# Project 2 – Final Report

## Supply Chain Network Simulation

Rodrigo Borela, Sangy Hanumasagar, Fangzhou Liu, Nimisha Roy

## Project Statement and Goals

A manufacturing company produces and sells a consumer product and wishes to utilize its production and supply chain resources to maximize profits. A time stepped simulation is proposed to investigate the supply-demand relations over the span of one year in the system, based on the interplay of constraints associated with the manufacturing plants, warehouses, stores, and consumers, which are all serviced by the delivery services. The revenue in the system is to be generated from sales of the product and the cost is incurred from the delivery services. The aim of this study is to maximize the differences of the two quantities (i.e. profit) in the system, which are collectively dependent on size of stocks maintained in each store, frequency of delivery trips and the choice of route for each delivery.

The problem statement is originated from the 2017-2018 math competition organized by the French Federation of Mathematical Games (http://www.ffjm.org/index.php?option=com_content&task=view&id=51&Itemid=6). The product sales information utilized in the simulation is based on data supplied by the organization. Therefore, by utilizing the provided sale records and associated costs in maintaining stocks at the stores and warehouses, we wish to provide an optimal strategy for the company to operate in the future years.

## Project Task Demarcation

Rodrigo Borela: Data visualization + Software Development + V&V
Sangy Hanumasagar: Software Development + Preliminary Results + Parametric Studies
Fangzhou Liu: Software Development + Parametric Studies + Report Organization
Nimisha Roy: Software Development + Data Processing + Parametric Studies

## Project Description

A manufacturing industry contains 2 plants, 2 warehouses, and 20 stores, manufacturing, stocking, and selling the same product (Product 1), respectively. The geolocations of the entities involved are shown in Figure 1 and important features specific to each entity is described below.

- Supply trucks operate between the plants and warehouses and stores and are assumed to be readily available as the need arises.
- Each grid shown in Figure 1 represent a 1 km by 1 km distance. The grid can only be traversed along the grid lines, i.e. in an orthogonal fashion.
- Each km of distance traversed by a delivery truck shall cost $1. There are no other incurred costs on the trips. Each truck can carry a maximum of 50 products on any given trip.
- The plants may produce the product with no limitations on the rate of production. Each plant supplies to one warehouse as depicted in Figure 1.
- Warehouses have a storage capacity of 650 units each, and unless low on stock themselves, can supply products to the stores with no delay. The warehouses are free to supply to any of the twenty stores, subject to delivery costs explained above.
- The stocking capacity of each store varies as per Table 1, but considering it is much lower than that of the warehouse, the delivery costs between warehouses and stores formulate the major component of the overall cost.
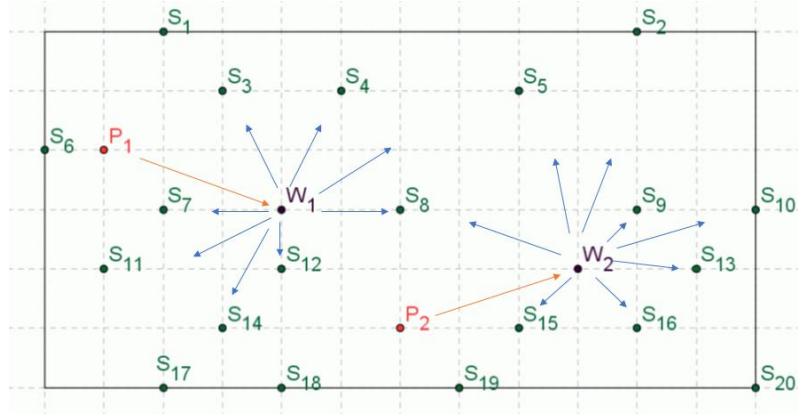- The daily sale records of each store from 2015 is used to model the consumer demand.

Figure 1. Geolocations of the plants (P), warehouses (W), and stores (S). Each horizontal and vertical grid line represents 1 km. A Manhattan distance is enforced in determining the shortest distance between organizations.

Table 1. Storage capacity of each store

| S | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|
| C | 15 | 15 | 20 | 20 | 15 | 20 | 30 | 3 | 35 | 25 | 30 | 30 | 30 | 35 | 30 | 40 | 25 | 20 | 15 | 20 |

*Note C represents capacity*

Each shop, warehouse, and plant are open six days a week for the whole year. A customer should find, in each store, in each day, at least one product is available (lower bound of the demand). The goal is that, at any given point, every single demand from any number of customers is satisfied at each store (upper bound of the demand). The price of the product is $100, and the delivery cost is $1 per km (a constant value throughout the simulation). To better simulate a real-life scenario, we also introduce a 'base' fee for hiring the truck agency, which is a fixed cost of $100 per day, regardless whether a delivery service incurred that day.

**Conceptual Model**

The SUI is a combination of the entities previously mentioned, over the course of 12 months. A detailed SUI block diagram illustrating the entities and their attributes is shown in Figure 2.
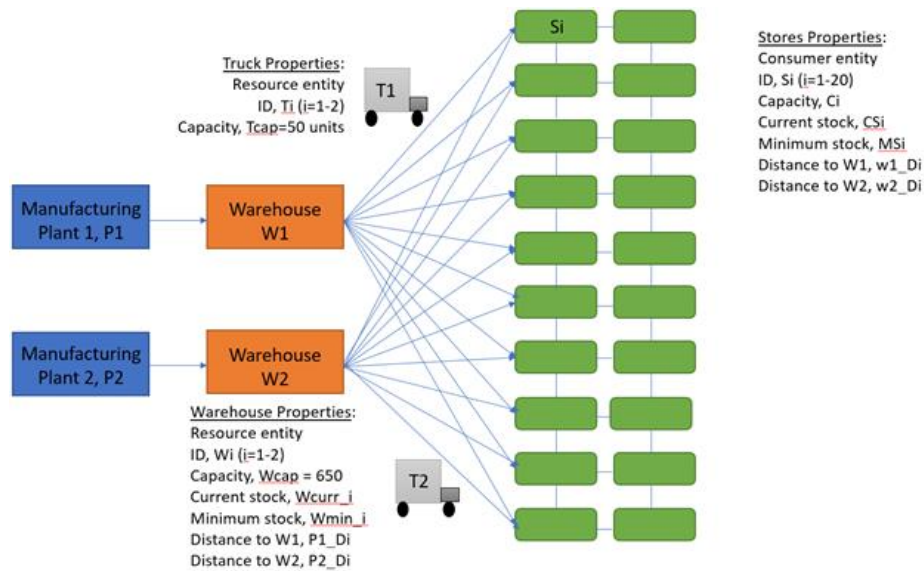


Figure 2. Network topology of the supply chains describing the structure of the SUI

More specifically, the entities and their attributes in the SUI are detailed in Table 2 below:

Table 2. Model entities

| Entity | Type | Attributes |
|---|---|---|
| Consumer | Consumer entity | Numbers |
| Store | Consumer entity | Id, current stock, minimum stock, storage capacity, location, distance to w1, distance to w2, revenue, loss |
| Truck | Resource entity | Id, capacity, products carried, route, schedule, cost, maximum travel distance per trip |
| Warehouse | Resource entity | Id, current stock, storage capacity, location, distance to stores, distance to plants |
| Plant | Resource entity | Id, location, production rate (no. of products/day), distance to warehouses |

**Store**

The behavior of the SUI is an iteration over all days in the year. The following sequence of events is repeated for every day of the year (except Sundays) which is also graphically depicted in
Figure 3.

- Each store makes sales as per a probability distribution of expected sales. This distribution is modeled individually for each store and explained in detail in the following section.
- Once the stock in a given store drops below a pre-determined minimum stock value, it triggers a low-stock warning. Similarly, if the stock falls to zero, it triggers a zero-stock warning.
- These warnings are communicated to the warehouses. Stores with zero-stock warnings are prioritized for restocking by delivery trucks.
- The nearest warehouse with sufficient stock will dispatch a truck with 50 products, which is the maximum capacity of each truck.
- As per the constraints applied on the simulation study, the truck supplies the entire shipment to the store with greatest need or distributes the shipment to multiple stores which all need restocking.
- Once, a warehouse runs low on stock, the nearest plant dispatches multiple shipments to the warehouse to restock it to its capacity.

```
for  day in 1 to N

    SUPPLY-CHAIN

    restock warehouses
        for  truck in fleet
            check warehouses stocks
            schedule plant-warehouse deliveries
            update warehouses stocks
            update delivery variables (costs, mileage, #)

    restock stores
        for  truck in fleet
            check low-stock store list
            schedule warehouse-stores delivery
                optimal delivery route selection
            update stores stocks & low-stock list
            update delivery variables (costs, mileage, #)
```

```
STORE SALES

    for store in franchise
        get demand from data model
        balance sales based on stock
            calculate effective and missed sales
            update store stock
        inventory check
            include store in low-stock list if necessary
        update sales, revenue, opportunity cost
```
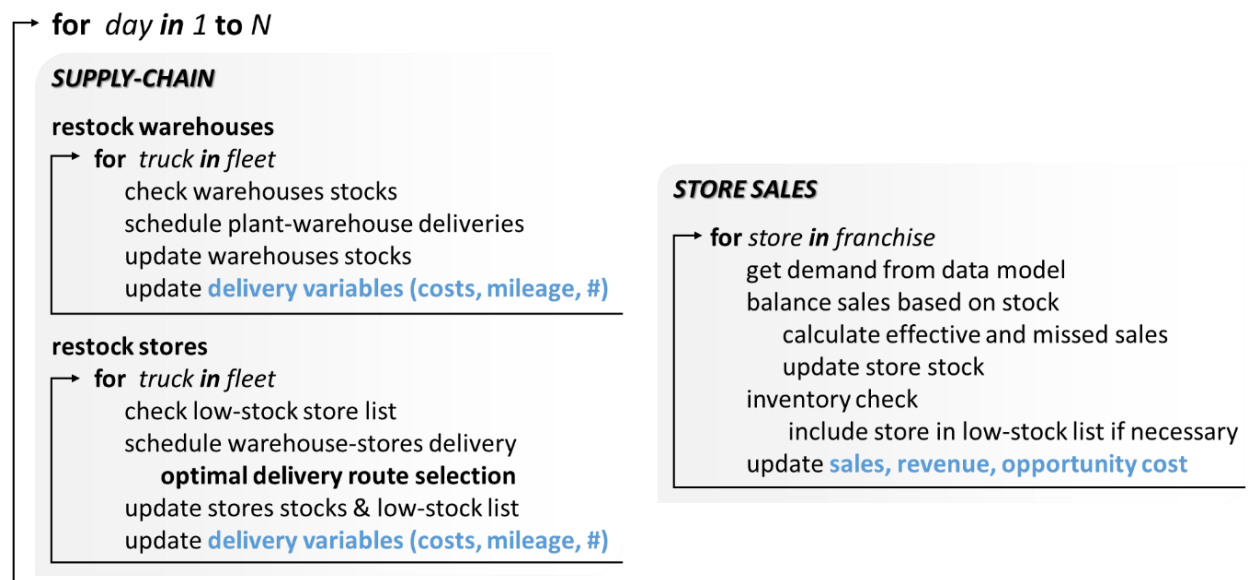
Figure 3. Flowchart demonstrating the behavior of SUI

At every iteration of the loop, the sales made, and delivery costs incurred are tracked to estimate profits. The optimal strategy in order to maximize profits will be evaluated and defined based on the following aspects.

- Minimum stock in each store: By defining a high value of minimum stock in each store versus a lower value, the focus is varying from a maximizing-sales approach to a minimizing-costs approach.
- Any lost sale or sale demand that is not met is considered to be bad for the reputation of the company. Thus, the number of such occurrences are tracked as an additional metric of optimal performance along with profits earned.
- Extent of restocking: When delivery is made to a store which is running low on stock, there is a decision to be made regarding partial versus full restocking. A full restocking would probably be appropriate for stores located far from the warehouse to minimize future trips, while stores closer can be visited frequently en-route to other stores, and thereby can do with partial restocking.
- Delivery route optimization: Each trip can be considered a trip to a single store and back to the warehouse (single-point delivery). However, considering that the stock capacity of each store is lower than the capacity of the truck, this can be solved to optimally serve multiple stores each in need of or having room for additional stock (multi-point delivery).

**Simulation Software**

Overall Structure

The simulation was implemented using Python 3.6, comprising of three main modules:

- **Topology**: This module defines the entities of the simulation, viz. plants, warehouses, stores and trucks. It converts the Cartesian grid present in the problem statement into a network, which is used for optimization of the delivery routes.
- **Application**: This module comprises all entities implemented as class objects, the event handlers which are coded as functions, data models which are used to generate the demand of the stores and the main function which runs the simulation through a period of 1 year and computes the total finances at the end of the simulation.
- **Visualization**: This module contains all functions used to visualize the data obtained from network analysis of the SUI. The visualization is carried out in terms of evolution of the total sales distribution, optimization of the routes based on certain attributes of the entities, etc.

Pseudo Code

1. State variables/parameters (also implemented as class objects):

day_revenue: total daily revenue collected from all the stores
day_loss: total loss incurred per day throughout all stores
truck.schedule: schedule of the truck for delivery for the particular day
entitiy.current_stock: the current stock of the entities

2. Topology:

class Store:
_init_()
Initializes the attributes of the store entity, viz. store ID, x coordinate and y coordinate of its location, capacity, minimum stock, refill percentage and expected sale

class warehouse:
_init_()

Initializes the attributes of the warehouse entity, viz. store ID, x coordinate and y coordinate of its location, capacity, minimum stock, refill percentage and parent plant

class plant:
_init_()
Initializes the attributes of the plant entity, viz. store ID, x coordinate and y coordinate of its location, capacity and production rate

class truck:
_init_()
Initializes the attributes of the warehouse entity, viz. store ID, capacity, actual load, route taken, schedule time, cost and maximum stores spanned

Uses network X to create a graph and adds nodes to the graph as entities, plant (P), warehouse (W), Truck (T) and Stores (S) and adds edges to the graph as paths from plants to warehouses and warehouses to stores.

Prints the graph information

3. Application:

generate_demand ()
Generates and stores expected demand for each store based on the modelled exponential probability distribution data

Class simulation:
_init_()
Initializes the finance variables (price per unit, total daily demand, daily revenue, daily loss, cumulative revenue, cumulative loss, total delivery cost, delivery cost per product, cumulative delivery cost, daily profit, cumulative profit), logistics variables and data structures (deliveries, low stock warning array, zero stock warning array) for the simulation

advance_time ()
Runs simulation for all the entities for 1 year, 6 days a week and computes the total finances at the end of the simulation

store_iterator ()
Computes the total daily revenue or loss of each store based on its demand and capacity

warehouse_check ()
if current warehouse stock < minimum stock
        restock the warehouse

balance_sales ()
if demand < current store stock
        revenue = demand* price per unit
        loss = 0
        update current store stock
else     revenue = store current stock* price per unit
        loss = (demand - store current stock)* price per unit
                store current stock = 0
update store revenue and loss

inventory_check ()

Checks if stock is low in any entity and updates the low stock warning array and zero stock warning array accordingly

restock_stores ()
Schedules the delivery of truck to stores from nearest warehouse with preference to stores with low stock or zero stock warning and updates the schedule of the truck entity

restock_warehouse ()
Schedules the delivery of truck to warehouse from nearest plant and updates the schedule of the truck entity

delivery_route ()
Finds optimal delivery route to service under-stocked stores

delivery ()
updates the stock of entities after dispatching of trucks

4. Visualization

Define products and paths
Load data
Generate total sales grid
Generate figure objects
Loop over the year and update the graph

Logic Flow

The simulation of the maintenance of stocks in the stores and warehouses in the given problem is implemented in the form of a continuous event simulation. The application module comprises eight event handlers and a main function which iterates throughout the span of a year to arrive at the total finances of the entities. The main function first calls the store_iterator () function which is an event handler that computes the revenue or loss across all stores based on the demand and maximum capacity of each store. The function also checks if the stock of stores and warehouses is below a minimum value and returns the daily revenue or loss across stores. The function also updates the state variables, viz. the daily revenue and daily loss of stores. The balance_sales () function computes the local revenue/loss of each store and passes it to the store_iterator () function to iterate over all the stores. The inventory_check () function checks if the stock is low for any given entity and if true, appends a low stock warning or zero stock warning signal to it so that the entity is prioritized over other entities while restocking. The warehouse_check () function checks if the stock of the warehouses has plummeted below the minimum threshold value and if true, calls for the restock_warehouse () function to restock the warehouses.

The restock_warehouse () function checks if a truck can leave from a plant and start stocking the warehouse in close proximity to it based on the interval between deliveries. The function computes the Manhattan distance between the plant and warehouse to and for and schedules the delivery via the truck by calling the delivery () function. It also updates the state variable, truck_schedule. After updating the daily revenue and daily cost for that day, the main function calls the restock_stores () function which checks and updates the schedule of the truck to deliver from each warehouse to the nearest stores based on the low stock and zero stock warning signals flagged off by the stores. The delivery of the products from the plant to the nearest warehouse or warehouse to the corresponding stores are carried out by finding the optimized route. This is done using the delivery_route () function which computes all the possible permutations among the routes and arrives at the shortest route according to the stored distances between the entities. The delivery () function updates the stock of entities along the route after the process of restocking is complete. After iterating the above process for 313 days (all days of the year except Sundays) for all stores, the main function computes the total year end finances in terms of the cumulative revenue, cumulative loss and cumulative profit.

<u>Optimal Delivery Routes</u>

To minimize costs associated with truck deliveries, an algorithm has been devised to generate and select the shortest routes for warehouse delivery to multiple stores. In the initialization of the model, given the maximum number of stores a truck can visit on a single trip, possible combinations of all the stores are obtained. For each combination, sequence of stores on potential routes are obtained by permuting the elements in the combination. Appending the warehouse to the beginning and end of the sequence, all possible routes given a subset of stores can be calculated, out of which only the shortest route is stored. This is exemplified in Figure 4, which illustrates possible permutations for a fixed combination set, in which only the rightmost sequence would be stored in the optimal routes array.

During the execution of the model, a delivery route assigned to a truck is done by first obtaining the combinations of stores in the low-stock list and consulting the corresponding optimal routes. The shortest of the optimal delivery routes for the combinations is then assigned to a truck for delivery. This procedure can be visualized on Figure 5, in which the optimal routes of different combinations of stores would be compared and the route in the center would be assigned to the next available truck for delivery.
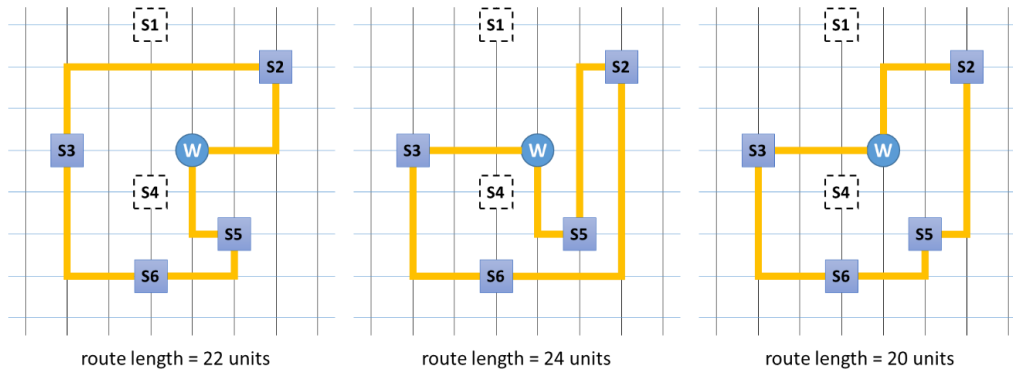


Figure 4. Optimal routes calculation example for a fixed subset of stores.
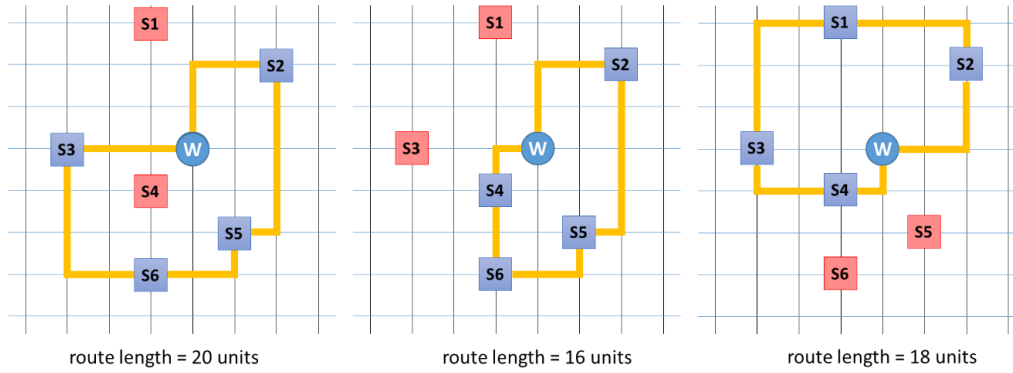


Figure 5. Delivery route selection

**Data Collection and Modelling**

The expected sales for each shop per day is derived from the data provided as a part of the problem statement of the competition. Based on the provided data, three theoretical probability distribution models, viz. normal, exponential and gamma distribution models, were used to formulate an appropriate distribution for the estimated annual sale of each store. Based on the analysis results from several chi-squared tests for the determination of goodness-of-fit of each parametrized theoretical distribution (Birta and Arbez, 2013; Ghosh and Bowden, 2004),

it was observed that the exponential probability distribution best fits the sales data. For visualization purposes, the actual data and the fitted exponential distribution curve superposed on the data for store 1 is shown in
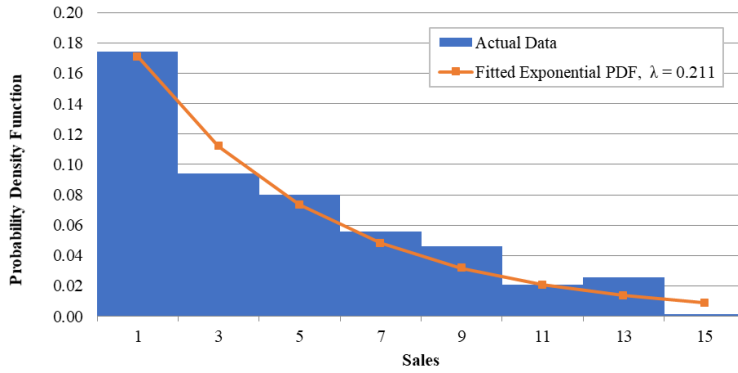


Figure 6. The exponential distribution has only a single parameter, λ, used for defining the distribution curve, the value of which is computed for each store, as represented in Figure 5. Average weekly sales throughout the year are presented in Figure 8. Additionally, to better comprehend the sales dynamics in terms of locations, an interactive algorithm was implemented that shows the evolution of the total number of sales in each store as the year progresses. A sample of three different days is presented in Figure 9. (Note: the code will be submitted as part of the initial software package later). It can be observed that the highest number of sales takes place at the stores close to the warehouses.
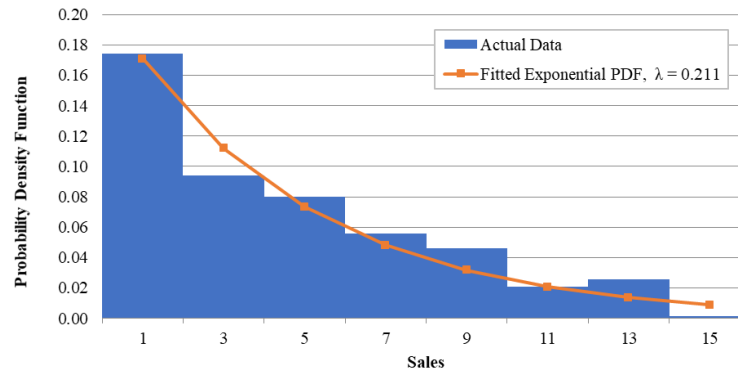


Figure 6. Original data and superposed exponential distribution curve for store S1
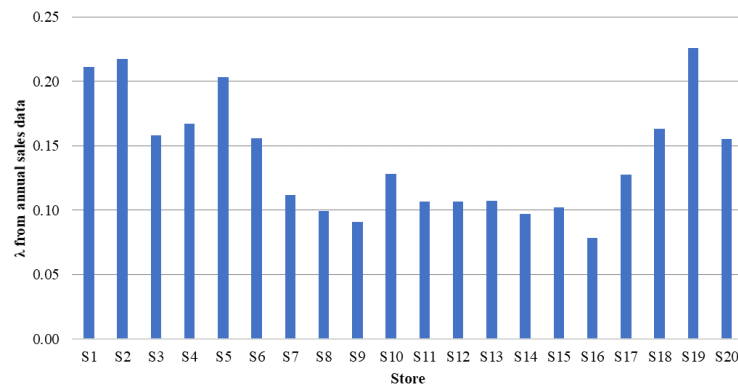


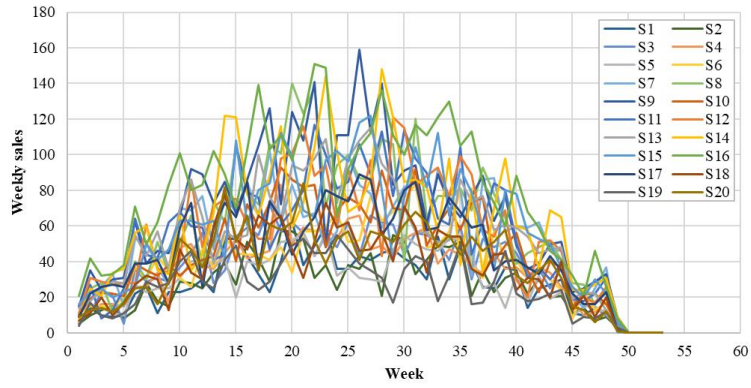Figure 7. Computed parametric value, λ of exponential probability distribution representing sales in each store
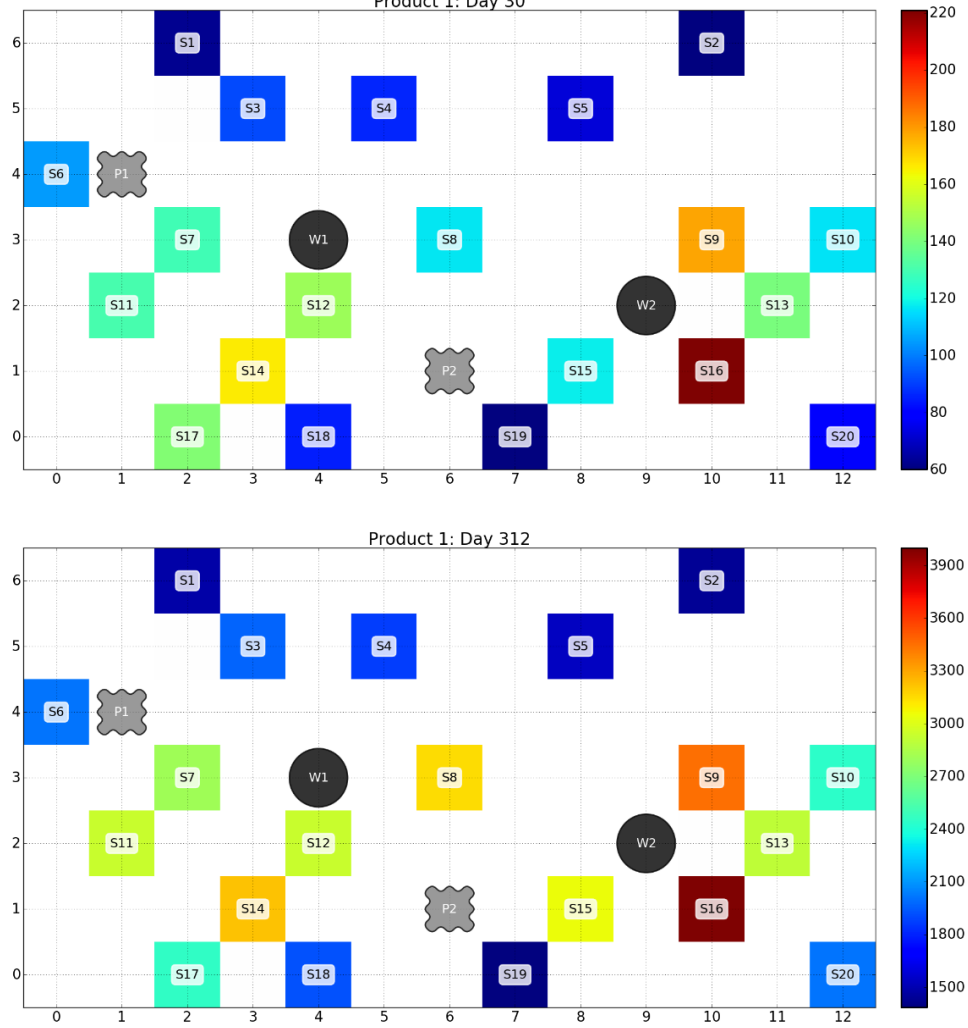
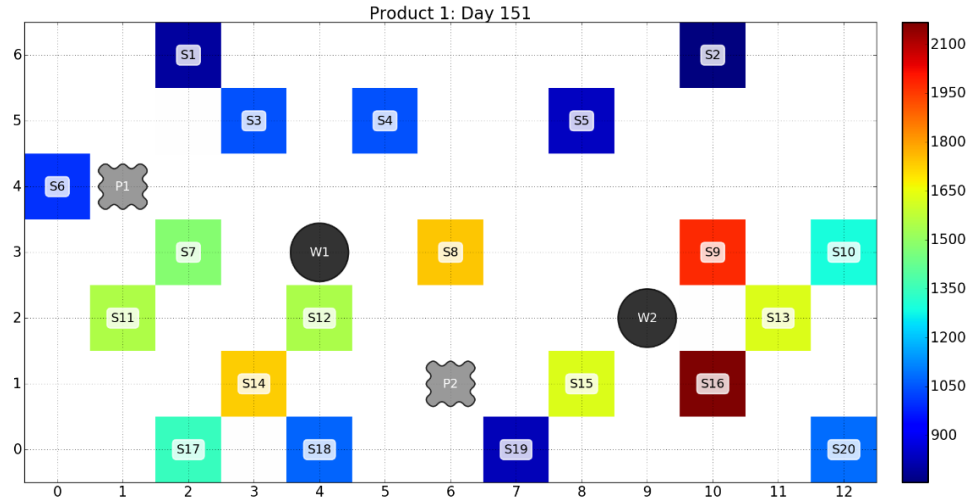Figure 8. Average weekly sales during the year for each store

Figure 9. Total sales distribution evolution in each store throughout the year. Colors correspond to the total number of sales.

**Verification and Validation**

Two sets of simulations are conducted to verify that the execution loop was operating adequately:

1. Constant demand of 10 products, daily deliveries to maximum of 5 stores;
2. Constant demand of 10 products, no deliveries for the full year;

The following figures demonstrate the correct execution of the simulation, in which constant demand and daily restocking results in linear cumulative sales, costs, profits and revenue.
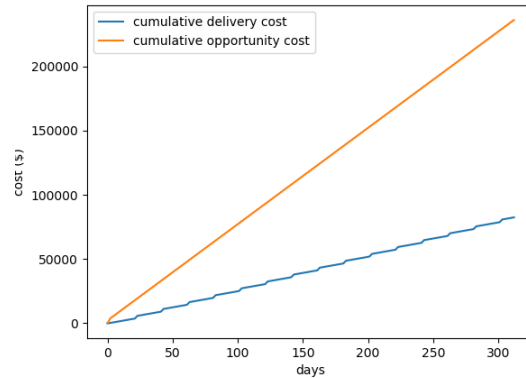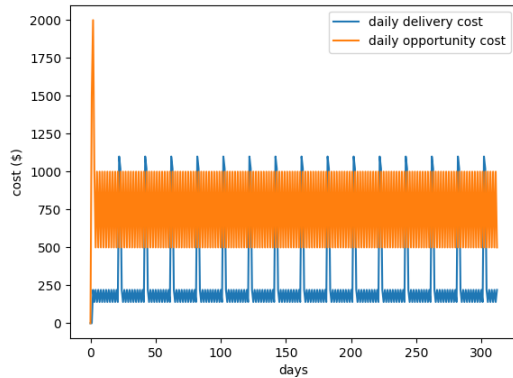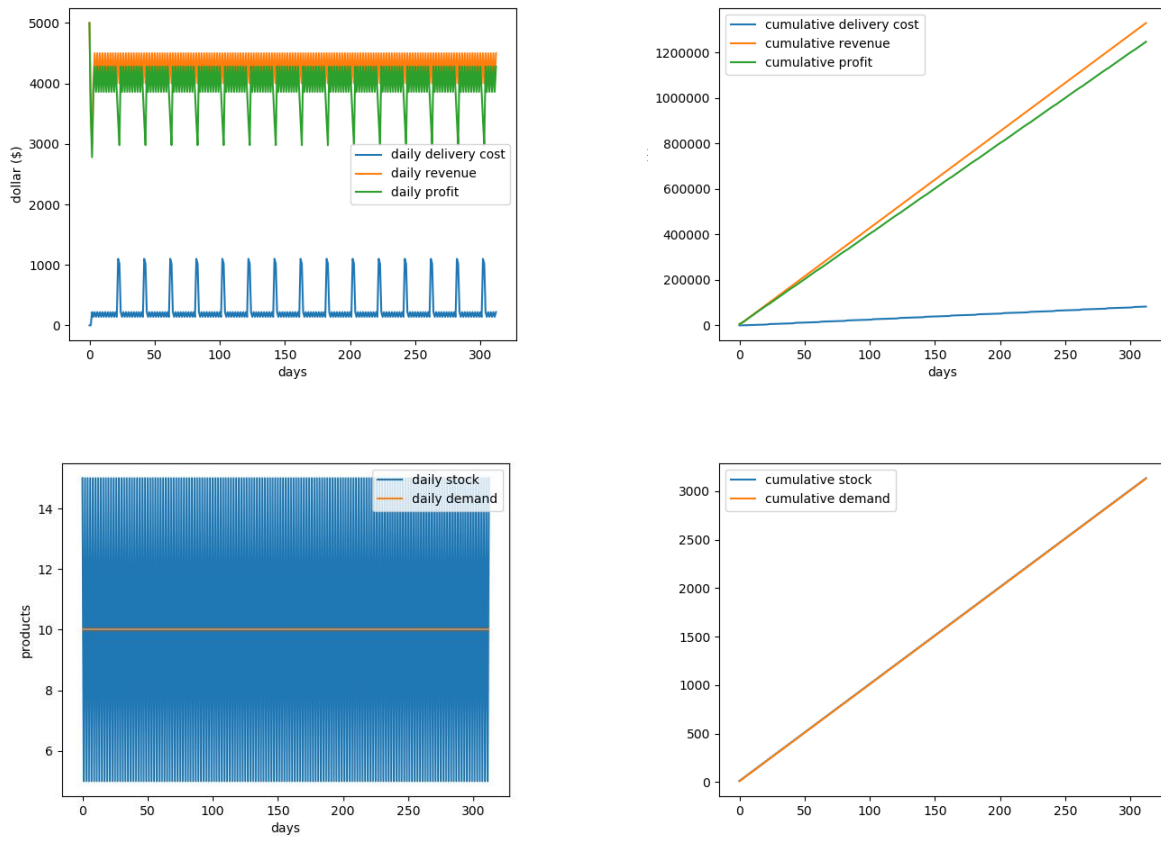
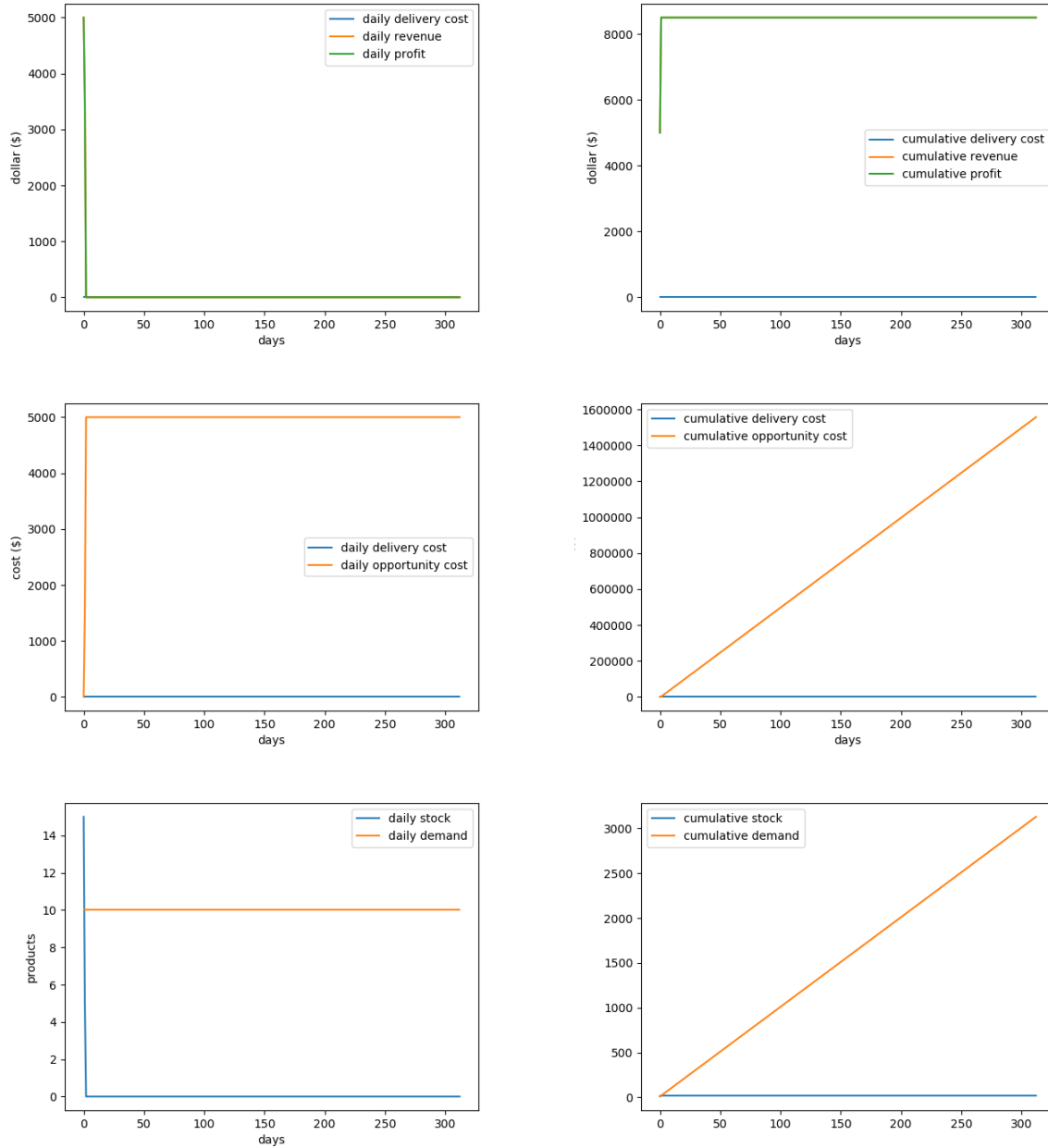Figure 10. Scenario 1, constant demand, daily deliveries to max of 5 stores.

Figure 11. Scenario 2, constant demand, no deliveries

In the first case, the output for no deliveries for entire year was observed. As expected, the delivery cost was observed to be zero throughout, while the profit was constant based on the initial sales of the existing products in the stores and the opportunity cost kept increasing linearly. The daily stock plummeted to zero and daily demands rose to a constant value of 10. This verifies that the conceptual model and simulation model are performing in a synchronized fashion with each other. From the second case, a linear increase in cumulative delivery and opportunity costs were observed. In addition, the daily demand, daily delivery cost and profit showed an oscillating behavior while the daily revenue remained constant due to a constant demand. This is the expected behavior of the actual sales that should be observed by the company. These results indicate the proper functioning of the algorithm.

**Preliminary Studies**

This section presents some preliminary results obtained from the simulation program. Two basic parameters that can affect the sales were analyzed and evaluated before carrying out the parametric studies. The two cases considered for preliminary studies are as follows:

1) For each restocking event, what percentage of the store is refilled: By varying the refill-percentage parameter (refill_pc) of each store, the extent of restocking can be controlled, which affects the frequency and behavior of the delivering trucks; a low value of refill-percentage would result in the trucks scheduling more frequent visits to the store, while serving more stores in each trip. On the other hand, a high value of refill percentage would result in more dedicated trips for each store.
2) Is it worth the money to expand the Warehouse: Buying a new warehouse can be expensive, however expanding the eating warehouses could be potentially a worthwhile avenue to pursue; in this section, cost-savings for a potential expansion from 650 to 1000 is evaluated.

The effect of store restocking percentage has been discussed in the checkpoint report and will be considered in depth in the following sections. Only the discussion on the effect of warehouse size is provided in this section.

Effect of Warehouse Size

While a big warehouse can be beneficial towards readily stocking the stores, they themselves need to be stocked form the manufacturing plant as well, which could curtail the benefits. The following results with warehouse size of 1000 versus the existing 650 indicate that that there isn't much to be gained from the expansion. While revenue does show an increase, the delivery cost of restocking the warehouses also increases thereby actually reducing the profits for a warehouse with capacity of 1000. Figures 12-13 show the daily trends for the two analyzed cases. The spikes in the delivery costs indicate the tips made to the warehouse from the respective plant. The warehouse with size 1000 clearly shows larger spikes owing to the larger number of trips required to restock the warehouse (since each truck can only carry 50 products per trip). Hence, the **warehouse capacity is retained at 650** for the subsequent studies.

Table 3: Results of study varying warehouse size

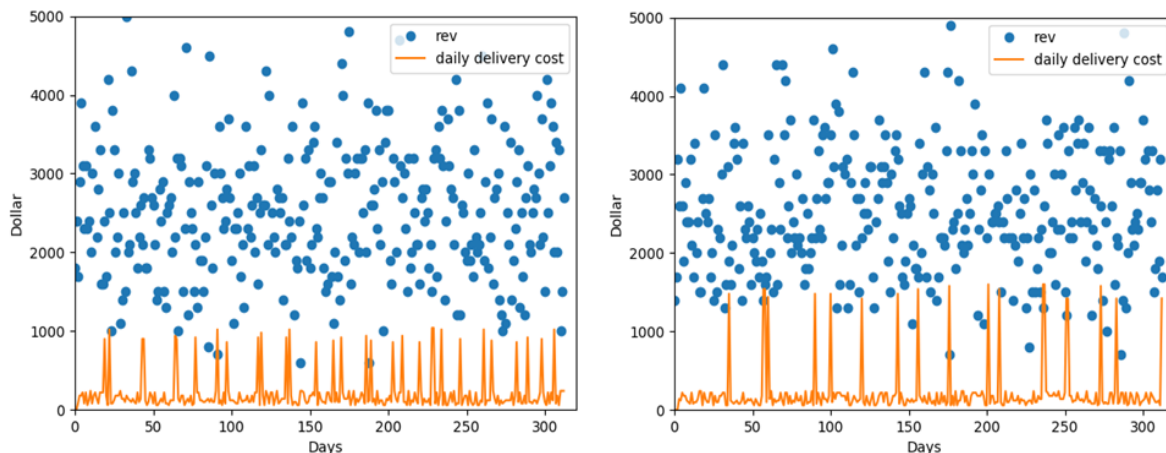| Parameter | Unit | | |
|---|---|---|---|
| Warehouse Size | % | 650 | 1000 |
| Cumulative Revenue | $ | 796700 | 797700 |
| Cumulative Delivery Cost | $ | 66400 | 68460 |
| Cumulative Profit | $ | 730300 | 729240 |



Figure 12. Daily trends in revenue and incurred delivery costs for warehouse of size 650 (left) and 1000 (right)

**Results: Parametric Studies**

This section presents a comprehensive analysis of the influence of some critical parameters obtained from the simulation program on the company finances. Three vital parameters were selected for the analysis as follows:

1. *The minimum restocking percentage:* The optimizing parameter is the minimum stock value defined as a class property (*min_stock*) for each store, which in turn, estimated as a minimum percentage (*min_percent*) of the capacity of the stores. By varying the minimum percentage, a low-stock threshold can be developed, which affects the frequency of delivery-trips to the stores. A low minimum percentage implies infrequent trips made by the trucks to the stores and covering less number of stores in one trip. This would reduce the revenue and increase the opportunity cost but reduce the delivery cost. The lowest minimum restocking percentage without appreciable loss of sales is important to be determined. The values of *min_percent* were varied from 0-90% by keeping the base cost of the trucks as $100, the cost per mile for delivery as $10 and maximum number of stores that a truck can cover per trip as 3. The number of trucks in this case was varied at 1, 5 and 9.

2. *Number of trucks:* By varying the number of trucks (*n_trucks*) and the resulted change in the frequency of the delivering trucks on any given day, the extent of restocking of stores can be controlled based on the priority for the low stocked stores. A low number of trucks delivering the goods would imply dedicated trips to less number of stores but higher overall loss in sales while a high number of trucks would result in more frequent visits to the stores resulting in an overall profit to the stores. However, if the number of trucks is too high, the profits made by increased revenues might be overshadowed by the cost due to increased deliveries. Hence, an optimum higher limit of this value can positively influence the revenue of the stores. The values of *n_truck* was varied from 1-10 keeping the base cost of the trucks as $100, the cost per mile for delivery as $10 and maximum number of stores that a truck can cover per trip as 3. The minimum restocking percentage in this case was varied at 20, 50 and 90%.

3. *Maximum number of stores that can be traversed by a truck in one trip:* By varying the maximum number of stores that a truck can traverse in one trip (*max_stores*), the extent of restocking of stores can be controlled based on the priority of stocking. If more number of stores are visited by a truck, then the stores cannot be restocked at lower minimum percentages, which means incur of high revenue cost, but high delivery cost at the same time. The tradeoff between these two costs decide the extent of profit for the company. Thus, an optimal maximum value of this parameter governs the sales of the company. The values of *max_stores* were varied 1-5 keeping the base cost of the trucks as $100, the cost per mile for delivery as $10 and number of trucks as 5. The minimum restocking percentage in this case was varied at 20, 50 and 90%.

A summary of the parameters that to be analyzed in this study along with the corresponding baseline parameters is presented in Table 4. The base cost per day (base_cost) and delivery cost per mile (cost_per_mile) are kept constant throughout the parametric studies.

Table 4: Analyzing parameters with the corresponding baseline parameters

| Analyzing Parameter | Values | Baseline Parameters | Values |
|---|---|---|---|
| **min_percent** | 0 – 90 | n_trucks | 1, 5, 9 |
| | | max_store | 3 |
| **n_trucks** | 1 – 10 | min_percent | 20, 50, 80 |
| | | max_store | 3 |
| **max_store** | 1 – 5 | min_percent | 20, 50, 80 |
| | | n_trucks | 5 |

<u>Effect of Minimum stock threshold for stores</u>

The minimum stock threshold is expressed as percentage of capacity, below which a low-stock warning is triggered in the simulation. For evaluating the effect of this parameter on the macro-financials, the simulation program was executed with threshold values from 0 to 90%. In other words, when the stock of the store falls below this threshold, the store requests for restocking at the next available opportunity. The analysis of influence of this parameter on company's finance at three different baselines is computed, as specified in Table 4. Figures 13 presents the results from varying the minimum stock value of the stores on the cumulative revenue, lost revenue due to missed sales, deliver costs and the net profit at the end of the year for the three baseline conditions.
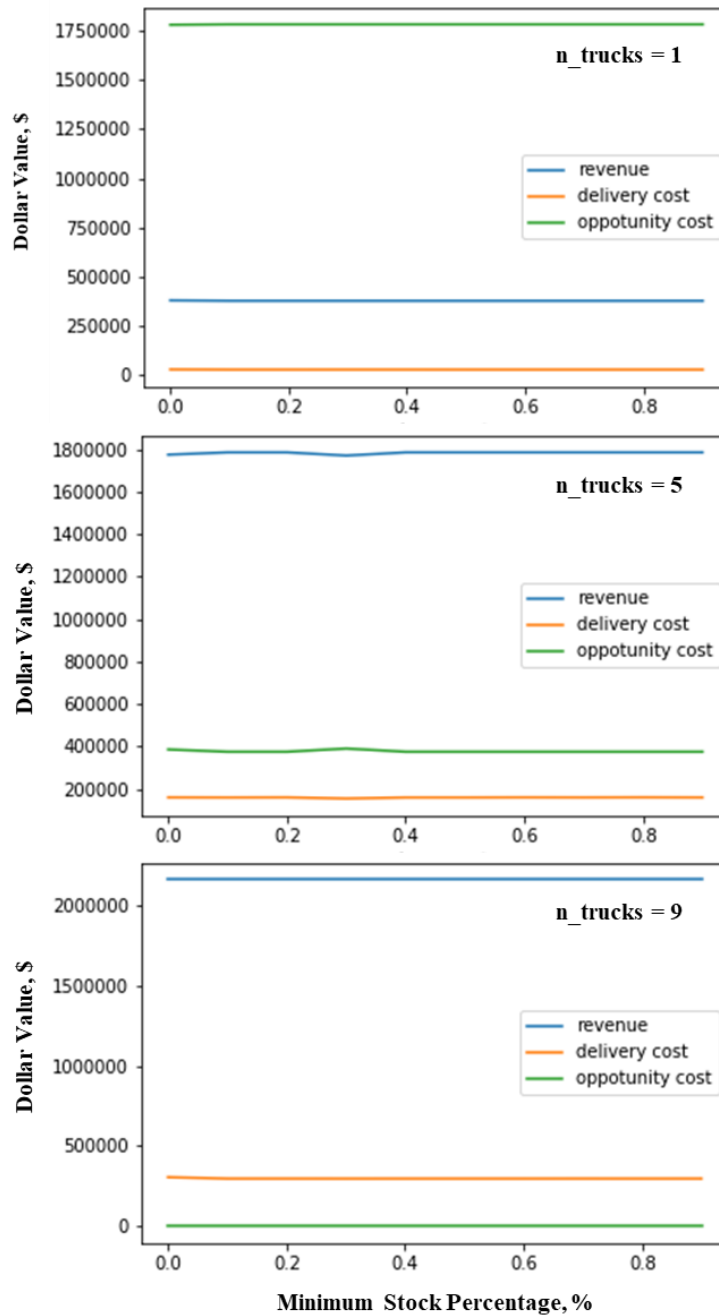


Figure 13. Trends of cumulative finances against minimum stock percentage for different number of trucks.

As shown in the figures, the parameter does not seem to affect the sales. However, an increase in the cumulative revenue and delivery cost and a decrease in opportunity cost is observed as the baseline parameter, *n_trucks* is increased. This is obviously because of the increase in frequency of deliveries. Figures 14 presents the results from varying the minimum stock value of the stores on the cumulative profit at the end of the year for the three baseline conditions.
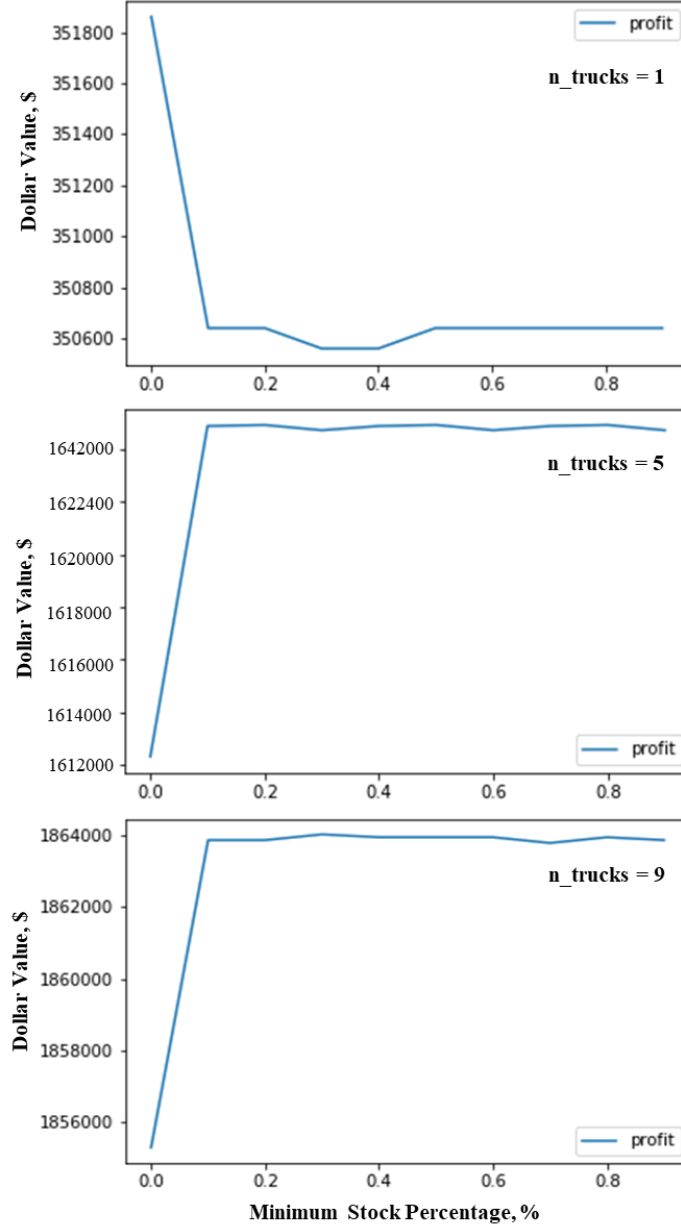


Figure 14. Trends of cumulative profit against minimum stock percentage for different number of trucks.

When the variation in cumulative profit of the company against the minimum restocking percentage is observed, it is found that for one truck, the profit decreases and then becomes almost constant while for more trucks the profit increases and then becomes almost constant, because higher minimum restock percentage correlates with frequent deliveries implying low opportunity cost; however in the case of 1 truck, the deliver cost is so high that it outweighs the high revenues for high minimum restock percentage.

The cumulative profit is observed to increase with increase in number of trucks, although the difference in increase reduces as number of trucks increase (Profit of $350,600 for 1 truck vs profit of $1,642,000 for 5 trucks vs profit of $1,864,000 for 9 trucks). This hints that the number of trucks can be increased up to a certain value after which the rate of increase of cumulative profit will reduce a lot. Also, the profit for minimum stock warning higher than 10% is almost constant for a given number of truck, so it is economically feasible to lower the minimum stock percentage as low as 10% without missing significant sales.

Effect of number of trucks

The following section presents the analysis of influence of number of trucks on the company finances, at three different baselines. The truck parameters considered are number of trucks operating for delivery and the maximum stores each truck can visit per trip. The three baselines are defined using minimum percent parameter of the stores and warehouses, which refers to the stock threshold below which a low-stock warning signal is transmitted. The scenarios are described in the table below.

Table 5 Baseline Cases

| Baseline Parameter | Values | Description |
|---|---|---|
| min_percent | 20, 50, 80 | Low stock warning when current stock at 20% of capacity |

The following sections describes the two case studies in detail.

The number of trucks operating to deliver products to the warehouses and stores is a critical part of the supply chain logistics. The delivery cost as well as the revenue made from sales are heavily influenced by this parameter. In this series of tests, the number of trucks were varied 1-10 and the resulting financial outcomes are studied.

Table 6 Study cases for analysis involving number of trucks

| Parameter | Values |
|---|---|
| No of trucks | Variable (1 – 10) |
| min_percent | 20%, 50%, 80% |
| max_store | 3 |

Figure 15 presents the trends associated with total revenue made from sales, total delivery cost incurred and the total lost opportunity cost resulting from sales not met over the calendar year versus number of trucks. The three graphs correspond to the three baseline cases enlisted in Table 6. In all three graphs, the delivery cost and revenue increase with increasing number of trucks, while the lost-opportunity cost decreases. This trend is aligning with expectation that more sales can be made following timely restocking of stores with greater number of trucks.

Comparing the baseline cases, a lower threshold for low-stock warning (min_stock) results in slower rate of increase of revenue. This follows from a greater lost sale because of infrequent deliveries. A restocking threshold of 20% shows that the revenue curve intersects the lost-opportunity curve at seven trucks, while the same quantity at 80% restocking threshold is two trucks. This information can be greatly useful when planning expansion of delivery fleets or designing stocking policies. Another notable aspect from Figure 15 is that the revenue curve increases to a point and plateaus, which corresponds to the lost-opportunity cost reaching zero, i.e. no sales being lost due to insufficient store stocks.

Figure 16 presents the year-end profits plotted against the number of trucks. The trend is to increase proportional to the number of trucks, however, the rate of increase slows down after a certain value. At a min_percent value of 50%, three is limited additional profit to be made beyond 7 trucks, while at min_percent of 80%, three trucks are sufficient to make a considerable profit. This is because additional trucks beyond requirements do not add much value.
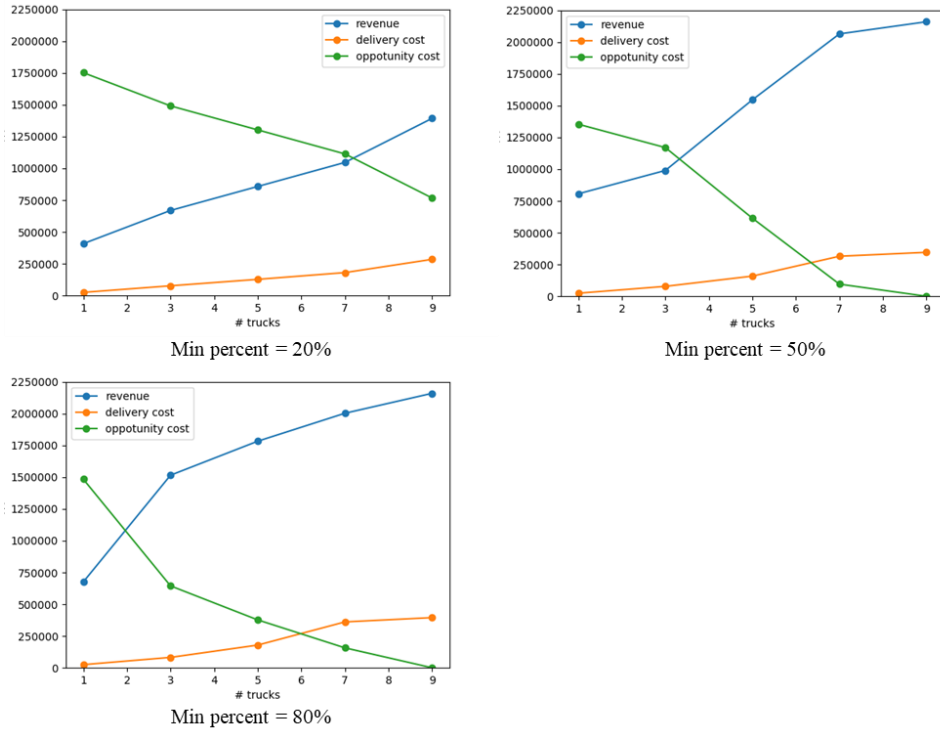
Figure 15. Variation of total revenue, delivery cost and opportunity-loss cost over one year versus number of delivery trucks operating at three baseline values of min_percent.
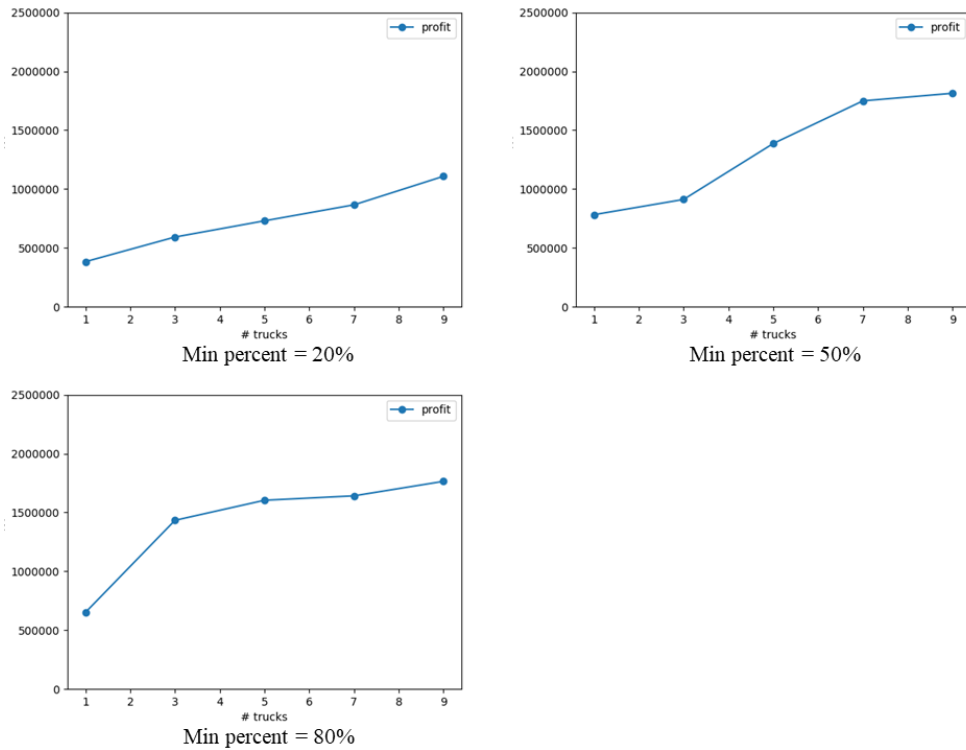


Figure 16. Variation of total profit over one year versus number of delivery trucks operating at three baseline values of min_percent.

Figure 17 presents the same study but with data plotted against number of days in each sub plot. The subplots correspond to 1, 3, 5, 7 and 9 trucks respectively. The orange curve represents the cumulative lost-opportunity cost while the blue curve is the cumulative revenue. As the truck number increases, the gap between the two curves decreases, and eventually at 7 trucks the revenue overtakes the losses. Therefore, a 7-truck delivery fleet is recommended for achieving the most sales.
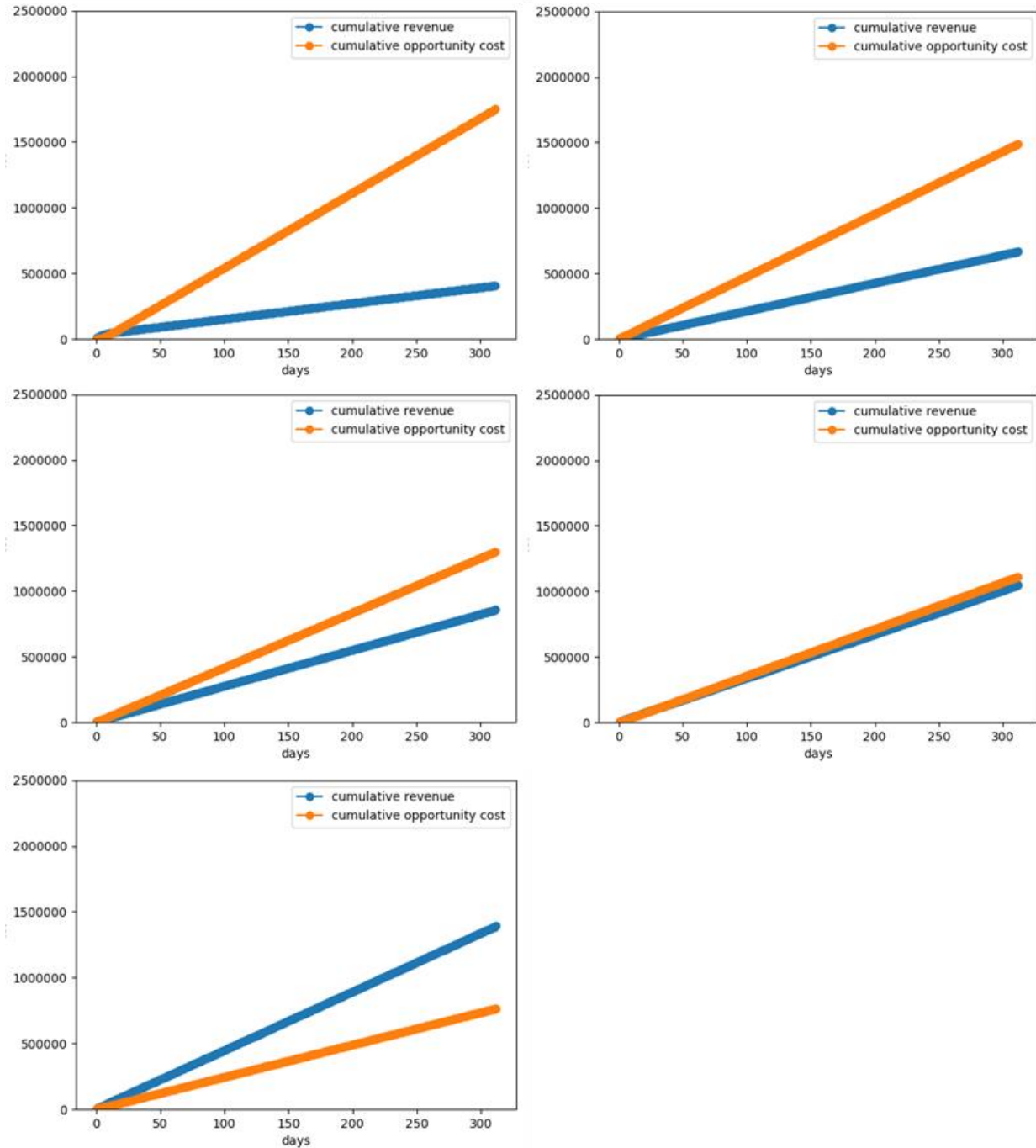


Figure17. Cumulative revenue and lost-opportunity cost {1,3,5,7,9} trucks at min_percent=0.2.

Effect of Number of Stores per Visit

The number of stops a truck is restricted to is another crucial parameter that affects the delivery process. If the trucks are restricted to only one or two stores, then these stores get a larger stock refill, while the other stores in need of restocking will have to wait for another truck to stop by. On the other hand, a max_store value of five or more would mean that the trucks have more liberty in choosing a route based on the needs of the stores. Therefore, the max_store parameter is to be analyzed in conjunction with store-refill percentage parameter and the number of trucks parameter.

Table 7 Study cases for analysis involving maximum allowable store-visits per trip of the truck

| Parameter | Values |
|---|---|
| max_store | Variable (1 – 5) |
| min_percent | As per baseline |
| No of trucks | 5 |

Figure 18 shows the trends that are associated with various finance-related parameters of the company. The first observation is that the revenue is above the opportunity-loss curve, which is favorable. Second, a max_store value of 1, is ineffective in delivery trips as seen in the high lost-sales value and low revenue, because each truck is being forced to go to a store and back to the warehouse in every trip, leading to a very slow restocking process. For higher values of max_store, the results depend on the value of minimum percentage of restocking of stores. At 20% min_percent, the stores are 80% empty before a restocking request is placed. Therefore, a truck can only adequately restock about 2-3 stores as it has a capacity of only 50 products.



Min percent = 20%

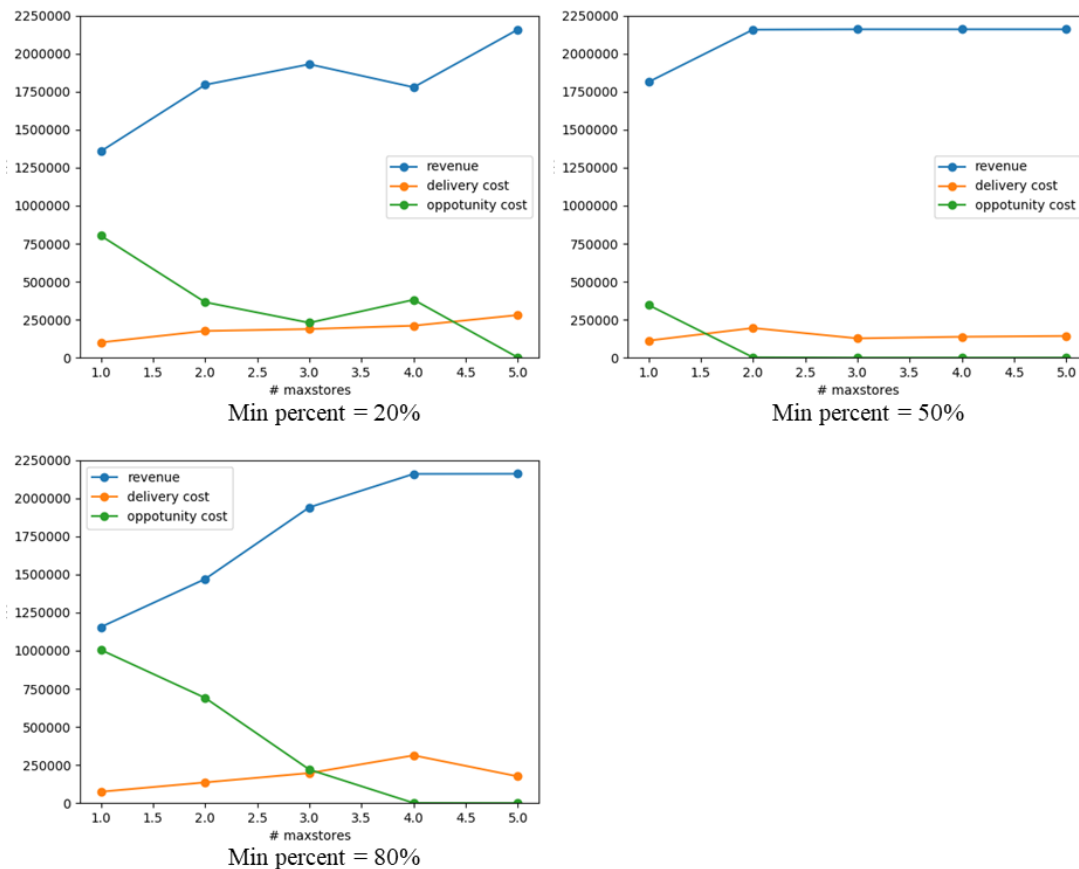Min percent = 50%

Min percent = 80%

Figure 18. Variation of total revenue, delivery cost and opportunity-loss cost over one year versus number of maximum store-stops per delivery

It follows that even at higher values of max_store, each delivery trip is only concerning about 2-3 stores and hence there isn't a great variation in revenue beyond this value. At a min_percent value of 80%, the stores will request restocking very frequently and only need restocking for 20% of its capacity, so a truck can easily service more than 2 stores in each trip. Following this reasoning, if the max_store is restricted to 2-4, the truck is not unloading all its products before coming back to the warehouse, which is inefficient. At greater values of about 4, the trucks can visit many stores and facilitate more sales. This trend is also reflected in Figure 19 with the profits.
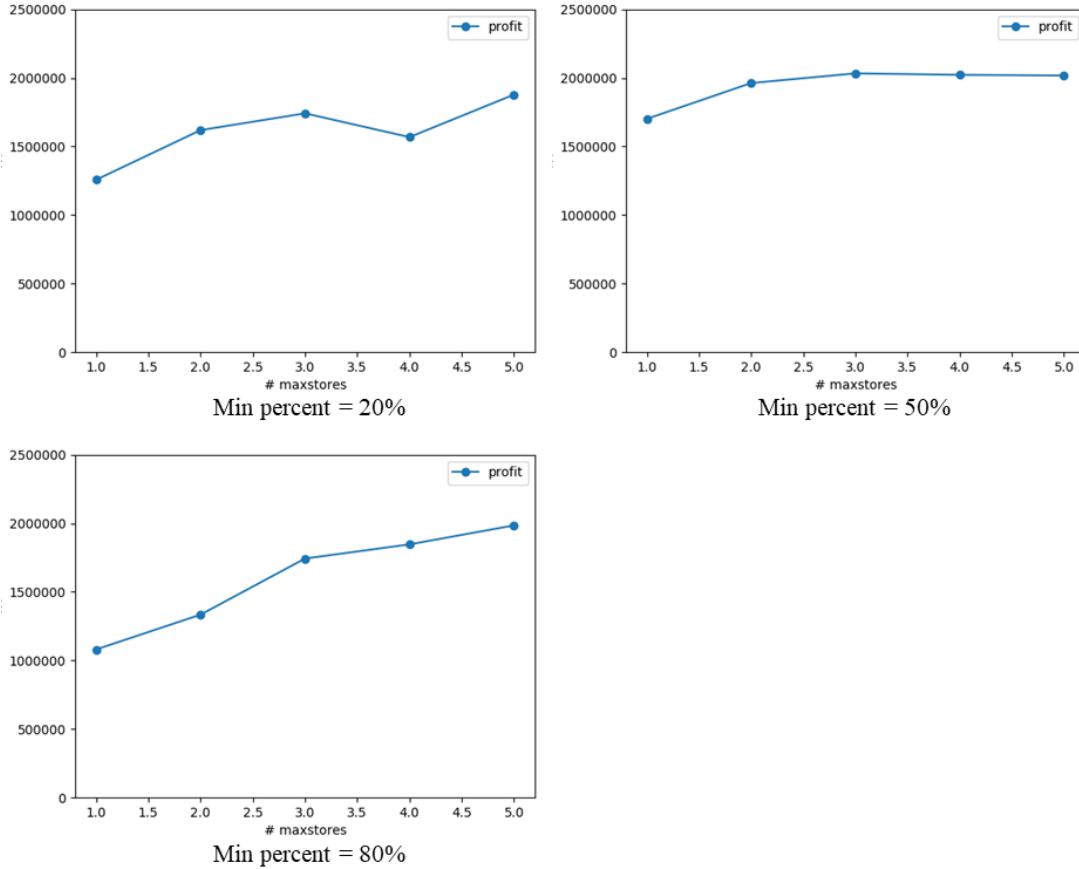


Figure 19. Variation of total profit over one year versus number of maximum store-stops per delivery

Since the max_store parameter depends closely depends on both the number of trucks and the min_percent, another set of results were generate, this time keeping the min_percent a constant at 20% and varying the number of trucks between 1 and 9. The previous section already presents results for number of trucks of 5. Figure 20 presents results from this study and helps explain the physical system better. When the number of trucks is 1 (Figure 20a), having a higher limit on max_store is better to generate revenue the one truck can service more stores with each trip. If the limit is lower, the stores must wait longer before restocking. The profits increase and plateau at about a max_store value of 3. When the number of trucks is 9, the revenue is plateaus at a max_store value of 2, since more trucks can independently service all stores.

The combinations of min_percent and number of trucks, i.e. 50% with 5 trucks, and 20% with 9 trucks, have shown a similar behavior from Figures 18 and 19, which matches the expectation that when the min_percent is higher, fewer trucks are needed as the refill volume is lower.
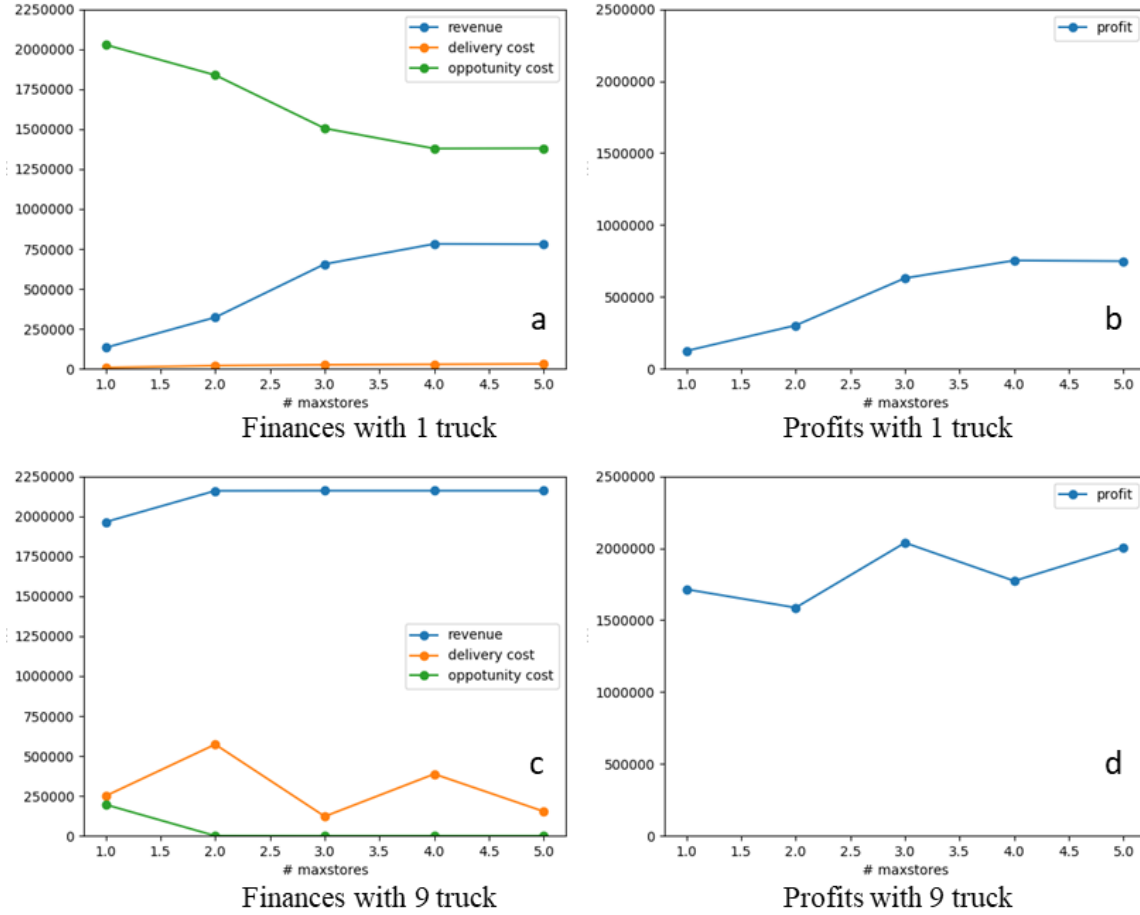
Figure 20. Results showing max_stores versus finances with 20% min_percent and varying number of trucks.

**Conclusions**

By modifying some of the constraints in the original problem statement, a new supply chain network system was designed and simulated by using a time-stepped model. The interplays of the critical parameters that are closely related to the profits, including the minimum restocking threshold of each store, the number of trucks, and the number of store visitations, are captured in the parametric studies. The results have clearly demonstrated that the importance of utilizing simulation model to study the potential behavior that may appear in the system and to optimize the properties of the system to achieve the desired objectives.

In the model presented here, we identified that in the supply chain network under the aforementioned conditions, we can maximize the profit of the industry by using the minimum restocking threshold of 10% with a 7-truck fleet performing delivery for equal or less than 9 stores per trip.

**References**

Birta, L. G., & Arbez, G. (2013). *Modelling and simulation*. London: Springer.
Harrell C, Ghosh BK, Bowden RO Jr (2004) Simulation using ProModel, 2nd ed. McGraw-Hill, New York