Maximizing Profit When Expanding the New York Thruway: A Simulation Study

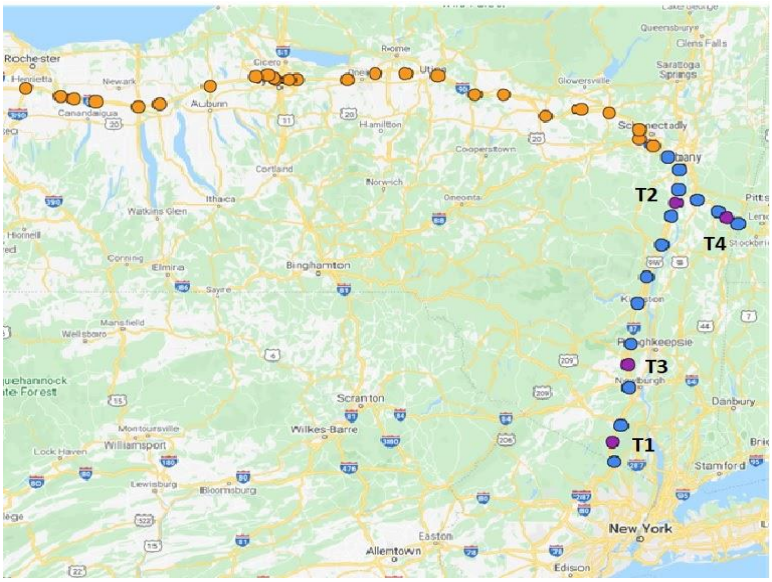
---

Introduction

The New York thruway is a 570 mile “superhighway” that allows travel between many areas of New York state and may require tolls to use [1]. What makes it an interesting subject for a simulation study is that the thruway authority and New York State has amassed a large, well-kept library of publicly available data sets regarding the types of vehicles that use it, as well as where they entered the system and subsequently exited it, in 15 minute or hour intervals [2,3]. In this report, we will present the main steps to achieve this goal. Briefly they are: 1) Data collection, 2) Input Analysis, 3) Simulation parameter selection, 4) Simulation code Creation/Execution, 5) Output analysis.

Problem Summary

The main goal of our project is to compare two scenarios where we add “new” exits to see which configuration we create would maximize revenue. These new exits are named “T1”, “T2'', “T3” and “T4”, and are marked by purple dots in the following map. Out of these four options, we must choose three. We also simulate profits made on the section of thruway marked by blue dots, as it is.



Map showing the locations of the “new” exit sites taken into consideration, labelled as T1, T2, T3 and T4

Results

For the former of our goals, i.e., determining three out of 4 profitable toll stations that are being proposed, we conclude that it is more profitable to build tolls T1, T3 and T4, and their combination will yield the most profit. Its annual profit exceeds that of the second-best configuration by about $22,000. The coverage percentages are approx. 89% for 10000 runs.

For the simulation of the existing model, the average error was the same when we perform analysis on input data pertaining to “Cash” paying motorists and “E-ZPass” using motorists. Thus, payment type does not affect the profits.

A possible cause of error would be that we only used E-ZPass data to calculate the adjustment, however, when we redid the analysis the adjustment value did not change significantly (about 2%) so we divided to keep the original adjustment amount.

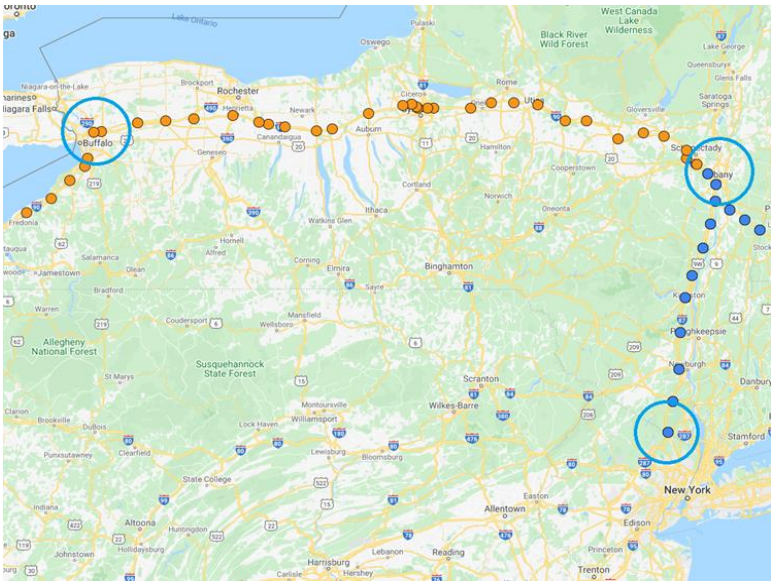
Data Collection

The Data for this project was gathered from two major sources the first was the New York state database [2]. From this website we were able to obtain the number of vehicles that had the same (type, entrance, exit) grouped by hour. Here we used the provided API to page through the data and aggregate it by week.

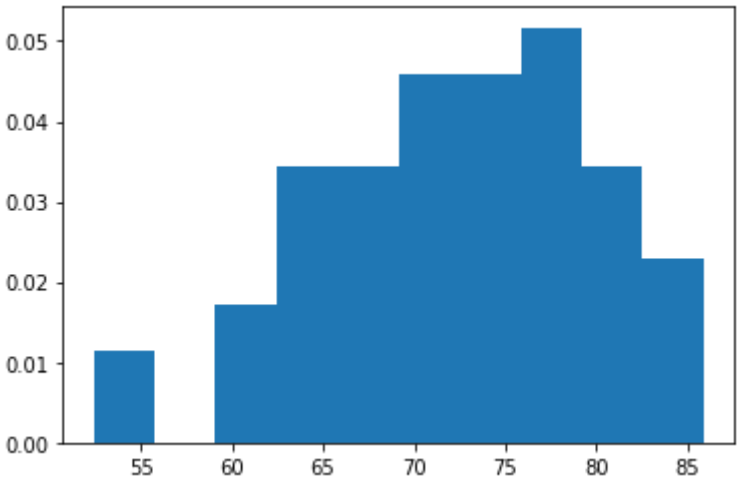
The second major data source was the New York Thruway authority website [3]. From this website we were able to obtain the toll data used in our simulation. This website did not contain an API so we downloaded the HTML of each (type, entrance, exit) tuple page to obtain the associated price of the trip.

Input Analysis and Data Visualization

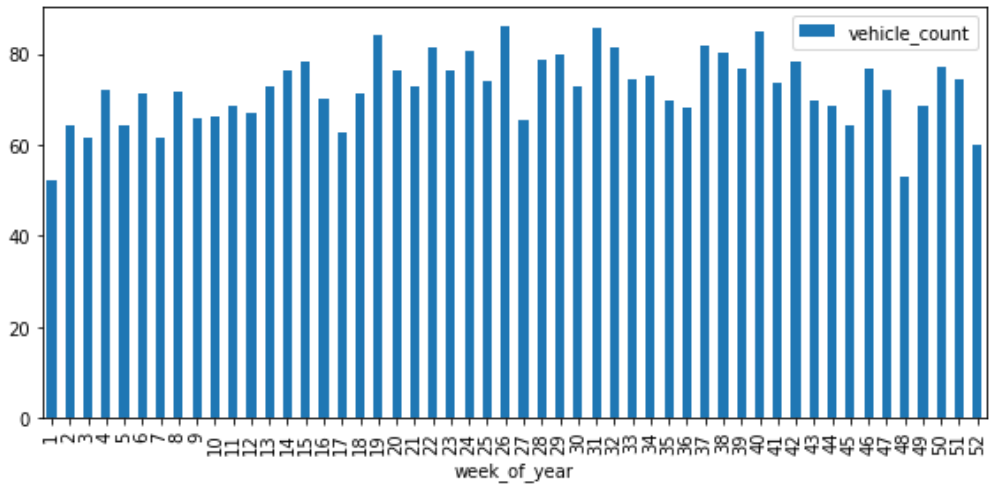
For determining the probability distributions for each incoming vehicle class at each toll station, we computed all 9*52 = 468 summary statistics, where 9 is the number of vehicle classes and 52 is the number of entrance sites or tolls. We performed a preliminary analysis on the existing toll stations to compare profits, and found toll stations 50, 24 and 15 to be more profitable than others, in the years 2017 and 2019. Based on this information, we decided to choose the toll stations marked by blue dots for the former goal of our project. Two of these toll stations, 24 and 15 lie on the branch of stations that we chose for placing new toll stations.



Entrance Sites 50, 24 and 15 in clockwise order

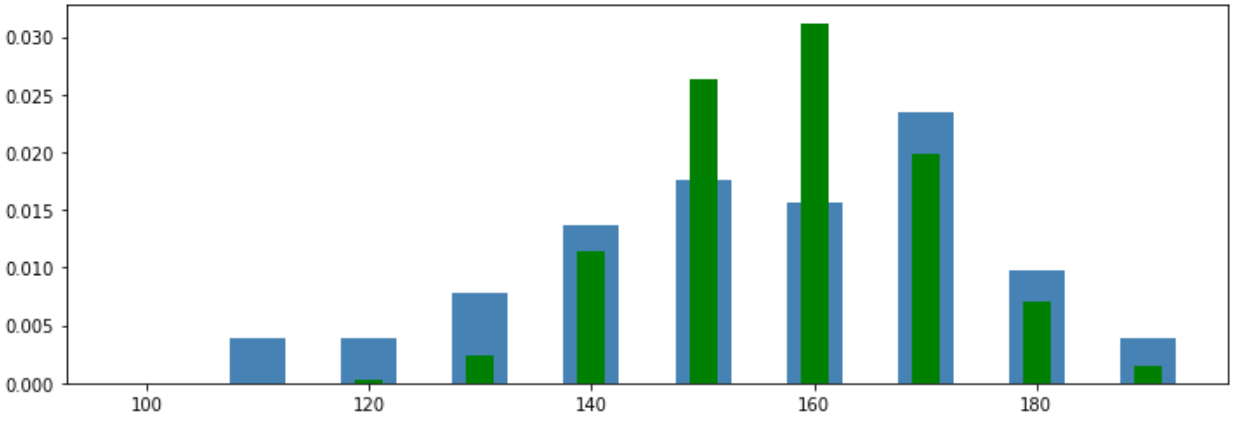


Normalized histogram for the average number of vehicles of class 2H at entrance site 15 in 2019

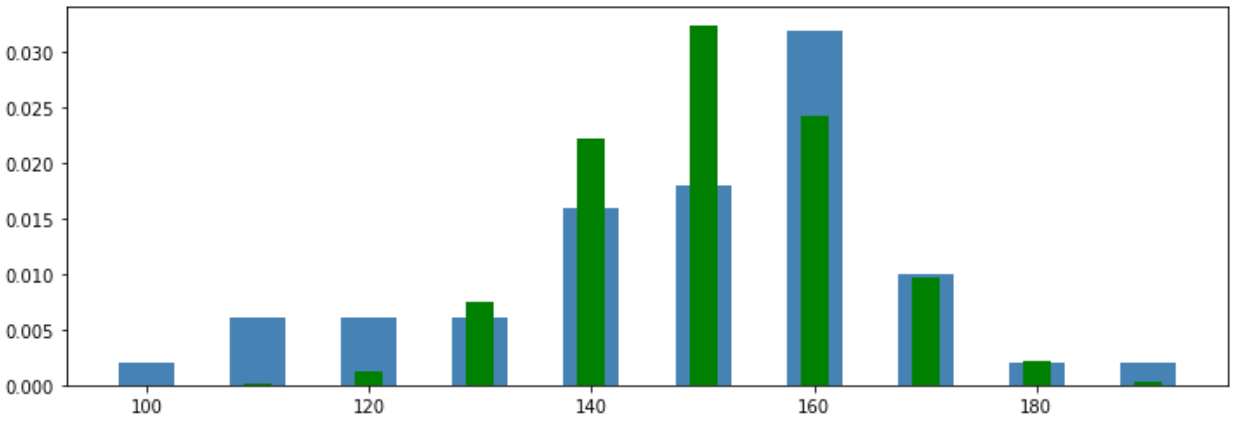


bar plots for average number of vehicles of class 2H at entrance site 15 in the year 2019

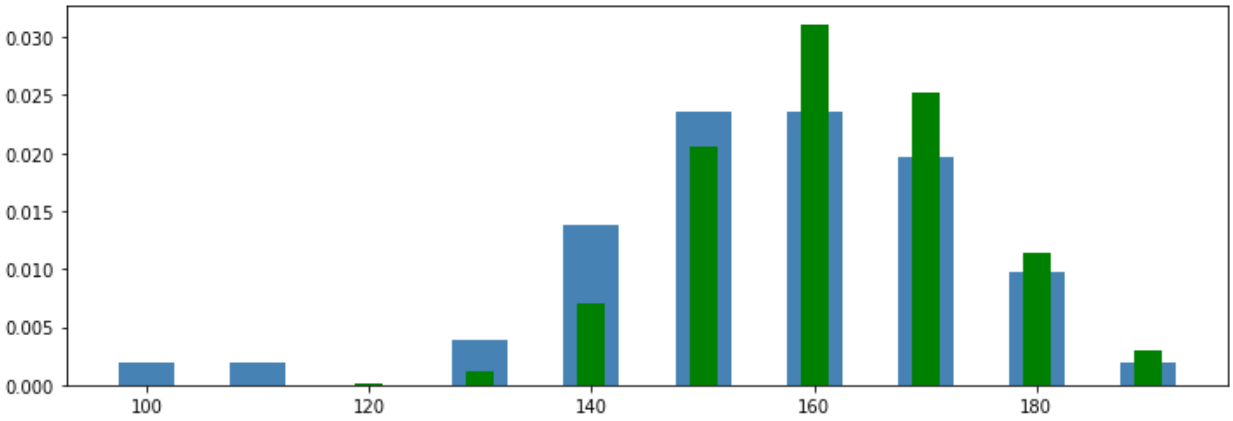
Since we have count data at hand, we fit Poisson and Negative Binomial Distributions and determine these parameters for all vehicle class and toll station combinations for the years 2017, 2018 and 2019. The Poisson fit for the average number of vehicles of class 2H at entrance site 15 for the year 2017, 2018 and 2019 are given below:



Poisson fit for the average number of vehicles of class 2H at entrance site 15 in 2017



Poisson fit for the average number of vehicles of class 2H at entrance site 15 in 2018



Poisson fit for the average number of vehicles of class 2H at entrance site 15 in 2019

We generate parameters for Poisson and Negative Binomial distributions and perform Chi-Square Goodness-of-Fit tests. Judging by the p-values, for most input distributions, Poisson distribution is a good fit, so we go ahead with Poisson.

Algorithm Intuition

We will begin by giving some intuition on how our algorithm works. With this intuition established we will move on to a more rigorous discussion of the terminology and strategies used in our simulation's implementation.

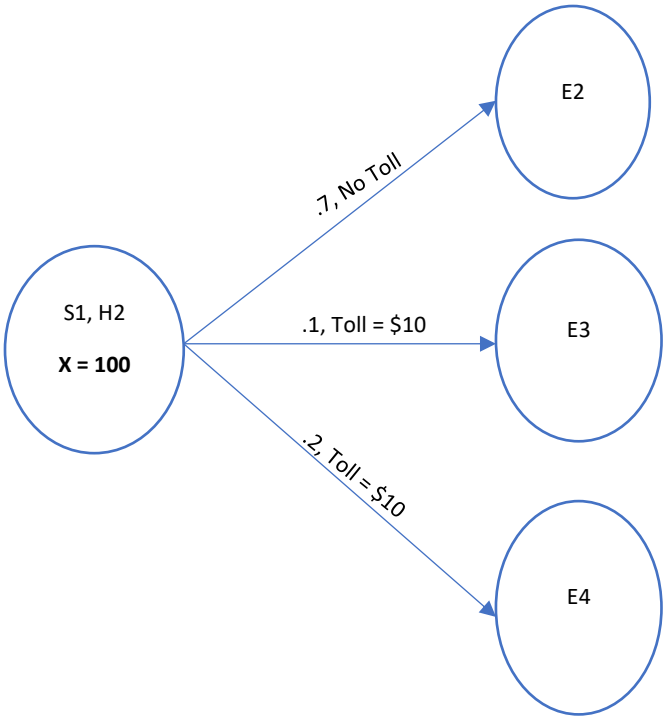


Fig 1. Transition Graph

Our algorithm can be thought of as a bipartite graph with all the (entrance, vehicle type) tuples, what we call "configurations", as the left set of nodes and all the exits as the right set of nodes. The left node will also contain the number of vehicles of that configuration type that showed up in one week. This arrival variable X will be our random variate.

The edges contain two pieces of information: 1) the proportion of cars that, given the configuration on the left node, go to the specified exit on the right node. 2) The total sum of toll prices one must pay on route from the entrance on the left node to the exit on the right node. An example of part of the graph discussed can be seen in figure 1.

Our goal is to simulate how much profit our new tolls can make in a given week. Breaking this down into smaller sub-problems we can first calculate how much each configuration makes in a certain week and then sum up all these profits to calculate the grad total weekly profit.

Given our graph in figure 1, the first sub problem becomes easy to solve. We see that we have 100 cars arrive this week (X = 100). We know that 10% of cars will go to exit E3 and another 20% will go to exit E4.

On the path from entrance S1 to exit E2 we will have to pay a total of $10 in tolls. Since only 10% of cars pass through here on average, we know we will make an average of $1 per car for a total of 100 * .1 = $10 in estimated profit for this week. Using similar logic, we see that on the S1 to E4 path we will get an estimated profit of $20.

On the S1 to E2 path there is no new toll, so we do not get any profit from 70% of the cars in this configuration. Therefore, to calculate the total profit for this configuration we just sum up all the profits from each path. Giving us a configuration total profit of $30.

To calculate the grad total profit, you could just repeat the procedure above for each configuration and sum up all the configuration totals for the week. The reason that we use "configurations" instead of just entrance and exits is that the toll depends on the type of vehicle that is being driven.

Algorithm Terminology:

S = Set of entrances

V= Set of vehicle types

E = Set of exits

C = $S \times V$, Entrance Configurations

T = Set of new tolls

$\alpha_{c,t}$= Proportion of vehicles from $c \epsilon C$ that pass through $t \epsilon T$

$F_{t,v}$= Fee of toll $t \epsilon T$ for $v \epsilon V$

$r_{c,e}$= Proportion of vehicles from configuration $c \epsilon C$ that will take exit $e \epsilon E$

$P_{s,e}$ = Set of all $t \epsilon T$ on route between $s \epsilon S$ and $e \epsilon E$

β_c = Fee proportion constant

X_c = Random Variate for number of vehicles from configuration $c \epsilon C$ in one week

Algorithm

Although the explanation given above is intuitive, it was found to not be the most computationally efficient. Instead, what one must do is group configurations by tolls passed not by paths taken. The advantage of this is that in this mathematical model the proportion of vehicles that pass-through a given path, and therefore a given toll, is constant. Furthermore, the price one must pay at a given toll is also constant. Therefore, one can precompute all Fee proportion constants, how much of a toll fee is "actually" paid on average, thus reducing the amount of calculation needed to compute a whole week's profit.

What we mean by "actual" profit is the following. If we know that for configuration (S1, H2) only 30% of cars pass through a toll T1 that charges $10, we know that if 10 cars arrive for that configuration, we will make 10 * $10 * .3 = $30. If 100 cars show up in a week then we make 100 * $10 * .3 = $300 in profit. In both these cases, the proportion of the total toll fee that we get as profit is always 30% of $10 dollars because only 30% of the cars ever take this path. So, the actual profit we make from this toll is $3 dollars for each car that arrives for the given configuration. Therefore, if we can find the "actual" profit for every toll, given a configuration, we can sum them all up to precompute each of these fee proportion constants. This will save the number of computations needed to find the total profit made.

The first step to calculate total profit with this strategy is to compute what percentage of vehicles from a configuration pass-through a given toll:

$$\alpha_{c,t} = \sum_{e \in E} I\{r_{c,e}\}_{t \in P_{s,e}}$$

Equation 1

As seen in Equation 1, to calculate this one can iterate through all exits and use an indicator variable to sum only those paths that contain the target toll.

Once all $\alpha_{c,t}$ have been calculated one can, for each toll, multiply these proportions with the full toll fee associated with the vehicle type to obtain the fee proportionality constant for a configuration. This is shown in Equation 2.

$$\beta_c = \sum_{t \in T} \alpha_{c,t} F_{t,v}$$

Equation 2

Finally, to calculate total profit for one week one can sum the product of the proportionality constant with the random variate of arrivals as shown in Equation 3.

$$\text{Profit} = \sum_{c \in C} \beta_c X_c$$

Equation 3

Using this precomputing approach brings the time complexity of the algorithm from $O(|C| \times |E|)$ to $O(|C|)$.

The full algorithm is as follows:

A. Precompute β_c
    1. Build graph G of exits/new tolls
    2. Find path between each entrance exit in G using Dijkstra with each edge having a weight of 1
    3. Using the obtained paths compute R, a matrix which stores what paths go through which tolls
    4. Load exit probabilities and store them in matrix B
    5. Load toll fees into a matrix F
    6. Use R and B to compute $\alpha_{c,t}$ for all $(c,t) \epsilon\ C \times T$ and store results in matrix A
    7. Using F and A, compute β_c for all $c \epsilon\ C$ and store in matrix
B. For each week:
    1. Profit = 0
    2. For each configuration:
        i. Generate X_c
        ii. Look up β_c
        iii. Profit += $\beta_c X_c$

Running the Simulation

To make sure that the parameters we fit for the Poisson distribution generated realistic data we first ran a simulation that generated profits based on (type, entrance, exit) data we had already collected and could compare our results to. What we found is that the percent error was larger than expected with the average being ~-90%. We decided to adjust the random variate we generated by $X\ /\ (1\ +\ error)$ because we saw that:

$$\frac{Observed - Actual}{Actual} = error$$

$$\frac{Observed}{(error + 1)} = Actual$$

Would get us close to the actual values we gathered if we used the average error value on all varieties. The results of the errors are as follows:

	NON-ADJUSTED	ADJUSTED
MEAN	-.90	1.2e-15
STD.	**0.12**	**1.23**
MIN	-1.00	-1.00
25TH PERC.	-.955	-.524
50TH PERC.	**-.942**	**-.381**
75TH PERC.	-.904	.016
MAX	2.49	35.9

Table 1.

As can be seen in table 1. we where able to keep or reduce most errors but at the expanse of increasing the standard deviation having the top ~5% of values in the adjusted data being over 200% inaccurate. However, since the reduction worked well for most values, and due to time constraints, we moved forward with the adjusted simulation distributions.

We ran our extended toll simulation using the algorithm explained earlier and by using a new stream of random numbers each run to ensure independence among runs.

Output Analysis

For each of the toll configurations T1, T2, T3 and T4, we first computed the 90% CI for average profits per annum (μ). Next, we calculated the 90% CI values for profits for each week in each run, based on which we computed the coverage percentages and the average coverage percentages.

T1	T2	T3	T4
(444443.7091,444464.1968)	(485287.9594,485310.2953)	(462505.0496,462526.8662)	(452793.5510,468009.2490)

90% CI for average profits per annum (μ)

T1	T2	T3	T4
89.39365384615387	89.63230769230769	88.72173076923075	88.43326923076924

Average coverage percentages for profits per annum

Thus, we conclude that 10000 simulation runs give acceptable coverage percentages and the configuration T2, which excludes toll T2 and includes tolls T1, T3 and T4 will be the most profitable configuration. This result is in line with our preliminary analysis which hinted that tolls present at the ends of branches are possibly more profitable than others.

Works Cited

[1] New York State. "DATA.NY.GOV" Accessed December 15, 2020. https://data.ny.gov/

[2] Thruway Authority. "Toll and Distance Calculator". Accessed December 15, 2020. https://wwwapps.thruway.ny.gov/tollcalculator