CPSC 320: Final Project

Donald Parker
Dylan Lafontsee
Ahmad Naqshbandi
April 27, 2023

Abstract

This is a simulation of a large baseball stadium modeled after the LA Dodgers stadium. The simulation models customers buying products from the vendors at the baseball game. Each vendor has their own line. The average wait time and total number of people served is being tracked from each vendor. The goal is to determine the number of vendors best suited to reduce time spent waiting in line and to serve a larger number of customers while keeping the number of vendors to a low but reasonable amount. Though the optimal value may vary depending on the specific wants of the user simulating.

1 Method

- The simulation assumes each vendor will be open for three hours.
- The only parameter for this simulation is the number of vendors.
- The number of people attending the event will be chosen by a gamma random variable with a mean of 45,000 attendees.
- Of those attendees based on historical data (from Dodger Stadium) we assume about 72% of them will be serviced by a vendor during the baseball game.
- The arrival rate of attendees to the vendors is determined by the number of people who show up to the game. The arrival rate will be an exponential random variable with a rate determined by the 72% of attendees who will want to go to a vendor divided by the total time of the simulation.
- Departure times or the time spent actually with a vendor and not in line will be determined by a gamma random variable based upon a mean service time of 90 seconds.
- Each Vendor exists as a struct in an array with their own variables being tracked vendors will have the following attributes.
  - int line: This will keep track of the number of attendees in line at the particular vendor.
  - double nextDeparture: This is where the actual time of the nextDeparture is stored for each vendor, set to INFINITY if there is no one in line.
  - int totalServed: This tracks the total number of attendees served for the vendor so far.
  - queue<double> ArrivalQ: This is a queue that stores arrival times of each person in line.This arrival queue is needed because when the vendor closes they will not help the people still in line.
- Vendors are assumed to be uniformly distributed throughout the stadium and so the vendor each attendee will go to is chosen uniformly. After the original uniformly chosen vendor the two vendors adjacent to this one are judged to see if it is better to switch. This encourages smaller lines and attempts to also simulate a small amount of human behavior. This process is explained in more detail in subsection 1.1.
- There is also a scoring system that quantifies the efficiency of the vendors (higher score the better.) This score helps track which number of vendors is better given how many people who wanted to be served, how many of those were served, and number of vendors. This is explained in more detail in subsection 1.2.

1.1 Choosing a Vendor

1. The nearest vendor is picked at random.
2. This vendor is compared with its two neighbors. Comparison is made based on the number of people in line and distance to be travelled to get to the neighboring vendors. A point scale is used, which incorporates both

distance and no of people in line with appropriate weightage. Following is the math and the logic behind the point scale:

- Average time between subsequent departures = 90 secs
  - On an average, each person in line adds 90 seconds to the wait time.
- Approximate average walking speed (across age groups) = 3 miles/hour = 3/3600 miles/sec
- Distance that can be traveled in 90 seconds at average walking speed = (3/3600)*90 miles = 0.075 miles = 396 feet
  - Therefore, 0.075 miles is going to increase the wait time the same amount as 1 person in line (both 90 secs)
- This implies, the distance equivalent of 1 person, in terms of wait time, is 0.075 miles = 396 feet
- This in turns gives us the following formula for calculating distance points, provided one person is equal to 1 point:
  - distancePoints = distance/396
- In the simulation, all the vendors are equidistant.
  - This implies, distance between vendors = (Circumference of the field)/(No. of vendors)
  - Average circumference of a Major League basketball field is 3770 feet.
  - This implies, distancePoints = (3770/No. of vendors)/396

3. Vendor with minimum points gets chosen by the customer.

1.2 Vendor Efficiency Score

1. Generate the number of people attending.
2. Find the subset of people who will use a vendor.
3. Initialize all starting variables.
4. Check if any departures have happened since the current time.
5. Process next arrival and update current time to the time of this arrival.
6. If the current time is less than or equal to closing time go to step 4.
7. If the current time is greater than closing time check all departures that happen before close.

2 Outcomes

Here are the results of specific numbers of vendors chosen:

1. Number of Vendors: 50
   Total Attendees: 41758
   Global Total Served: 5931
   Global AVG Time in System: 1 Hours 13 Minutes 14 Seconds
   Vendor Efficiency Score: 23.2782 (Higher is better)
2. Number of Vendors: 100
   Total Attendees: 41758
   Global Total Served: 11885
   Global AVG Time in System: 0 Hours 55 Minutes 19 Seconds
   Vendor Efficiency Score: 46.6466 (Higher is better)
3. Number of Vendors: 150
   Total Attendees: 41758

   Global Total Served: 17815
   Global AVG Time in System: 0 Hours 36 Minutes 54 Seconds
   Vendor Efficiency Score: 69.9208 (Higher is better)

4. Number of Vendors: 200
   Total Attendees: 41758
   Global Total Served: 23682
   Global AVG Time in System: 0 Hours 21 Minutes 4 Seconds
   Vendor Efficiency Score: 92.9478 (Higher is better)

5. Number of Vendors: 250
   Total Attendees: 41758
   Global Total Served: 28987
   Global AVG Time in System: 0 Hours 4 Minutes 28 Seconds
   Vendor Efficiency Score: 110.877 (Higher is better)

6. Number of Vendors: 300
   Total Attendees: 41758
   Global Total Served: 30166
   Global AVG Time in System: 0 Hours 2 Minutes 23 Seconds
   Vendor Efficiency Score: 100.336 (Higher is better)

7. Number of Vendors: 350
   Total Attendees: 41758
   Global Total Served: 29708
   Global AVG Time in System: 0 Hours 1 Minutes 56 Seconds
   Vendor Efficiency Score: 83.0026 (Higher is better)

8. Number of Vendors: 400
   Total Attendees: 41758
   Global Total Served: 29922
   Global AVG Time in System: 0 Hours 1 Minutes 46 Seconds
   Vendor Efficiency Score: 73.648 (Higher is better)

As we can see, once the number of vendors hits around 250, the wait times start decreasing at as lower rate and the number of people served total starts to even out. This is indicating that everyone who wants to be served is getting served and not waiting very long at these numbers. The vendor efficiency score is also reflecting these changes.

3 Conclusion / Recommendations

This simulation program could be customized for a variety of different multiple server simulation purposes. The variables could become something more generic and you can already vary the number of servers which makes this program very versatile. You could also run a loop checking up to any number of servers or vendors in the system and display that output for a more exact measurement. Products sold and a price point could be established in the simulation and those could be varied as well. Adding some of those extra features as well as a labor value to the number of servers such as dollars per hour and you could have a simulation that attempts to maximize profit.