Team Control Number

858

For office use only		For office use only
T1 ______		F1 ______
T2 ______		F2 ______
T3 ______		F3 ______
T4 ______		F4 ______

Problem Chosen

B

As a team of analysts in consultation with a generic toll booth, we are charged with the task of determining how to minimize traffic congestion within our toll plaza by deploying an optimal number of toll booths.

We first consider our options short of major renovations, concluding that the only variable we can adjust is the number of toll booths open. We aim to minimize congestion by exhausting the physical constraints.

In solving this problem, we begin with the Primitive Model that serves as both a reality check and a landmark by which to measure our more developed models. A special case of the Primitive Model deals with the scenario of one toll booth per lane.

The Primitive Model is developed into a more accurate depiction of the problem: our 2-Queue Model. Motorists must first wait in a queue to pay the toll, and then in another queue to leave the plaza. We found through a simulation generated in Excel that the rate at which cars may leave the second queue and the maximum size of the second queue are binding constraints when minimizing congestion. Observing that toll booths must halt when the second queue is filled, we optimize the secondary objective of scheduling open toll booths by keeping the second queue as "close" to capacity as possible.

The Toll Booth Utilization Model deals with the issue of deploying different quantities of toll booths. Using a Java simulation, we were able to generate a random arrival of cars per minute. Then we used this distribution of cars on a different number of toll booths, allowing us to analyze the average and maximum wait times per queue – an issue not focused upon in the other models – with a granulation in minutes.

Though the previous two models were adequate under an assumption of no significant renovations, we relax that condition in order to formulate the Dual Flux Model. Motorists have the option of buying a One Time Fast Track pass in one lane before entering the first queue. Upon exiting the first queue, they are channeled down a side road that bypasses the second queue and then merged into the highway. When implemented in Excel, this creates two flows of traffic that relax the binding constraints and cut down on congestion markedly. The required investment is an additional toll booth, one Automated Vehicle Identification scanner, and one kilometer of extra road.

Relieving Toll Booth Congestion

February 7, 2005

Contents

1 Introduction

When traveling on a toll road, a traveler is interrupted at certain intervals by toll plazas. In the case of a barrier-toll road, every vehicle must stop to pay the toll, and as a result traffic congestion occurs. Our conclusion in regards to this is that congestion is unavoidable. Since congestion is inevitable, we want to confine or limit the congestion in some way. There are three possible scenarios to this point of view:

- allow congestion upon entry to the toll plaza, and ensure that little congestion occurs when exiting the toll plaza,
- attempt to ensure that there is very little or no congestion upon entry to the toll plaza and allow congestion in the exit, or
- allow congestion at both the entry and exit of the toll plaza.

The primary goal is to minimize the annoyance of the motorists by minimizing traffic congestion. We must determine how many toll booths to deploy in the toll plaza to minimize motorist annoyance by limiting congestion and maximizing traffic flow. As a secondary objective, we deal with the issue of maximizing a schedule of opening toll booths.

Through our research, we found that a number of toll booths roughly three times the number of traffic lanes is fairly standard. We chose to impose 12 as the maximum for our generic toll booth area, assuming a four-lane road.

2 The Primitive Model

2.1 The Limiting Case

Here we examine the limiting case where there are precisely as many toll booths as traffic lanes.

In order to model this, we make the following assumptions:

- Number of cars that pass through the toll booth per day: 42,000
- Standard number of lanes of the highway: 4.
- Rate the toll booths can serve cars: 3 cars per minute per booth.
- Rate the plaza can empty itself out: 1750 cars per hour.
- Rate cars are entering the first area of congestion: variable

To elaborate on what these values are and our justification for using them: according to the New Jersey Turnpike Authority, one of the most densely traveled highways in the USA, the New Jersey Turnpike, receives over 45,000 cars over the course of a day. Since most highways do not have that much traffic, and we are considering a generic toll booth, we have reduced that number to 42,000 vehicles for our models, which still implies a very busy highway and thus makes it a sufficient model for almost any highway.

We estimated that the toll plaza extends some 500m beyond the booth area, subsequently generating a rough estimation of 1300 cars that can be contained in this part of the plaza at a given point in time.

Based on personal experience, we imposed the service rate at the booths.

To determine the rate at which the plaza can empty itself back onto the four lanes of the highway, we say it is just 42000/24, or 1750 cars per hour.

The rate at which vehicles enter the toll area is variable due to the rise and fall of traffic volume into peak and non-peak times. We split a 24 hour period up into 8 units and assign to it the average arrival rate for the time period. Consequently, we get a piece-wise uniform probability distribution over the course of the day. Since 42,000 cars pass through the toll in a day, by our informed assumption. It is reasonable to assume, then, that the departure rate from the toll area can be bounded from above by

$$\frac{42000}{24} = 1750 \quad \text{cars per hour} \tag{1}$$

Operating on the assumption that each toll booth can serve 3 cars per minute, or 180 per hour, and given our four lane highway has four toll booths, we find that the average number of vehicles that can pass through the toll area every hour is given by

$$N = 4(180) \tag{2}$$

where N is the number of cars passing through per hour. So every hour, an average of 720 cars can pass through the toll area. However, we have an average of 1750 vehicles per hour entering the toll area. So we have a far greater number of vehicles entering the toll area than can leave. This presents a problem, and clearly a great deal of congestion will occur as the vehicles arrive at the toll area.

The following table (Table 1) shows the build up of vehicles waiting to pass through the toll area over time:

Table 1: The effects of having one toll booth per lane.

t	$Q_1(t)$	$Q_{1in}(t)$	$T(t)$	N_1
t_0	0	1750	720	720
t_1	1020	1750	720	720
t_2	2050	1750	720	720
t_3	3080	1750	720	720
t_4	4110	1750	720	720
t_5	5040	1750	720	720
t_6	6070	1750	720	720
t_7	7100	1750	720	720

where

$$
\begin{aligned}
t &= \text{the time interval under consideration} \\
Q_1(t) &= \text{The number of vehicle waiting to pay the toll} \\
Q_{1in}(t) &= \text{The rate vehicles that arrive during the time interval} \\
T(t) &= \text{The rate vehicles are passed through the toll booths} \\
N_1(t) &= \text{The number of vehicle passed though the toll booths}
\end{aligned}
$$

Based on the data from Table 1, it is clear that one toll booth per lane of traffic is not adequate, even when all the toll booths are operating at capacity.

One thing we have chosen to ignore for this preliminary model is what happens on the other side of the plaza, after the toll is paid. We assume for the sake of simplicity that once the toll is paid, where they go is of little concern to us.

2.2 The Generalized Case

As seen from our last model, one toll booth per lane of traffic is not sufficient to deal with a large number of vehicles. It seems clear that we need more toll booths.

In order to establish a framework on which to build, we came up with an simplified model, and used this to analyze the different scenarios so as to determine the optimal number of toll booths. To begin, we assumed an average daily volume of 42,000 vehicles (the New Jersey Turnpike's Interchange 1 serves 45,000), and each of these cars must stop to pay the toll. We then segmented the day into 8 three-hour time blocks. We make the assumption that traffic volume follows a normal distribution, with peak volumes during the morning and afternoon rush hours. In addition, we assume a slightly elevated traffic volume between the hours of 12:00 pm and 3:00 pm, (the lunch rush), and between the hours of 12:00 am and 3:00 am, (when the bars close).

The rate at which cars enter the congestion area will be variable, for certainly peak and non-peak times will arise out of the scenario. We will, however, assume a rough distribution over the course of a 24 hour period. The number of cars that pass through our toll booth is projected as follows in Table 2.

Table 2: The volume distribution of vehicles entering the toll plaza.

t	$Q_{1in}(t)$
t_0	3000
t_1	7500
t_2	4500
t_3	6000
t_4	7500
t_5	3000
t_6	4500
t_7	6000

where

$$t = \text{the time interval under consideration}$$
$$Q_{1in}(t) = \text{The rate that vehicles arrive during the time interval}$$

Through our research, we found that a number of toll booths roughly three times the number of traffic lanes is fairly standard. We chose to impose 12 as the maximum for our generic toll booth area, assuming a four-lane road.

To get a sense of the dynamics of the problem, we began by formulating a quick model by which we can make decisions on how to alleviate congestion and get motorists through the toll area. For the sake of simplicity, at this stage we assume that the number of toll booths open remains constant throughout the 24 hour period. Additionally, we use Table 1 to dictate the entrance cars into the model by a uniform distribution. The time intervals of the day are split up into discrete intervals of three hours. The final assumption we make explicit here is that what happens after the toll is paid is ignored in this primitive model. With this model, we are concerned with exactly how many toll booths we need to minimize congestion in the queue building up in front of the booths.

As in the previous model, we assume that all toll booths are open and operating at capacity. We have imposed a maximum of 12 toll booths, and as indicated previously, we must have more than 4.

Table 2 above addresses the variable rates at which vehicles arrive in the toll area. The numbers are based on the trends typically observed, taking into account the elevated volumes during morning and afternoon rush hours, as well as lunchtime and when the bars close.

2.3 Analyzing the Model

We now analyze the effects of increasing the number of toll booths in the toll area. We have imposed a maximum of 12 toll booths, or 3 per lane, and we have already concluded that we need more than one per lane. To begin, we have broken up a 24-hour day into 8 three-hour intervals, and denote by t_0 the interval from 3:00AM to 6:00AM. We begin by assessing the effects of increasing the number of toll booths to our assumed maximum of 12 booths. (Table 3)

Table 3: The effects of having 12 toll booths.

t	$Q_1(t)$	$Q_{1in}(t)$	$T(t)$	N_1
t_0	0	3000	6480	3000
t_1	0	7500	6480	6480
t_2	1020	4500	6480	4520
t_3	0	6000	6480	6000
t_4	0	7500	6480	6480
t_5	1020	3000	6480	4020
t_6	0	4500	6480	4500
t_7	0	3000	6480	3000

where

$$\begin{aligned}
t &= \text{the time interval under consideration} \\
Q_1(t) &= \text{The number of vehicle waiting to pay the toll} \\
Q_{1in}(t) &= \text{The rate that vehicles arrive during the time interval} \\
T(t) &= \text{The rate vehicles are passed through the toll booths} \\
N_1(t) &= \text{The number of vehicle passed though the toll booths}
\end{aligned}$$

We immediately notice a vast improvement over the Limiting Case in the flow of traffic. There is some congestion when entering the toll plaza during the rush periods, but otherwise congestion is nonexistent. We are not necessarily looking for no congestion at all, because as previously stated, we have concluded that some congestion is unavoidable. Let's see what happens if we reduce the number of operating toll booths down to 11.

The following table (Table 4) displays the results of having only 11 operating toll booths with the same traffic volume distribution.

The results of having only 11 operating toll booths provides for a great deal more congestion which is longer lasting. Since the goal is to attempt to alleviate congestion upon entry to our toll area, having only 11 toll booths in operation is not a good solution.

Table 4: The effects of having 11 toll booths.

t	$Q_1(t)$	$Q_{1in}(t)$	$T(t)$	N_1
t_0	0	3000	5940	3000
t_1	0	7500	5940	5940
t_2	1560	4500	5940	5940
t_3	120	6000	5940	5940
t_4	180	7500	5940	5940
t_5	1740	3000	5940	4740
t_6	0	4500	5940	4500
t_7	0	6000	5940	5940
t_8	60	3000	5940	3060

As a point of interest, we now examine the case where we have 10 toll booths operating with the same vehicle volume distribution. The following table displays the results. (Table 5)

Table 5: The effects of having 10 toll booths.

t	$Q_1(t)$	$Q_{1in}(t)$	$T(t)$	N_1
t_0	0	3000	5400	3000
t_1	0	7500	5400	5400
t_2	2100	4500	5400	5400
t_3	1200	6000	5400	5400
t_4	1800	7500	5400	5400
t_5	3900	3000	5400	5400
t_6	1500	4500	5400	5400
t_7	600	6000	5400	5400
t_8	1200	3000	5400	5400

Having only 10 toll booths in operation results in a degenerate situation in which congestion is never alleviated. Instead of eventually clearing out, any vehicle that arrives in the toll area will have to wait. As time progresses, the congestion increases and so does the corresponding wait time.

It would seem then that 12 toll booths operating at their capacity of 3 vehicles per minute per booth is a good solution as far as this model goes. We would therefore conclude that 12 toll booths is an optimal number, which corresponds to 3 toll booths per lane of highway.

2.4 Comments on the Primitive Model

The major drawback to our primitive model the fact that it does not address certain issues, all of which should be of concern. Issues such as:

- What sort of congestion might occur on the other side of the toll area?
- Can 12 toll booths really operate at capacity constantly?
- Is there a need for all 12 toll booths to be operating all the time?

Another consideration we should pay attention to is the fact that we recommended 12 toll booths in operation strictly because we imposed a maximum of three times the number of lanes on the highway and we are considering a four lane road. Since congestion upon entry to the toll plaza was our only concern, we could conceivably remove that constraint and say that as far as toll booths go, the more the better!

The major disadvantage to this model is in the liberal assumptions we have made. The next model we present attempts to justify some of what was taken for granted in the Primitive Model.

3 The 2-Queue Model

3.1 The Queuing Theory Facet of the 2-Queue Model

As an improvement upon the Primitive Model, the 2-Queue Model interprets the problem as a sequence of two queues:

- First queue: cars waiting to pay their toll
- Second queue: cars in the plaza beyond the toll booths waiting to merge back onto the 4 lane highway

We call the region where the second queue lies the "after-plaza", simply to avoid confusion with the portion of the plaza where cars fan out to enter the toll booths.

A key observation to make with this model that drives our approach to a solution is that the first queue does not have a practical bound on its size while the second one does. For the first queue, cars can back up down the highway for what we'll assume is an infinite distance. In a practical sense the size of the first queue, denoted Q_1, is at least bounded by the number of cars that pass through the model.

The characteristics of the second queue are fundamentally different because there is a binding constraint on how large the queue, denoted Q_2, can get. We estimate that the length of the after-plaza is approximately 500 meters and from that estimation concluded that the after-plaza has an area of approximately 13000 square meters and therefore has a maximum capacity of 1300 cars, based on the assumption that each vehicle encompasses an average of 10 square meters. Therefore, one constraint for our model is given by

$$Q_2 \leq 1300 = Q_{2\text{cap}}. \tag{3}$$

The first queue of the model is essentially the same as the single queue used in the Primitive Model. We were able to alleviate congestion by adjusting the number of open toll booths. In the terminology of Queueing Theory, we are able to control the service rate of the first queue.

The second queue, because of our initial assumptions about what lies within our control, becomes more problematic. We've assumed for the 2-Queue

Model that we do not have the funding to undertake major construction on the physical properties of the toll area – in particular the size of the after-plaza and the size of the bottleneck created by its contraction into four lanes of highway. In other words, we cannot adjust the service rate of the second queue, but can only control the inflow rate of cars into the queue – which is precisely the same as the service rate for the first queue, T. Consequently, when T is large, Q_2 is likely to fill to capacity as motorists merge into four lanes more slowly than they are released collectively from the toll booths. When Q_2 has reached its upper bound, the toll booths grind to a halt – a motorist who has completed his transaction won't have room to move from the booth into the after-plaza, and so disables another motorist from driving up and making their transaction.

The Queue Theoretical aspects of our model can be pictorially represented as in Figure 1:

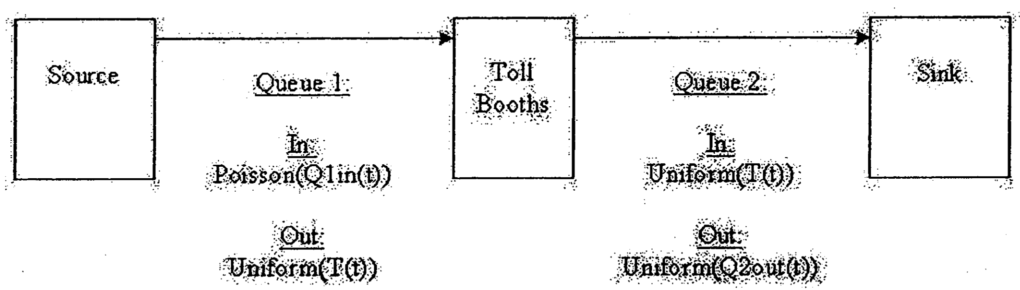

Figure 1 : The Queue Theoretical aspect of the 2-Queue Model.

3.2 The Operations Research Facet of the 2-Queue Model

With the established notion of what is occurring, our 2-Queue Model becomes a dual optimization model. The primary objective is to minimize congestion. Secondary to that is keeping unnecessary toll booths closed: when considering the situation where $T = i$ yields a constantly filled up second queue, opening the $(i+1)^{\text{th}}$ booth does not add anymore to the alleviation of congestion, for the inflow rate into the second queue would remain unchanged. Therefore, the service rate of the second queue and the bound on Q_2 become binding constraints on the primary objective of the model. The secondary objective of minimizing redundant toll booths then kicks in.

Also different from the Primitive Model is the probability distribution of the inflow of vehicles into the first queue. Instead of a uniform distribution on arrival times, we more appropriately model it by a Poisson process, with the mean being the same as was given in Table 1.

The way in which we optimize the primary objective is to ensure that the service rate of the second queue is binding whenever possible throughout the course of the day we are examining. As long as cars are leaving the plaza area at the maximum allowed rate, denoted $Q_{2\text{out\_cap}}$, we cannot do any better as far as alleviating congestion. In keeping $Q_{2\text{out}}$ as close to $Q_{2\text{out\_cap}}$ as possible with minimal risk, Q_2 should be as full as possible throughout the day.

As mentioned, it is the scenario where Q_2 is at capacity that we run into the issue of unnecessary open toll booths. We can now concern ourselves with the secondary objective of the model: if we can, by adjusting T, keep Q_2 as close to $Q_{2\text{cap}}$ as possible, without constant saturation of the after-plaza, then we can achieve both objectives optimally. More precisely, we wish to minimize

$$Q_{2\text{cap}} - Q_2 > 0. \tag{4}$$

The obvious course of action in minimizing the consolidated objective is to assign a penalty based on the size of the second queue. There should be a high penalty for $Q_2 = Q_{2\text{cap}}$ and a high penalty for $Q_2 = 0$. If a low Q_2 is unavoidable, our penalty may be high, but it would remain optimal with respect to alleviating congestion, i.e., the penalty would only increase if we increased congestion. Additionally, we want a low penalty centered around

$$Q_2 = Q_{2\text{cap}} - T_{\text{cap}}. \tag{5}$$

The risk function, denoted $Q_{2\text{penalty}}(Q_2)$, is modeled by a pairwise linear function as shown in Figure 2.

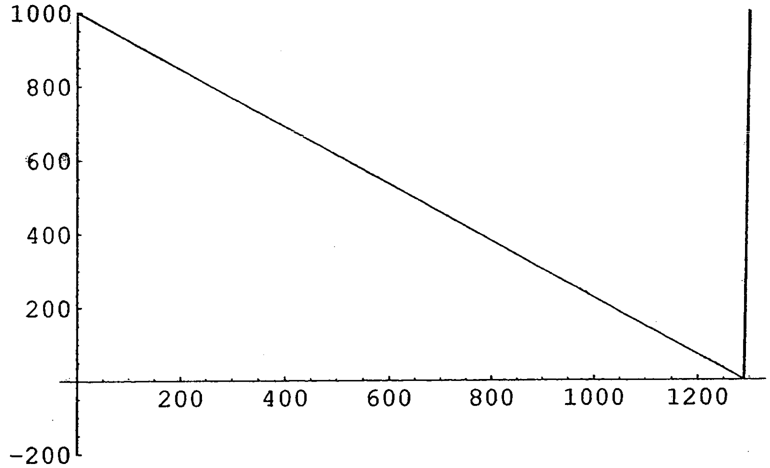

Figure 2 : Risk function $Q_{2\text{penalty}}(Q_2)$.

Since scaling is arbitrary, a maximum penalty of 1000 and a minimum of 0 was chosen. As shown in Figure 2, we have the linear function

$$Q_{2\text{penalty}}(Q_2) = -\frac{100}{129}Q_2 + 1000 \tag{6}$$

on the interval $[0, 1290]$ and

$$Q_{2\text{penalty}}(Q_2) = 100Q_2 - 12900 \tag{7}$$

on the interval $[1290, 1300]$.

In the event that the queues are hovering around empty, there remains the possibility that toll booths can be redundant without Q_2 being filled. For example, it would not be optimal to have all 12 booths open when fewer than 12 would still handle all of the cars in the first queue. To account for this, a penalty of 1 is accumulated for each toll booth open, a penalty that won't interfere with the more significant $Q_{2\text{penalty}}$ function, but still solve the problem of redundant toll booths in this specific instance.

3.3 Solving the 2-Queue Model in Excel

The dual optimization 2-Queue Model as applied to a 24-hour time period was solved using Excel, minimizing the penalty function

$$z = \sum_{t=1}^{24} Q_{2\text{penalty}}(Q_2(t)) + \sum_{t=1}^{24} T_{\text{open}}(t). \tag{8}$$

A short MatLab function was used to produce a stream of Poisson distributed numbers.

Because the algorithms implemented in Excel are not suitable for printing as an Appendix, the process by which the model was solved is reasoned from the articulation of the model from above and from the results given below in Table 6 – a basic reproduction of the solved spreadsheet.

Table 6 : Results of the 2-Queue Model's solution.

t	$Q_{1\text{in}}$	Q_1	T	Q_2	$Q_{2\text{out}}$	T_{open}	$Q_{2\text{penalty}}$	T_{penalty}
1	2488	0	0	2160	1750	**12**	1000	12
2	2510	328	2160	410	1750	**12**	681.677	12
3	2518	678	2160	820	1750	**12**	363.354	12
4	1493	1036	1800	1230	1750	**10**	45.031	10
5	1557	729	1620	1280	1750	**9**	6.211	9
6	1486	666	1800	1150	1750	**10**	107.143	10
7	1936	352	1800	1200	1750	**10**	38.323	10
8	2001	488	1620	1250	1750	**9**	29.503	9
9	2056	869	1800	1120	1750	**10**	130.435	10
10	2438	1125	1800	1170	1750	**10**	91.615	10
11	2490	1763	1800	1200	1750	**10**	52.795	10
12	2472	2453	1620	1270	1750	**9**	13.975	9
13	894	3305	1800	1140	1750	**10**	114.907	10
14	960	2399	1800	1190	1750	**10**	76.087	10
15	976	1559	1620	1240	1750	**9**	37.267	9
16	1420	915	1800	1110	1750	**10**	138.199	10
17	1440	535	1800	1160	1750	**10**	99.379	10
18	1509	175	1800	1210	1750	**10**	60.559	10
19	2035	0	1620	1260	1750	**9**	21.739	9
20	1996	415	1800	1130	1750	**10**	122.671	10
21	2009	611	1800	1180	1750	**10**	83.851	10
22	997	820	1800	1230	1750	**10**	45.031	10
23	955	17	1620	1280	1750	**9**	6.211	9
24	1051	0	—	1150	1750	—	107.143	—

The output answer is emphasized by the boldfaced entries. Given the arrival rates $Q_{1\text{in}}(t)$, the optimal schedule for opening toll booths is given in the T_{open} column.

The MatLab program `poisson_stream(x)` generated the arrival rates, $Q_{1\text{in}}(t)$, using the uniform arrival times of the Primitive Model as the mean `x` to the Poisson distribution.

The "—" entries of the table indicate an inability to calculate a value. In particular, we cannot determine how many toll booths to open without a projection of the data for the next hour.

Analyzing $Q_1(t)$, we can see that by the end of the day, the toll booth was able to handle the 42,000 cars that went through it. An initial seed started the queue at zero, and by the time the non-peak hours roll around at the end of the day, the queue is cleared out – justifying the seed of $Q_1(1) = 0$ for the next day. Since there is no surplus for the next day, we conclude that the toll booths can adequately handle their economic demand.

Likewise, by closer inspection of the Q_1 column, we can draw some conclusions about how long motorists had to wait to pay the toll. Since the granulation is in one-hour intervals, we can only conclude that from 6:00am to 6:00pm and 9:00pm to 6:00am, the wait was less than an hour. For the three hours of 6:00pm to 9:00pm, since the size of the first queue is greater than what the toll booths are able to take, there is a wait of more than an hour.

The final column worth commenting on is Q_2, which indicates that the after-plaza is, as desired, close to capacity for most of the day. When traffic is generally low, a lot of toll booths are open to fill up the queue.

When comparing the 2-Queue Model to the Primitive one, we make an interesting, but not unexpected observation: the average number of toll booths that we deploy throughout the day in this model is 10 – the number at which things started to get out of hand when we had 10 toll booths open throughout the day. The conclusions we draw from that is that this model better allocates toll booths somewhat and that the size of the second queue, and the rate at which cars can leave the after-plaza is the binding constraint of the model's solution.

It should also be mentioned that the maximum number of booths open during the day in the 2-Queue model is 12, which corresponds to our conclusion in the Primitive Model where we determined we needed 12 toll booths, or three booths per lane of traffic as an optimal number of booths.

3.4 Verification of the 2-Queue Model

The 2-Queue Model has its advantages and disadvantages. The model excels at planning the toll booth schedule during the non-peak times. Congestion is minimized and toll booth scheduling is optimized to avoid unnecessary open toll booths.

Problems still occur at peak times, and may arise in part because of the constraints imposed on the system – in particular that the only variable we have control over is T_{open}. Since we have assumed that long-term solutions, such as those that would involve construction work, are out of the scope of control, the binding constraints of $Q_{2\text{cap}}$ and more so of $Q_{2\text{out\_cap}}$ are the principle cause of the congestion.

An additional shortcoming of the model is the lack of rigor in the modeling of the arrival times. We distributed them appropriately by a Poisson distribution, but the figures used still may not accurately represent a case study of what happens in the real world. Every three hours we get big jumps in the arrival rate, an artifact of the original data we explicitly assumed that was given in time intervals of three hours. A different distribution of the 42,000 cars over the course of the day may have resulted in more or less serious peak time congestion. The results for non-peak times appear stable enough that the model could handle an adjusted influx rate ($Q_{1in}(t)$).

4 The Toll Booth Utilization Model

4.1 The Motivation for the Model

Although in the previous models we appropriately modeled the distribution of cars by a Poisson distribution, as stated previously this might not necessarily concur with real-world phenomena. Everyone who has driven on a highway knows that some days traffic is worse than other days. This is due to the fact that the number of vehicles fluctuates in a random manner. An attempt to model this behavior by necessity must be done using a simulation model whereby random figures can be generated. We developed this model because a simulation process is intended to enable us to understand the general behavior of a random system. Using a computer to imitate the system's behavior, we can gain an understanding of the system's behavior over time. We were interested in seeing the effects of differing the number of toll booths used in the simulation model.

In the sections that follow, we present the results of the simulation process for the entire range of booths from 4 to 14. Although we imposed a maximum of twelve booths in the previous models, for the simulation experiment we removed that constraint as a verification of our conclusions in the previous sections. We also removed this constraint because we assume we can build the plaza as large as we need it to be.

The class of assumptions made in the Simulation Model is the same as for the previous models.

4.2 The Formulation of the Model

For this model, we use simulation to formulate the random process by which vehicles would arrive at the toll plaza. We wish to examine the average time a car will have to wait in line before being able to pass through the toll plaza. To answer this, the simulation generates a random distribution of vehicles entering the toll plaza.

To explain the random nature of the numbers generated, we have assumed as before that the number of vehicles entering the toll plaza follows a normal distribution as offered in the Primitive Model and modeled more explicitly by a Poisson distribution in the 2-Queue Model. What makes this model different is that we have a assumed a standard deviation from the normal distribution, and the simulation selects at random the deviation, up to a certain limit, and adds to or subtracts from the normal distribution. This accounts for the real-world scenarios of people calling in sick to work, or being on vacation or perhaps just running late. The point is that this process determines the number of vehicles that arrive at the toll area in our model.

We call the service rate the flow of vehicles from Q_1 into Q_2. The service rate is determined by the number of toll booths open and operating.

The granulation for this model is offered in minutes instead of hours. The average arrival rate per minute is calculated by taking an average hourly rate and then dividing by 60. We assumed a standard deviation of at most 250 vehicles per hour, which would correspond to the rush hours. For the decrease in the traffic flow during non-peak times we assumed a lower deviation and incorporated this into our model.

We presume that an in-depth explanation of the model is unnecessary due to the fact that we have the same assumptions and are simply modeling them in a slightly different manner. We used an iterative process to get an average over a 25 day period. The 25-day average is sufficient for the scope of this project, however, if given more time to run the simulations we could have obtained more accurate results.

4.3 Analysis of the Model

Using the simulation program, the source code for which is presented in Appendix A, we obtained the results presented in Table 7.

Table 7: The random distribution of vehicles and varying number of tollbooths

number of tolls	average wait Q_1	average wait Q_2	average wait $Q_1 + Q_2$	average length Q_1	average length Q_2
4	161	0	80	10228	0
5	160	0	80	8691	0
6	146	0	73	6115	0
7	119	0	59	4328	0
8	106	0	53	4066	0
9	55	0	27	1570	0
10	41	0	20	1219	0
11	6	13	9	180	348
12	1	18	9	36	479
13	1	36	18	47	1055
14	0	42	21	0	1215

Although in the previous models we imposed a maximum of 12 toll booths, corresponding to 3 booths per lane of traffic (assuming a four-lane highway), for the simulation model we allowed up to 14. We did this simply as a verification of the previous models, because we have not yet explored the possibility of more toll booths. We see in Table 8 that while more toll booths is excellent for those waiting to pay the toll, it is not as good for those trying to leave the toll plaza. Based on the data in the table, we determined that perhaps we should further explore the ramifications of having 10, 11, or 12 toll booths.

We make note again that the limiting case of 4 toll booths, corresponding to one booth per lane, is not a good solution at all, as indicated in the Primitive Model.

In order to examine the cases where we have between 10 and 12 toll booths, we ran the simulation ten times for each case. We then averaged the results and determined that this average was the best solution for each case. The results are given in the following table. (Table 8)

Table 8: The average cases.

number of tolls	average wait Q_1	average wait Q_2	average wait $Q_1 + Q_2$	average length Q_1	average length Q_2
10	31.5	0	15.6	912.9	0
11	16.2	20.8	18.4	474.2	627.7
12	3.8	27.8	15.4	121.9	806.8

Based on the averages in Table 8, we find again that 12 toll booths fits our definition of optimal. The average wait time and length of Q_1 is minimized, which has been our primary objective in the previous models. We sacrificed Q_2, in terms of length and waiting times, so that Q_1 would be as small as possible. This is consistent with our previous models. We offer this model as a verification of our resolutions as presented in the previous sections.

Given more time, we could have examined the notion of a priority queue in this model. Each model presented thus far deals with a first in, first out scenario (FIFO), and therefore all customers are treated equally. In reality there exists a priority queueing system such that customers arriving are given different weights. The priority customers are then treated first, or perhaps offered more expediency. A priority queueing system would enable the toll plaza to deal with more cars. We treat a possible scenario of priority queueing in the next section.

Also, we perceive a direct correlation between the length of Q_2 and the time it takes a car to leave the toll area. Given more time we would have liked to analyze this further, and present the correlation and some possible ways to address this issue.

5 The Dual Flux Model

Though the 2-Queue Model adequately takes into account the physical constraints of our toll plaza, we still get an unacceptable amount of congestion because of the binding constraints imposed by the size of the after-plaza and of the rate at which vehicles can funnel out of the after-plaza. Up until now, we have assumed limited control over the variables of the model: that the only thing we could change is the number of open toll booths. If the city is in fact interested in alleviating congestion, it may be interested in making more fundamental changes to the infrastructure of the toll area. Our group of modelers will present a system that should require a reasonable amount of investment, with a marked improvement in the effectiveness of the 2-Queue Model that is embedded in the system. We dub this model the Dual Flux Model because it involves two separate flows of motorists. First we discuss some preliminary issues, and then explain the formulation of the model. A solution and verification of the model follows.

5.1 Preliminary Considerations and Financial Investments

There are several ways one could enhance the structure of the toll plaza:

- increase number of toll booths
- build a larger plaza
- expand the four lane highway
- implement something more unusual

Clearly from what we have seen in the 2-Queue Model, the maximum number of toll booths is not a binding constraint and would not be the smartest course of action to invest funding in. Instead, expanding the after-plaza would lead to results in alleviating congestion. Expanding the number of lanes on the highway, though helpful, is impractical, for then you get into the issue of how far to extend the expansion beyond the toll plaza and the construction costs therein.

What we propose is a bit more unusual, and whose effectiveness must be measured against an investment into the expansion of the after-plaza. It involves a variety of relatively modest investments:

- One more toll booth
- One Fast Track automated booth
- One kilometer of one lane road

Fast Track is a system inspired by the FasTrak system used by the Golden Gate Bridge toll booth. [8]. FasTrak allows motorists who frequent the toll road to place an electronic tag in their windshield so as every time they go through the toll area, they may enter a special lane where their tag is read quickly and sends them on their way. Our version, Fast Track, is not a long term purchase that motorists can make but a one time use pass that works in the same way as described for FasTrak. The reason we will not consider a multi-use FasTrak explicitly in the model is because it does not benefit the alleviation of congestion overall, [9] and so does not play into our considerations. A multi-use FasTrak is certainly a vital implementation for any congested toll booth area, however, in order to satisfy a small subset of toll-paying commuters.

The additional toll booth is installed along the right lane of the highway before the toll area and before the end of the first queue. A Fast Track automated booth is added to the toll area, with the one kilometer road running from it to the highway where it merges. The purpose of these implementations is a modeling issue, and so covered in the model formulation section to follow.

5.2 Formulation of the Dual Flux Model

Let's consider the ramifications of the proposed implementations. A motorist will have two choices, depending on what lane they are in. The rightmost lane, lane 4, will purchase a "One Time Fast Track" (OTFT) while the other three lanes continue their commute. When the OTFT motorists reach the toll plaza, their lane – lane 4 – will channel to the new Fast Track automated booth. By this point, they have placed their OTFT pass inside the windshield and can make their transaction without coming to a complete stop. The gate several meters in front of them opens and they gain access to the one lane road that bypasses the second queue and then merges with traffic.

A flow chart of the model is given below in Figure 3.

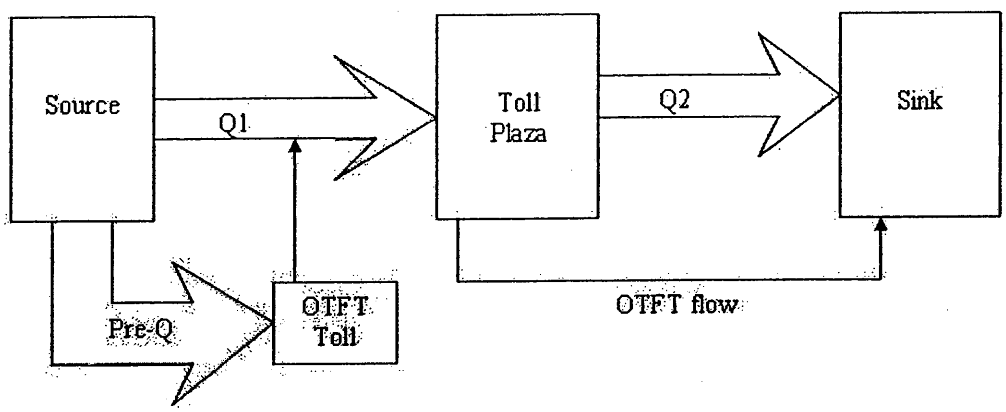

Figure 3 : A depiction of vehicle flow in the Dual Flux Model. $Q_{2\text{penalty}}(Q_2)$.

What occurs here is an alleviation of the binding constraint of the size of the after-plaza and the abrupt merging of a lot of traffic into four lanes. A proportion, η of cars coming into the first queue will never see the second queue; only those that have the presence of mind to wait in a line prior to arriving at the toll booth will actually bypass it. Consequently, we have a $1 - \eta$ proportion of motorists being handled by the same number of maximum toll booths and the same size after-plaza merging into four lanes as before. Motivating this formulation is the prospect of alleviating congestion by relaxing the load on the binding constraints.

It is worth noting that the queue, Q_{pre}, that is established for the purchase of OTFTs is similar to the limiting case of our Primitive Model, save that we have one toll booth for one lane instead of four and four.

There are no additional queues added to the model save Q_{pre}. We assume that at the OTFT lane of the toll plaza, verification of the OTFT is quick enough to not build up a queue. Likewise merging into traffic of the highway beyond the after-plaza is trivial enough to ignore, since it would not be dissimilar from a standard highway entrance-ramp.

Since we do not model this thing called "presence of mind", we make a standard assumption that people will make the most economic choice. When you consider that at one extreme when nobody purchases an OTFT, one person can purchase it and not wait in the "pre-queue" nor the second queue. It should be obvious too for the other extreme that if everyone utilized the system, then it would be less desirable than going through the plaza as before. Therefore, η is going to have a specific value dictated by the behavior of the queues for each flow of motorists.

The final assumption we mention before moving on to the solution of the model is that the same data that we have used for the Primitive and 2-Queue models will apply to this model as well. This will make comparison and contrasting straight forward in the analysis.

5.3 Solving the Dual Flux Model in Excel

First, we determine η, and then formalize the behavior of the queues for each flow of motorists.

As alluded too, the pre-queue is similar in structure to the limiting case of the Primitive Model. Because of that, the Excel program used for that model was tweaked to consider a one lane road (really one of four lanes) with one toll booth. We then examined different proportions of the $Q_{1\text{in}}$ stream as fed through this sub-model. What we found was that a proportion of $\eta = .10$ made the queue have a stable ebb and flow in the sense that the queue was able to empty itself out by the end of the 24 hour period. The results are given below in Table 9.

Table 9 : Results of the sub-model to determine η

t	Q_{1in}	Q_{pre}	T	T_{open}
1	249	0	180	1
2	251	69	180	1
3	252	140	180	1
4	149	212	180	1
5	156	181	180	1
6	149	157	180	1
7	194	125	180	1
8	200	139	180	1
9	206	160	180	1
10	244	185	180	1
11	249	248	180	1
12	247	317	180	1
13	89	385	180	1
14	96	294	180	1
15	98	210	180	1
16	142	128	180	1
17	144	90	180	1
18	151	54	180	1
19	204	24	180	1
20	200	48	180	1
21	201	68	180	1
22	100	88	180	1
23	96	8	180	1
24	105	0	180	1

Multiplying the Q_{1in} stream by the chosen η yielded non integer results. These were rounded to give us the figures seen in the Q_{1in} column. The number of toll booths open was not within our control this time. There is assumed just one booth open throughout the day. What we did adjust is η, in increments of hundredths and found that .10 gave us the ebb and flow of Q_{pre} as seen in the table: stable in that it doesn't increase out of control over time.

The next task was to go back to the 2-Queue Model with some appropriate modifications: scale the Q_{1in} stream by $1-\eta$ and subtract out those who used the OTFT system from entry into the second queue. What we get is a bind on the same constraints, but with a distinctly reduced level of congestion. The congestion in the main plaza flow of motorists is given in Table 10

Table 10 : Results for the plaza flow of the Dual Flux Model η

t	$Q_{1\text{in}}$	Q_1	T	Q_2	$Q_{2\text{out}}$	T_{open}	$Q_{2\text{penalty}}$	T_{penalty}
1	2488	0	2160	0	1750	**12**	1000	12
2	2510	328	2160	410	1750	**12**	681.677	12
3	2518	678	2160	787	1750	**12**	388.820	12
4	1493	1036	1980	1129	1750	**11**	123.137	11
5	1557	549	1800	1256	1750	**10**	25	10
6	1486	306	1800	1251	1750	**10**	28.804	10
7	1936	0	1620	1270	1750	**9**	13.742	9
8	2001	316	1800	1140	1750	**10**	114.674	10
9	2056	517	1800	1159	1750	**10**	100.388	10
10	2438	773	1800	1157	1750	**10**	101.708	10
11	2490	1411	1980	1130	1750	**11**	122.904	11
12	2472	1921	1980	1219	1750	**11**	53.882	11
13	894	2413	1980	1257	1750	**11**	24.457	11
14	960	1327	1800	1245	1750	**10**	33.230	10
15	976	487	1800	1163	1750	**10**	97.438	10
16	1420	0	1800	1164	1750	**10**	96.429	10
17	1440	0	1800	1214	1750	**10**	57.609	10
18	1509	0	1620	1264	1750	**9**	18.789	9
19	2035	0	1800	1134	1750	**10**	119.721	10
20	1996	235	1800	1184	1750	**10**	80.901	10
21	2009	431	1800	1210	1750	**10**	60.326	10
22	997	640	1800	1217	1750	**10**	54.969	10
23	955	0	1800	1203	1750	**10**	65.839	10
24	1051	0	-	1253	1750	-	27.019	-

In producing Table 10, we simplify the model somewhat by assuming effectively that there is no wait in Q_{pre}. The justification for this assumption is that, since we are only considering one hour intervals, it is negligible enough to ignore. The results given in Table 9 give us an accurate idea of what is going on in the pre-queue, considered independently from the program run of the main plaza flow that yielded Table 10. For Table 10, it should be noted that the number of toll booths open includes the OTFT booth up the highway, since the OTFT motorists are a part of the first queue of the model.

5.4 Verification of the Dual Flux Model

By looking at $Q_{1(t)}$ in the table, and comparing to the results in Table 6 – the results of the 2-Queue Model's solution – we can see that instead of three time intervals of wait times over an hour, there is just one in the Dual Flux Model at $t = 13$ where there are 2413 motorists in the first queue with only 1980 that can be serviced. The 2-Queue Model had its worse hour at $t = 13$ with 3305 motorists waiting and only 1800 that can be serviced. The reason the Dual Flux Model is able to handle an 11th open toll booth is precisely because of the relief the model provides on the after-plaza and its outflow rate – the original motivation for this model's formulation.

Non-peak times were affected too. The first queue was emptied out during 8 time intervals: $t = 1, 7, 16, 17, 18, 19, 23, 24$ while the 2-Queue Model had 3 such time intervals. The conclusion then is that the Dual Flux Model alleviates congestion over the entire course of the 24 hour period.

As was stated previously, the drawback to the model is that it requires an investment of time and money to add the functionality to the toll area. It is difficult to measure how economical this course of action is when compared with the idea of simply expanding the after-plaza. The Dual Flux investment does have the added benefit of allowing a multi-use Fast Track to be easily implemented, since the specialized toll booth would already be built at the plaza.

Another disadvantage to the model that should not be overlooked is the assumption that people will make the best decision over the scope of the entire scenario. Those who are unfamiliar with the system may not be willing to try out the OTFT system. Likewise those who rely on immediate gratification will tend away from the lane that distributes the OTFT passes, not realizing the congestion that awaits them up the road in the toll plaza.

6 Possibilities for Future Work

Through our exploration of the posed problem of minimizing toll booth congestion, some possibilities for future work have arisen.

The biggest issue that needs exploring is a whole other modeling problem of deciding exactly how to use funds to expand the toll booth area for the best desired results, be those short term, somewhat ad hoc alleviation or long term planning with anticipation of emerging technology. The possibilities for investment include increasing the number of toll booths, expanding the size of the after-plaza, implementing the Dual Flux Model, and automating and making-electronic the payment process. In such a modeling problem, the costs and benefits of each course of action would need to be meticulously analyzed.

There has already been a great deal of work done in automated and electronic toll booths. We have mentioned in the paper the FasTrak system that the Golden Gate Bridge toll booths use to allow frequent motorists to quickly pass through by having a tag installed inside their windshield for quick reading.

There also exists the technology to read license plates of speeding cars. This method should be investigated as a means of allowing non-regular commuters of toll roads to pay quickly by receiving a bill in the mail based on the license plate read at points along the toll road. As part of this investigation, the legal issues should be explored, for motorists could find themselves getting unexpected bills in the mail because of possibly too-effective tolling methods.

7 Conclusions

A barrier toll road is such that any vehicle traveling along must stop at certain intervals to pay a toll. In all the sections previous, we made certain assumptions so that we could determine an optimal number of toll booths to deploy in a barrier toll plaza so as to minimize the traffic congestion inherent to such a system. We chose to define "optimal" as minimal congestion.

Our first model, the Primitive Model, gave a rough overview of the situation and offered a solution to the problem. We determined that an optimal number of toll booths to deploy would correspond to three booths per lane of traffic.

In the 2-Queue Model, we addressed the consideration that it is not optimal to deploy three toll booths per lane of traffic constantly. The traffic at 1:00 o'clock in the morning is never as busy as the morning rush hour. In the model, we used a Poisson process to estimate the variable rate of traffic flow and used that to determine how many toll booths to have in operation. The optimal number of booths to deploy as indicated in the 2-Queue Model, and under the assumption of a cap on the size of the after-plaza, was an average of 10 booths, with a maximum of 12, corresponding to approximately three toll booths per lane of traffic.

The Dual Flux Model offered a possible alternative to the need for all the vehicles to have to stop by implementing a "day pass" system that would allow some of the vehicles to avoid the toll plaza. We found, using the same process to estimate traffic flow as in the 2-Queue Model, that again we would need an average of 10 toll booths, with a maximum of 12, corresponding again to three booths per lane of traffic.

The Toll Booth Utilization Model addressed the one dynamic we had not addressed in our previous model, namely the issue of wait times. To simulate this we used a Java program that would randomly generate a deviation in the proposed normal traffic flow, due to considerations such as people missing work or running late. Using this program, we were able to estimate the average wait time for vehicles attempting to pass through the toll plaza, and using the average wait times, were able to determine that minimal congestion and wait time was obtained when we deployed 12 toll booths, corresponding to three booths per lane of traffic.

Our conclusion therefore is that an optimal number of toll booths to deploy in a barrier toll road would be approximately three booths per lane of traffic.

The section on Future Work addresses a few options to consider that might replace a barrier toll system, because the only true way to eliminate the congestion caused by toll plazas is to eliminate the need to have a barrier toll system.

References

[1] Allen, Arnold O.. *Probability, Statistics and Queueing Theory with Computer Science Applications* Academic Press, Inc., 1978

[2] Bertsimas, Dimitris and John N. Tsitsiklis. *Introduction to Linear Optimization* Massachusetts: Athena Scientific, 1997

[3] Detel, H. M. and P. J. *Java: How to Program* 6$^{\text{th}}$ ed. New Jersey: Pearson Education, Inc., 2005

[4] Hillier, Frederick S. and Gerald J. Lieberman. *Introduction to Operations Research* 3$^{\text{rd}}$ ed. San Fransisco: Holden-Day, Inc., 1980

[5] Maki, Daniel and Maynard Thompson. *Mathematical Modeling and Computer Simulation* Thomson Brookes/Cole 2005

[6] Nash, Stephan G. and Ariela Sofer. *Linear and Nonlinear Programming* McGraw-Hill, 1996

[7] Taha, Hamdy A.. *Operations Research: An Introduction* 6$^{\text{th}}$ ed. New Jersey: Prentice-Hall, 1997

[8] Golden Gate Bridge, Highway and Transportation District. 2005

[9] Bernstein, David and Owen J. Chen. *AM/PM Congestion Pricing with a Single Toll Plaza* Transportation Research Record No. 1498, 1995

[10] Tolling Solutions < www.chichwys.com > Feb. 6, 2005

[11] Automatic Vehicle Identification < www.society.omron.com > Feb. 6, 2005

[12] The New Jersey Turnpike Authority < www.state.nj.us/turnpike > Feb. 6, 2005

[13] E-470 Public Highway Authority < www.e-470.com > Feb. 6, 2005

[14] The History of Orlando-Orange County Expressway Authority Chapter 13

[15] F. Glover, D. Klingman, and N. Phillips *Network Models in Optimization and their Applications in Practice* John Wiley & Sons NY 1992

[16] R. Ahuja, T. Magnauti, and J. Orlin *Network Flows* Printice Hall, NJ, 1993

TEAM #...

1 Appendix A

```
import java.util.*;  //package for LinkedList
import java.lang.*;  //package for Integer

class Simulation
{
   public static void main(String args[])
   {
      int i;                      //generic loop counter
      int currentMinute;          //represents current state (minute) of model
      int totalCars          =0;  //sum of all cars that pass through the simulation
      int totalWaitTimeQ1    =0;  //sum of all individual car wait times in Q1
      int totalWaitTimeQ2    =0;  //sum of all individual car wait times in Q2
      int totalQueueLengthQ1 =0;  //sum of the size of Q1 over all the minutes
      int totalQueueLengthQ2 =0;  //sum of the size of Q2 over all the minutes
      int maxWaitQ1          =0;  //longest time a car waited in Q1
      int maxWaitQ2          =0;  //longest time a car waited in Q2
      int maxLengthQ1        =0;  //largest line in Q1
      int maxLengthQ2        =0;  //largest line in Q2
      int serviceRate        =36; //average number of cars that pass through the Tolls per minute
      int leaveRate          =30; //average number of cars that leave the model per minute
      int thisWaitTime;           //individual car wait time
      int exitTime;               //minute a car left the model
      LinkedList preToll = new LinkedList();  //This is Q1
      LinkedList postToll= new LinkedList();  //This is Q2

      for(i=0;i<100;i++) //loop 25 times to average variance
      {

         for(currentMinute=0;currentMinute<(60*24);currentMinute++)
         {      //loop for every minute of one day

            for(i=0;i<carsArrivedThisMinute(currentMinute/60);i++)
            {      //loop until all cars enter queue for that minute
               preToll.add(new Integer(currentMinute));    //enqueue arrive at a certain minute
               totalCars++;    //add one to sum of cars
            }

            for(i=0;i<serviceRate;i++)  //loop until the maximum number of cars have received serviced (dequeue)
            {
               if(preToll.size()>0)    //if there are cars waiting to be serviced
               {
                  exitTime=((Integer)preToll.removeFirst()).intValue();   //minute that a car exits Q1
                  thisWaitTime=currentMinute - exitTime;  //total wait in minutes that a car had to wait in Q1
                  totalWaitTimeQ1=totalWaitTimeQ1 + thisWaitTime; //update total

            //update date max if neccessary
                  if(thisWaitTime>maxWaitQ1)
                  {
                     maxWaitQ1=thisWaitTime;
                  }

                  postToll.add(new Integer(currentMinute));   //enqueue service time
               }
            }   //end of servoce this minute

            for(i=0;i<leaveRate;i++)    //loop until the maximun number of cars exit the model
            {
               if(postToll.size()>0)   //if there are cars waiting to leave
               {
                  exitTime=((Integer)postToll.removeFirst()).intValue();  //minute that a car exits Q2
                  thisWaitTime=currentMinute - exitTime;  //total wait in minutes that a car had to wait in Q2
                  totalWaitTimeQ2=totalWaitTimeQ2 + thisWaitTime; //update total

            //update maximum if necessary
                  if(thisWaitTime>maxWaitQ2)
                  {
                     maxWaitQ2=thisWaitTime;
                  }
               }
            }
```

```
//update maximums if neccessary
                totalQueueLengthQ1=preToll.size()+totalQueueLengthQ1;
                if(preToll.size()>maxLengthQ1)
                {
                    maxLengthQ1=preToll.size();
                }

                totalQueueLengthQ2=postToll.size()+totalQueueLengthQ2;
                if(postToll.size()>maxLengthQ2)
                {
                    maxLengthQ2=postToll.size();
                }

            }//end of day
        }//end of 25 iterations

//Display results

        System.out.print("Avg wait time Q1: ");
        System.out.println(totalWaitTimeQ1/totalCars);

        System.out.print("Avg wait time Q2: ");
        System.out.println(totalWaitTimeQ2/totalCars);

        System.out.print("Avg wait time in both Queues: ");
        System.out.println(((totalWaitTimeQ1/totalCars)+(totalWaitTimeQ2/totalCars))/2);

        System.out.print("Max wait time Q1: ");
        System.out.println(maxWaitQ1);

        System.out.print("Max wait time Q2: ");
        System.out.println(maxWaitQ2);

        System.out.print("Queue length per minute Q1: ");
        System.out.println(totalQueueLengthQ1/(60*24));


        System.out.print("Queue length per minute Q2: ");
        System.out.println(totalQueueLengthQ2/(60*24));

        System.out.print("Max queue length Q1: ");
        System.out.println(maxLengthQ1);

        System.out.print("Max queue length Q2: ");
        System.out.println(maxLengthQ2);


    }


    private static int carsArrivedThisMinute(int hour)
    {
        Random myRandomNumberGenerator = new Random();  //random number generator
        int variance;
        int[][] flowTable = //row 1=average arrival rate; row 2=standard deviation; Column = hour
            {{18, 18, 25, 25, 18, 35, 40, 40, 40, 35, 25, 25, 18, 25, 18, 35, 35, 40, 40, 35, 25, 25, 25, 18},
             { 2,  2,  2,  3,  3,  3,  4,  4,  4,  3,  3,  3,  2,  2,  2,  4,  4,  4,  3,  3,  3,  2,  2,  2}};
//this will allow the variance to effect the mean arrival rate either positively or negitevly
        variance = myRandomNumberGenerator.nextInt(2*flowTable[1][hour]) - flowTable[1][hour];

        return flowTable[0][hour] + variance;   //
    }
}
```