. It is a measure of how congested traffic is locally around a car. The third is Speed, containing information on how fast the car is moving. The fourth and fifth are called TotalTimeOnGrid and TotalTimeInToll respectively. These will be discussed at a latter point. There are also member functions which allow one to change these private variables.

Figure 3: Car Class Variables in C++

In our code, the highway is represented as a large 50 x n array of these car variables, where n represents the number of max lanes. When initialized, this array contains empty grid spaces. As cars enter in from the left, grid spaces are activated and infused with information about the cars. Then, with this information, the state of the system at the next time step can be determined.

Figure 4: Grid Definition

To formalize, we let ?(i, j, t) denote the state of the car located in the (i, j) position at time t. To avoid ambiguity, we present in Fig. 4 possible (i,j) positions on the grid. We see that i represents the number of spaces from the left of the grid and j represents the number of spaces down. As we are dealing with discrete time, we note that t ? N (we also note that t refers to the time in seconds multiplied by two, as discussed above). The state of the car can be thought of as a five-dimensional vector, as follows:

$$?(i, j, t) := \begin{matrix} o(i, j, t) \\ c(i, j, t) \\ s(i, j, t) \\ t_1(i, j, t) \\ t_2(i, j, t) \end{matrix} \quad ($$

where o(i, j, t) ? {0, 1} denotes whether the car is real or null, c(i, j, t) ? {0, $\frac{1}{5}$, $\frac{2}{5}$, $\frac{3}{5}$, $\frac{4}{5}$, 1} denotes the congestion level of the car, s(i, j, t) ? {0, 1, 2, 3} denotes the speed of the vehicle, $t_1(i, j, t)$? N denotes the amount of time the car has existed in the array, and $t_2(i, j, t)$? N denotes the amount of time the car has spent in the toll booth.

Vehicular Speed

As mentioned before, the speeds of the cars (not in the tollbooth regime) are dictated by the following rule:

• Car separation should be one car length for every ten miles per hour of driving speed.

As our model is discrete both in space and time, this result must be generalized. We find that moving one grid space ahead in one temporal step corresponds to a speed of about 8.5 MPH, moving two grid spaces represents 17 MPH, and moving three grid spaces corresponds to a speed of 25.6 MPH. If we approximate one grid space as one car length and 8.5 MPH ~ 10 MPH, we can generalize the speeds of the cars in the following manner:

$$0 \text{ if } \min_{x>i} (x|o(x, j, t) = 1) = i + 1$$

$$s(i, j, t) := \begin{cases} 1 \text{ if } \min_{x>i} (x|o(x, j, t) = 1) = i + 2 \\ 2 \text{ if } \min_{x>i} (x|o(x, j, t) = 1) = i + 3 \\ 3 \text{ otherwise} \end{cases} \qquad ($$

These results are presented in tabular form in Fig. 5. We enforce 25.6 MPH as an upper limit to our speed as the vehicles must slow down in an appropriate manner as they begin to near the toll. At each time step, the speed for a car is updated just before it initiates movement. This is shown in the Movement Function for the Transition Regions flow chart, Fig. 7.

Figure 5: Vehicular Speed

Congestion

An important factor a driver takes into consideration when maneuvering his or her car in a toll plaza is his local congestion. As a driver is far more forward focused than rearward focused, we consider the congestion to be determined only on the cars immediately in front of him or her. In particular, we estimate that the congestion is a factor of the nearest five cars. We then write congestion for the car located in grid cell ?(i, j, t) as:

$$c(i, j, t) := \frac{1}{5} \sum_{k=1}^{5} o(i + k, j) \qquad ($$

where

$$o(i, j, t) := \begin{cases} 1 \text{ if grid cell } (i, j) \text{ contains a car} \\ 0 \text{ otherwise} \end{cases} \qquad ($$

Sequencing

As mentioned before, cells are updated in a sequential manner as opposed to a uniform manner. This is because cars have to make decisions based on the cars in front of them. Hence, during a temporal step of simulation, each car is moved in a sequential manner. The cars farthest to the right are moved first as the cars to the left, in reality, would respond to their actions. Furthermore, in a given column of our array, that is, one spatial location across four lanes, the car with the largest velocity would have the first initiative. The car with the second largest velocity would move second, et cetera. In the case of a tie, the car closer to the top of the grid would move first. This is due to psychological characteristics. The car closer to the top would be in "the fast lane." On the road, drivers would often move in response to his actions.

Figure 6: The Sequencing Algorithm for a Column of Cars

Movement

What follows is a discussion of how a car moves in accordance to the regime it is in.

Transition Regimes

The transition regimes are the regions where traffic comes in from the highway or leaves to the highway. In these regimes, drivers maneuver in a manner such that they can optimize their travel time. However, drivers would like to move in a manner such that it requires minimum effort. Thus, movement possibilities in the transition regimes can be described by Fig. 5.

Figure 7: Movement in the Transition Regimes: Center Lane, Far Left Lane, Far Right Lane

The optimal maneuver for a driver to do is to move forward. However, he or she will drive to a lane to his right or left provided the move will minimize his congestion. This move, in the mind in the driver, will minimize the time he or she will spend in line waiting to pay the toll.

In two locations of the transition regimes, special consideration must be made. The first is located in the transition from highway regime. There must be some way to accurately depict the arrival of traffic from the highway. We will discuss how to do this in a couple of sections. The second location where consideration must be made is in at the tail end of the transition to highway regime. Here, provided a car has sufficient speed, we eliminate the car from the grid. We also keep a record of its TotalTimeOnGrid variable, as this provides us with vital information about the results from our simulations.

A general flow chart for movement in the transition regions is presented in a couple of subsections.

Tollbooth Regime

In the tollbooth regime, drivers no longer veer to the right or left. Instead, they move forward in line until they reach the tollbooth. In this region, spanning the 100 feet in front of the tollbooth, cars move at a maximum rate of one grid space per temporal element. Once in the tollbooth, they wait two entire temporal elements solely in the booth, or about four seconds, until they move on to the transition regime. This is implemented by incrementing a car's TotalTimeInToll variable (which is initialized as equal to zero when a vehicle enters the map) every temporal step a car is in the booth (for the entire step) and checking if it is greater than two. Often in this region, lines will form. As soon as a car emerges from the tollbooth, all of the cars behind will move immediately

into the spot in front of them. The dynamics of this regime are quite a bit different and simpler than the dynamics of the transition regime.

In Fig. 8, this situation is illustrated. The red cars in lanes one and four are stopped, waiting behind cars located in the booth. The green cars ahead of the toll are transitioning to the highway regime. The yellow car is moving into the tollbooth, and the blue car is moving further inside the region. The green car before the toll is just now moving into the tollbooth region. While its current speed is 25.6 MPH, once inside the region, it decelerates to 8.5 MPH.

Figure 8: Movement in the Tollbooth Regime

A general flow chart for movement in the tollbooth regime region is presented in a couple of subsections.

Modeling the Incoming Traffic Flow

In order to make our model more accurate, it was determined that a statistical distribution should be utilized in order to predict incoming flow. There are many statistical distributions that model random traffic flow. There are two distributions that are particularly common in traffic flow theory: the Poisson distribution and the negative exponential distribution. The Poisson distribution apthe Poisson distribution and the negative exponential distribution. The Poisson distribution approximates a binomial distribution by taking its limit. This limit is useful as there are often a very large number of chances for something to happen, but the probability of any one of those chances being successful is very little. However, the Poisson distribution only fits well with observations for light traffic flow [1]. The negative exponential distribution is a good fit for heavy traffic flow. The negative exponential distribution is commonly used in traffic theory to model the variations of gap length in a traffic stream over distance and random arrivals. The probability density of the negative exponential distribution is given by:

$$f(t) = qe^{-qt}$$

where t is the time between arrivals and q is the rate of arrival with units of cars per second. Next we will use a cumulative distribution function $P(t < T)$, (T is a random time deviate) which “specifies obtaining a given value or less from the distribution.” By integrating

$$P(t < T) = \int_0^T f(t)dt$$

$$P(t < T) = \int_0^T qe^{-qt}\, dt = 1 - e^{-qT}$$

or

$$P(h < t) = \int_h^t qe^{-qt} = 1 - e^{-qt}$$

where h is the time between arrival of successive cars. The left side of the equations is equal to a uniformly distributed random fraction (R) between 0 and 1 ($0 < R = 1$), we can say that

$$R = 1 - e^{-qt} \tag{10}$$

which can be rewritten as

$$1 - R = e^{-qt}. \tag{11}$$

The next step is to solve for t,

$$t = \frac{-1}{q} \ln R. \tag{12}$$

In general, once one knows the time between arrivals they can shift said times accomodate for the fact that no two cars can arrive at the same time [15].

To implement this arrival time into our simulation, the first course of action, is to assign it to a site of entry (a space) into the grid. A random number generator creates the random fraction (R), which is then inserted into equation (12). To ensure equation (12) produced a value, we threw out the R = 0 case. The value solved for, t, is then assigned to a ‘spawn site’, a place where ’cars’ are created. A counter is utilized to keep track of the time between different spawnings of cars. If this counter is greater than R and the ’spawn site’ is empty (contains a null car), then a car is created at the spawning site. Otherwise, the counter is incremented until one of these two conditions are met. Cars ‘arrive’ in each lane of the simulation using this method. This process is outlined in Fig. 12. It is also noteworthy that for our model, we will be using a modified q such that is in units of car per two seconds per lane. Calculating the flow rate in total cars per second is a simple conversion.

Flow Charts

This first flow chart outlines the general flow of a simulation using the model presented in this paper. We note that for this flow diagram, the location of the tollbooth is at i = 30.

Figure 9: Flow Chart for Entire Simulation

Page 11

This flow chart outlines how a car will move if it is in either the Transition to Highway regime or the Transition from Highway regime. We note that if a car has sufficient speed to leave the grid, then it is removed and its TotalTimeOnGrid value is recorded.

Figure 10: Flow Chart for Movement Function in Transition Regimes for an Element ?(i, j, t)

This flow chart outlines how a car will move if it is in the Tollbooth regime.

Figure 11: Flow Chart for Movement Function in Tollbooth Regimes for an Element ?(i, j, t)

This flow chart oulines the incoming flow function. As the function is called at the end of the temporal step, it acts on Elements ?(0, j, t + 1) rather than ?(0, j, t).

Figure 12: Incoming Flow Function

Results

We ran a simulation of our model for varying values of q, the flow rate of cars per two seconds per lane, for a four lane highway with an equal number of toll booths. The range we ran simulations for was from q = 0.01 cars per second per lane, or, q = 0.02 cars per second overall, to q = 1 cars per scond per lane, or about 2 cars per second overall. Fig. 13 outlines a given time evolution for a small value of q:

Figure 13: Time Evolution of a Simulation, Four Temporal Steps

As mentioned before, we note that the time through which the cars move through the grid (or toll plaza) is an appropriate measure of congestion. Thus, we plotted, as shown in Fig. 14, the average time a car spent getting through the grid versus the flow rate. The average time was obtained from a simulation accounting for an hour of traffic. We also plotted the maximum amount of time anyone spent getting through the grid for each flow value.

From our results, we note that between values of q ranging from 0.01 to 0.37 cars per two seconds per lane (0.02 and 0.74 cars per second overall), drivers enjoy an average time through the grid below 50 seconds. We will consider this an optimal situation. However, at around a q value of 0.36 cars per two seconds per lane, there is what appears to be a boundary layer. Anytime $q > 0.37$, it could take drivers potentially an average of 2 minutes or more to get through the quarter-mile long grid. This corresponds to an average speed of less than 10 MPH. This is not desirable in the least bit. We will demonstrate later that by adding more tollbooths, one actually shifts the boundary layer and lowers the time cars move through for larger q. Thus, a good strategy in determining the number of toll booths to use is to determine what your anticipated max flow rate and choose an appropriate number of lanes such that you never have a value of q after the boundary layer.]

Congestion is at its worst during ’rush hour’, the time when a majority of people are travelling to or from work. Toll plazas during these times create a bottleneck phenomena, “a narrow or obstructed section, as of a highway...”. But what do these congestion levels mean in total time through the plaza, are the number of tollbooths optimal?

The Hiawassee M/L Toll Plaza utilizes a 4 tollbooth plaza. In October 2003, the Eastbound car count between 7-8 am at this toll plaza was 3403 cars distributed over 4 lanes with one booth per lane. From the data provided by the Orlando-Orange County Expressway Authority it is clear

Figure 14: Average Time Through Grid Versus Flow Rate

Figure 15: Maximum Time Through Grid Versus Flow Rate

that this is when rush hour traffic occurs. Rush hour traffic happens once a day on each side of the toll plaza(Eastbound and Westbound), we will only consider the Eastbound case. This means that cars arrive at a rate of .945 cars/second/lane [10]. If we use our assumption from earlier that a car is 17.5 feet long, clearly, four lanes of tollbooths are not enough to handle this heavy demand. However, this is tough to judge. In this day and age, EZPass and other such programs allow one to minimize the amount of time they spend in a tollbooth. As a matter of fact, a car using one of these devices could get through a tollbooth without even stopping. If even a small portion of the cars utilize the EZPass system, the value of q for which a boundary layer vastly grows. If we were to accurately determine an optimal value of tollbooths for a certain value of q for a highway using such a system, we would have to approach the problem in a slightly different fashion. In particular, we would have to vary the time drivers spend at the booth and designate certain lanes as having a quick pass system.

A Quasi-SCA Model of Toll Plaza Dynamics

Case Study Two: More Tollbooths Than Incoming Lanes

In this section, we explain how our previous model can be modified to account for a greater number of tollbooths than incoming lanes.

Preliminaries

The situation changes quite a bit once you start adding more tollbooths than incoming lanes. Drivers in the far left and right lanes start moving into the new tollbooth lanes in order to optimize their travel time, and once they leave the tollbooth, they stop at nothing to get back into their original lane or a new one. Hence, we introduce a new scheme for our regimes, as presented in Fig. 16.

Figure 16: Possible Regimes

Life in our three regimes from before, the Transition from Highway, the Transition to Highway, and Tollbooth regimes, work in the same way that we discussed in the last section. In this section of the paper, we briefly describe how the two new regimes work in our model.

Movement in the Expansion Regime

The expansion regime corresponds to the region where the incoming traffic lanes fan out to a greater number of tollbooth lanes. As in the transition regimes, drivers maneuver in a manner such that they can optimize their travel time, and drivers would also like to move in a manner such that it requires minimum effort. For the center lanes, movement is identical to the transition regimes. On the outer lanes, however, movement is slightly different. The movement possibilities are outlined in Fig. 17.

Figure 17: Movement in the Expansion Regime

If a driver is on the outside lane, his optimal maneuver is to move into one of the newly created tollbooth lanes. However, if the congestion is less in his lane, he will continue to drive forward. As before, in the mind of the driver, this will minimize the amount of time he or she will spend in line waiting to pay the toll. Another new addition is that the driver will not try to move in to one of the inner lanes. This is more for psychological reasons than practical reasons. According to the model, the driver assumes that the most outside lanes are the least dense (and fastest) as the lanes did not

 Page 17

exist on the highway. This thinking does contribute to jamming situations, as is demonstrated by our simulation results. Drivers on the newly created lanes are only allowed to move forward in our model. While a driver may move to an outside lane just to move back again, we consider the chance of this occurring as very slim.

We do not provide a flow chart for this regime as its logic is very similar to that of the transition regimes. The algorithm works in a very similar manner.

Movement in the Compression Regime

The compression regime corresponds to the region where a greater number of toll booth lanes collapse onto a smaller number of highway lanes. As our drivers are pretty much concerned only with themselves, they do not move in a fashion such that they do not box out the cars on the outermost lanes. We thus have the movement possibilities presented in Fig. 18.

Figure 18: Movement in the Compression Regime

If a driver is not in a tollbooth lane that is not a highway lane, they follow the same rules presented in Fig. 9. If a driver is in one of these lanes, however, they try as hard as possible to move back onto the highway lanes. If this is not possible, they keep driving forward and trying again until they are forced to stop at the end of the tollbooth lane. This phenomenon provides for some hectic situations.

The flow chart in Fig. 19 demonstrates how the movement function works for a driver in one of the outer lanes. In particular, we assume the outer lane is on the top of the grid, so $j = 1$ for some Elements ?(i, j, t). Special considerations were made in order to ensure a vehicle would not drive off the road if forced to stay on the tollbooth lane by other traffic.

Results

We ran our simulation for our second model for varying values of q, the flow rate of cars per two seconds per lane, for two cases: 4 highway lanes with 5 tollbooths and 4 highway lanes with 6 tollbooths. The range we ran simulations for was the same as our first model, from $q = 0.01$ cars per second per lane, or, $q = 0.02$ cars per second overall, to $q = 1$ cars per scond per lane, or about 2 cars per second overall. Fig.'s 20 and 21 show the results for these two cases. For these cases, the expansion and compression regimes were taken to be 125 feet in length.

As we can see by the figures, the boundary layer we saw from before is moved to the right as the number of toll lanes increase. Furthermore, we see that the value for q on the right side of the boundary layer decreases with increasing toll lanes as well. Thus, as suggested, one should choose a sufficient number of lanes that correlates to this behavior. If the maximum flow rate one expects is a certain value, one can run a simulation for a certain number of tollbooths and choose the least number of tollbooths such that the maximum flow is to the left of the boundary layer. Then, there should not be any problems with congestion.

An interesting point to add, however, is that with an increased number of lanes comes an increased maximum individual travel time. At times, people will become stuck in the toll lanes, and as such, have to wait for an opportune moment to move over. In our model, this was reflected by the fact that while the four tollbooth case resulted in a maximum travel time of about 3.4 minutes as demonstrated on the plot in Fig. 16, the 5 and 6 lane case would sometimes have a maximum individual travel

Figure 19: Flow Chart for Movement Function in Compression Regime

Figure 20: Average Time Through Grid Versus Flow Rate - 5 Lanes

Figure 21: Average Time Through Grid Versus Flow Rate - 6 Lanes

time upwards of 4 minutes. However, as these are the maximum travel time through the plaza one traveled over the hour, it is a rarity.

Model Improvements and Discussion

There are several factors we did not take into account while making our model due to time and knowledge constraints. For example, drivers do not always move in a predictable manner. A probabilistic model taking into account the unpredictabile nature of humans could further improve our model. Our model also did not take into account the possibility of accidents. For example, when there are more tollbooths than highway lanes, people could get frustrated in the outside tollbooth lanes and accidentally veer to the right when a car is actually there. An accident model would surely improve our model. While we did take into account the random nature of incoming traffic flow, one could develop an even better model to approximate the flow rate. Lastly, our model could include a probablistic model for the time a car waits at a tollbooth. Whether a driver pays exact change or with a 20 dollar bill definitely influences the amount of time it takes an employee to process the toll.

As there was not much readily available information on tollbooths not containing quickpass systems, it was hard to verify our model. However, if such data were made available, our model could be verified and or improved to accomodate such information. If given data on how long it often takes a driver to get through a toll plaza area which does not contain a quickpass system, we could easily check our model with the number of incoming lanes and tollbooths. As far as other information, it would be convenient to know the following: more sources on average time it takes to go through a toll, how people pay tolls, human behavior near a tollbooth, and speeds near a tollbooth. With this added information, our model could be brought to a higher level. Furthermore, based on additional experimental data, our model could be modified to more appropriately model different tollbooth situations, such as incorporating the quickpass system.

Conclusion

Entering and exiting a toll plaza can be hazardous and annoying experience for a motorist. However, adding additional tollbooths cost labor, money, and time. Thus, it is the goal of those who design tollbooth systems to try to optimize the number of tollbooths such that the motorists remain content and the number of tollbooths are minimized.

In this paper, we developed a quasi-SCA model for toll plaza dynamics. In our model, we treated time and space in a discrete manner in order to properly capture the motivation and actions of drivers on the road. We also considered a random negative exponential distribution for the incoming flow rate of cars. To implement this model, we used object-oriented C++ code to model the cars as they moved through the toll plaza area. By keeping track of the amount of time each driver spent moving through the area, we were able to compute the average time it took for different traffic flow rates. Through the results of simulation, we saw that at a certain flow rate, unique to the number of tollbooths in use, there would be a boundary layer in which the travel time would increase substantially from one flow rate to the next. Thus, an optimal solution to the tollbooth problem would be to choose the minimum number of tollbooths such that the expected rate of incoming flow would correspond to a point before the boundary layer. This way, “everyone wins.”

Works Cited

Works Cited

1. Aston, Winifred. The Theory of Traffic Flow. New York: John Wiley & Sons Inc., 1966.

2. Campbell, Paul. “Special Issue: The 1999 Mathematical Contest in Modeling.” The UMAP Journal. Vol. 21: COMAP Inc., 1999.

3. Campbell, Paul. “Special Issue: The 2000 Mathematical Contest in Modeling.” The UMAP Journal. Vol. 21: COMAP Inc., 2001.

4. Design and Evaluation of Toll Plaza Systems. Chao, Xiuli. < http : //www.transportation.njit.edu/nctip/finalreport/TollPlazaDesign.htm > .

5. Designing Traffic Signals to Accommodate Pedestrian Travel. 1997. TranSafety Inc. Site. < http : //www.usroads.com/journals/p/rej/9710/re971002.htm > .

6. Drew, Donald. Traffic Flow Theory and Control. New York: McGraw-Hill Inc., 1968.

7. Edie, A. C. (1954). Traffic delays at toll booths, Journal of Operations Research Society of America, 2, 107-138.

8. Hristova, Hristina. ”Lecture Notes.” MIT OpenCorseWare. < http : //ocw.mit.edu/OcwWeb/Mathematics/18 - 306Spring2004/LectureNotes/ > .

9. “Locality”. Rauch, Erik. June 26, 1996. < http : //www.swiss.ai.mit.edu/ rauch/dapm/node1.html > .

10. “MAINLINE PLAZA CHARACTERISTICS.” Orlando-Orange County Expressway Authority Traffic and Revenue Statistics. 2003. < http : //www.expresswayauthority.com/assets/STDStats20Manual/Mainline20Toll20Plazas.pdf > .

11. Nagel, Kai, Dietrich Wolf, Peter Wagner, and Patrice Simon. ”Two-lane traffic rules for cellular automata: A systematic approach.” Physical Review E. Vol. 58-2. August 1998. < http : //www.sim.inf.ethz.ch/papers/nagel - etc - 2lane/nagel - etc - 2lane.pdf > .

12. Sveshnikov, AA. Problems in Probability Theory, Mathematical Statistics, and Theory of Random Functions. New York: Dover Publications, 1968.

13. Timed Cell-DEVS: modeling and simulation of cell spaces. **Wainer**, Gabriel and Norbert Giambiasi.
< http : //www.sce.carleton.ca/faculty/**wainer**/papers/timcd.PDF > .

14. “Toll Roads” Wikipedia. February 2, 2005.
< http : //en.wikipedia.org/wiki/Tollroad > .

15. Tosic, Predrag and Gul Agha. “Concurrency vs. Sequential Interleavings in 1-D Threshold **Cellular Automata**.” < http : //osl.cs.uiuc.edu/docs/ipdps04/ipdps.pdf > .

16. “Traffic Congestion.” Answers.com. Wikipedia.
< http : //www.answers.com/traffic + congestion&r = 67 > .

17. Traffic Flow with **Cellular Automata**. NKS SJSU. Jiang, Henry.
< http : //sjsu.rudyrucker.com/ han.jiang/paper/ > .

18. “Traffic Simulations with Cellular Automaton Model.” April 30, 2000.
< http : //www.newmedialab.cuny.edu/traffic/gridCA1/sld001.htm > .

19. Wolfram, Steven. A New Kind of Science. Canada: Wolfram Media Inc., 2002.

20. “1997 Work Program.” DelDot. 1997.
< http : //www.deldot.net/static/work/objectives/objectives.html > .