ISE 533: Integrative Analytics, Spring 2020

Multi-Location Transshipment Draft

Group 1:

Xinjie Sun

Xinrui Yan

Yiwen Cao

Zelong Chen

Abstract **3**

Goal and Scope of Project **3**

Assumptions 3

Overview of Models/Algorithms **4**

Original Retail Optimization Formulation: 4

Updated Retail Optimization Formulation: 5

Stochastic Decomposition (SD) Solver: 6

Stochastic Quasi-Gradient (SQG) Method: 6

Data Sources **7**

Discussion of Results **7**

Original Transshipment Formulation vs Updated Formulation 7

Stochastic Decomposition (SD) vs S
tochastic Quasi-Gradient (SQG) 8

Future Work **9**

Data Source 9

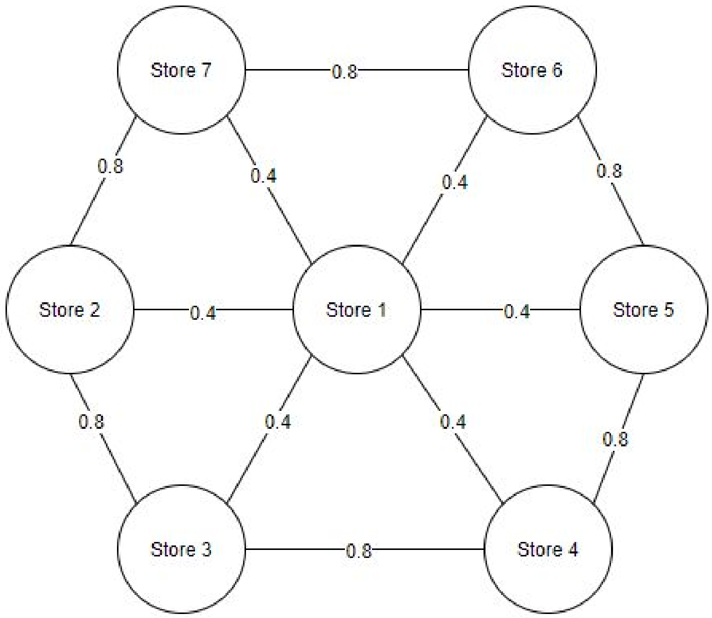
Abstract

This project is aimed at comparing the performance of different models while solving the multi-location transshipment problems. This kind of problem is designed to minimize the collective long-run average costs for multiple retailers supported by one supplier. In order to do the comparison, we applied both SD solver and SQG method to the original model as well as the updated model. The difference between the two models is that we encourage retailers to transship between each other to meet the demands and the transshipment cost between retailers is calculated based on distances in the updated one. After comparison, we found the SD solver takes significantly less time to get almost the same result as the SQG method.

Goal and Scope of Project

Assumptions

- Below is a figure that describes our assumptions of transshipment cost per unit, c_{ij} , in our Updated Retail Optimization Formulation. The original transshipment cost was uniformed between all stores with the value of 0.1 per unit. We wanted our new model to simulate the real world, where transshipment cost varies based on the distances from retailers. Thus, we created the figure below where Store 1 is a super-store that is located in between all the other stores with the lowest transshipment cost. While other retailers will have double the transshipment cost due to their further distance from one another. (i.e. Store 1 to Store 6 transshipment cost is 0.4 per unit while Store 6 to Store 3 is 0.8)



Overview of Models/Algorithms

Original Retail Optimization Formulation:

$$\text{minimize } \sum_i h_i a_i + \sum_{i \neq j} c_{ij} t_{ij} + \sum_i p_i r_i$$

$$\text{subject to} \quad f_i + \sum_{j \neq i} t_{ij} + a_i = s_i\,, \ \forall i \qquad (1a)$$

$$f_i + \sum_{j \neq i} t_{ji} + r_i = d_i\,, \ \forall i \qquad (1b)$$

$$\sum_i r_i + \sum_i q_i = \sum_i d_i \qquad (1c)$$

$$a_i + q_i = s_i\,, \ \forall i \qquad (1d)$$

$$a_i,\ f_i,\ q_i,\ r_i,\ s_i,\ t_{ij} \geq 0,\ \forall i, j$$

Variables:

- a_i = ending inventory held at retailer i
- f_i = stock at retailer i used to satisfy demand at retailer i
- q_i = inventory at retailer i increased through replenishment
- r_i = amount of shortage met after replenishment at retailer i
- t_{ij} = stock at retailer i used to meet demand at retailer j, using transshipment
- d_i = demand observed at retailer i

Objective Function Variables:

- h_i = unit cost of holding inventory at retailer i (1 unit cost)
- c_{ij} = unit cost of transshipment from retailer i to j (0.1 unit cost)
- p_i = penalty cost for shortage at retailer i (4 unit cost)

Formulation Description

In the original optimization formulation of the multilocation transshipment problem, the objective was to minimize the cost of operation. The cost of operation includes holding cost from surplus inventory, transshipment cost of stock transfer to other retailers, and penalty cost for not meeting demand. We are minimizing this cost with respect to the constraints stated above (1a, 1b, 1c, 1d).

Updated Retail Optimization Formulation:

$$minimize \sum_i h_i a_i + \sum_{i \neq j} c_{ij}\, t_{ij} + \sum_i p_i r_i - \sum_i const \times f_i$$

$$subject\ to \quad f_i + \sum_{j \neq i} t_{ij} + a_i = s_i\ , \ \forall i \quad (1a)$$

$$f_i + \sum_{j \neq i} t_{ji} + r_i = d_i\ , \ \forall i \quad (1b)$$

$$\sum_i r_i + \sum_i q_i = \sum_i d_i \quad (1c)$$

$$a_i + q_i = s_i\ , \ \forall i \quad (1d)$$

$$a_i,\ f_i,\ q_i,\ r_i,\ s_i,\ t_{ij}\ \geq\ 0,\ \ \forall i, j$$

Variables:

- a_i = ending inventory held at retailer i
- f_i = stock at retailer i used to satisfy demand at retailer i
- q_i = inventory at retailer i increased through replenishment
- r_i = amount of shortage met after replenishment at retailer i
- t_{ij} = stock at retailer i used to meet demand at retailer j, using transshipment
- d_i = demand observed at retailer i

Objective Function Variables:

- h_i = unit cost of holding inventory at retailer i (1 unit cost)
- c_{ij} = unit cost of transshipment from retailer i to j (vary based on distance)
- p_i = penalty cost for shortage at retailer i (4 unit cost)

Formulation Description

The updated formulation is similar to the original formulation with only a couple small changes. We added term $-\sum_i const \times f_i$, where $const = 0.01$ in this implementation, in the objective function to relax the penalty costs when a large demand is met by the retailer. This term provides an incentive for retailers to meet as much demand as they can and avoid shortages of demand. We also wanted to change and varied the term c_{ij} to reflect the actual world, where transshipment cost is varied based on distances between different retailers. Refer to the assumption section above for the distances we set.

Stochastic Decomposition (SD) Solver:

The Stochastic Decomposition Solver is an algorithm designed to solve two-stage optimization problems with randomness or uncertainty introduced to only the right-hand-side *r* (in this problem, the demand at each retailer). At each iteration, the SD solver introduces a new sampled outcome from the outcome sample space and generates a lower bounding linear approximation. Then using this lower bounding linear approximation, we can then solve only one second-stage LP in any iteration to effectively generate a similar result that can be obtained through asymptotic convergence of all possible second-stage LPs without solving a large amount of LPs.

For more details please refer to these papers:

- *Higle, J. L., and S. Sen. 1991. Stochastic decomposition: an algorithm for two-stage linear programs with recourse.*
- *Higle, J. L., and S. Sen. 1999. Statistical approximations for stochastic linear programming problems*

Stochastic Quasi-Gradient (SQG) Method:

This method is designed to calculate the subgradient of piecewise objective function. The first step is to draw M i.i.d samples of random vectors which denote M different scenior of demand. Then use these realizations to solve the dual problem M times in order to obtain the average of optimal dual multipliers. And the estimate of subgradients of the primal problem could be calculated by summing up the average of optimal dual multipliers. The last step is choosing the step size alpha and updating the given policy S. After repeating these steps for K times, we are able to get the optimal value.

Data Sources

The original Multilocation Transshipment Optimization Formulation were based on the papers:

- *Yale T. Herer , Michal Tzur & Enver Yücesan (2006) The multilocation transshipment problem*
- *Suvarjeet Sen, Lei Zhao (2006) A Comparison of Sample-Path-Based Simulation-Optimization and Stochastic Decomposition for Multi-Location Transshipment Problems*

We obtained the Retail dataset from the cORe Dev website:

- https://core.isrd.isi.edu/chaise/record/#1/Core:Instance/RID=W19A
- SMPS files includes:
  - .cor file
    - Cost variable values and original Multilocation Transshipment Optimization formulation
  - .sto file
    - The amount of demands observed at each retailer along with the probability of occuring
  - .tim file
    - Breaking down the core file into nodes corresponding to the individual stages.
- Individual_loc.csv file:
  - Including the latitude and longitude of each retailer which allow us to calculate the distances between retailers

Discussion of Results

Original Transshipment Formulation vs Updated Formulation

SD Solver	s_1	s_2	s_3	s_4	s_5	s_6	s_7	Obj
Original Formulation	108	220	162	191	197	181	190	154.4
Updated Formulation	134	218	159	189	194	179	188	175.8

The solutions provided by the SD solver for both the original and updated formulation was just as we expected. We can see an increase in order-up-to quantity (s_1) as Store 1 is considered to be a superstore that provides the cheapest transshipment cost to all the other retailers. In addition, all the other stores had a decrease in order-up-to quantities due to the increase in transshipment cost when compared to the original transshipment formulation. The increase in transshipment cost also increased the total operating cost of all retailers, which in terms caused an increase in the objective value.

The term $-\sum_i const \times f_i$ did not seem to play a big effect on the optimization problem as the constant might be too small. Thus, did not provide a big enough incentive for retailers to satisfy a larger demand.

SQG Method	s_1	s_2	s_3	s_4	s_5	s_6	s_7	Obj
Original Formulation	106.5	212.1	160.0	180.9	191.2	180.9	181.2	175.71
Updated Formulation	111.6	217.5	164.8	186.4	194.6	184.8	183.9	175.95

After tuning the hyperparameters, we applied the SQG method with M=200, K= 1500 and step size $\alpha = K/i$ and obtained the above result. As can be seen from the table, the result is similar to the solution provided by SD solver. However, it's easy to figure out that the quality of the solution is worse than that of SD because the objective value is higher than the value of SD.

Stochastic Decomposition (SD) vs Stochastic Quasi-Gradient (SQG)

Run Time (Seconds)	**SD Solver**	**SQG Solver**
Original Formulation	58.716	828.805
Updated Formulation	38.889	822.834

From the run time table as well as the comparison of solutions, we are able to find that SD solver not only provides solutions of better quality, but also takes less time. To conclude, SD solver would be a good fit to solve this problem.

Future Work

Data Source

- Obtain real world data from retailers such as Target or Walmart and do an in depth analysis in retailer transshipment
- Apply different holding cost as well as shortage penalty to different retailers according to the sales

Algorithm

- Implement Infinitesimal Perturbation Analysis (IPA) from the original Multilocation Transshipment Paper to see the implementation difference and run time difference between our current algorithms.