# SIMULATION STUDY OF A STOCHASTIC VEHICLE ROUTING AND PRODUCTION SYSTEM

Submitted in partial fulfillment of the
requirements of the course

ISEN625 Simulation Methods & Applications

by

Ishank Gupta 832000398

Herin Savla 931002085

Project Guide: Dr. David Eckman

Wm Michael Barnes '64 Department of Industrial & Systems Engineering

Texas A&M University, College Station

# TABLE OF CONTENT

# Executive Summary

In a vehicle routing problem, the primary goal is to find optimal routes for the delivery vehicles to follow such that all the demand is met on time with minimum expenditure in cost. After obtaining historical shipment data of a company in Texas for 51 stores (customers), an attempt is made to simulate and analyze this system and thus, solve such a problem.

A stochastic vehicle routing system is coded and analyzed in Python and the operations preceding the delivery of goods such as production, packaging and procurement of goods from a production facility using road transport are studied in Simio. In the earlier, the last mile delivery process is optimized by calculating optimum delivery routes to minimize the expected transportation costs while ensuring minimum number of stockouts. In the latter, a simulation optimization study is carried out by varying parameters like pallet capacity and vehicle capacity to maximize and minimize the expected response characteristics like throughput of system and transportation cost respectively. The linking factor between these two studies is the input analysis done on the demand data obtained, on which the distribution fit serves as the rate at which parts are produced at the production facility.

The linearly interpolated CDF is used to generate random demand values based on historical data obtained. After finding GPS coordinates of all the stores using Google Maps, the location of a central depot is obtained using Center of Gravity method. It serves as the reference point in the routing system from which trucks set on their journey to deliver parts following fixed calculated routes. In the study carried out in Simio, the same central depot serves as the destination for all parts to be delivered to.

Savings Algorithm and Linear Programming approaches are used to determine optimal routes with more emphasis placed on the latter when creating experiments. After assigning varying priority levels to a certain percentage of stores with higher demand, the following two models result in the most optimum response values:

1. For Minimizing Expected Weekly Transportation Cost:
    - **Optimum Model**: Priority Level 2 assigned to stores with top 20% demand (Scenario 10)
    - **Objective Values**: Expected Weekly Cost = **8512.230 USD**; Expected No. of Stores = **3.558**
2. For Minimizing Expected No. of Stores with Unmet Demand:
    - **Optimum Model**: Priority Level 5 assigned to stores with top 30% demand (Scenario 15)
    - **Objective Values**: Expected Weekly Cost = **10167.633 USD**; Expected No. of Stores = **1.942**

For the simulation optimization study carried out in Simio, a pallet capacity of **500 parts** and vehicle capacity of **13 pallets** leads to maximum expected throughput of **43,550 parts** and transportation cost of **51,049.9 USD**.

# Introduction

Imagine yourself as the logistics manager of a UPS warehouse with one or more different types of delivery vehicles, in charge of satisfying the demand of multiple customers at different locations. Your task is to assign vehicles to a fixed set of locations (a route) to transport the goods at minimum cost. Problems of these types are known in mathematics as NP hard. The time required to solve such problems is an exponential function of the number of locations and is a popular area of research in the industry.

Almost every business decision is taken with the aim of maximizing some form of profit under the constraint of time, budget, space and other finite resources. The Vehicle Routing Problem (VRP) is all around us with companies like UPS facing a challenge in efficiently facilitating the flow of goods and services from one place to other while ensuring maximum possible savings in money and time spent.

In this study, we will be attempting to solve one type of VRP known as Capacitated Vehicle Routing Problem (CVRP) in which a fleet of identical vehicles with limited capacity, located at a central depot must be optimally routed to supply a set of customers with known demands. Due to certain limitations mentioned at the end of the report, this study is carried out in two sections using two different software: Simio and Python.

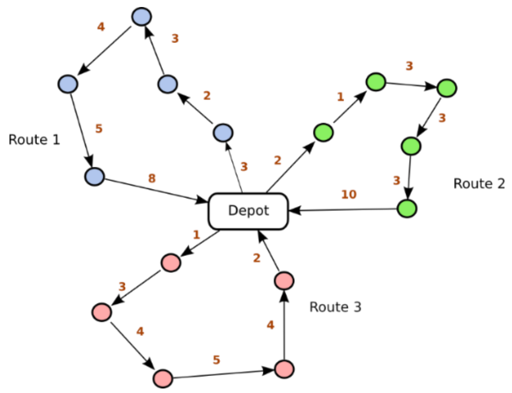


*Fig 1: Capacitated Vehicle Routing Problem (CVRP)*

In the first section, a production facility supplies parts to a central depot in Texas. There are various operations like inspection, palletization, wrapping, transportation and depalletization going on in this system. Here, simulation optimization is carried out to determine the most optimum value of parameters like vehicle capacity and pallet capacity resulting in the maximum expected value of throughput of system and minimum expected value of total transportation cost accrued.

In the second section, the central depot in Texas then delivers the parts to each store. Here, our goal is to optimize the last mile delivery process by calculating optimum delivery routes and minimizing the transportation costs while ensuring minimum number of stockouts.

# Methods

For this study, the methods used to solve the problem have been limited to exact methods (Linear Programming) and a Heuristic method (Wright Clark Savings Algorithm). To find the most optimum delivery route, various scenarios such as fixed routes, assigning stores with priority level based on historical demand, have been created.

## 2.1 Data

With the help of Dr. Joseph Guenes (Professor at ISEN Department, Texas A&M University), routing data is obtained for a company containing the following fields: shipment date, demand, store number, store address, name of store. The shipment date varies from 12-30-1996 to 08-17-1997 with orders from stores like Home Depot, Lowe's, etc.

## 2.2 Center of Gravity

Using the address of the stores GPS coordinates were found for each store using Google Maps and used as 'X' and 'Y' coordinates. The distance between these points is calculated using Haversine Formula. After that optimum central location is calculated based on the stores that are ordering that week and inter-store distances, using the Centre of Gravity method (iterative method). After finding the location based on each week's data, an average of all the central coordinates is taken to find the optimum centre location. We earlier observed that there is no seasonal pattern in the demand, and it is assumed to follow similar trend in the future. That's why it made more sense to consider both, weekly demand and location of stores to find the location of the central depot in our system.

## 2.3 Input Analysis

The dataset obtained is thoroughly cleaned by removing missing, negative demand values. A few stores based in Los Angeles are filtered out since the focus of study is only on stores based in Texas. To fit the best distribution to this data, demand is aggregated weekly. An assumption is made the store are being delivered the parts weekly. Various distributions like Normal, Gaussian, Triangular, Exponential distributions are fitted to the data. After plotting the histogram of weekly demand data, Q-Q plots and K-s tests are conducted for various distributions. In the K-s test, p-value of 0.9663 is obtained for the normal distribution. Thus, the weekly demand is approximated as a normal distribution. Normal, Gamma distributions were tried to fit onto demand of individual stores as well but no suitable fit is found. Finally, we built our models at individual store level using empirical data.

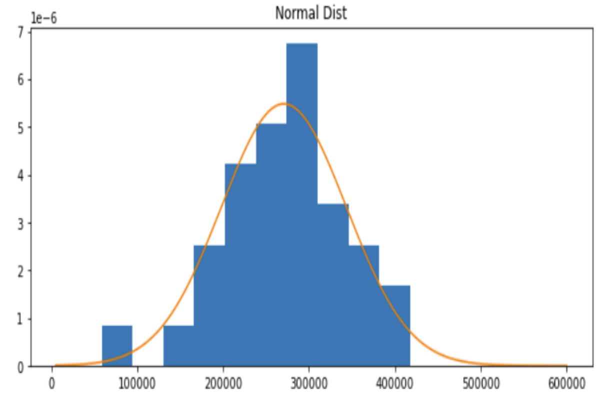


*Fig 2: Histogram for Weekly Demand Data*

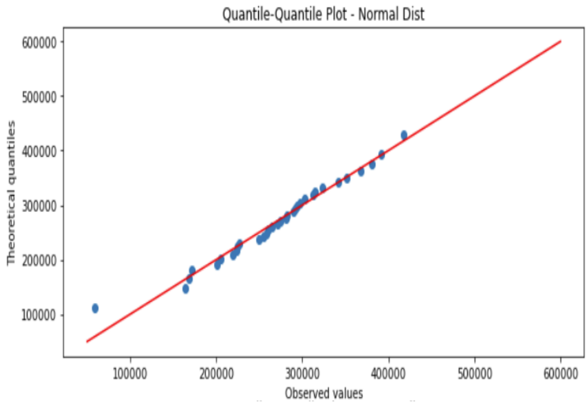


*Fig 3: Q-Q Plot for Normal Distribution*

## 2.4 Linearly Interpolated CDF

The linearly interpolated empirical CDF (The IECDF is a non-decreasing piece-wise linear function) is used for each of the stores to generate random demand instead of empirical CDF method to get more data points. To do this, the probability of a store ordering in a week from historical data is calculated using the number of times the store ordered across 33 weeks of data (the data is aggregated to count multiple orders by a store in a week as a single order only). So, assuming same probability of the stores ordering as during the historical period, demand is calculated for each store for 52 weeks of data using the linearly Interpolated Empirical CDF.

## 2.5 Key Decision & Assumptions

Now, to deal with stockouts, that is, cased where the truck with fixed capacity cannot meet the demand of all stores on a route, two decisions can be made:

1. Send multiple trucks as required to satisfy the unmet demand, upon the condition that it will need to travel the entire route
2. Paying a fixed shortage cost to each of the stores for which demand is unmet.

The following cost assumptions are made: Renting Cost of Truck = 75 USD; Fuel Cost per mile = 3 USD, Vehicle Capacity = 40,000 parts; Labour Cost = 20 USD/ hour; Unloading Time = 30 minutes; Operating Cost = 0.25 USD/mile; Extra Wage for Drivers for heavy lifting = 30 USD/hour; Truck Speed = 45 miles/hr

### 2.6.1 Vehicle Routing Algorithm (Savings Method)

This method is a heuristic method in which for each pair of nodes (stores), the savings in money spent if they are included in one route is calculated (since less distance is travelled) by

$$S_{i,j} = C_{0,i} + C_{0,j} - C_{i,j} \quad \text{where } C_{i,j} \text{ is the cost of travelling from node i to node j}$$

Based on this, each pairing of stores is ordered in a descending manner from highest to lowest savings. Then, routes are formed using this ranking while ensuring that number of stores being added to the route does not exceed the truck capacity. Although this method is very quick in giving results, they are very not optimum.

### 2.6.2 Vehicle Routing Algorithm (Linear Programming)

Gurobi solver in python is used to solve the routing problem. Fixed routes are determined based on inter-store distance. The objective function and constraints used to solve this linear programming problem is as follows:

**Objective Function**: $Minimize \sum C_{i,j} * X_{i,j}$

**Constraints:**

$\sum X_{i,j} = 1$ (this ensures only one outgoing edge is selected for each node)

$\sum X_{j,i} = 1$ (this ensures only one incoming edge is selected for each node)

$q[i] \leq u[i]$ (this ensures that the flow into the node satisfies the demand at that node)

u[i] <= Q (this ensures that the flow cannot exceeds the vehicles capacity)

$X_{i,j} = \{0,1\}$ ($X_{i,j}$ is a binary variable with 1 means vehicle travels from node i to j; 0 means otherwise)

$If\ X_{i,j} = 1\ \ then\ \ u_i + q_j = u_j$ (if a route exist between two nodes, then capacity accumulated at the 2$^{nd}$ node is sum of capacity accumulated at first 1$^{st}$ node and demand at the 2$^{nd}$ node)

### 2.7 Experiments (Python)

The first experiment is based on Trace-driven simulation, that is, finding optimum routes for historical demand data. The second is based on savings algorithm and the third is is based on randomly generated demand data. Upon observing a very high number of stores being unsatisfied, priority levels or weights are assigned to stores. Thus, 6 more experiments with multiple scenarios are created where in each scenario, the maximum number of stores allowed in a route is changed. Out of the 6, the first 3 are where the top 20% stores by demand (based on historical data) are assigned a higher priority/weight of 2,3 and 5. The next 3 are where the top 30% stores by demand are assigned weights of 2,3 and 5. This means that if a store is assigned a weight of 5 then only one store with higher demand will be added to the route instead of 5 unprioritized stores. The aim of this is to reduce stockouts, because with fewer stores in routes, both the operating cost and number of stores not being satisfied will reduce. Finally, for each scenario in each experiment expected weekly cost and expected number of stores left unsatisfied is calculated. Then for each experiment, the optimum route is determined by considering the trade-off between cost and number of stores being missed out.

## 2.8 Simio

To study the effect of varying parameters like pallet capacity and vehicle capacity on response characteristics like throughput of system and the total transportation cost, a simulation model is built using the academic edition of the software Simio 14 to replicate, optimize and derive insights about the system. Response characteristics mentioned above are created and then maximized and minimized respectively using the OptQuest add-in for Simio.

Firstly, as per the demand distribution obtained during Input Analysis, parts (Model Entity) are produced at the production facility (Source) following a normal distribution with a mean and standard deviation of 27,076.84 and 7,278.47 respectively. To stay within our computational limitations, one part is assumed to weigh 10 lbs. Connectors transport these parts through two rounds of inspection processes. After the final inspection process, about 1% of the parts are discarded due to defects, 4% parts are sent back to the first inspection station (Server) for re-inspection and the remaining 95% are sent to the palletizing station (Combiner).

Palletization process is assumed to follow an exponential distribution with a mean of 2 minutes. The pallet capacity is set as a parameter whose optimum value will be obtained during the experiment. Now, the palletizer receives another model entity, that is, pallets. The facility has a fixed number of pallets, that is, 80 which circulates in the system. Then, the units are palletized if a pallet and units equivalent to the pallet capacity is available at that point in time during the simulation run and sent to the wrapping station (Server). It takes approximately 1 minute to wrap a pallet. The wrapped pallets are then transported to the depalletizing station (Separator) of the Home Depot (Sink) which is approximately 100 miles away using a truck travelling at a speed of 45 mph. Once the parts are depalletized following an exponential distribution with a mean of 2 minutes, the empty pallets are transported back to palletizing station of the production facility using an identical vehicle. Both trucks are modelled such that they do not leave their station until their capacity is reached. And after delivering the goods, they immediately start the return journey back to pick up more goods. Loading and unloading of entities onto and from the truck takes an average of 2 minutes per pallet. The vehicle capacity is set as a parameter with varying levels of input, out of which the most optimum value will, once again, be determined during the experiment. The simulation is run for 2 weeks.

Various cost components associated with operating a truck are defined: operating cost of 25 USD/hr, cost of loading and transporting of 10 USD/pallet, a one-time usage cost of 250 USD, idle cost rate of 3 USD/hr and non-transport usage cost of 15 USD/hr. Almost all the assumptions made while defining the fixed parameters above have a reasonable reference point which can be found in the References section of the report.

# Analysis

In simio, the experiment created and the following response characteristics are defined: Throughput at the Central Depot and the transportation costs associated with both the Trucks. Pallet Capacity is varied through five levels of inputs: 100, 200, 300, 400 and 500 parts; Vehicle Capacity is varied through four levels of inputs: 13, 26, 39 and 52 pallets. Throughput and Cost have been given equal weightage in this multi-objective optimization problem but that can be varied as per the preference of the user.

The minimum and maximum number of replications are set as 7 and 20 respectively. The maximum number of scenarios are limited to 50 and a 95% level of accuracy is used when statistically comparing one response value to another. After a run time of 3646.8 seconds, the estimated optimal values turn out to be: throughput of 42,714.3 parts and transportation cost of 51,135.8 USD, at a vehicle capacity of 13 pallets and pallet capacity of 500 parts.

| Scenario | | | Replications | | Controls | | Responses | |
|---|---|---|---|---|---|---|---|---|
| ■ | Name | Status | Required | Completed | PalletCapacity | Vehicle_Capacity | Transportation_Cost (USD) | Throughput |
| ☑ | 006 | Compl... | 7 | 7 of 7 | 500 | 13 | 51135.8 | 42714.3 |
| ☐ | 011 | Compl... | 7 | 7 of 7 | 400 | 13 | 52963 | 41600 |
| ☐ | 009 | Compl... | 7 | 7 of 7 | 300 | 13 | 57345.6 | 42342.9 |
| ☐ | 012 | Compl... | 7 | 7 of 7 | 200 | 13 | 65300.8 | 43457.1 |
| ☐ | 002 | Compl... | 7 | 7 of 7 | 100 | 13 | 84885.1 | 43428.6 |
| ☐ | 020 | Compl... | 9 | 9 of 9 | 500 | 26 | 88919.6 | 37555.6 |
| ☐ | 008 | Compl... | 7 | 7 of 7 | 400 | 26 | 93995.2 | 41600 |
| ☐ | 018 | Compl... | 7 | 7 of 7 | 300 | 26 | 102003 | 37885.7 |
| ☐ | 004 | Compl... | 7 | 7 of 7 | 200 | 26 | 110627 | 41600 |
| ☐ | 015 | Compl... | 7 | 7 of 7 | 500 | 39 | 127224 | 39000 |
| ☐ | 014 | Compl... | 7 | 7 of 7 | 100 | 26 | 135882 | 42342.9 |
| ☐ | 005 | Compl... | 7 | 7 of 7 | 400 | 39 | 137720 | 31200 |
| ☐ | 001 | Compl... | 7 | 7 of 7 | 300 | 39 | 143846 | 35100 |
| ☐ | 017 | Compl... | 7 | 7 of 7 | 200 | 39 | 151085 | 37885.7 |
| ☐ | 003 | Compl... | 7 | 7 of 7 | 500 | 52 | 160415 | 26000 |
| ☐ | 013 | Compl... | 16 | 16 of 16 | 400 | 52 | 177195 | 39000 |
| ☐ | 019 | Compl... | 7 | 7 of 7 | 300 | 52 | 185801 | 31200 |
| ☐ | 016 | Compl... | 7 | 7 of 7 | 100 | 39 | 186692 | 42900 |
| ☐ | 010 | Compl... | 7 | 7 of 7 | 200 | 52 | 194547 | 41600 |
| ☐ | 007 | Compl... | 7 | 7 of 7 | 100 | 52 | 233098 | 41600 |

*Fig 4: Results of Simulation in Simio*

To eliminate optimization bias, the optimal solution of vehicle capacity = 13 pallets and pallet capacity = 500 parts is simulated in a different experiment with the number of replications set as 10. The estimated optimal values turn out to be: throughput of **43,550 parts** and transportation cost of **51,049.9 USD**. The 95% confidence interval for throughput and cost is [39000, 45500] and [49710.61, 51490.52]

Now, for the simulation study conducted in Python, to compare between the best models, Common Random Numbers (CRN) are used because each week's data is independent of each other. The best model out of the 6 experiments where priority level was kept as 2, 3, and 5 for top 20% and top 30% of data is determined. Each experiment has multiple scenarios.

To find the best scenario for each experiment, we analyzed various scenarios by plotting the graph of cost and number of stores being stocked out. Assuming the company's threshold of maximum number of unsatisfied stores allowed to be less than 10, the best models were obtained.

| **Scenario (LP)** | **Best Model (Scenario No, Max Stores in a Route)** | **Expected Weekly Cost (USD)** | **Expected No. of stores being Unsatisfied** |
|---|---|---|---|
| Trace Driven Simulation), (No Priority) | (10, 10) | 8457.10 | 7.697 |
| Random Generated Data, (No Priority) | (10, 10) | 8233.676 | 3.615 |
| **Priority Assigned - 2, Top 20% Stores** | **(11, 11)** | **8512.2305** | **3.558** |
| Priority Assigned - 3, Top 20% Stores | (12, 12) | 9116.366 | 3.019 |
| Priority Assigned - 5, Top 20% Stores | (15, 15) | 9239.820 | 6.038 |
| Priority Assigned - 2, Top 30% Stores | (13, 13) | 8492.192 | 3.730 |
| Priority Assigned - 3, Top 30% Stores | (13, 13) | 9196.248 | 3.615 |
| **Priority Assigned - 5, Top 30% Stores** | **(15, 15)** | **10167.633** | **1.942** |
| Saving's Method (No Priority or LP) | (6, 6) | 36450.758 | 6.667 |

*Table 1: Results of Simulation in Python*

There are lot of uncontrollable factors in this study such as price of fuel/mile, average salary of workers, sudden spikes in demand, roadblocks affecting routes and many more. How will the best model react to these factors? To answer that, the effect of sudden fuel price hikes and spikes in demand on our optimum models is studied.

| **Uncontrollable factors** | **Cost (USD)** | | | | **Stores with Unmet Demand** | | | |
|---|---|---|---|---|---|---|---|---|
| | **Model with Priority Level 2, Top 20% Stores** | | **Model with Priority Level 5, Top 30% Stores** | | **Model with Priority Level 2, Top 20% Stores** | | **Model with Priority Level 5, Top 30% Stores** | |
| | **Old** | **New** | **Old** | **New** | **Old** | **New** | **Old** | **New** |
| Demand Spikes | 8512.231 | 10829.173 | 10167.63 | 10286.371 | 3.558 | 4.942 | 1.942 | 4.211 |
| Fuel Price Hike | 8512.231 | 12672.329 | 10167.63 | 15330.709 | 3.558 | 6.423 | 1.942 | 4.211 |

*Table 2: Effect of Uncontrollable Factors on Optimum Models*

# Conclusions

For the experiments where top 20% stores are assigned a priority level, if saving on cost is the company's focus, the model with priority level 2 should be used. If reducing the number of stores with unmet demand is the objective, then model with Priority 3 is the most preferred. Similarly, for the 3 experiments where priority level was changed as 2, 3, and 5 for top 30% of data, if saving on cost is the company's focus, then model with priority level 3 is most preferred. If reducing the number of stores with unmet demand is the objective, then the model with priority level 5 is most preferred. Finally using CRNs again across all these optimum models we concluded that:

- Most Cost-Effective Model: Priority Level 2 for top 20% stores (Scenario 10)
- Least Stock-outs Model: Priority Level 5 for top 30% of stores (Scenario 15)

Observations were made that assigning a greater number of stores, a higher priority will result in a smaller number of stores being unsatisfied because there are lesser number of stores in the route itself but correspondingly cost will also increase as number of routes increase, so, there is a trade-off. Thus, there is a need to assign priority to it find optimum number of stores. Also, as increase the Priority/weight to the stores generally the cost will increase and number of stockouts tend to decrease. So, finding right Priority level for each store is also important which can be done by assigning each store a priority by how much larger is it compared to store with smallest demand.

### 5.1 Future Study

For the system studied in Simio, the model can be further developed by including the manufacturing processes in creating the part and then simulate the movement of parts through the system to each and every final destination of 51 stores. In doing so, a better understanding and control of the system will be obtained. Various additional factors like Inventory Level, Reorder Point, Production Capacity, Multiple Trucks, can also be factored in.

Currently, mixed linear programming method takes up a lot of computational time and power to solve due to which each scenario was limited to a solving time of 90 seconds leading to sub-optimum results. Therefore, there should be more focus towards using more advanced meta heuristic algorithms that will give even more optimum results.

# Limitations of Study

1. The academic edition of Simio 14 (Version 14.221.2.283, 32 bit) has certain computational and memory limitations which we realized after running an experiment with a model where the source produces parts with a normal distribution with $\mu = 270768.445$ and $\sigma = 72784.715$ as derived from the input analysis. The number of entities in the system and the simultaneous processes going on during a simulation run consume too much memory and it can destabilise the application. To accommodate this limitation, 1 part = 10 lbs is assumed and the parts per arrival at the source are modelled as a normal distribution with $\mu = 27076.844$ and $\sigma = 7278.471$
2. A vehicle routing system consisting of 51 stores with varying weekly demand which dynamically affects the route every week is complex to model on Simio. Multiple attempts were made to enter the weekly demand data using data tables, state variables, work schedules or using process logic but they did not lead to the desired system of study. Moreover, the practical problem of feeding in demand data at a sink or a server for it to consume made it even more difficult to model. Finally, the decision was made to split the study in two parts as mentioned in the Introduction section
3. For the routing study conducted in Python, linear programming is used where we limited each scenario out of a total of 20 scenarios to a solving time of 90 seconds each. This causes variations in the results every time the simulation is run. Wright-Clark Savings Algorithm is also used which is a heuristic method which does not guarantee an optimum solution despite its quickness. Other robust, meta-heuristic methods like Genetic Algorithms, Tabu Search and Simulated Annealing have not been explored in this study.
4. In the real-world, companies like Amazon have fixed but at the same time flexible delivery routes, that is, they can be readjusted based on varying incoming demand. Also, to satisfy sudden spikes in demand they can utilize vehicles with higher capacities. In our study, the overage vehicle being send out to satisfy unmet demand travels the entire route instead of visiting the particular stores.
5. In this study, to satisfy unmet demand, either extra trucks are sent out to satisfy unmet demand or a stock-out fee is paid to each of the missed-out stores (to retain the customer). However, in a real-world scenario, this is an optimization problem with a mixed solution, that is, some stores are paid while other stores have their demand met by sending out additional vehicles.
6. This study fixes routes based on distance and in a particular method, higher priority is assigned to stores with greater historical demand. However, the number of stockouts occurring in the system could have been further decreased by adding an additional constraint that the sum of average historical demand of stores being added to a route is less than the vehicle capacity.

# References

[1] Gurobi & Python. Capacitated Vehicle Routing Problem - YouTube

[2] Combiner and Separator Objects, Batch and Unbatch Steps - YouTube

[3] Commercial Truck Driver Hourly Pay in Dallas, Texas | PayScale

[4] Calculating The Real Delivery Costs In Your Rental Business: Trucks - RW Elephant

[5] Houston, TX to Weimar Independent School District, Texas - Google Maps

[6] Standard Pallet Sizes & Dimensions | Freightquote | Freightquote

[7] Steven Nahmias, Tava Lennon Olsen (2005). Production and Operation Analytics. Waveland Press

[8] Jeffery S.Smith, David T. Sturrock (2021). Simio and Simulation - Modelling, Analysis, Applications

# Appendix

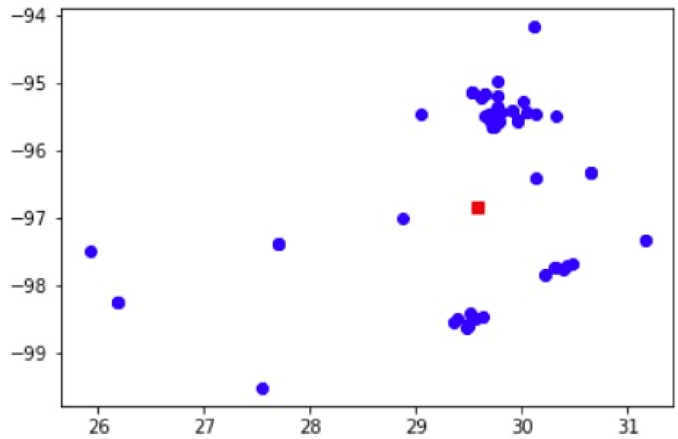


*Fig 5: X & Y Coordinates of 51 Stores and Central Depot*

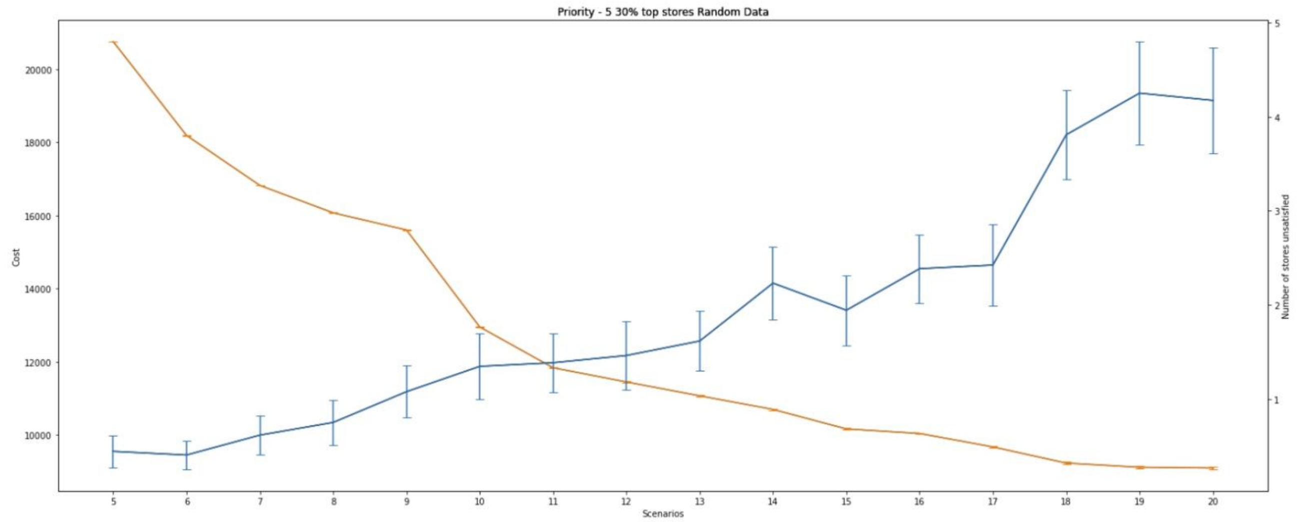


*Fig 6: Priority 5 to Top 30% of Stores (Scenario 15)*

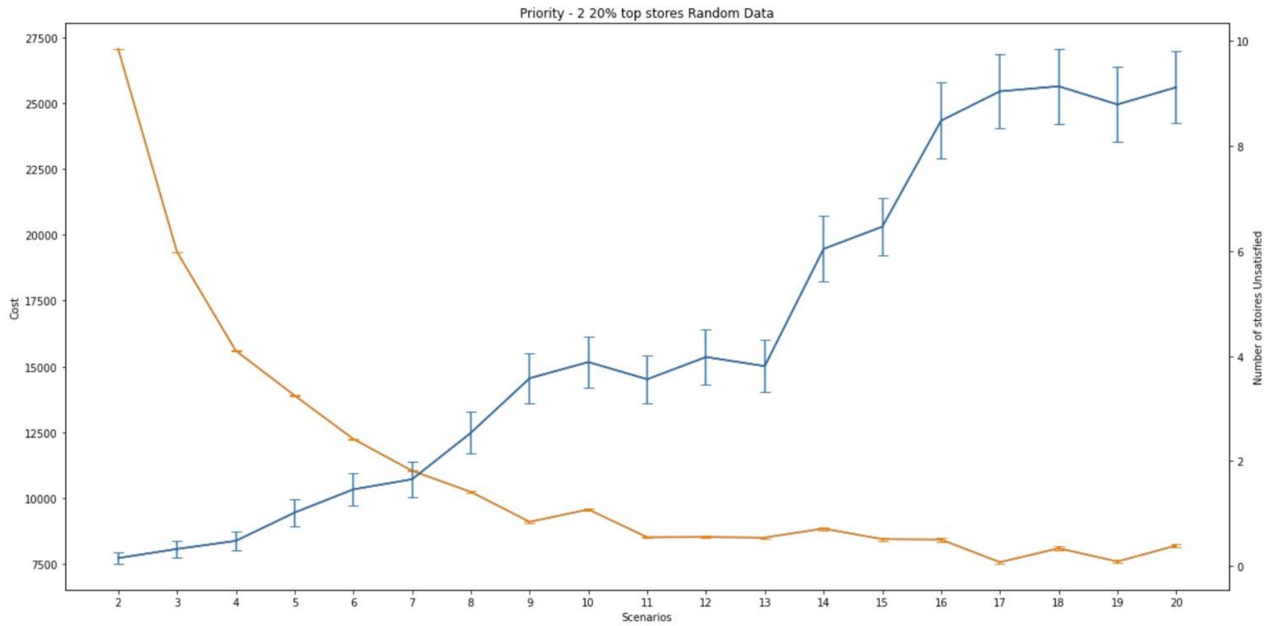


*Fig 7: Priority 2 to Top 20% of Stores (Scenario 11)*

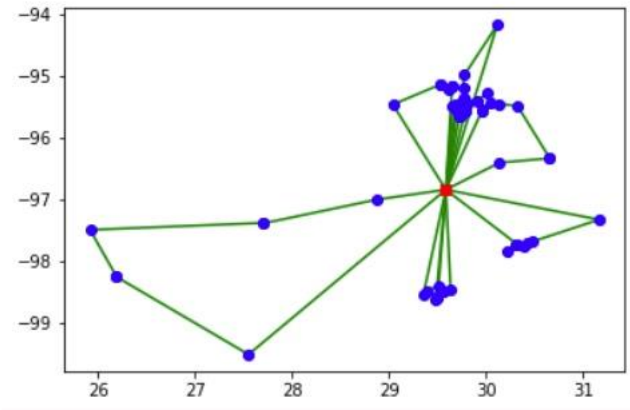

*Fig 8: Final Routes for Priority 5 to Top 30% of Stores (Scenario 15)*

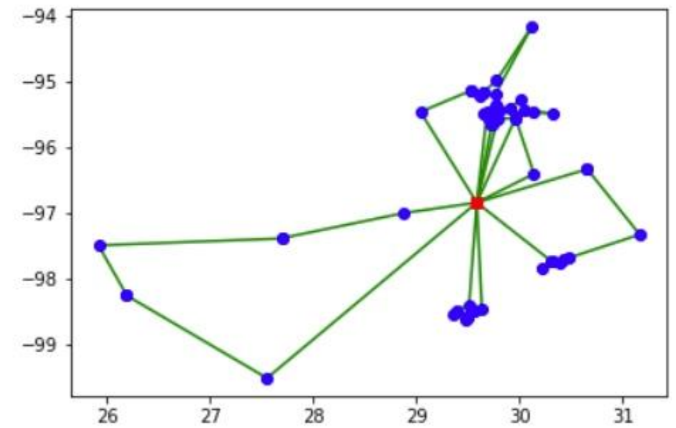

*Fig 9: Final Routes for Priority 2 to Top 20% of Stores (Scenario 11)*