Uber Driver Optimization Via Queueing Models

Maxwell Wagner

Abstract: Uber and similar platforms are becoming more commonplace in urban areas and have the problem of determining how they should operate. Driver optimization is crucial to success and short wait times. Drivers and Uber simultaneously need to turn a profit and provide proper service to passengers. There are three leading options for driver behavior after a fare; move randomly, remain stationary, or move toward a hotspot.

A model has been created where pseudo drivers and passengers are placed into a block based fictional city, and behave in one of the above three ways. The number of failed fares and fuel usage is recorded for 10,000 iterations of pickup and drop-offs, and then averaged. The results are inconclusive and two options present opposing viewpoints. Remaining stationary benefits drivers due to the decreased fuel usage, and eventual higher profits, while moving toward hotspots benefits Uber due to a lower amount of dropped fares.

Keywords: Queueing Theory, Uber, Location, Availability

Introduction: Uber has a unique market when it comes to business models. They are essentially a public service, however their drivers act independently from the company itself. Each driver is a contractor that Uber pays for their work. The problem for Uber arises due to them having two very different goals. One is to satisfy their customers and produce their rides as quickly as possible. The other is to sustain a healthy and professional group of drivers. The first goal is accomplished by successfully solving the second. However, drivers have an agenda of their own; to make money for themselves, with the secondary goal of making money for Uber. In **Figure 1** below, the basic model of Uber can be seen.

Figure 1: Uber's general model.

In order for a driver to productively make money from fares, they must have a strategy for when they drive, how they drive, and what fares they decide to take. There are three main trains of thought when it comes to taxi and pick-up services. The first is take the closest fare to the driver and remain stationary after the drop-off. The second is to target hotspots in the city, such as large events or known busy areas. The third is to drive randomly through the city after a drop-off in hopes that the driver will run into another fare. In optimum conditions, drivers can both profit, and provide fast pickup times.

The goal of this paper is give a glimpse at how an Uber driver should behave in order to benefit both themselves and Uber.

Literature Review: The most applicable field in comparison to taxi services is emergency vehicle distribution. The location of both stations, and individual vehicle locations is important in maximizing coverage. Erkut et al (2008) provide a framework for a node-like system with certain nodes having higher probabilities for high volume than others. Additionally, the research provides the idea that in order to service a node, the vehicle or station does not necessarily need to be on that node. It is purely required to be within a serviceable distance from it.

The ability to not be in the exact location of the destination is a key concept for the problem Uber faces. By positioning drivers at strategic locations and changing their methods, all blocks of a city can be covered with minimal passenger downtime and high driver profits. The advantage of a node system is that it allows for multiple connections to each location in the structure. Each imaginary city block has the potential to be closer or further from its neighbor, and can be connected to multiple other blocks.

Ball and Lin (1993) introduces the 0-1 integer programming technique to solving emergency vehicle reliability problems. That is to reduce more complex issues to simple whole number integers. For example, using a branch and bound method becomes much more feasible while using integers. The obvious downside in this situation is the potential for oversimplification and the loss of accuracy. In the case of this study, we are limited to whole blocks, instead of something more precise like addresses or GPS coordinates.

A theme between Erkut et al (2008) and Ball and Lin (1993) is the focus on consistency and reliability. The simplification of complex processes makes the model more stable and predictable. Unlike the case of emergency vehicles where response is necessary regardless of distance, taxi services like Uber can take advantage of the ability to drop or lose a small percentage of calls. By doing so, they are able to optimize response time to other passengers and by relation, increase profits.

Methodology: The preferred methodology in the case of Uber driver optimization is a fairly simple one. The setup itself relies on a location based idea where there are a certain number of drivers, and a certain number of passengers. The difference from a standard queue model is that there is location to take into consideration.

In order to simulate a city, or at least a block-based city, each passenger and driver was given a location from 1 to 400, simulating a 20x20 block city. To do this each was given a random number from 1 to 400, with replacement, meaning that drivers could potentially stack up on each other, and create a pseudo-hotspot for activity.

From this point, the approach split into three distinct groups. The first is a situation where each driver begins at their starting location, and remains stationary. They approach their nearest passenger, pick them up, and head to their final location. The amount of gas used, and number of passengers that didn't receive a ride is then measured.

The penultimate option has the drivers start at a location, but then drive a random distance away from their location in hopes that they find a passenger. The difference in this situation is the inclusion of a pre-fare travel time that consumes gas. An advantage is potential better placement for picking up passengers. Below in **Figure 2** is a short example of the random distance a driver will take before his/her fare.

```
d.start.loc <- sample(d.quadrant, d.start, replace = T)
d.change <- sample(-d.dist:d.dist,d.start, replace = TRUE)
d.start.loc <- d.start.loc + d.change
for (i in 1:length(d.start.loc)) {
  if (d.start.loc[i] > 400)
    d.start.loc[i] <- 400
  else if (d.start.loc[i] < 1)
    d.start.loc[i] <- 1
}
```

Figure 2: Example code for option two.

The final option for drivers to gravitate toward a hotspot in the city. In this case it is the median of passenger locations. Drivers will move a set distance toward the median depending on whether they are above or below the median. This uses slightly more gas than option one, less than option two, and places the drivers in a more optimal position for passenger pickups.

Passenger trip length follows a simpler method than the above and works the same for every driver strategy. They begin at their random start location, and have a random end location assigned within the city block.

Instead of repeating the cycle with the same driver and passenger crop, the process is simply randomized again from the beginning, which is reason for the movement before fares instead of post. In this way it is possible to see how many fares would be missed at each round, and then repeat. The missed passengers represent passengers who became impatient while waiting for a driver to arrive.

Results: After 10,000 iterations for each driving strategy was simulated, a general idea of the advantages and disadvantages of each method become visible, as there is no definite winner between option one and option three. Drivers remaining stationary and drivers moving toward hotspots in the city were clearly more suitable options than drivers moving randomly.

titles	leftovers	gas
"stationary"	"21.7394"	"115.477643747725"
"random"	"22.6617"	"128.51025366149"
"hotspot"	"21.4243"	"117.109562309126"

Figure 3: End results from 10,000 iterations of each method.

From **Figure 3** above, it is clear that there is a very marginal difference between the three methods' leftover amount, meaning that the amount of passengers remaining of the 250 initial is almost always in the 21 to 22 range. In most cases however, the random driving method has a slightly elevated amount of leftovers. In the case of gas usage, the random method comes out as the clear worst option, typically using 10 to 12 more units of gas on each fare. The stationary and hotspot methods are relatively similar, with the stationary method using slightly less on average.

The key concept from these results is that Uber and their drivers remain in a slight contradiction. For drivers, remaining stationary provides a slight edge in profit due to less gas being used, and a relatively similar amount of time taken on each fare. Uber's best interest is to pick up as many passengers as possible, and to do this they would prefer drivers use the hotspot method in order to reduce the amount of leftover passengers, or at least limit the amount of time they need to wait for a driver.

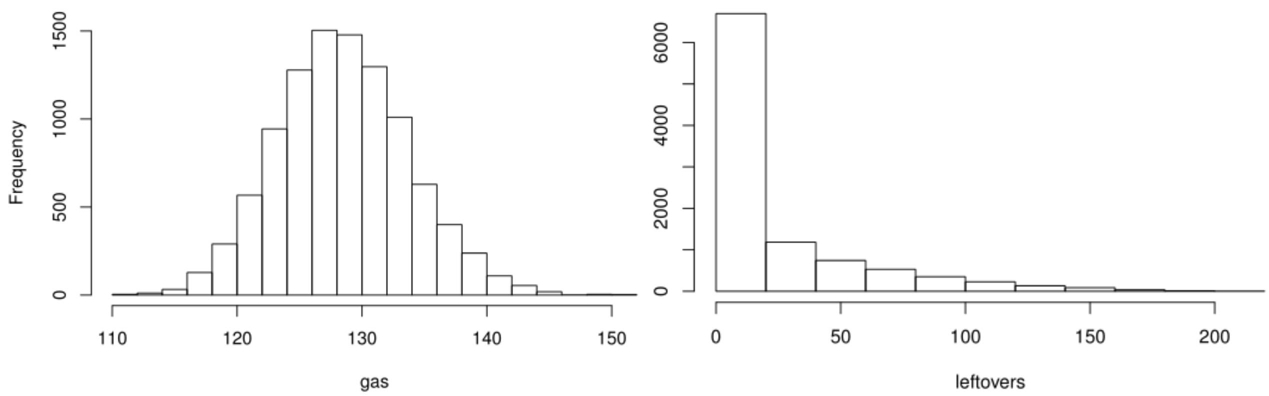

Figure 4: Frequency of gas usage and leftover amounts for option two (random).

Above in **Figure 4** are two graphs showing the frequency of both gas usage and leftover amounts for option two, random driving. The importance here is that gas usage depicts a normal bell curve, and that leftovers is predominately near the mean, with a few outliers.

Summary and Future Works: While the results show promise for deciphering how an Uber driver should behave, the model itself could use improvement and would benefit from a realistic data source. The lack of credible data from any of the major ride service companies hindered the ability to estimate genuine data and thus forced the use of artificial variables for trip length, number of cabs, number of passengers, and other time constraints.

However even with a relatively simple method, it is possible to rule out at least one of the driving styles. Random driving, while nearly as efficient in pickups, left much to be desired in the amount of gas used per fare. The stationary and hotspot methods offer opposing ideology that could benefit both drivers, passengers, and Uber.

A future and more beneficial model would include information based on swarm theory. Meaning that the simulation would learn from itself in a more natural way, and use other drivers in order to better estimate how to behave. Using other drivers' successes and failures would benefit the group as a whole, and eventually all units in the system would perform better as a group, instead of focusing on individual gain.

A priority system accompanies a swarm theory based model with the idea that each driver could prioritize passengers based on proximity. There is also the idea of ignoring short or long fares, in favor of the opposite. Traffic also plays a role in priority and drivers would realistically choose fares that avoid common congested routes. The randomness in fare length in this paper's model does allow for some "bad traffic" but in general follows a bell curve, with very few outliers.

There is also a complete lack of human behavior in the model. Instances where a driver makes a mistake and doesn't pick up a call, or a passenger doesn't show up is likely to happen in a model based on real data. There are other underlying issues like vehicle breakdowns that can take a driver out of commission of a set amount of time.

References:

Ball, M.O., F. L. Lin. 1993. A reliability model applied to emergency service vehicle location. *Operations Research* 41:18-36.

Erkut, E., Ingolfsson, A., Budge, S. 2008. Maxium availability/reliability models for selecting ambulance station and vehicle location: a critique.

Further Notes: The R code required to run the simulation can be found on GitHub.